Eye-Tracking Technology in Vehicles: Application and Design

By
Vasileios Selimis

Thesis submitted for the degree of

Master of Philosophy

City University London

Department of Electrical and Electronic Engineering

August 2015

THE FOLLOWING PARTS OF THIS THESIS HAVE BEEN REDACTED FOR COPYRIGHT REASONS:

p 9: **Fig 1.1.** Image of human eye.
p 22: **Fig 2.2.** Image of Digital Camera Signal Flow.
p 43: **Fig 2.6.** Image of pupil.
p 98: **Fig 3.1.** Seeing Machines' DSS suite.
p 99: **Fig 3.2.** Image of DSS software.
p 100: **Fig 3.3.** Image of Seeing Machines' faceLAB approach.
p 101: **Fig 3.4.** Image of Seeing Machines' faceLAB approach.
p 130: **Fig 5.11.** Algorithm for Driver Fatigue Detection System.

Abstract

This work analyses the eye-tracking technology and, as an outcome, it presents an idea of implementing it, along with other kinds of technology, in vehicles. The main advantage of such an implementation would be to augment safety while driving. The setup and the methodology used for detecting human activity and interaction using the means of the eye-tracking technology are investigated. Research in that area is growing rapidly and its results are used in a variety of cases. The main reasons for that growth are the constant lowering of prices of the special equipment that is necessary, the portability that is available in some cases as well as the easiness of use that make the usage of that technology more user-friendly than it was a few years ago.
The whole idea of eye-tracking is to track the movements of the eyes in an effort to determine the direction of the gaze, using sophisticated software and purpose built hardware. This manuscript, makes a brief introduction in the history of eye monitoring presenting the very early scientific approaches used in an effort to better understand the movements of the human while tracking an object or during an activity. Following, there is an overview of the theory and the methodology used to track a specific object. As a result there exists a short presentation of the image processing and the machine learning procedures that are used to accomplish such tasks. Thereafter, we further analyze the specific eye-tracking technologies and techniques that are used nowadays and the characteristics that affect the exact choice of eye-tracking equipment. For the appropriate choice we have to take into account the area of research-interest in which the equipment will be used. In addition, the main categories of eye-tracking applications are presented and we shortlist the latest state of the art eye-tracking commercial systems. Following, we present our first approach, trying to describe an eye-tracking device that could be used in vehicles offering much better safety standards, controlling various parameters, continuously checking the readiness of the driver and alerting him for potential imminent collision incidents. Finally, we describe the existing way of connecting a device, in our case an eye-tracker, can be connected to an automobile's system.

Contents

Figures

Tables

Acknowledgments

I would like to thank Dr. Ioannis M. Aslanides, MD, PhD, FRCOphth, Medical Director and founder of Emmetropia, Mediterranean Eye Institute, who supported me mentally and also, sponsored my studies. Without him this thesis would never become true.

Also the staff of Emmetropia, Mediterranean Eye Institute and its research laboratory –Optotech- that mostly aims its research in eye-tracking applications and technologies, must be thanked for their support and help.

Furthermore, I really owe many thanks to my supervisor Veselin Rakocevic from the City University of London for his assistance, support and endless patience in supervising the writing of this thesis.

Above all, my thanks go to my family as their continuous help and support were vital for the creation of this thesis. I would like to especially dedicate this thesis to my late father.

Chapter 1

Introduction

1.1 A general introduction into the eye-tracking technology

Eye-tracking is the process of calculating the motion of the eye relatively to the head. An eye-tracker is a device for measuring eye positions and eye movement. Eye-trackers are used in research mainly on the visual system, in marketing, psychology/psycholinguistics, marketing, product design and as input device for human computer interaction.

Our eyes are vital in order to accomplish even the simplest tasks that are just part of our routine during everyday tasks. Specifically, we need our sight to read, learn, watch, navigate etc. again and again but in most cases we do not really control the movements of our eyes which seem to happen automatically thanks to orders coming from our brain. Even though, watching seems a rather simple task to all of us, it is in fact a very complex process which requires constant sampling of the various stimuli that exist around us and then make the movement required to fixate the eye accordingly. Our brain, gathers that series of images, perceives the scene and processes the information obtained.

To better understand eye-tracking a brief introduction along with key technical details will follow in this section and those subjects will be further analysed, in detail, in Chapter 2. When we are seeing something or someone what really happens is that the light gets through the eye via the pupil, the image is turned upside down in the crystalline lens and it is finally projected on the retina at the back of the eye. The retina is full with special cells separated into two groups, the cones (that are responsible for the high visual detail and the colour vision) and the rods (that are responsible for vision under dim light as they are very sensitive to it).

Figure 1. 1: The cones and rods on the human eye[1]

Those cells convert the incoming light into electrical signals which are sent, through the optic nerve, to the visual cortex. Interestingly enough, the most important characteristic in eye-tracking is not an eye movement, but the occasions in which the eye stays still over a certain period of time. This lack of movement is called a fixation and it takes place when we are gathering information from our environment. Nevertheless, even during a fixation the eye is never totally still as three separate types of tiny movements are present. Those are drifts, microsaccades and tremor and are so fast and small that cannot be easily, if ever, detected by all the eye-trackers (mainly because of the high frequency needed by the hardware). Along with the fixation another type of eye movement is present which is called saccade and is the very fast movement of the eye from one fixation point to another.

[1] Image source: http://www.mhhe.com/biosci/esp/2001_gbio/folder_structure/an/m3/s3/anm3s3_4.htm

Nowadays, the most common method for recording the eye moments, in order to estimate the gaze point of a subject, is by tracking the pupil and a corneal reflection (some systems require more than one) of the eye. The corneal reflection results after illuminating the eye with one (or more) infrared source in order to avoid natural light reflections. A typical video-based eye-tracker in order to calculate where someone looks follows three major steps: image acquisition, image analysis, and gaze estimation. The eye-tracker's software takes into account the fact that the relative positions of the pupil and the corneal reflection change in a very specific way. The pupil moves faster and the corneal reflection moves slowly. Based on that fact, the eye-tracker measures the relative distance between them and calculates the gaze position each time.

By understanding the way the eyes are moving, while reading or looking at an image, we can export useful information about the processes that take place during those tasks. In general, eye-tracking is used in a variety of fields, like designing human-computer interfaces, monitoring the training evolution of pilots/drivers, in marketing researches and in defensive systems to name a few.

Specifically, concerning the human-computer interfaces, except of making the software or the hardware more user-friendly this kind of technology can be used to broaden the communication, learning and in general interacting capacities of people who are disabled. When, used in a simulator the eye-tracking can offer to the trainer important information about the scanning patterns of the trainee reducing significantly the time needed to fully train an apprentice. In marketing, the researchers have the ability to understand what really catches the attention of a potential customer, where to put their products and so on. Finally, eye-tracking is already used in some cases by the defense industry mainly helping the pilots to control (select target, open fire) their weapon systems using their eyes only.

1.2 Task and Contribution of the Thesis

The main objectives of this thesis are:

- To present a thorough investigation of the eye-tracking technology along with its scientific background.
- To describe the scenario of a new vehicular application, based on the eye-tracking technology, which potentially can make the whole driving procedure a lot safer than it is today.
- To present the existing ways a device can be fused in the systems of an automobile.

In the literature, and nowadays in the market, there are some applications proposed and used that have the same goal as the one mentioned above. The vast majority of those approaches are using the eye-tracking technology as a way to detect the gaze of the driver and determine if the driver is about to fall asleep, if he spends lots of time not looking at the road or the mirrors of the vehicle but to other things (e.g. radio, mobile phone, other passengers in the car) and in general to detect any kind of concentration loss by the driver. In case such a loss is detected usually a sort of alarm sounds and a slower velocity is proposed by the system. In reality though, the vast majority of those devices just detect the eye, checking if it is open or not, and as a consequence the device is not aware of where the driver is looking each specific moment.

The goal of proposing a design framework of a new approach is to add extra safety control measures using the eye-tracking. In combination with other technologies, that modality could help the driver avoid looking around to gather information concerning the car and make it easier to remain focused on the road.

In general, the thesis presents in detail the eye-tracking technology and its way of operation and finally introduces a concept that could increase the safety standards during driving a vehicle. With the combined advantages of existing and well-known modalities and technologies, it is possible to make driving, an everyday task of millions of people, significantly safer without having to spend lots of money to acquire any kind of special and expensive equipment.

1.3 Structure of the Thesis

The thesis is structured as follows. In chapter 2 a comprehensive analysis of existing theoretical solutions for tracking and eye-tracking especially is described. Existing eye-tracking applications and especially some used in vehicle systems are presented in chapter 3. In chapter 4, the new approach proposed in this thesis is introduced as well as an initial algorithm design and its theoretical challenges. This new approach is based on existing technologies such as the eye-tracking, the HUDs one and the one that uses GPS devices in vehicles to detect the exact position of a car every specific moment. Following, in chapter 5, the way a device can be integrated in the systems of an automobile is presented. Finally in chapter 6, where the thesis concludes, there is a summary of the previous chapters and the main advantages and drawbacks of the approach are discussed.

Chapter 2

2.1 A Century of Eye Monitoring

Scientific study of human eye movements began in the late 19th century, and employed a variety of measurement techniques. Many systems, such as that of Delabarre [1] were mechanical, while others like Dodge and Cline [2] used photography. Since their inception, eye monitoring systems have not been considered important innovations in-and-of themselves, but rather tools that allow scientists to study the gaze behaviour of experimental subjects. While much information concerning basic mechanisms can be gleaned from responses to the controlled laboratory studies, there is also much interest in behaviours occurring in the “real world”. Over time, technological advances have allowed eye monitoring systems to emerge from the laboratory, and today the use of eye-tracking in the study of the performance of everyday tasks is routine in a number of applied disciplines [3]. To name some of those disciplines, the eye-tracking is used in research concerning the reading process in order to understand the way the reader is scanning a text, the human computer interaction, in vehicle simulators and also in professional training (mainly athletes) as a mean to see through the eye of the user and analyse the eye movements pattern used in a task.

In the late 1940s, researchers used cameras to record the eye movements of pilots in the cockpit [4]. Eye-tracking systems such as these were refined in the following decades, and a host of techniques appeared, including electro-oculography (EOG) and magnetic search coil [5, 6]. But it was the advent of digital technology and image processing in the 1970s which marked the opening of a new era of video-based eye monitoring. Research continued in the 1970s under sponsorship from the U.S. Air Force to improve cockpit usability [4]. Today, many companies offer video -based eye monitoring systems at affordable prices.

Historically, eye monitoring systems were developed in support of physiological research on the oculomotor system. Eye monitoring systems could be classified into two categories: invasive and active vs. non-invasive and passive. In this latter category the experimental subject is often not aware of the presence of the eye monitoring system as no device is being actively attached to the physical body.

In an early study of fixational eye movements, Horace Barlow placed a drop of mercury in his eye, while an iron bar pressed his head firmly against a granites lab [6]. Fortunately, it is unnecessary to go to such heroic lengths today! Nevertheless, it is still difficult to make measurements having a precision comparable to the physiological noise level: subjects can generally maintain fixation to within a few minutes of arc, while a typical video-based system will produce errors as large as a degree unless special care is taken. To make ultra-precise measurements, a number of approaches have been developed over the years. For example, in the magnetic search coil system [5], a small loop of wire is placed in the eye; in humans this is usually done with a special contact lens, while for animal research the coils are usually surgically placed under the conjunctiva. The position and orientation of the coil (and hence the eye) is determined by measuring the currents induced by three, mutually orthogonal external magnetic fields. The search coil has the advantage of not requiring the head to be fixed, although it must remain in the volume enclosed by the field coils. It is fairly complex and expensive, but is unique in enabling very accurate determination of gaze while still allowing free head movement. But the placement of the coil in the eye is intrusive and can cause discomfort to the subject.

Another high-precision eye-tracker is the Dual Purkinje Image (DPI) tracker [7]. It uses fast optical servos to track the first and fourth Purkinje images (reflections of the illuminator from the refracting surfaces of the eye). The first Purkinje image is a virtual image formed by the front surface of the cornea, while the fourth Purkinje image is a real image formed by the (concave) rear surface of the crystalline lens. These two images fortuitously both fall more-or-less in the plane of focus of the pupil, and can sometimes be observed in video images of the eye, although good sharp focus is required to see the dim fourth image. The design of the DPI tracker is such that there is no relative motion between the two images if the eye translates without rotating, while the relative positions encode the rotational state. While the DPI tracker provides excellent performance in terms of sensitivity and temporal bandwidth, it requires stabilization of the head (usually with a dental impression or "bite-bar"), and is thus unsuitable for measurements in "natural" conditions.

The electro-oculogram (EOG) is a measurement made using electrodes attached to the skin around the eye region. The accumulation of electrical charges in the retina gives the eye a dipole moment, and motion of the eye causes the electrical potential to vary in the surrounding region. After calibration, readings of these voltages can be used to infer into eye gaze direction. Unfortunately, the dipole moment changes as the visual stimulation impinging on the retina changes, which limits the accuracy in practical situations. The EOG is simple to implement, and relatively low in cost, but the electrodes are somewhat intrusive, and require a bit of setup time.

The aforementioned methods are relatively invasive and can cause discomfort to the subject. Moreover, they provide indirect measures of what is falling on the retina, and are subject to mechanical artifacts. For instance, the DPI tracker generates spurious transients caused by "wobble" of the eye's lens following rapid saccadic movements, while users of search coil systems must worry about slippage of the coil relative to the eye. Retinal imaging, on the other hand, provides a direct measure of what is on the retina. In the scanning laser ophthalmoscope (SLO) [8], a laser beam provides illumination for imaging retinal structures, while simultaneously allowing modulation of the beam intensity to deliver patterned stimulation. Recent SLO designs have incorporated adaptive optics to correct the eye's aberrations [9], enabling the acquisition of diffraction-limited images of retinal structures, and images of the foveal cone mosaic. Using an adaptive-optics-enhanced SLO (AOSLO), it has been shown that the functional foveal "center" (defining the line-of-sight) has a slightly different position for steady fixation target versus smooth pursuit of a moving target [10].

Today, the majority of eye monitoring systems in general use are based on digital images of the front of the eye, captured with a remote video camera and coupled with image processing and machine vision hardware and software. Such systems are called passive eye monitors, or video-based eye-trackers. When these systems first appeared, they generally required special-purpose image-processing hardware to allow real-time measurement of gaze. Today, thanks to the steady increase of microprocessor power, it is possible to do a decent job entirely with software.

Passive eye monitoring technology became popular by being completely remote and non-intrusive, while offering reasonable accuracy at an affordable cost. It can be slower (sampling at about 30 to 60Hz in most of the times) and less accurate than invasive eye monitoring techniques, but provides a more natural experience for the subjects. In many applications, the available accuracy may be adequate to answer the questions of interest. This technology, for example, allows a disabled person to compose electronic messages, browse and navigate through web pages, turn on-off monitors, or call an assistant. Installed in a motor vehicle, a passive eye monitor can continuously evaluate the driver's fatigue and distraction [11, 12, 13], and generate appropriate vigilance warnings. Numerous other application areas have benefitted from the recent advances in the theory and practice of video based approaches: military, medicine, information security and retrieval, typing and reading, online search, marketing, augmented and virtual reality, and video games, to name a few.

2.2 Human Eye Physiology

For the human eye to work properly the light of the environment has to enter the visual system and be focused on the fovea. The human eyes act, most of the times, as motion detectors even though they can detect only still images by identifying objects and giving information to the brain in order to calculate the spatial relationships among different stimuli. Whenever we are fixating on a target our eyes are making small, very fast movements, called saccades, during which we are blind. It is the brain that processes the obtained information and so that we can recognize objects. The fact that we recognize things is the result of the learning process that exists as long as we live and use our eyes and is performed by the brain's neural organization [84].

2.2.1 Structure of the Human Eye

In Figure 2.1 we can see the basic structure of the eye, which is the organ responsible for our sight. Light reflected from an object enters the eye through the cornea, which is its first layer (cornea can be divided in more layers e.g. Bowman's, but a full anatomic description is not needed for the purposes of this thesis). Then the light enters through the pupil, in the centre of the iris, and passes through the crystalline lens, which is behind it. At this stage, right after the crystalline lens, the image, in the form of light waves, is reversed and turned upside-down, and in that form passes through the vitreous humor (a sort of gelatine that practically fills the eyeball). Finally, the light waves are focused on the retina at the back end of the eye.

At this point, the acquired signal travels to the brain thanks to a vast network of optic nerves. The retina itself is formed by photoreceptors which are divided in groups. Firstly, at the centre of the fovea there are those called cones (colour sensors). The cones are responsible for the colour vision and there are three kinds of them that are more sensitive in different colours (i.e. red, green, blue). The density of the cones is high centrally and gets lower at the periphery.

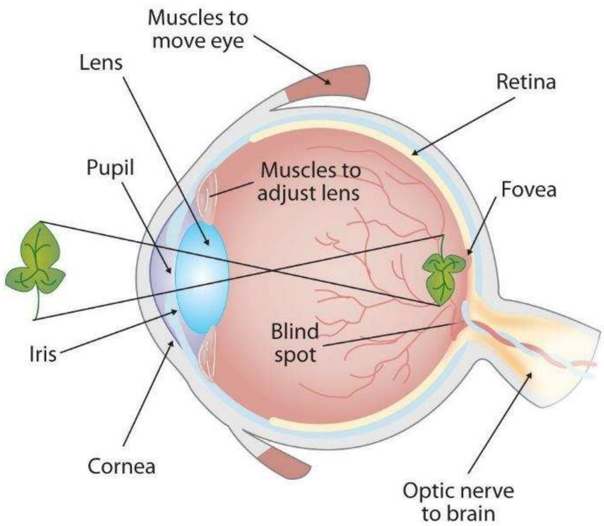


Figure 2. 1: The structure of the eye [78]

In the free space among cones there is the other kind of photoreceptors, called rods. Those cannot "detect" colour and offer grey, peripheral vision as well as the ability to see in mesopic or scotopic conditions (dim light). In general, the cones need high illumination environment to work properly offering maximum resolution but reduced sensitivity [78].

The number of the cones varies from six to seven millions, when in the same time the number of the rods is exceptionally bigger ranging from 110 to 130 million. The optical signal is transferred from the photoreceptors to the optic nerve, using fibres, at the optic chiasma and then to the brain [84]. Each half of the brain gets half an image, preventing blindness in case one of the eyes is lost. The two parts of the visual field are connected in the brain area where the visual regions of the two hemispheres are connected.

2.3 The Human Eye as a Digital Camera

In general, we can think the way the human eye is functioning as a digital camera that continuously takes images from the environment. Those images are sent via the optic nerve to the brain for further processing. The same way, an eye-tracking device captures the eye of the subject at all times and transmits the acquired images to a computer for further processing.

The digital cameras are divided in two groups; those using Charged Coupled Devices (CCD) and those using Complementary Metal Oxide Semiconductor (CMOS) sensors.

In the first case there is an $n \times m$ grid of photo diodes (photosensors) in a rectangular shape. Each one of those diodes is sensitive to light intensity, which is used as a measure to perceive the power emitted by a light source in a particular direction, converting light energy into a voltage. On the other hand the CMOS camera is similar to the CCD one; differentiating in that the photo diode is replaced by a CMOS sensor.

Each CMOS sensor is made by a number of transistors that are used for the electric signal amplification. In practice, CMOS sensors are noisier than CCD sensors, but they consume less power and are less expensive. Those photo diodes or sensors that detect the light and capture the images from the environment are much like the cones and rods that were mentioned above in the section about structure of the human eye.

The camera works like the human eye also when the light conditions are changing as the pupil automatically dilates or shrinks according to the light conditions. When there is enough light the camera aperture (controlling the amount of light that reaches the camera sensor) does not have to be large. On the contrary, if the light is not enough then the camera aperture has to be enlarged since the camera will need more light to form the image. The distance between the focal plane of the lens and the surface of the sensor grid gives us the focal length of a digital camera. The focal length value is required to select the magnification degree which is requested to the digital camera. Again, the human eye is functioning in a similar way but automatically. Depending on the stimulus we are fixating on the eye lens accommodates and focuses as needed to get the clearer possible image.

The elementary unit of a digital image is named pixel, which is an abbreviation of “picture element”. Different resolutions can be used to capture images by using a different amount of pixels. When we are talking about low resolution we usually have to do with an image using a matrix of 320 × 240pixels, whereas in medium resolution each image is generally represented by means of 640 × 480 pixels. At high resolution the image is represented by 1216 × 912 pixels. Even though that kind of resolutions are nowadays obsolete they are still in use in various eye-tracking applications. The image size in pixels, which corresponds to the size of the CCD or the CMOS grid, is called spatial resolution of the camera.

Finally, there are two other important parameters of the digital camera, the field of view and the sensor resolution. The field of view (or FOV) is the area of the scene that the digital camera can acquire. This is equal to the horizontal dimension of the region that includes all the objects of interest. The sensor resolution (SR) of a digital camera is given by:

$$\boldsymbol{SR} = \mathbf{2}\,\frac{\boldsymbol{FOV}}{\boldsymbol{OR}} \quad \mathbf{(1)}$$

where OR stands for the minimum object resolution (the dimension of the smallest object that can be seen by the camera).

[redacted]

Figure 2. 2: Digital Camera Signal Processing Flow[2]

2.4 Image Formats

There is a variety of file types used to store digital images based on particular standards called image file format standards. In the majority of cases the images are compressed as otherwise a lot of memory would be required. Those image file formats can be divided into two main categories: non-lossy image file formats and lossy image file formats. In the non-lossy image file formats the compression stage does not imply

[2] Image source: http://av.jpn.support.panasonic.com/support/global/cs/dsc/knowhow/knowhow30.html

an information loss. Hence after the decompression we obtain the original file before the compression. Respectively, in the lossy formats the compression stage implies an information loss.

2.4.1 Image File Format Standards

In this section the most popular image file formats are mentioned, with a brief description of them, with the exception of JPEG. This standard, as the most common, will be presented in more detail in Section 2.4.2. Those formats are the following: a) **Tagged Image File Format (TIFF)**, this format, its extension is .tif or .tiff, can be used to manage a variety of different types of images such as bitmaps and compressed colour images in an efficient way. TIFF is generally a non-lossy compression format (TIFF also provides lossy compression schemes, even though they are less popular), b) **Portable Network Graphics (PNG**), its extension is .png, is a format that provides lossless storage of raster images. c) **Graphics Interchange Format (GIF)**, supports 8-bit colour images and is generally used in application programs (e.g. word processors, Internet Explorer) in the Windows environment, d) **Postscript**, this format, developed in the UNIX environment, is used for printing. In this format grey-level images are represented by decimal or hexadecimal numerals written in the ASCII format and finally e) P**ortable Image File Formats** which are popular image file formats that include portable bitmap, portable greymap, portable pixmap and portable network map, whose file extensions are, accordingly, .pbm, .pgm, .ppm and .pnm. Those file formats are convenient format to store the images since they support a variety of images with increasing complexity, ranging from bitmaps to colour images.

2.4.2 JPEG Standard

JPEG, whose file extension is .jpg, is the acronym of “Joint Photographic Experts Group”. JPEG is the first international image compression standard for continuous-tone still images. This standard is the result of joint efforts by the International Telecommunication Union (ITU), the International Organization for Standardization (ISO) and the International Electrotechnical Commission (IEC) and its official name is: ISO/IEC IS 10918:1: Digital Compression and Coding of Continuous-tone Still Images. JPEG is very important since the widely used video standard MPEG is based on it. JPEG generally performs a lossy compression and it has four modes (sequential lossless, sequential DCT-based mode, progressive DCT-based mode, hierarchical mode) and several options. Here we will describe the JPEG basic coding algorithm (baseline JPEG algorithm) which is based on the Huffman coding for entropy encoding.

2.4.3 Baseline JPEG Algorithm

The baseline JPEG algorithm is formed by the following steps:

1. Colour space transformation: Firstly, the image is converted from RGB space into a space similar to the colour spaces used in NTSC and PAL systems. A matrix for each single component is built. Each matrix is formed by elements with range from 0 to 255.

2. Downsampling: The chrominance components are downsized. Each one of the matrices used is reduced by a factor of two in horizontal and vertical directions (JPEG offers the possibility of reducing by a factor of 2 only in the horizontal direction). The

element values are then centred around zero by subtracting 128 from each one of them. Finally each matrix is divided in blocks of 8×8 pixels.

3. Discrete cosine transform: The 8×8 blocks of each component are converted to the frequency space using a two-dimensional Discrete Cosine Transform (DCT). DCT output is an 8×8 matrix of DCT coefficients. Theoretically, DCT is non-lossy, but in practice there is some information loss because of the inevitable approximation errors.

4. Quantization: The physiological fact that the human eye cannot discriminate the exact magnitude of a high-frequency brightness variation is used in JPEG to reduce the amount of information in the high frequencies. This is performed in the Quantization step, where less important DCT coefficients, in general those that are related to high frequencies, are deleted. This lossy transformation is performed by dividing each DCT coefficient by a weight taken from a table (quantization table). If all weights are 1, then the transformation does not really have any effect, but if the weights increase quickly from the origin then the coefficients related to high frequency, are significantly downsized.

5. Average value reduction: In this step the value (0,0) (average value) of each block, which is given by the value at the top left corner, is reduced, by replacing it with the difference between actual average value and the average value of the previous block. This difference is generally small since the average values of the block do not differ a lot with each other. As a result, replacing each average value with its difference with the average value of the previous block implies that most of the new average values are very small. During average value reduction, the other DCT coefficients do not change.

6. Linearization: In this step the linearization of the block is performed. The block is linearized using a particular zigzag scheme. The zigzag scheme produces a high density of zeros at the end of the block making possible to code a relatively big area of

the block with a unique value (zero at this time). After the linearization process, the image can be represented as a unique list of numbers.

7. Huffman coding: Finally, the list of number is coded by using the above mentioned Huffman coding.

JPEG is very popular mainly because its compression rate is very high (most of the times no less than 20:1). The decoding procedure of a JPEG image requires performing the above-described algorithm backwards. In general, encoding and decoding a JPEG image requires the same computational resources [84].

2.5 Video Principles

Our ability to see, without understanding that in reality we are looking a sequence of discrete images, comes from the fact that each projected image is “held” for a few milliseconds before being deleted. For that illusion to work properly a minimum projection speed of 25 images per second is necessary. A good way to understand video is to consider the black-and-white television model [81]. In order to represent the 2D image, the camera scans fast from the left to the right and slower from up to down recording the light intensity on the screen, using a beam of electrons. Each completed scan constitutes a frame and after getting one the beam starts all-over again. To reproduce the image, the receivers repeat the scanning using the transmitted signal which is the intensity, in function of time. Nowadays, CCD video cameras are integrated, but still there are some CRTs (Cathode Ray Tube) screens that work that way. The exact parameters of the scanning procedure change according to the television standard in use each time. For example, the European standards, PAL (Phase

Alternative Line) and SECAM (SEquentiel Couleur Avec Memoire /Sequential Colour With Memory) have 625 scanning lines and a ratio between the horizontal and the vertical equal to 4, making 25 frames per second. On the other hand, the NTSC (National Television Standard Committee), which is used in the U.S.A., has 525 scanning lines, the same ratio between horizontal and the vertical dimension, which is 4, producing 30 frames per second. In the case of colour television, the very same scanning pattern is used but there are three synchronised electron beams- one for each of the primary colours (Red, Green Blue). After the scanning procedure, no matter which television standard is used, the three colour signals are transformed to one luminance and two chrominance signals (to obtain those each standard uses different transformations). As the human eye is more sensitive to luminance it is the luminance signal that has to be transmitted more accurately out of those three.

Concerning the digital video which is a sequence of frames; each of them is a digital image, whose basic element, as we have seen, is the pixel. In digital video colour, each one of the primary colours is represented by eight bits. As a consequence, more than sixteen millions of colours can be represented in the digital colour videos, whereas the human eye can detect about 17,000 of them.

In order to produce a uniform movement, digital video has to display at least 25 frames per second. In digital video, the rate between the horizontal and the vertical dimension is 4 and the digital screen resolution usually has 640×480 (or 800×600) pixels. High-definition television standards have different parameters though, the digital screen has 1280×720 pixels and the ratio between the horizontal and the vertical dimension is 16, like the European standard for Digital Video Broadcasting (DVB). It is obvious that the digital video, in order to be used in everyday computers and tasks has to be compressed. The most commonly used standard is the MPEG one.

2.6 MPEG Standard

Compression is a crucial topic for the storage of digital video. In this section, we briefly describe the MPEG (Motion Picture Experts Group) standard [82], paying particular attention to MPEG-2. The MPEG-1 (International Standard 11172) was designed for a video-recorder at 1.2 Mbps. MPEG-2 (International Standard 13118) was designed in a way to compress video signals from 4 to 6 Mbps in order to be used in NTSC and PAL television systems. Both MPEG-1 and MPEG-2 use spatial and temporal redundancies in the video. A spatial redundancy can be exploited coding separately each frame by means of JPEG. A further compression can be obtained by the fact that consecutive frames are most of the times very similar (temporal redundancy) [84]. The digital video system (DV), used in digital video cameras, codes each frame separately by means of JPEG. Since coding has to be performed in real time, it is a lot faster to code each frame separately. Nevertheless, when the two consecutive frames are not similar the above strategy fails. As a result, a method of motion compensation is needed to compress the information. In particular, MPEG-2 produces three different types of frame:

- I-frame (Intra-frame): still images coded using the JPEG methodology
- P-frame (Predictive frame): this corresponds to the difference between the actual frame and its predecessor.
- B-frame (Bidirectional frame): this corresponds to the differences between the actual frame, its predecessor and its successor.

I-frames are still images compressed the JPEG approach as it was described above. Those I-frames have to be produced periodically, mainly because MPEG can be used for the transmission of television signal. Given that viewers get connected to a channel randomly, if each frame was depending on the preceding one then anyone who has missed the first frame could never decode the succeeding ones. In addition, that repetition corrects any kind of problems caused in case the frame was originally received wrongly.

P-frames code the differences between two consecutive frames. They are based on the idea of macroblocks, (16×16 pixels in luminance and 8×8 pixels in the chrominance components). A macroblock is coded looking for in the preceding frame the very same macroblock or another macroblock which differs slightly from it.

The B-frames are similar to P-frames, with the difference that they code both the differences of the actual frame with the preceding and the succeeding frame.

2.7 Machine Learning

2.7.1 Introduction

In order to automate the eye detection and create a robust eye-tracker we can use machine learning. Thanks to machine learning the acquired video can be analysed by a computer that will detect the eye without needing any kind of involvement by the personnel in charge of the measurements.

Machine learning can be defined as the process of programming computers in order to get optimal results on a specific task by using taught experience or data that act as examples for certain cases [91]. In other words, machine learning is the effort made to study and model, usually using computers, the learning process itself.

In those cases we already have a working computer model but in order to ameliorate its effectiveness certain methods are used, to transfer past experience or training data, in an effort to optimise the parameters the model uses to make a decision. Those computer programs can be used either in order to extract important information from the used/input data (descriptive model) or in an effort to predict the results that will occur in the future (predictive model)). In some cases there are models that are both descriptive and predictive [91].

There are some main research areas around which machine learning has been developed:

• Task-oriented studies, which have to do with the development of learning systems to improve performance in a set of tasks which in already known.

• Cognitive simulation, which has to do with research and computer simulation of learning processes as they are used by humans.

• Theoretical analysis, which has to do with research investigating the potential use of learning algorithms and methods no matter what is the applicative domain.

Machine learning strives to make possible using already existing examples or already solved analogue problems as input to computers programs and therefore make the computers learn from them. In addition, another challenge that researchers are facing is to make the computers learn by observing and imitating certain facts and actions or based on the analysis of past mistakes. Another important aspect of machine learning is that since those machines are going to interact with people their concept and acquired skills must be understandable making the interaction with humans as friendly as possible. [84]

2.7.2 Classification of Machine Learning

The different learning types can be classified into four different categories according to the effort that is required by the teacher and learner: rote learning, learning from instruction, learning by analogy and learning from examples.

2.7.2.1 Rote Learning

In rote learning, the new knowledge is directly implanted in the learner that does not have to reach any deduction. The variants of this method include [84]:

• The machine learns by being programmed or modified by the user. In that case the learner's side has to do nothing at all except follow the predefined instructions (an obvious example here is the usual style of computer programming).

• The machine learns by memorizing given facts and data. In that case there is no deduction from the learner's side (that is practically how the very first database systems used to work).

2.7.2.2 Learning from Instruction

Learning from instruction consists, as its very name suggests, of acquiring knowledge from a teacher. The new information is integrated with prior knowledge for effective use. The learner is required to perform some deduction, but it is the teacher that has to present and offer the knowledge in a way that the learner can receive and absorb it. As learning from instruction is similar to education methods widely used, the system must be built accordingly, so it can store and use the acquired knowledge. In that case the task would be to build a system that can accept and store a set of instructions so it can apply this learned knowledge effectively when needed. An example of such case would be when a computer system learns the rules of a card game along with certain advice on how to win. D.J. Mostow described such a system in [92] back in 1983.

2.7.2.3 Learning by Analogy

Learning by analogy consists in using existing knowledge and transforming it in a form that can be used in the new situation. In such a case, the learner has to make more deductions, in order to retrieve the pre-acquired knowledge, transform it accordingly, apply it to the new situation and then store the new knowledge for potential future use [84]. Examples concerning such systems have been described by Bayoudh et al. at [94] and by J.G. Carbonell [93] who used as an example the case in which a person wanted to travel in another city but all the flights were booked. Even though that person had never travelled again with a train he knew it was possible and using in an analogous way the knowledge he had for plane travelling he managed to check the availability of a train ticket to his destination, make the payment, arrive at the station on time and get on board the correct train to his destination.

2.7.2.4 Learning from Examples

In this case the amount of deduction required by the learner is a lot bigger compared to the previous ones as the computer system has to induce a general concept description after analysing a set of examples of that concept. As a result, the main problem of the learning procedure, is to determine a general rule that explains the results exploiting the limited sample size of the examples that were used as input [84]. There are three general categories in which the learning techniques fit: supervised learning, reinforcement learning and unsupervised learning.

2.7.2.5 Supervised Machine Learning

In this category, the data is a sample of input-output patterns (recognition of handwritten letters and digits). The task here is to find a function that maps any input to an output that can foresee future input- output observations, minimizing the errors. According to the outputs type, supervised learning can be distinguished in classification and regression learning [91].

2.7.2.5.1 *Classification Learning*

In case the output has no structure, except whether two elements are equal or not, then we are talking about classification learning (each one of the elements of the output space is called a class). The majority of the pattern recognition tasks belong to that category and the algorithm that is used to solve a problem of that kind is called classifier [91]. A very well-known type of a classification problem is when an Optical Character

Recognition (OCR) software tries to recognize a character in an image format and recognises it as a specific letter of the alphabet [84].

2.7.2.5.2 Regression

In case the output is the values of a series of continuous variables, then we are talking about regression learning. A typical example of a regression problem would be to estimate the price of a used car using as input the car's characteristics such as the brand, the year it was manufactured, its mileage and other information that might affect its worth (e.g. performed services, accident record etc.) [91].

2.7.2.6 Reinforcement Learning

In certain occasions, the output of our system is not just one but a sequence of actions. As a consequence, the importance of a single action is minimal and what matters is to perform a sequence of correct actions in order to accomplish the original goal. In other words, an action is correct as long as it is part of a good set of actions. In such a case, the machine learning program evaluates the results of a set of actions and learns from past correct ones to create a general policy. Those learning methods are known as reinforcement learning algorithms [91].

A typical example of that case is a learning to play chess game. If the right piece is moved each time the reward will be to win the game. On the contrary, a bad decision about a piece movement will result in losing the game. The fact that the reward is delayed and does not occur right after the decision taken is also typical for reinforcement learning. As a consequence the algorithm must take decisions balancing between exploitation (to get the reward) and exploration (to acquire and absorb more knowledge) [84].

2.7.2.7 Machine Learning used in eye-tracking

In an effort to ameliorate the eye detection results of an eye-tracker the usage of Support Vector Machines (SVM) [105] has been introduced [106, 108, 109]. SVM is a two-class classification method of supervised learning that finds the optimal decision hyper-plane [107]. At the very beginning of their appearance, SVM models were used for the classification of linearly separable classes of objects. To give an example we can suppose that we have some objects that they belong to either one of two possible classes. Those classes would be class +1 and class -1. As it is obvious in Fig. 2.3 it is possible to discriminate the two classes and separate them.

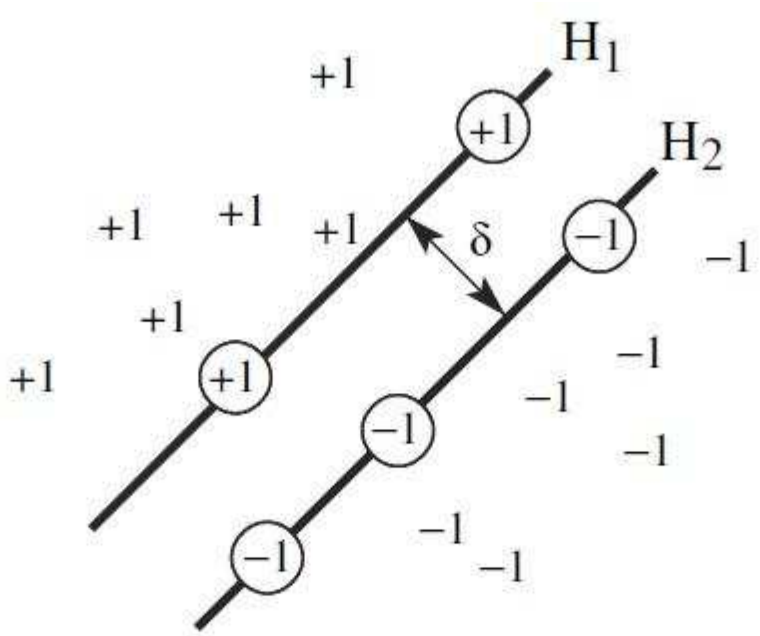


Figure 2. 3: Linear separable classes [107]

For each of the classes the SVM finds a hyperplane having the maximum margin (δ in Fig2.3). H1 is called the hyperplane defining the border of class +1, and H2 is the hyperplane H2 defining the border of class -1. The two circled objects on the borderline of class +1 and the three circled objects on the borderline of class -1 are called support vectors.

Nevertheless, there are cases where the classes cannot be separated with a linear classifier. In those cases, nonlinear functions (known as kernels) are used in order to map the objects into a feature space, where classes are linearly separable. The SVM kernels that are more widely used are the Gaussian kernels, the polynomial kernels and the Radial Based Function (RBF) kernels.

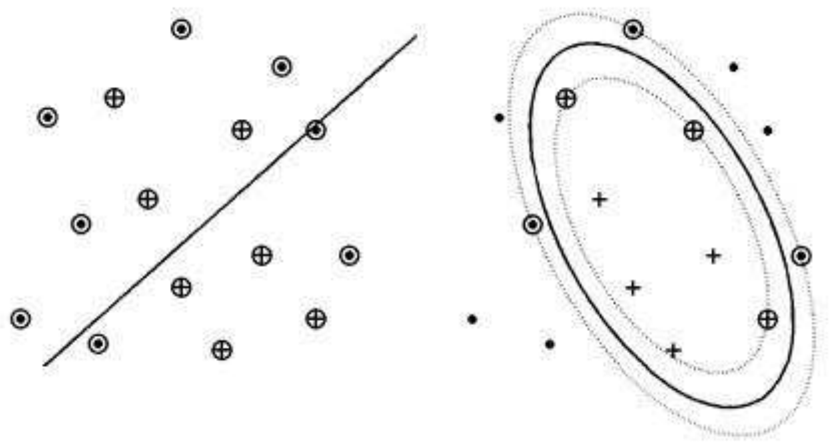


Figure 2. 4: Classification of a given set of data a) linear and b) polynomial (σ=2) [107]

Unfortunately, as there is no theoretical way to predict the best resulting kernel the identification of the best function to use comes after experimentation. In all cases, training data are needed for the SVM to be trained and result the optimal hyperplane.

In machine learning and eye-tracking the training data consists of images that are used as input to the SVM. This training set has two sub-sets of images, one with eye images (class +1) and one with "no-eye" images (class -1). The first set of images (class +1) has to be as generic as possible in order to train the SVM in recognising an eye in as many situations as possible. As a result, the eye images have to be from a variety of people with and without glasses and they have to be taken from different postures. After, finalising the training data the optimal parameters for the SVM have to be identified. This procedure can be rather time consuming since as it was mentioned before only with experimentation we can find the best set of parameters. For instance, Zhu and Ji in [106] experimented with three kernel types (linear, polynomial-of 3 different degrees- and Gaussian-of 4 different σ) and achieved the highest accuracy score when using a Gaussian kernel with a σ of 3. In another case though [108], Huang et al. had the best performance using polynomial kernels of 2^{nd} degree.

Furthermore, the SVM training is not completed in this stage. In most cases, to get a high level of accuracy thousands of data (in our case eye images) have to be used and then the SVM is tested on other data that are still unlabelled. Subsequently, the administrator has to evaluate the results of the SVM, find the mislabelled data, label them correctly and add them to their correct training set. Then the SVM is retrained, with the new set of data, from the very beginning. This whole procedure might be repeated several times until the accuracy level is as high and stable as it gets.

2.7.2.8 Unsupervised Learning

As it was shown above in supervised learning, the aim is to learn how to connect the input to the correct output given that an administrator (teacher) has provided the algorithm with the necessary data. On the contrary, in unsupervised learning, we only have input data. The goal is to detect similarities in the input and we want to see what generally occurs as a result and what does not [91]. Thus, whenever the input data consists of objects that are not associated with target values we have a case of unsupervised learning (there is no teacher). Typical example of an unsupervised learning task is the problem of image segmentation. For instance our input could be several images taken by a camera and the task is to be able to identify which of them were acquired indoors and which outdoors.

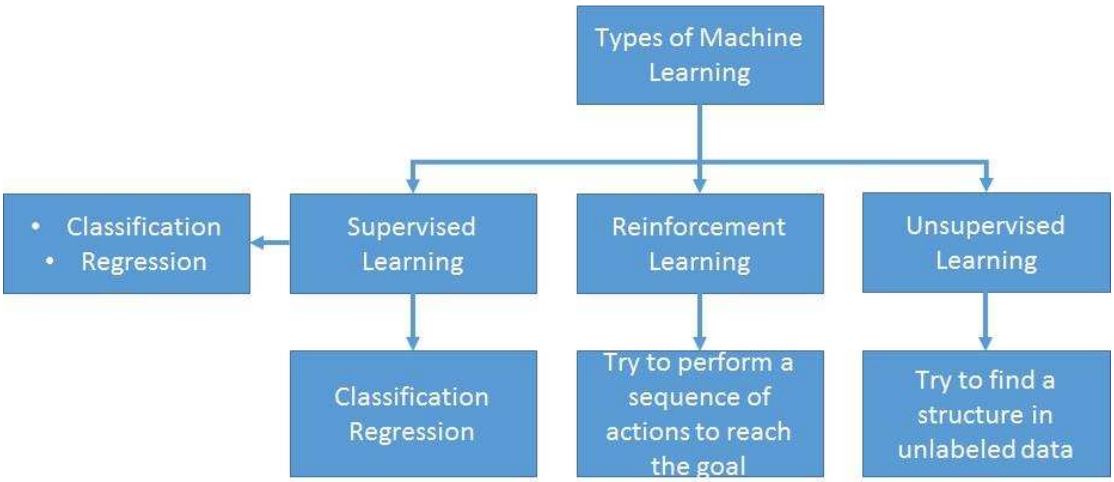


Figure 2. 5: Types of Machine Learning

2.8 Tracking and Eye-Tracking solutions

2.8.1 Tracking Overview

In this chapter the existing tracking solutions will be analyzed by emphasizing especially on the eye-tracking approaches used until now. One of the most challenging problems in computer vision is tracking an object through a video camera input. There is a variety of factors that can affect the difficulty in tracking such as the camera motion, obstacles that may appear in the scene changing the appearance pattern of an object, blunt-curt changes in the motion of the object etc. Tracking is usually used as part of other applications that require the exact location of a specific object all the time. [14]

Specifically, in eye-tracking applications what we are trying to detect the whole time is the eye(s) of a user and through that his gaze in order to know where exactly he sees its specific moment. Even though we shall see that this kind of technology is rapidly expanding in a variety of areas, at the moment is mainly in research on the visual system, in psychology, in cognitive linguistics and in product design [58].

2.8.2 Tracking an object

2.8.2.1 Introduction

As it was mentioned above trying to track an object is rather important and interesting in computer science. As the computers get better and better nowadays with constant increase of their analysis power and the optical means (video cameras) get cheaper and of higher quality, the need to develop new, more rapid and more robust object tracking algorithms became possible, in order to use them in automated video analyses of several types. The tracking procedure using a video input, can be described as an effort to anticipate the path that the object of interest is going to follow, as it keeps moving in the area of interest, and then redetect it. To do so, the tracker tries to recognize and label all the tracked objects in each frame of the video input. In general there are three key steps in order to perform a tracking procedure using a video input: a) detect the object that interests us, b) follow-track that object in each frame and c) analysis of the tracks of that object usually trying to recognize its behavior [14]. As a result, tracking an object is used in vital tasks such as those mentioned below:

Tasks	Rough Description
HCI (Human Computer Interaction)	Eye or gaze tracking and mostly use of that data as input info to computer software
Monitor the traffic	Collect information about the traffic flow so the responsible agency can act accordingly
Recognition based on motion	Identification of people by using unique facial or walking characteristics and patterns
Automated surveillance	Automatic detection of suspicious in sensitive areas that are monitored continuously

Table 2.1: Tasks in which tracking an object is required

The main reasons that cause problems in tracking an object are those:

Noise in the video signal
Hardware failure to process data in real-time
Quick and significant changes in the light of the scene (illumination)
Objects with abnormal or complex shapes
Non lineal or abnormal motion of the object of interest
Obstacles “hiding” part of the object we want to scan
Information loss as the object moves in a 3D environment and we are analyzing it using 2D images

Table 2.2: Reasons causing tracking procedure problems

To make things simpler we can add some sort of limitations in our systems concerning what we want to detect every time. Generally, knowing some of the characteristics of the object we want to track can really make things easier. Those characteristics can be the shape of the object, its size, if it is going to be one or more of them etc. Additionally, we can make assumptions, providing that we have a minimum of information about the movement of the object. In that case, we can assume for example that the object that interests us will have a smooth motion with a steady velocity.

2.8.2.2 Representing *an object*

The object of interest that we want to track depends every time on the task that we want to fulfill. That could, for instance, either the eye(s) of a user, people wandering in a room or walking on a road, vehicles on the roads of a city. The major characteristic used for the purpose of tracking an object is its shape. For the purposes of this thesis the most common shape representations are going to be briefly described.

Way of Shape Representation	Brief Description
By using a point (centroid) [110] **By using a number of points**	This kind of representation is working better when trying to track objects that represent a small portion of the image
By using geometric shapes [111]	In that case basic geometric shapes (circle, ellipse, rectangular, etc.) are used to represent the object of interest which is mostly of compact nature
By using the contour or the silhouette of the object [112]	This way of representation is more suitable in order to track objects of complex shape, whereas the contour practically is the line that defines the boundary of the object and silhouette is

	its “negative”-the area inside the contour
Object’s skeleton [113]	With this method firstly the object’s silhouette is found and then a medial axis transform is applied
Combination of basic geometric shapes [114]	Basic geometric shapes (circles, ellipses, etc.) are combined and group in order to track an object. Mainly used to track human bodies or other objects of articulated shape

Table 2.3: Ways of representing the shape of an object

2.8.2.3 Tracking by selecting a feature

When trying to track an object another critical step is to determine if that object has any kind of unique feature which can make its discrimination and thus the effort to track it easier. This kind of feature is affected by the way we represent the appearance of the objects in the application we are using. Therefore, in case the objects are represented by using histograms the color can be this unique feature [14]. On the other hand, the texture can be used to track an object. We take information about the texture of an object by quantifying properties like its regularity and its smoothness and thus measuring the intensity variation of a surface [115]. Also, another feature used is the edges of the object. Its boundaries can generate a unique shape which makes it easily detected by the application [116]. Finally, the last feature that can be used is the optical flow one. The optical flow is measured thanks to the brightness limitations as we can assume that the brightness of neighboring pixels is consistent in a frame sequence [117].

The truth is that at this moment lots of the tracking algorithms in use have the ability to combine more than one of those features, which at the time are mostly chosen in a manually way. However, more and more research and effort is dedicated to solve that problem and make the feature selection fully automatic [118].

Whichever method is going to be used to track an object it is going to need a sort of mechanism in order to detect it. A rather common solution is trying to detect an object in one frame; but to avoid false positive detections other methods take information from a sequence of frames. In that case, the algorithm tries to detect differences in consecutive frames [14].

2.9 Eye-tracking Technologies and Techniques

The most widely used current designs of eye-trackers are video-based. A camera focuses on one or both eyes and records the movement as the viewer looks at some kind of stimulus. Most modern eye-trackers use contrast to locate the center of the pupil and use infrared and near-infrared non-collimated light to create a corneal reflection (CR). The vector between these two features can be used to compute gaze intersection with a surface after a simple calibration for an individual [57].

Figure 2. 6: The pupil and the corneal reflection

Two general types of eye-tracking techniques are used: Bright Pupil and Dark Pupil. Their difference is based on the location of the illumination source with respect to the optics. If the illumination is coaxial with the optical path, then the eye acts as a retroreflector as the light reflects off the retina creating a bright pupil effect similar to red eye. If the illumination source is offset from the optical path, then the pupil appears dark because the retroreflection from the retina is directed away from the camera [57].

Bright Pupil tracking creates greater iris/pupil contrast allowing for more robust eye-tracking with all iris pigmentation and greatly reduces interference caused by eyelashes and other obscuring features. It also allows for tracking in lighting conditions ranging from total darkness to very bright. But bright pupil techniques are not effective for tracking outdoors as extraneous IR sources interfere with monitoring.

Eye-tracking setups vary greatly; some are head-mounted, some require the head to be stable (for example, with a chin rest), and some function remotely and automatically track the head during motion. Most use a sampling rate of at least 30 Hz. Although 50/60 Hz is most common, today many video-based eye-trackers run at 240, 360 or even 1000/1250 Hz, which is needed in order to capture the detail of the very rapid eye movements during reading, or during studies in neurology [52, 57].

Eye movements are typically divided into fixations (moments when the eyes are relatively stationary, taking in or “encoding” information) and saccades, when the eye gaze pauses in a certain position, and when it moves to another position, respectively. The resulting series of fixations and saccades is called a scanpath. Most information from the eye is made available during a fixation, but not during a saccade. The central one or two degrees of the visual angle (the fovea) provide the bulk of visual information; the input from larger eccentricities (the periphery) is less informative. Hence, the locations of fixations along a scanpath show what information located on the stimulus were processed during an eye-tracking session. On average, fixations last for around 200ms during the reading of linguistic text and 350ms during the viewing of a scene. Preparing a saccade towards a new goal takes around 200ms [57].

Scanpaths are useful for analyzing cognitive intent, interest, and salience. Other biological factors (some as simple as gender) may affect the scanpath as well. Eye-tracking in HCI typically investigates the scanpath for usability purposes, or as a method of input in gaze-contingent displays, also known as gaze-based interfaces [16].

Details on Eye-Movement Metrics

The main measurements used in eye-tracking research, as it was mentioned before, are fixations and saccades, which are quick eye movements occurring between fixations. There are also a multitude of derived metrics that stem from these basic measures, including gaze and scanpath measurements. Pupil size and blink rate are also studied.

Fixations: Fixations can be interpreted quite differently depending on the context. In an encoding task (e.g., browsing a web page), higher fixation frequency on a particular area can be indicative of greater interest in the target, such as a photograph in a news report, or it can be a sign that the target is complex in some way and more difficult to encode [17, 18]. However, these interpretations may be reversed in a search task: A higher number of single fixations, or clusters of fixations, are often an index of greater uncertainty in recognizing a target item [17]. The duration of a fixation is also linked to the processing-time applied to the object being fixated [18]. It is widely accepted that external representations associated with long fixations are not as meaningful to the user as those associated with short fixations [19]. Fixation-derived metrics are described in the following Table.

Eye-Movement Metric	What it Measures
Number of fixations overall	More overall fixations indicate less efficient search (perhaps due to sub-optimal layout of the interface).
Fixations per area of interest	More fixations on a particular area indicate that it is more noticeable, or more important, to the viewer than other areas.
Fixations per area of interest and adjusted for text length	If areas of interest are comprised of text only, the mean number of fixations per area of interest should be divided by the mean number of words in the text. This is necessary to separate out: (i) a higher fixation count simply because there are more words to read, from (ii) a higher fixation count because an item is actually harder to recognize.
Fixation duration	Longer fixation duration indicates difficulty in extracting information, or it means that the object is more engaging in some way.
Gaze (also referred to as “dwell, fixation cluster” and “fixation cycle”)	Gaze is usually the sum of all fixation durations within a prescribed area. It is best used to compare attention distributed between targets. It can also be used as a measure of anticipation in situation awareness if longer gazes fall on an area of interest before a possible event occurring.
Fixation spatial density	Fixations concentrated in a small area indicate focused and efficient searching. Evenly spread fixations reflect widespread and inefficient search.
Repeat fixations (also called “post target fixations”)	Higher numbers of fixations off-target after the target has been fixated indicate that it lacks meaningfulness or visibility.
Time to first fixation on-target	Faster times to first-fixation on an object or area mean that it has better attention-getting properties.
Percentage of participants fixating an area of interest	If a low proportion of participants is fixating an area that is important to the task, it may need to be highlighted or removed.
On-target (all target fixations)	Fixations on-target divided by total number of fixations. A lower ratio indicates lower search efficiency.

Table 2.4: Fixation-derived metrics and how they can be interpreted in the context of interface design and usability evaluation. References are given to examples of studies that have used each metric.

Saccades: No encoding takes place during saccades, so they cannot tell us anything about the complexity or salience of an object in the interface. However, regressive saccades (i.e., backtracking eye-movements) can act as a measure of processing difficulty during encoding [20]. Although most regressive saccades (or "regressions") are very small, only skipping back two or three letters in reading tasks, much larger phrase-length regressions can represent confusion in higher level processing of the text [20]. Regressions could equally be used as a measure of recognition value, in that there should be an inverse relationship between the number of regressions and the salience of the phrase. Saccade-derived metrics are described in Table 2.5.

Eye-Movement Metric	What it Measures
Number of saccades	More saccades indicate more searching.
Saccade amplitude	Larger saccades indicate more meaningful cues, as attention is drawn from a distance.
Regressive saccades (regressions)	Regressions indicate the presence of less meaningful cues.
Saccades revealing marked directional shifts	Any saccade larger than 90 degrees from the saccade that preceded it shows a rapid change in direction. This could mean that the user's goals have changed or the interface layout does not match the user's expectations.

Table 2.5: Saccade-derived metrics and how they can be interpreted in the context of interface design and usability evaluation. References are given to examples of studies that have used each metric.

Scanpaths: A scanpath describes a complete saccade-fixate-saccade sequence. In a search task, an optimal scan path is viewed as being a straight line to a desired target, with relatively short fixation duration at the target [19]. Scanpaths can be analyzed quantitatively with the derived measures described in Table 2.6.

Eye-Movement Metric	What it Measures
Scanpath duration	A longer-lasting scanpath indicates less efficient scanning.
Scanpath length	A longer scanpath indicates less efficient searching (perhaps due to a sub-optimal layout).

Spatial density	Smaller spatial density indicates more direct search.
Transition matrix	The transition matrix reveals search order in terms of transitions from one area to another. Scanpaths with an identical spatial density and convex hull area can have completely different transition values – one is efficient and direct whilst the other goes back and forth between areas, indicating uncertainty.
Scanpath regularity	Once "cyclic scanning behaviour" is defined, deviation from a "normal" scanpath can indicate search problems due to lack of user training or bad interface layout.
Spatial coverage calculated with convex hull area	Scanpath length plus convex hull area define scanning in a localised or larger area.
Scanpath direction	This can determine a participant's search strategy with menus, lists and other interface elements (e.g. top-down vs. bottom-up scanpaths). "Sweep" denotes a scanpath progressing in the same direction.
Saccade/fixation Ratio	This compares time spent searching (saccades) to time spent processing (fixating). A higher ratio indicates more processing or less searching.

Table 2.6: Scanpath-derived metrics and how they can be interpreted in the context of interface design and usability evaluation.

Blink rate and pupil size: Blink rate and pupil size can be used as an index of cognitive workload. A lower blink rate is assumed to indicate a higher workload, and a higher blink rate may indicate fatigue [21, 22]. Larger pupils may also indicate more cognitive effort [23, 24]. However, pupil size and blink rate can be determined by many other factors, such as ambient light levels, so are open to contamination [25]. For these reasons, pupil size and blink rate are less often used in eye-tracking research [26].

2.10 Eye-tracking vs. Gaze-tracking

Eye-trackers necessarily measure the rotation of the eye with respect to the measuring system. If the measuring system is head mounted, as with EOG, then eye-in-head angles are measured. If the measuring system is table mounted, as with scleral search coils, or table mounted camera ("remote") systems, then gaze angles are measured.

In many applications, the head position is fixed using a bite bar, a forehead support or something similar, so that eye position and gaze are the same. In other cases, the head is free to move, and head movements are measured with systems such as magnetic or video based head trackers.

For head-mounted trackers, head position and direction are added to eye-in-head direction to determine gaze direction. For table-mounted systems, such as search coils, head direction is subtracted from gaze direction to determine eye-in-head position. [16]

In [27] Yu and Eizenman, presented a new methodology to determine the point-of-gaze with a head-mounted eye-tracking system. It combined the well-known homography algorithm with distortion compensation, to determine the point-of-gaze from point correspondences in images obtained by the eye-tracker's scene camera. This methodology does not require either a separate head tracking system or accurate 3-D measurements of objects in the subject's field of view to determine the visual scanning behavior (i.e. viewing time and viewing frequency of each object). The point-of-gaze estimation methodology can be used to assess visual scanning patterns accurately (to less than 0.90º). As such, it can provide insights into selective attention processes that can aid in the diagnosis and evaluation of subjects with mood disorders. The reduced complexity of the methodology allows it to be used in applications that require portability, flexibility, and a changing visual scene [27].

2.11 Eye-tracking in practice

A great deal of research has gone into studies of the mechanisms and dynamics of eye rotation, but the goal of eye-tracking is most often to estimate gaze direction. Users may be interested in what features of an image draw the eye, for example. It is important to realize that the eye-tracker does not provide absolute gaze direction, but rather can only measure changes in gaze direction. In order to know precisely what a subject is looking at, some calibration procedure is required in which the subject looks at a point or series of points, while the eye-tracker records the value that corresponds to each gaze position. (Even those techniques that track features of the retina cannot provide exact gaze direction because there is no specific anatomical feature that marks the exact point where the visual axis meets the retina, if indeed there is such a single, stable point.) An accurate and reliable calibration is essential for obtaining valid and repeatable eye movement data, and this can be a significant challenge for non-verbal subjects or those who have unstable gaze.

Each method of eye-tracking has advantages and disadvantages, and the choice of an eye-tracking system depends on considerations of cost and application. There is a trade-off between cost and sensitivity, with the most sensitive systems costing many tens of thousands of dollars and requiring considerable expertise to operate properly. Advances in computer and video technology have led to the development of relatively low cost systems that are useful for many applications and fairly easy to use. Interpretation of the results still requires some level of expertise, however, because a misaligned or poorly calibrated system can produce wildly erroneous data [16].

2.12 Choosing an eye-tracker

One difficulty in evaluating an eye-tracking system is that the eye is never still, and it can be difficult to distinguish the tiny, but rapid and somewhat chaotic movements associated with fixation from noise sources in the eye-tracking mechanism itself. One useful evaluation technique is to record from the two eyes simultaneously and compare the vertical rotation records. The two eyes of a normal subject are very tightly coordinated and vertical gaze directions typically agree to within +/- 2 minutes of arc (RMS of vertical position difference) during steady fixation. A properly functioning and sensitive eye-tracking system will show this level of agreement between the two eyes, and any differences much larger than this can usually be attributed to measurement error.

2.13 Video based eye-tracking algorithms

2.13.1 Feature-based and model-based approaches

Eye-tracking algorithms can be classified into two approaches typically: the feature-based and the model-based. The first approaches detect and localize image features related to the position of the eye. A commonality among feature-based approaches is that a criterion (e.g., a threshold) is needed to decide when a feature is present or absent. The determination of an appropriate threshold is most of the times adjusted by the user as a free parameter. The tracked features vary widely across algorithms but most often rely on intensity levels or intensity gradients. For example, in infrared imagery and the dual-threshold technique, an appropriately set intensity threshold can be used to extract the region corresponding to the pupil. The pupil center is concerned to be the geometric center of the identified region. The intensity gradient can also be used to detect the pupil contour in infrared spectrum images [28, 29] or the limbus in visible spectrum images [30, 31]. Least-square fitting [30, 28, 32] or circular Hough transform [33] can then be used to fit an ellipse or a circle to these feature points. However, as the feature point detection may be affected by the eyelashes and the eyelids of the user, some additional process is needed to eliminate false feature points (which are called outliers). Pupil feature points are detected along radial vectors in [29], but a method of rejecting outlines is not given. Feature points are delimited in a quadrilateral formed by the eye corners, the uppermost point of the upper eyelid and the lowermost of the lower eyelid [30]. A double ellipse fitting approach is used in [28]. First, roughly detected feature points are used for ellipse fitting. And then feature points are detected again by using the center of first ellipse as starting point. Finally, an ellipse is fitted to the feature points that are close enough to the first ellipse. A curvature function is applied to eliminate the artifacts of pupil edge in [32]. However, these methods may not be robust enough to a relatively large number of outliers and may not be able to remove all of them [34, 35].

On the other hand, model-based approaches do not explicitly detect features but mostly find the best fitting model that is consistent with the image. For example, integro-differential operators can be used to find the best-fitting circle [36] or ellipse [37] for the limbus and pupil contour. This approach requires a continuous search of the model parameter space that maximizes the integral of the derivative along the contour of the circle or ellipse. The model-based approach can provide a more precise estimate of the pupil center than a feature-based approach given that a feature criteria is not in use to compute the image data. However, gradient techniques cannot be used without a good first guess for the model parameters. As a result, the gain in accuracy of a model-based approach is obtained at a significant cost in terms of the time needed for the calculations and the flexibility. However, the use of multi-scale image processing methods [38], in combination with a model-based approach, is very promising for real-time performance [39, 34, 35].

Robust non-intrusive eye detection and tracking is a crucial step of any eye-tracking system. It is fundamental for human computer interaction, attentive user interfaces and understanding human affective states. Boost cascade detectors typically use Haar-like features that calculate very fast using an intermediate image representation called “Integral Image”. Nevertheless, Haar-like features are less discriminative for describing information when the texture is of high frequency. Levi and Weiss [40] proposed a set of features based on local edge orientation histograms (EOH). The discrete Adaboost algorithm used a decision stump, i.e. a kind of threshold-type weak hypothesis with binary output but a lot of information is lost when using this approach. The histograms are used to approximate complex distributions of positive and negative training samples by dividing the feature space into many subregions with the same width. Then a weak classifier based on a look-up table function is built by calculating the log-ratio on each subregion. Although the histogram weak classifier seems to achieve a better performance, two issues remain. The first one concerns how many bins should be used in order to build the weak classifier. The second one is that regular histograms with the same bin width are good to describe data that are generally uniform. If the data distribution is not uniform at all and the details of the high density portion of the data should be captured, the number of bins should be bigger.

This means that a large amount of bins are lost when in the low-density region as they are not needed. It seems that the key is dividing the strong classifier into steps-stages. This approach decouples the detecting learning problem into several sublearning problems. The decision taken for each specific stage classifier still may not be the best concerning the overall performance of the cascade. Xiao et al. [41] proposed an algorithm called the "Boosting Chain" to combine the classification score in earlier cascade stages into the boosting learning in later stages. The Boosting Chain still requires many training parameters in each cascade stage. Tuning an optimal set of these training parameters is difficult and takes lots time. Cascade of Feed-Forward Classifiers improved and modified from the Real AdaBoost Algorithm that developed by Chen and Chen [42] might be applied. There are systems, such as the one presented in [40], fully automatic that operate in real-time at a high level of accuracy. Automatic face detection has to be performed when a new user appears in the image or when the tracking process has to be restarted because the user has been out of track for a certain period of time [40].

In a try to build an eye-tracking system with a robust pupil center estimation algorithm Long et al. [15] developed an algorithm named "Two-Step Processing Algorithm" in order to reach an even higher speed. This algorithm first tracks the approximate position of the pupil and then determines a very small image window enclosing the pupil for further processing. The image is down sampled at $\frac{1}{4} * \frac{1}{4}$ rate of the original resolution to facilitate the location of the pupil center. Then, the system specifies a tiny trace window slightly larger than the detected pupil area so that no pixel gets lost. The tiny image is processed at full pixel density resolution using the symmetric mass center algorithm which was also developed by Long et al. to locate the accurate position of the pupil center. The center of mass algorithm safely assumes that the pupil area can be approximated by a circle or an ellipse. This assumption will be wrong whenever the eye lid covers part of the pupil area and is going to lead to large measurement error. The symmetric mass center algorithm utilizes only the non-occluded, symmetric portion of the pupil area to estimate the pupil center. Suppose that an ellipse is partially occluded as shown in the Figure below, the mass center of the non-occluded area will not yield the correct ellipse center. From geometric properties of ellipses, it is known that if we find the maximum parallelogram $S_1E_1S_3E_3$, the mass centre of the area enclosed by $\widehat{S1E1}\widehat{S3E3}$. According to the authors their method offered an improvement of the processing rate to about 230% − 300% of the original one. [15]

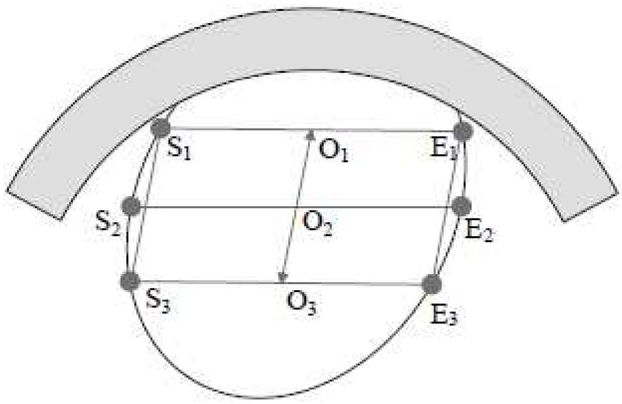


Figure 2. 7: Locating the centre for an ellipse that is partially occluded

2.14 Starburst Algorithm for Infrared Eye-tracking

As each company manufacturing eye-trackers uses customized algorithms to detect the eye, that are not presented publically for commercial reasons, we are going to see in more detail the open-source algorithm named Starburst. It is reasonable to assume that the general idea remains the same in each case and that the way those different algorithms function is more or less the same.

Starburst is a robust eye-tracking algorithm that combines feature-based and model-based approaches to achieve a good trade-off between run-time performance and accuracy for dark-pupil infrared imagery. The algorithm's goal is to locate the pupil center and the corneal reflection in order to relate the vector difference between these measures to specific coordinates in the scene image. To begin with, the algorithm locates and removes the corneal reflection from the image. At this point, the pupil edge points are located using a repetitious feature-based technique. The best possible ellipse is fit to a subset of the detected edge points using the Random Sample Consensus (RANSAC) algorithm [43]. The best fitting parameters from this feature-based approach are then used to begin a local model-based search for the ellipse parameters that maximizes the fit to the image data [34].

2.14.1 Noise Reduction

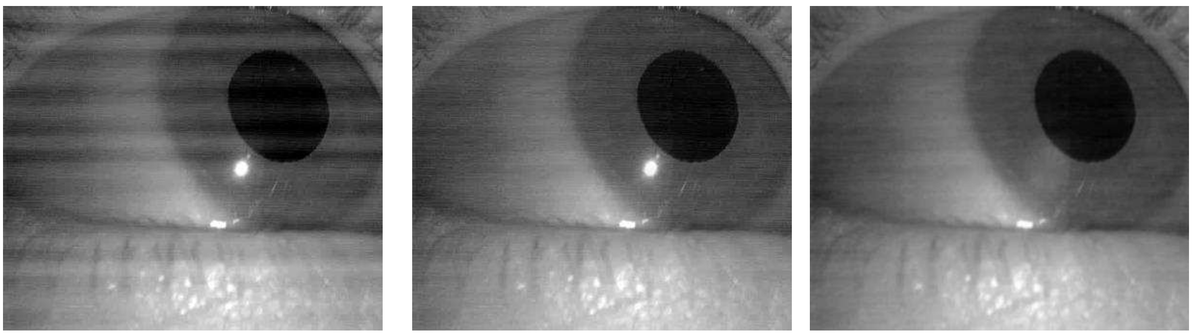
Due to the use of a low-cost head-mounted eye-tracker noise can cause significant problems. Thus, the algorithm reduces the noise present in the images. There are two types of noise, shot noise and line noise. To reduce the shot noise a 5 x 5 Gaussian filter is applied with a standard deviation of 2 pixels. The line noise is fictitious and a normalization factor can be applied in each line to shift the mean intensity of the line to the running average derived from previous frames. This factor C for each line l is:

$$\boldsymbol{C(i,l) = \beta\bar{I}(i,l) + (1-\beta)C(i-1,l)} \quad \textbf{(2)}$$

where $\boldsymbol{\bar{I}(i,l)}$ is the average line intensity of line l in frame i and $\beta = 0.2$. For i = 1, $\boldsymbol{C(i,l)} = \boldsymbol{\bar{I}(i,l)}$. This noise reduction technique is optional whenever the algorithm is used combined with an eye-tracker capable of capturing images with significantly less noise. The effect of the noise reduction can be seen in Figure 2.8 (compare (a) and (b)). [35]

2.14.2 Corneal reflection detection, localization and removal

In infrared spectrum, eye-tracking using the dark-pupil technique, the corneal reflection corresponds to one of the brightest regions in the eye image. As a result, the corneal reflection can be obtained by using thresholds. However, a steady threshold across observers and even within observers is not the best solution. Therefore, an adaptive thresholding technique is used in each frame in order to localize the corneal reflection. In addition, as the cornea roughly extends up to the limbus, the search for the corneal reflection can be limited to a square region of interest with a half-width of h = 150 pixels. To start with, the maximum threshold is used to create a binary image in which only the values above the threshold are taken as potential corneal reflection. It is likely that the largest candidate region is attributable to the corneal reflection, as any other specular



(a) (b) (c)

Figure 2. 8: (a) The original image. (b) The image with noise reduction. (c) The image with the corneal reflection removed after noise reduction. [35]

reflections tend to be quite small and located of the cornea (e.g., near the corner of the image where the eye lids meet). The ratio between the area of the largest candidate region and the average area of other regions is computed as the threshold is lowered.

Firstly, the ratio increases because the corneal reflection will grow in size faster than other areas. This growth happens as the intensity of the corneal reflection decreases towards its edges. Generally, a lower threshold will cause an increase in false candidates. The ratio will begin to drop as the false candidates become more prominent and the size of the corneal reflection region becomes large. The threshold that generates the highest ratio is taken as the optimal one. The location of the corneal reflection is then given by the geometric center (x_c, y_c) of the largest region in the image using the adaptively determined threshold.

While the approximate size of the corneal reflection can be derived using the thresholded region from the localization step, this region does not typically include the entire profile of the corneal reflection. To determine the full extent of the corneal reflection, there is the assumption that the intensity profile of the corneal reflection follows a symmetric bivariate Gaussian distribution. Thus, the radius r is calculated where the average decline in intensity is maximized and then related to the radius with maximum decline for a symmetric bivariate Gaussian (i.e. a radius of one standard deviation). As a result, the full extent of the corneal reflection can be taken as 2.5r to capture 98% of the corneal reflection profile. A Nelder-Mead Simplex search that minimizes is used to find r

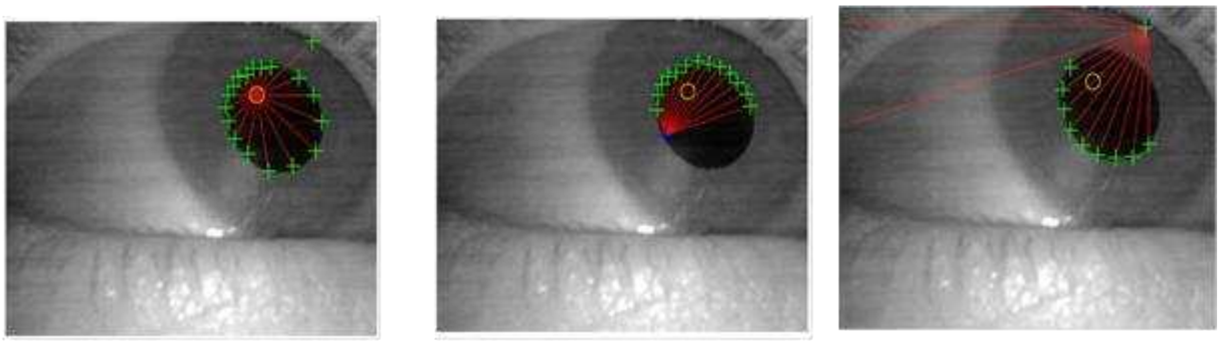
$$\boldsymbol{\int I\,(r+\delta,\, x_c,y_c,\theta)\, d\theta \,/ \int I\,(r-\delta,\, x_c,y_c,\theta)\, d\theta \quad (3)}$$

where $\delta = 1$, and $I(r, x, y, \theta)$ is the pixel intensity at angle θ on the contour of a circle defined by the parameters r, x and y. The search is initialized with $r=\sqrt{area/pi}$, where area is the number of pixels in the thresholded region. The search requires on average only 2.3 percent of the algorithm's runtime.

Radial interpolation is then used to remove the corneal reflection. Firstly, the central pixel of the identified corneal reflection region is set to the average of the intensities along the contour of the region. Then for each pixel between the center and the contour, the pixel intensity is determined via linear interpolation. An example of this process can be seen in Figure 2.9 (compare (b) and (c)) [34, 35].

2.14.3 Pupil Contour detection

For the starburst algorithm a novel feature-based method was developed in order to detect the pupil contour. While other feature-based approaches apply edge detection to the entire eye image or to a region of interest around the estimated pupil location, these approaches can be computationally wasteful as the pupil contour frequently occupies very little of the image and not all the pupil contour points are necessarily needed for accurate estimation of the pupil contour.



(a) (b) (c)

Figure 2. 9: Pupil feature detection. (a) Pupil contour edge candidates are detected along the length of a series of rays extending from a best guess of the pupil centre. Pupil contour candidates are marked using crosses. Note that two contour candidates are incorrect - one ray reaches the border and does not generate a candidate. (b) For each pupil contour candidate another set of a rays are generated that create a second set of pupil contour candidates (c) pupil contour candidates not on the pupil contour can lead to additional feature points not on the contour however these are typically not consistent with any single ellipse. [35]

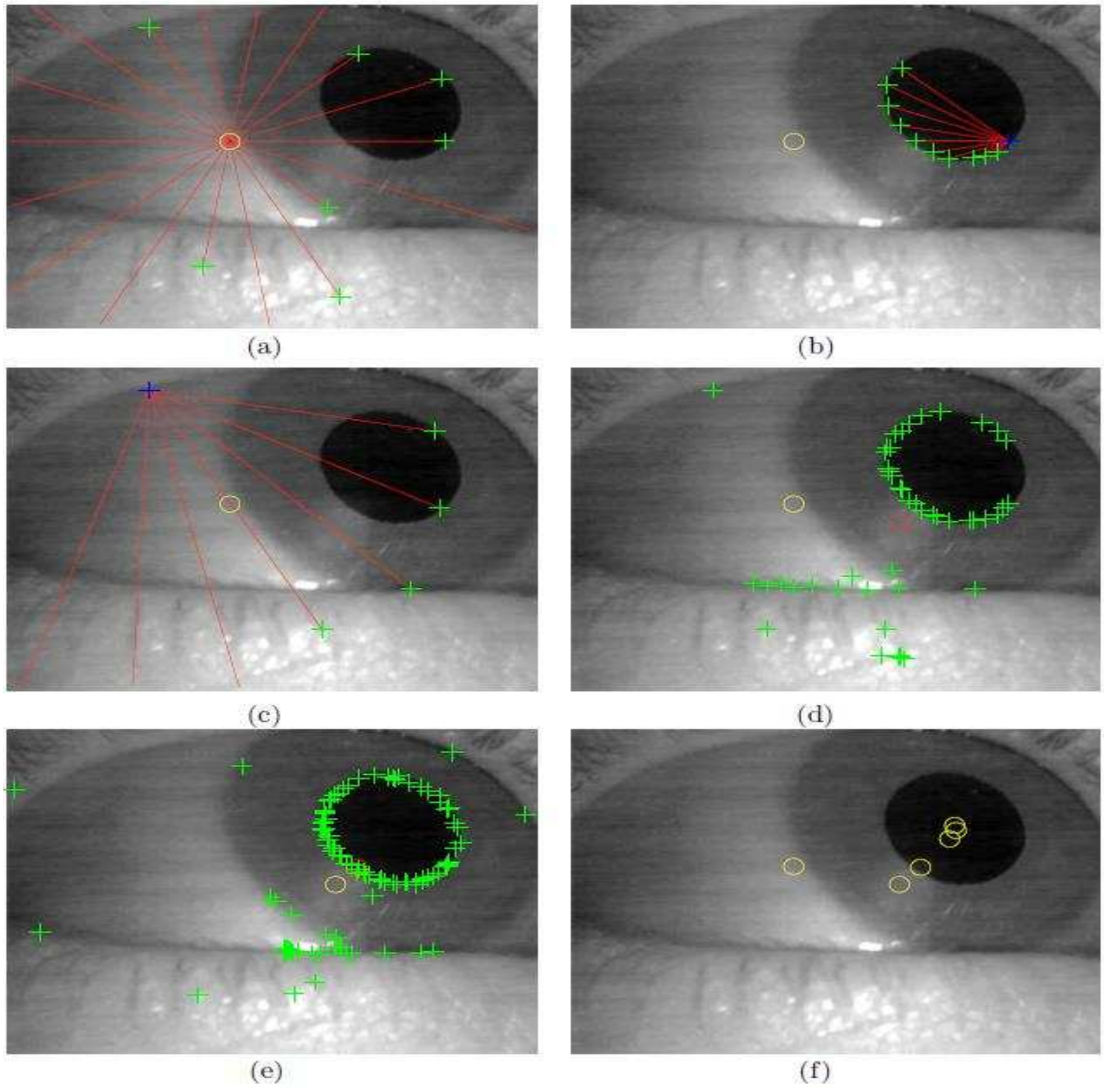


Figure 2. 10: Pupil feature detection. (a) The original starting point (yellow circle) shoots rays (blue) to generate candidate pupil points (green crosses). (b&c) The candidate pupil points shoot rays back towards the start point to detect more candidate pupil points. (d) All the candidate pupil points are shown. The average of these locations is shown as a red circle. This location seeds the next iteration. (e) The results of the second iteration. (f) The starting locations from all iterations show a rapid convergence. [34]

With this approach edges are detected along a small number of rays that extend from a central best guess of the pupil center. These rays can be seen in Figure 2.9 (a). For robustness to inaccuracy of the starting point, edges are also detected in a small number of rays extending from the initial set of detected features returning in the direction of the starting point. These returning rays can be seen in Figure 2.9 ((b) &(c)). This two-stage detection method takes advantage of the elliptical profile of the pupil contour to preferentially detect features on the pupil contour.

For each frame, a location is chosen that represents the best guess of the pupil center in the frame. For the first frame this can be determined either manually or taken as the center of the image. For subsequent frames, the location of the pupil center calculated from the previous frame is used. Next, the derivatives Δ along N rays, extending radially away from this starting point, are independently evaluated pixel by pixel until a threshold φ is exceeded. Given that the dark-pupil technique is used, only positive derivatives (increasing intensity as the ray extends) are considered. When the above-mentioned threshold is exceeded, a feature point is defined at that location and the processing along the ray is paused. If the ray extends to the border of the image, no feature point is defined. An example set of candidate feature points is shown in Figure 2.10 (a).

For each of the candidate feature points, the above described feature-detection process is repeated. However, rays are limited to $\gamma= \pm 50$ degrees around the ray that originally generated the feature point. The motivation for limiting the return rays in this way is that if the candidate feature point is indeed on the pupil contour (as shown in Figure 2.10 (b)), the returning rays will generate additional feature points on the opposite side of the pupil such that they are all consistent with a single ellipse (i.e., the pupil contour). However, if the candidate is not on the pupil (e.g., see Figure 2.10 (c)), this process will generate additional candidate feature points, that are not necessarily consistent with any single given ellipse. Thus, this procedure tends to increase ratio of the number of feature points on the pupil contour over the number of feature points not on the pupil contour. Given that feature points defined by a large Δ are more likely to be located on the pupil contour (as this is the strongest image contour), the number of returning rays is variable and set to 5Δ/φ. Note that the minimum number of rays is 5 because by definition a feature point is determined by Δ >= φ.

The two-stage feature-detection process improves the robustness of the method to poor initial guesses for the starting point. This is a problem when an eye movement is made as the eye can rapidly change positions from frame to frame. This is especially true for images obtained at low frame rates. For example, shown in Figure 2.10 (a) is such a case. While the initial set of rays only detects two feature points on the pupil contour, the return rays from these two points detect many more points on the contour (see Figure 2.10 (b)). The combined set of feature points is shown in Figure 2.10 (d) and the number of points on the contour well exceeds those off of the contour. However, the feature points are biased to the side of the pupil contour nearest the initialization point. Although another iteration of the ray process would minimize this bias, the computational burden grows exponentially, with each iteration, and thus that would be an inefficient strategy.

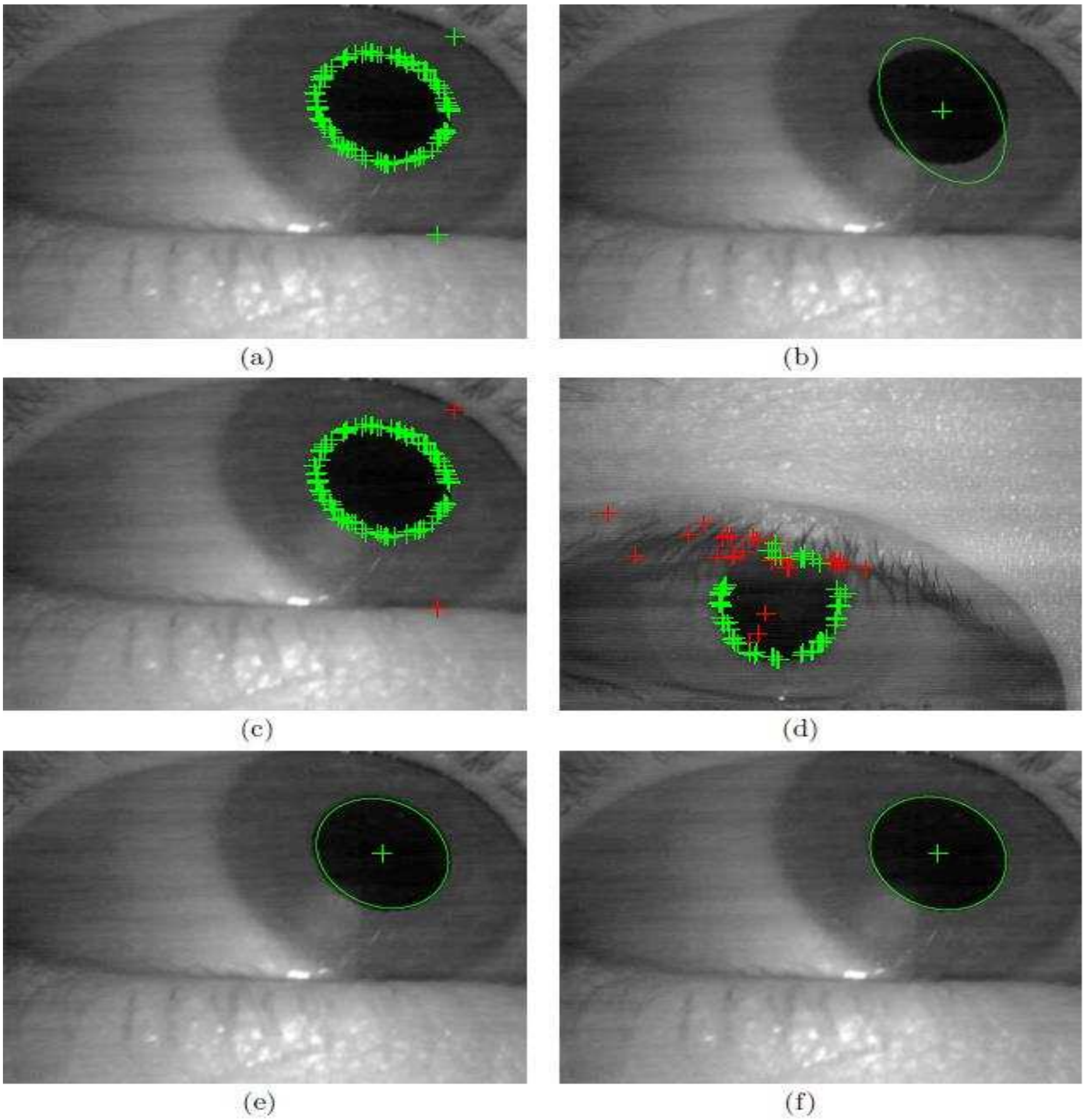
At this point, an ellipse could be fitted to the candidate points; however, the bias would induce a significant error into the fit. To eliminate this bias, the above described two-stage feature-detection process is iterated. For every iteration, after the first one, the average location of all the candidate feature points from the last iteration is taken as the next starting location. The red circle in Figure 2.10 (d) shows the starting point for the second iteration. The detected feature locations for the second iteration are shown in Figure 2.10 (e). Note the absence of a strong bias. Figure 2.10 (f) shows how the central locations rapidly converge to the actual pupil center. The iteration is halted when the center of the detected feature points changes less than d = 10 pixels (for this specific modality). When the initial guess is a good estimate of the pupil center, for example during eye fixations which occupy the majority of the frames, only one iteration is required. When the initial estimate is not good, typically only a few iterations are required for convergence.

The histogram of the iteration count is shown in Figure 2.12 (a) for the videos recorded. Note that 93% of the iteration counts are less than or equal to 5. If convergence is not reached within i = 10 iterations, as occurs sometimes during a blink when no pupil is visible, the algorithm halts and begins processing the next frame. On average, the feature-detection process requires 27 percent of the algorithm's runtime. [35]

2.14.4 Ellipse Fitting

Given a set of candidate feature points, the next step of the algorithm is to find the best fitting ellipse. While other algorithms commonly use least-squares fitting of an ellipse to all the feature points, gross errors made in the feature-detection stage can strongly influence the accuracy of the results. Consider the detected feature points shown in Figure 2.11 (a) and the resulting best-fit ellipse using the least-squares techniques shown in Figure 2.11 (b). Notice that a few feature points not on the pupil contour dramatically reduce the quality of the fit to an unacceptable level. To address this issue, the Random Sample Consensus (RANSAC) paradigm for model fitting is applied [43]. This case is the first application of RANSAC in the context of eye-tracking, however RANSAC is frequently applied to other computer-vision problems (e.g., see

[44]). RANSAC is an effective technique for model fitting in the presence of a large but unknown percentage of outliers in a measurement sample. An inlier is a sample in the data attributable to the mechanism being modeled whereas an outlier is a sample generated through error and is attributable to another mechanism not under consideration. Inliers are all of those detected feature points that correspond to the pupil contour and outliers are feature points that correspond to other contours, such as that between the eye lid and the eye. Least-squares methods use all available data to fit a model because it is assumed that all of the samples are inliers and that any error is attributable exclusively to measurement error. On the other hand, RANSAC admits the possibility of outliers and only uses a subset of the data to fit the model. In detail, RANSAC is an iterative procedure that selects many small but random subsets of the data, uses each subset to fit a model, and finds the model that has the most agreement with the data set as a whole. In most cases, the two stage feature-detection process results in very few outliers (e.g., see Figure 2.11 (c)) while in other cases, outliers are much more prevalent (e.g., see Figure 2.11 (d)). The distribution of outlier percentages for the tested videos is shown in Figure 2.12 (b).



2. 11: (a) Example set of feature points with only 2 outliers. (b) Poorly fit ellipse resulting from least-squares approach. (c) Inliers (green) and outliers (red) differentiated by RANSAC. (d) An example with more outliers. (e) Best-fitting ellipse using only inliers. (f) Best-fitting ellipse using model-based optimization. [34]

On average, 17 percent of the feature points are outliers. This relatively high amount of outliers is due to the fact that a low-cost eye-tracker was used, constructed from off-the-shelf parts, which introduces significant image noise into the videos. Given the presence of these outliers, it is important to use the RANSAC paradigm to find the ellipse that best fits the pupil contour.

The following procedure is repeated R times. First, five samples are randomly chosen from the detected feature set given that this is the minimum sample size required to determine all the parameters of an ellipse. Singular Value Decomposition (SVD) on the conic constraint matrix generated with normalized feature-point coordinates [45] is then used to find the parameters of the ellipse that perfectly fits these five points. The parameters of the ellipse must be real, the ellipse center must be inside of the image, and the major axis must be less than two times the minor axis. Otherwise, five more points are randomly chosen and a new ellipse fit, until these constraints are met. Then, the number of candidate feature points in the data set that agree with this model (i.e. the inliers) are counted. This set is called the consensus set. After the necessary number of iterations, an ellipse is fit to the largest consensus set (e.g., see Figure 2.11 (e)). Inliers are those sample points for which the algebraic distance to the ellipse is less than some threshold. This threshold is derived from a probabilistic model of the error expected based on the nature of our feature detector. It is assumed that the average error variance of the feature detector is approximately one pixel and that this error is distributed as a Gaussian with zero mean. Thus to obtain a 95% probability that a sample is correctly classified as an inlier, the threshold should be derived from a x^2 distribution with one degree of freedom [44]. This results in a threshold distance of 1:96 pixels. Because it is often computationally infeasible to evaluate all possible feature point combinations, the number of random subsets to try must be determined in a way that assures that at least one of the randomly selected subsets contains only inliers. This can be guaranteed with probability p = 0.99

$$\mathbf{R = \log(1\text{-}p)/\log(1\text{-}w^5)} \qquad \mathbf{(4)}$$

where w is the proportion of inliers in the sample. Although w is not known a priori, its lower bound is given by the size of the largest consensus set found. Thus R can initially be set very large and then set lower based on Equation 4 as the iteration proceeds. The number of necessary iterations can be further reduced each time that a new largest consensus set is detected, by iteratively re-estimating the model using all the members of the consensus set until the total number of inliers remains constant. The histogram of RANSAC iterations for the tested videos is shown in Figure 2.12 (c). Note that the median number of iterations is only 8 and the RANSAC model fitting on average utilizes 5.5 percent of the algorithm's runtime [34].

2.14.5 Model based Optimization

Although the accuracy of the RANSAC fit may be sufficient for many eye-tracking applications, the result of ellipse fitting can be improved through a model-based optimization that does not rely on feature detection. To find the parameters, the major and minor axis a and b, the center coordinate (x, y) and the orientation α of the best fitting ellipse, we minimize

$$- [((\int I (\alpha+\delta,b+\delta,a,x,y,\theta)d\theta)/ (\int (\alpha-\delta,b-\delta,a,x,y,\theta)d\theta))] \qquad (5)$$

using a Nelder-Mead Simplex search where $\delta = 1$ and I (a, b, α, x, y, θ) is the pixel intensity at angle θ on the contour of an ellipse defined by the parameters a, b, x, y and α. The search is initialized with the best-fitting ellipse parameters as determined by RANSAC. An example of model-based optimization can be seen in Figure 2.11 (f). The probability distribution of optimization iterations is shown in Figure 2.12 (d). The mean number of iterations is 74 and, on average, model-based optimization requires 17 percent of the algorithm's runtime [34, 35].

2.14.6 Calibration

In order to calculate the point of gaze in the scene image, a mapping must be constructed between eye-position coordinates and scene-image coordinates. Either the pupil center or the vector difference between the pupil center and the corneal reflection center can be used. The vector difference leads to

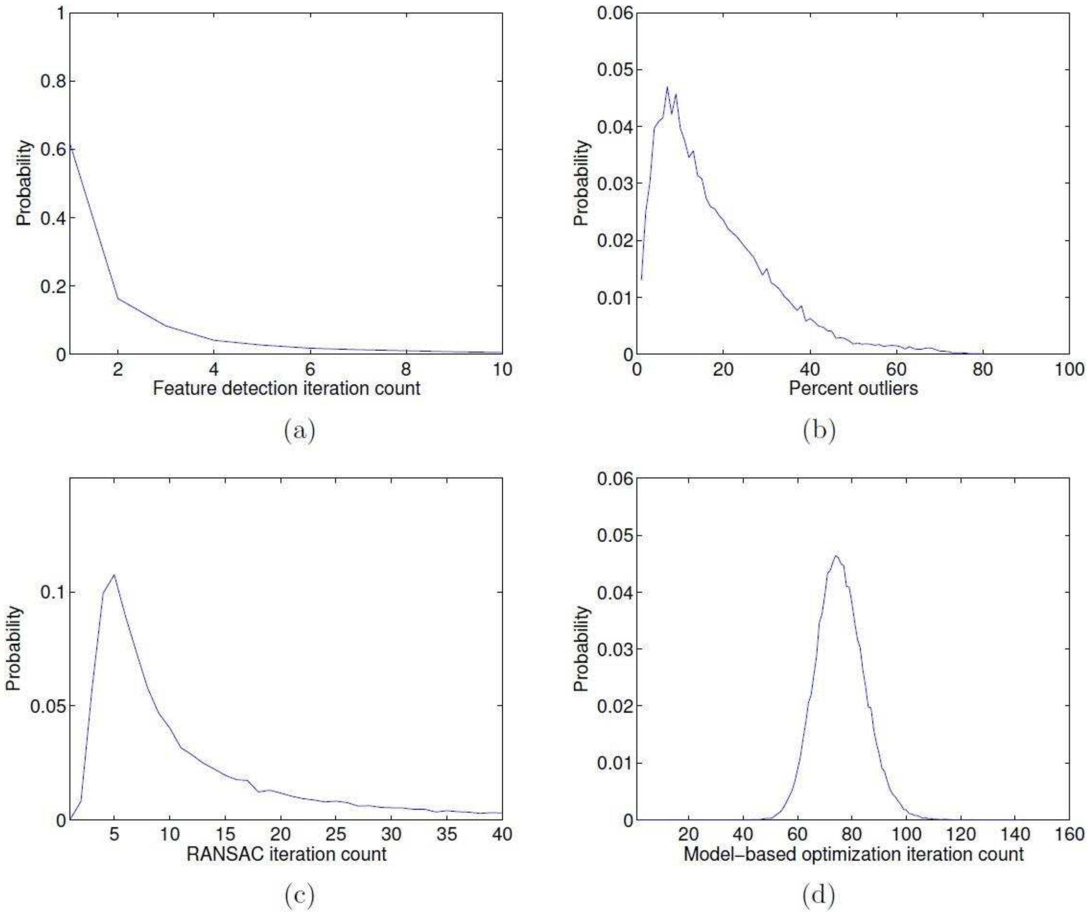


Figure 2. 12: (a) The histogram of iterations of pupil feature detection. (b) The percentage of outliers in processed videos. (c) The histogram of RANSAC iterations. (d) The histogram of iterations of model-based optimization. [34]

superior performance because it reduces sensitivity to the slippage of the headgear. The mapping can be initialized by relating known eye positions to known scene locations. The typical procedure in eye-tracking methodology is to measure this relationship through a calibration procedure. During calibration, the user is required to look at a 3x3 grid of scene points for which the positions in the scene image are known. While the user is fixating each scene point: $\overrightarrow{si} = (x_{ei}, y_{ei})$, the eye position: $\overrightarrow{ei} = (x_{ei}, y_{ei})$ is measured.

The particular mapping used by different eye-tracker manufacturers and different research group varies widely. The first mapping that was examined was a first-order linear mapping. For each correspondence between $\overrightarrow{si}$ and $\overrightarrow{ei}$, two equations are generated that constrain the mapping:

$$\boldsymbol{x_{si} = \alpha_{x0} + \alpha_{x1} x_{ei} + a_{x2} y_{ei}} \quad \textbf{(6)}$$

$$\boldsymbol{y_{si} = \alpha_{y0} + \alpha_{y1} x_{ei} + a_{y2} y_{ei}} \quad \textbf{(7)}$$

where α_{xi} and α_{yi} are undetermined coefficients of the linear mapping. This linear formulation results in six coefficients that need to be determined. Given the nine point correspondences from the calibration and the resulting 18 constraint equations, the coefficients can be solved for in the least-squares sense using SVD. Nonlinear mappings were also considered using this framework including second-order and third-order polynomial mappings. The second-order mapping included all six additional higher order terms.

Another non-linear method that was considered was to use a homographic mapping. In that case the mapping H is generated, a 3x3 matrix that has eight degrees of freedom, between the scene point $\vec{s} = (x_s, y_s, 1)$, and the pupil-CR vector $\vec{e} = (x_e, y_e, 1)$. To determine the entries of H, a constraint matrix is generated using measured point correspondences. Each correspondence generates two constraints and thus four correspondences are sufficient to solve for H up to scale. Finally the null space of the constraint matrix can be determined through SVD to provide H. SVD produces the mapping H that minimizes the algebraic error. Once the mapping is determined, the user's point of gaze in the scene for any frame can be established as $\vec{s} = H\,\vec{e}$.

Calibration Method	Error (degrees)
Linear	0.77
2nd-order polynomial	0.57
3rd-order polynomial	0.64
Homographic	0.58

Table 2.7: Accuracy of the different calibration methods. [34]

The average errors obtained are shown in Table 2.7. All mappings provide reasonable accuracy. However, the second-order mapping and homographic mappings result in the best performance. The lack of cross-terms hurts the third-order mapping. However, it is expected that the third-order mapping would result in accuracy comparable to the second-order mapping, given that sufficient correspondences will be available to include the cross-terms. Overall, the choice of mapping makes little difference but that of a non-linear model should be preferred. However, a more comprehensive investigation that examines the ability of these mappings to extrapolate outside of the nine-point calibration grid would be valuable [34, 35].

2.15 Image-Video Compression in eye-tracking

In our days the wide spread of mobile and portable devices, wireless sensor network technologies as well as cutting-edge biomedical microsystems (e.g. camera micro-pill), require imaging front-end that acquire the image, process, and transmit data using very low power.

In general, in a video communication application, a pair of encoder and decoder is required. The image encoder converts, at each time step $\mathbf{n}$, a sampled version $\mathbf{u_n}$ of the pixel value into a digital form $\mathbf{J_n}$. The codeword is then transmitted over the channel $\mathbf{C}$ to the decoder, which reconstructs the pixel value $\hat{\boldsymbol{u}}_{\mathbf{n}}$, as close as possible to the original image source. The most efficient way to handle nonstationary signals, such as pixel intensity, is to continuously adapt the encoder/decoder pair. In backward adaptation, the transmitted codeword is used to adjust the encoder parameters. [46]

For this purpose, and because of the very rapid emergence of CMOS imaging technology which is the technology of choice for portable digital imaging products, Shoushun et al. approached this field by proposing an adaptive quantization scheme based on a boundary adaptation procedure followed by an efficient online quadrant tree decomposition algorithm in order to achieve low-power and yet robust image compression integrated together with a digital CMOS image sensor. In that paper they reported the theory, simulation, VLSI design, and experimental measurements of a single-chip CMOS image sensor and a compression processor. The normalized power and the silicon area of their on-chip compression processor were compared with a number of image compression on-chip solutions. They showed that, while their processor is very compact (less than 55 k transistors implementing all modes of operation), it also features a much lower power consumption when compared with other processors realized in similar technological processes [46].

The modern CMOS technology was a matter of research also by Zamora et al. in [76]. In their study they presented new approaches and coordinated policies, in the area of Power Management, for wireless video sensor networks, they discussed techniques for analytically predicting their performance, and ran extensive simulations using each PM policy to obtain power-savings estimates in an effort to make those sensors less power consuming. Thus, it seems that CMOS technology might have lots to offer in the future eye-tracking.

Unfortunately, in eye-tracking devices it is not that easy to use Multi-View Tracking techniques in wireless sensor networks the same way they are used in a variety of occasions (e.g. in traffic surveillance, aircraft control and guide, security surveillance and so on) [47] as we can't afford to have a tracker with lots of nodes tracking the eye of the subject. In that case the wearer of the device would feel uncomfortably and in addition the power needed for more than one node to work might become a problem.

2.16 Building Blocks

The minimal hardware requirements for an eye monitoring system are a digital camera and a personal computer with a suitable interface. The feasibility of creating a low-cost eye monitoring device for personal use has been examined in [4]. Their study shows that the hardware components of their low cost system can be assembled from readily available consumer electronics and off-the-shelf parts for fewer than 30 dollars with an existing personal computer. On an Apple Power Book laptop, their proof of concept eye monitoring device operates in near real-time using prototype Jitter software at over 9 frames per second, and within 1 degree of gaze error.

Figure 2.13 illustrates a block diagram of a typical passive eye monitoring system including hardware and software. This system includes a video imaging camera and an infrared light illuminator, coupled to a vision processor which, in turn, is coupled to a host processor. The video imaging camera may include a CCD/CMOS active-pixel digital image sensor mounted as a chip onto a circuit board. One example of a CMOS active-pixel digital image sensor is Model No PB-0330, commercially available, which has a resolution of 640H × 480V. Other imaging cameras, like webcams, may be employed depending on the application field. To achieve robustness to environmental variations in the ambient illumination, it is common to image the eye in the near-infrared (NIR), with a filter on the camera to block visible light, and controlled NIR illumination provided by light emitting diodes (LEDs). In many situations useful images may be obtained with available natural light, but the use of controlled NIR illumination assures consistent lighting conditions across a range of environmental conditions. Lighting conditions are one of the most important problems to be addressed when the eye monitoring system is employed in real operating scenarios like driving a vehicle. In order to minimize the interference from light sources beyond the IR light emitted by the LEDs, a narrow band pass filter centered at the LED wavelength could be attached between the CCD camera and the lens [48]. Bergasa et al [48] reported that when such a filter is employed in their real-time video-based pupil tracking system for monitoring driver vigilance, the problem of artificial lights and vehicle light has been solved almost completely. Further, they made two important points: (1) this filter added a new drawback for it reduces the intensity of the image and the noise is considerably amplified by the automatic gain controller (AGC) integrated in the camera, and (2) this filter does not eliminate the sunlight interference in real-driving scenarios, except for cases when the light intensity is very low.

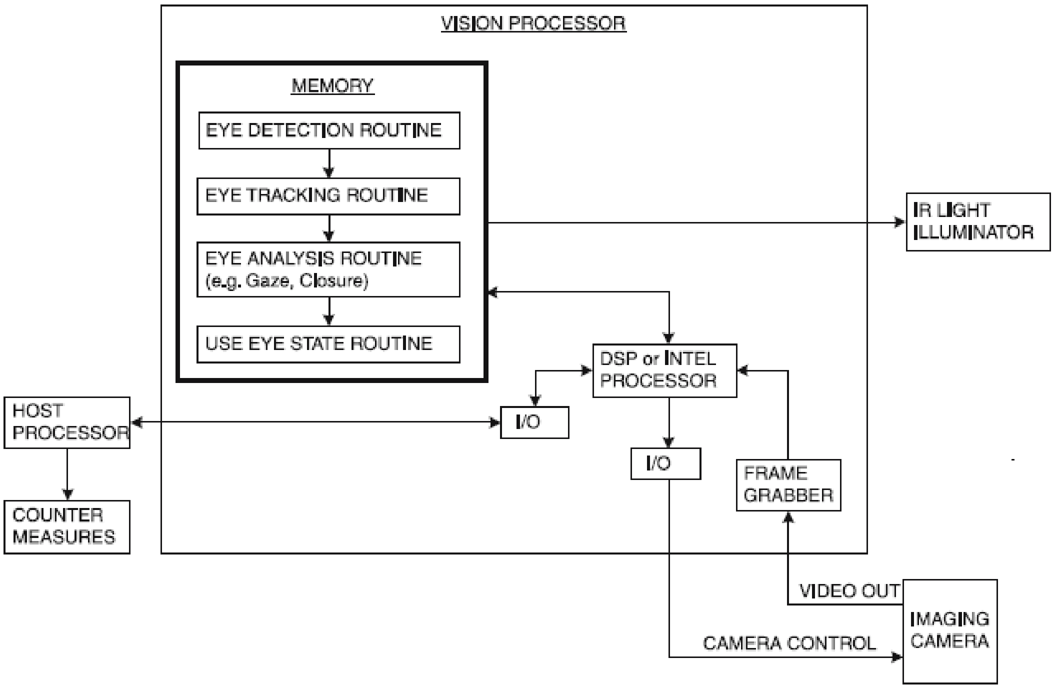


Figure 2. 13: Block diagram illustrating a passive eye monitoring system [36]

Once an image has been acquired by the camera, it is generally passed to a processor using a frame grabber or digital interface. In some systems, a "smart camera" may be employed which performs significant computations onboard using dedicated hardware such as a fully-programmable gate array (FPGA). Regardless of the specific implementation, the vision processor is generally responsible for controlling activation of the IR light illuminator(s), controlling the camera, and processing the acquired video images. Control of the video imaging camera may include automatic adjustment of the pointing orientation, focus, exposure, and magnification. Each video frame image is processed to detect and track the spatial location of one or both eyes of the subject (eye finding and eye-tracking routines). The detected eye(s) may be analyzed (eye analysis routine) to determine eye gaze vector(s) and eye closure state of one or both eye(s). After determining the gaze direction and/or the eye closure state, the eye monitoring system can then detect inattention and fatigue. The host processor may also interface to control devices that employ the determined eye information. For example, eye closure state maybe used in a drowsy driver application to initiate a countermeasure such as an audible warning.

The robustness and accuracy of the software building blocks is crucial in practice. False eye position detection or mistracking of the pupil will produce large errors in gaze vector estimation and noisy eye monitoring data. Thus, much effort has been expended in the computer vision community to develop effective low-level building blocks, including automatic detection of human eyes, modeling of the eyes' structure, appearance changes across subjects and lighting conditions, tracking of spatial eye coordinates, identifying pupil, iris, eyelids, and eye-corners locations, estimating, calibrating and tracking a person's gaze direction. In [49, parts I and II] Dr. Hammoud describes thoroughly the existing methods and analyzes new, very promising approaches, to overrun problems as: eye region modeling, the automatic eye position detection and spatial eye position tracking (Part I –Low level building blocks) and others like: gaze calibration, gaze & eye pose tracking and eye images interpretation (Part II- Mid level building blocks).

The accuracy of gaze tracking, pupillometry, or other measures, will generally scale linearly with the number of pixels per eye. Thus we are faced with a tradeoff: high accuracy demands high magnification (narrow field of view), while the ability to track moderate head movements requires a wide field of view. One approach is to satisfy both desires by having two or more cameras: a wide-field camera finds the face and the eyes within it, while a steerable, narrow-field camera provides a magnified image of the eye. Moderately-priced pan/tilt/zoom cameras are attractive, but do not move fast enough to follow rapid head movements. Mirror galvanometers allow redirection of the narrow-field camera's line-of-sight during vertical blanking, but add significantly to system cost. Thanks to the introduction of CMOS camera chips supporting area-of-interest readout, these mechanical solutions can now be superseded by a single high-resolution sensor with a wide angle view of the working volume. In this case, the entire image is searched to find the eyes, while the programmable area-of-interest provides a "digital pan".

In a pupil-imaging system, accuracy can be increased by increasing the optical magnification, up to the point at which the pupil fills the entire image – or slightly less than the entire image, to allow some range of movement. Under these conditions, a typical eye with a 7mm pupil, imaged by a 640 ×480 sensor, will produce 1 pixel of image motion for a rotational movement of around 5–10 minutes of arc. To obtain a higher optical gain, it is necessary to image a structure smaller than the pupil. Retinal imaging offers the possibility of imaging extremely small structures; even a crude imaging setup is capable of achieving an optical gain of 1 pixel of image motion per arc minute [50]. The AOSLO described above produces 1 pixel of image motion for a few arc seconds. It has also been suggested that blood vessels in the sclera might be imaged at high magnification for high-precision gaze tracking; while this approach deserves to be investigated, potential problems include shallow depth-of-field at high magnification (necessitating dynamic focusing), and the fact that the blood vessels are not rigidly embedded in the sclera, but are supported above it by a thin clear membrane known as the conjunctiva, and so may themselves move with respect to the eye as the eye moves.

The capability to actively control multiple illuminators can enhance the performance of an eye monitoring system. The pupil finding task can be simplified by exploiting the fact that the eye is a natural retroreflector. This approach is well known by dark-bright pupil technique. Light entering the eye is reflected by the retina, and passes back out towards the illuminator. When the eye is viewed from the direction of the illuminator the pupil appears filled with (red) light. This is the origin of the "red-eye" effect in flash photographs, which can be eliminated by moving the flash away from the camera lens. The pupil reflex can be isolated by subtracting an image collected with off-axis illumination from one collected with on-axis illumination [51, 52]. The two images may be collected using a single camera and temporal multiplexing, or multiple cameras using wavelength multiplexing. In an interesting twist on the idea of wavelength multiplexing, a sensor chip has been demonstrated which incorporates a checker board NIR filter array, providing a single-camera solution with wavelength multiplexing [53]. Wavelength multiplexing is generally superior to temporal multiplexing, which suffers from motion and interlace artifacts.

The subtraction of dark-pupil image from bright-pupil image results into a very short list of potential eye/pupil candidates which are further filtered using machine learning techniques. Besides the pupil detection algorithm, the system includes pupil tracking and glint localization in order to estimate the gaze vector. While clean images of the eye, generated by such an imaging system, are relatively easy to analyze, it is nevertheless difficult to build a passive eye monitoring system that works reliably with all members of a large population in all operating scenarios. Eyewear such as prescription glasses and sunglasses are particularly common challenges, introducing clutter around the eye and occlusion of some key eye features. Changes in magnification of the eye caused by the power of the eyeglass lens can generally be calibrated out, but bright specula highlights from metal frames can confound simple searches for the glint which are based on finding pixels whose values exceed a fixed threshold. Similarly, threshold-based approaches to finding the (dark) pupil can be foiled by mascara and cosmetic products applied to the eyelashes. These types of problems arise indoors under the best of conditions; outdoors, the problems are multiplied. For example, many people wear sunglasses while driving which make the eyes nearly invisible; while sunglasses are generally designed to block visible and ultraviolet(UV) light, little can be safely assumed about their NIR transmission. In [48] it is been reported that when the sunlight or the lights of moving vehicles directly illuminate the driver's face, an increase of the pixel levels is noticed, causing the pupil effect to disappear.

Chapter 3

3.1 Eye-tracking Applications

A wide variety of disciplines use eye-tracking techniques, including cognitive science, psychology (notably psycholinguistics, the visual world paradigm), human-computer interaction (HCI), marketing research and medical research (neurological diagnosis). Specific applications include the tracking eye movement in language reading, music reading, the perception of advertising, and the playing of sport. [54] Uses include:

- Cognitive Studies
- Medical Research
- Human Factors
- Computer Usability
- Translation Process Research
- Vehicle Simulators
- In-vehicle Research
- Training Simulators
- Virtual Reality
- Adult Research
- Infant Research
- Adolescent Research
- Geriatric Research
- Primate Research
- Sports Training
- fMRI / MEG / EEG
- Commercial eye-tracking (web usability, advertising, marketing, automotive, etc.)
- Finding good clues

- Communication systems for disabled
- Improved image and video communications

3.2 Commercial Applications

In recent years, the increased sophistication and accessibility of eye-tracking technologies have generated a great deal of interest in the commercial sector. Applications include web usability, advertising, sponsorship, package design and automotive engineering. In general, commercial eye-tracking studies function by presenting a target stimulus to a sample of consumers while an eye-tracker is used to record the activity of the eye. Examples of target stimuli may include websites, television programs, sporting events, films, commercials, magazines, newspapers, packages, shelf Displays, consumer systems (ATMs, checkout systems, kiosks), and software. The resulting data can be statistically analyzed and graphically rendered to provide evidence of specific visual patterns. By examining fixations, saccades, pupil dilation, blinks and a variety of other behaviors researchers can determine a great deal about the effectiveness of a given medium or product. While some companies complete this type of research internally, there are many private companies that offer eye-tracking services and analysis.

The most prominent field of commercial eye-tracking research is web usability. While traditional usability techniques are often quite powerful in providing information on clicking and scrolling patterns, eye-tracking offers the ability to analyze user interaction between the clicks. This provides valuable insight into which features are the most eye-catching, which features cause confusion and which ones are ignored altogether. Specifically, eye-tracking can be used to assess search efficiency, branding, online advertisements, navigation usability, overall design and many other site components. Analyses may target a prototype or competitor site in addition to the main client site.

Eye-tracking is commonly used in a variety of different advertising media. Commercials, print ads, online ads and sponsored programs are all conducive to analysis with current eye-tracking technology. Analyses focus on visibility of a target product or logo in the context of a magazine, newspaper, website, or televised event. This allows researchers to assess in great detail how often a sample of consumers fixates on the target logo, product or advertisements. In this way, an advertiser can quantify the success of a given campaign in terms of actual visual attention.

Eye-tracking provides package designers with the opportunity to examine the visual behavior of a consumer while interacting with a target package. This may be used to analyze distinctiveness, attractiveness and the tendency of the package to be chosen for purchase. Eye-tracking is often utilized while the target product is in the prototype stage. Prototypes are tested against each other and competitors to examine which specific elements are associated with high visibility and appeal.

One of the most promising applications of eye-tracking research is in the field of automotive design. Research is currently underway to integrate eye-tracking cameras into automobiles. The goal of this endeavor is to provide the vehicle with the capacity to assess in real-time the visual behavior of the driver. The National Highway Traffic Safety Administration (NHTSA) in the United States of America estimates that drowsiness is the primary causal factor in 100,000 police-reported accidents per year. Another NHTSA study suggests that 80% of collisions occur within three seconds of a distraction. By equipping automobiles with the ability to monitor drowsiness, inattention, and cognitive engagement driving safety could be dramatically enhanced. Lexus has equipped its LS 460 with the first driver monitor system in 2006, providing a warning if the driver takes his or her eye off the road [55].

Since 2005, eye-tracking is used in communication systems for disabled persons: allowing the user to speak, send e-mail, browse the Internet and perform other such activities, using only their eyes. Eye control works even when the user has involuntary movements as a result of Cerebral palsy or other disabilities, and for those who have glasses or other physical interference which would limit the effectiveness of older eye control systems.

Eye-tracking has also seen minute use in autofocus still camera equipment, where users can focus on a subject simply by looking at it through the viewfinder [16].

3.3 Eye-tracker types

Eye-trackers measure rotations of the eye in one of several ways, but principally they fall into three categories:

1. One type uses an attachment to the eye, such as a special contact lens with an embedded mirror or magnetic field sensor, and the movement of the attachment is measured with the assumption that it does not slip significantly as the eye rotates. Measurements with tight fitting contact lenses have provided extremely sensitive recordings of eye movement, and magnetic search coils are the method

of choice for researchers studying the dynamics and underlying physiology of eye movements.

2. The second broad category uses some non-contact, optical method for measuring eye motion. Light, typically infrared, is reflected from the eye and sensed by a video camera or some other specially designed optical sensor. The information is then analyzed to extract eye rotation from changes in reflections. Video based eye-trackers typically use the corneal reflection (the first Purkinje image) and the center of the pupil as features to track over time. A more sensitive type of eye-tracker, the dual-Purkinje eye-tracker [56], uses reflections from the front of the cornea (first Purkinje image) and the back of the lens (fourth Purkinje image) as features to track. A still more sensitive method of tracking is to image features from inside the eye, such as the retinal blood vessels, and follow these features as the eye rotates. Optical methods, particularly those based on video recording, are widely used for gaze tracking and are favored for being non-invasive and inexpensive.

3. The third category uses electric potentials measured with electrodes placed around the eyes. The eyes are the origin of a steady electric potential field, which can also be detected in total darkness and if the eyes are closed. It can be modeled to be generated by a dipole with its positive pole at the cornea and its negative pole at the retina. The electric signal that can be derived using two pairs of contact electrodes placed on the skin around one eye is called ElectroOculoGram (EOG). If the eyes move from the centre position towards the periphery, the retina approaches one electrode while the cornea approaches

the opposing one. This change in the orientation of the dipole and consequently the electric potential field results in a change in the measured EOG signal. Inversely, by analyzing these changes eye movements can be tracked. Due to the discretisation given by the common electrode setup two separate movement components - a horizontal and a vertical - can be identified. The potential difference is not constant and its variations make it challenging to use EOG for measuring slow eye movements and detecting gaze direction. EOG is, however, a very robust technique for measuring saccadic eye movements associated with gaze shifts and detecting blinks. It is a very light-weight approach that, in contrast to current video-based eye-trackers, only requires very low computational power, works under different lighting conditions and can be implemented as an embedded, self-contained wearable system [77]. It is thus the method of choice for measuring eye movements in mobile daily-life situations and REM phases during sleep. [16]

3.4 Commercial eye-tracking systems

Nowadays, there are several companies offering eye-tracking equipment for a variety of purposes. Some of the most well-known of them are shown on the table below:

Manufacturer	Type of Eye-tracking Device
Alphabio Technologies	Eyeputer, a head mounted 3-D video eye-tracker.
AmTech GmbH	Pupillograhic Sleepiness Test (PST), table mounted, monocular, video based systems.
Applied Science Laboratories	ASL, eye-tracking and pupillometry systems, both head mounted and remote.
Arrington Research	ViewPoint EyeTracker, both remote and head mounted, video based.
Cambridge Research Systems Ltd.	MR-Eyetracker, a low-cost, contact-free eye-tracker for fMRI & MEG.
Chronos Vision	Eye-tracking devices are used in e.g. neuroscience, ophthalmology, refractive surgery or clinical research. The classic Chronos Eye-tracker was deployed on the International Space Station (ISS) in early 2004 and is in continuous use for the study of eye and head coordination during long-term stays in the weightlessness of spaceflight.
Ergoneers Dikablis	Head-mounted lightweight eye-tracking system.
EyeTech Digital Systems	VT series, eye-tracking equipment
EyeTracking, Inc.	Technology developed by Marshall & CERF, San Diego State University.
Fourward Technologies, Inc.	Advanced Dual-Purkinje-Image (DPI) Eyetracker, mainly for research purposes.
Interactive Systems Labs	Model-based face and gaze tracking (from video image), Carnegie Mellon University.
ISCAN	Eye & Target Tracking Instrumentation, head mounted and remote eye-tracking systems, single and multiple target video tracking systems.
LC Technologies Inc.	Eye tracking systems and software for human factors research.
Mangold International	MangoldVision for lightweight, portable eye-tracking, solutions for both remote and head-mounted eye-tracking. Software for data recording and analysis.
NAC Image Technology	NAC EMR-9 eye path tracking.
Ober Consulting Poland, JAZZ-novo	Portable multisensor system with IR based eye-tracker. Head rotation and tilt measurement, blood pulse monitoring, voice recording and optional video context recording, designed to study human interaction with environment.
Ober Consulting Poland, Saccadometer	Portable eye movement laboratory for study on saccadic reactions using multiple diagnostic experiments, integrated stimulation and eye movement measurement and recording system, head mounted.

Optomotor Laboratories	Express-Eye, a stand-alone eye-tracker with saccade analysis, and FixTrain, a small hand held device for daily training of saccadic eye movement control.
Primelec, D. Florin	CS681, a digitally controlled scleral search coil system for the linear detection of 3D angular eye and head movements.
Seeing Machines	faceLAB, a 3D head position and eye-gaze direction tracking system (VOG based).
SensoMotoric Instruments GmbH	Remote and head mounted, Hi-Speed eye and gaze tracking for research and applied science, open programming interface and comprehensive stimulus/analysis software.
Smart Eye AB	Eye-tracking analysis based on any standard camera(s), analog or digital.
SR Research Ltd	EyeLink 1000 and Eyelink II, video based, head mounted eye-tracking system (EyeLink 1000 Plus has been announced).
Synthetic Environments, Inc.	EyeTalk integrates voice recognition and eye-tracking.
Thomas RECORDING GmbH	Eye-Tracking-System (ET-49/50) system constructed for neuro-scientific purposes and enables a laboratory to correlate the monkey's eye position.
Tobii Technology	Tobii X2 Light Eye Tracker, can be mounted on a laptop or PC monitor for a compact, portable eye tracking solution, Tobii Glasses 2, a new, unobtrusive and mobile eye tracker (head-mounted eye tracker) for capturing natural behavior in both real-world research, Tobii TX300, collects gaze data at 300 Hz yet allows large head movements. The system is designed for studies that require a higher sampling rate, Tobii T60 and T120 Eye-trackers - both integrated into a 17" TFT monitor also T60XL is available with a 24" TFT Monitor, and Tobii X60 and X120 Eye-tracker - a standalone eye-tracking unit designed for eye-tracking studies relative to any surface.

Table 3.1: Table list of the main eye-trackers for eye movement research, analysis and evaluation

3.5 Open source gaze-tracking, freeware and low cost eye-tracking

The following table contains low-cost, free and open source eye-tracking systems research prototypes and information that should help in building your own eye-tracker. Some of them are targeted at people with disabilities (eye-control systems), some for more general eye-tracking and research.

Open source eye-tracking	
Name	**Description**
openEyes	open-source open-hardware toolkit for low-cost real-time eye-tracking
Opengazer	open-source gaze tracker for ordinary webcams
TrackEye	real-time Tracking of Human Eyes Using a Webcam. Implemented in C++ using the OpenCV library
ITU Gaze Tracker	works with a webcam or video camera with night vision and infrared illumination
Freeware eye-tracking	
Name	**Description**

myEye	eye-tracking software to allow people with severe motor disabilities to use gaze as an input device for interacting with a computer
Open source and freeware eye movement analysis tools	
Name	**Description**
OGAMA (OpenGazeAndMouseAnalyzer)	open source software designed to analyze eye and mouse movements in slideshow study designs
RITCode	analysis tool for captured eye-tracker video files, created by the Rochester Institute of Technology Visual Perception Lab
ETU Driver	Eye-Tracking Universal (Standard) Driver, which helps the developer to build tracker-independent applications and test them off-line with a gaze data simulator
Low-cost eye-tracking	
Name	**Description**
I4Control®	a low-cost eye control system

Haytham Eye-tracker	a low cost eye-tracker
Research Prototypes	
Name	**Description**
FreeGaze	a gaze tracking system for everyday gaze interaction by Takehiko Ohno et al.
GoldenGaze	a low-cost IR-based eye-tracking system, developed at Universität Koblenz-Landau. Droege et al.
Miscellaneous	
Name	**Description**
Blink-It	a system for environment control and communication for entirely disabled people. Not eye-tracking but reacts to eye blinks
xuuk eyebox2™	it detects when an eye is looking at it at reasonable cost

Table 3.2: Table list of low-cost, free and open source eye-tracking systems

Practical tips on: Building a lightweight eyetracking headgear by J.S. Babcock and J.B. Pelz. Proceedings of ETRA 2004

Practical tips on: Real Time Eye-tracking and Blink Detection with USB Cameras by Chau, M. and Betke, M. Boston University Computer Science Technical Report No. 2005-12.

For more related references, see:

http://wiki.cogain.org/index.php/Bibliography_Gaze_Interaction

3.6 Driving and eye-tracking research

The use of mobile eye-tracking equipment in active moments of participants, like driving, is not a new idea. The first mobile eye-tracker was created in the 50s by Norman Mackworth. Actually, it was Mackworth and Thomas who firstly used a mobile eye-tracker to record the gaze of a driver in 1962. Nevertheless, we had to wait for 20 years, for user-friendlier devices of this kind (in the 80s the video cameras where significantly smaller and the usage of computers to analyse the recorded images was possible) [57]. A great review concerning the methods and ways used in eye-tracking experiments was created by Duchowski in 2002 [58].

During driving we expect lots and different eye movement strategies and patterns to be followed as it is a rather complex activity. As a driver someone has to keep an eye on the road itself (e.g. turns), to monitor the other users on the road (e.g. other vehicles and pedestrians), to pay attention on the road signs and of course to decide, after the feedback he gets by evaluating all the above information, where to steer and control the speed of his vehicle. Thus, driving is a very complex task and so very interesting to monitor with eye-tracking equipment.

The most important factor during driving is obviously the process of steering as everything else follows. The eyes give to the brain all the necessary information to decide when exactly to turn the steer and how much, a process that needs excellent eye-hand (and arm) coordination. In 1978, Edmund Donges, using a driving simulator, describes that basically there are two sorts of signal to the driver. Firstly, the feedback signals (lateral and angular deviation from the road centre-line, differences between the road curvature and the vehicle's path curvature), and feed-forward or anticipatory signals obtained from more distant regions of the road up to 2 s ahead in time (corresponding to 27.44 meters -90 feet- at 48.28 Km/h -30 mph) [59].

The findings of Donges have been confirmed almost 20 years afterwards on real roads and on a simulator by Land & Lee in 1994 [60] and Land & Horwood in 1995 [61] respectively. In general, several studies, except those mentioned above, took place during the last years (eg. Zwahlen, 1993 [62], Underwood, Chapman, Crundall, Cooper, & Wallın, 1999 [63], Land & Horwood, 1995 [61], Land, 1998 [64], Summala, Nieminen, and Punto, 1996 [65], Mourant and Rockwell, 1970 [66]) concerning the way the driver is looking around.

Studies, mainly on U.S. roads that had predominantly low curvatures, found only a weak relationship between gaze direction and steering. On the other hand, on a winding road in Scotland, where continuous visual control was essential, a much more precise relationship was seen. Moreover it was found that drivers spent much of their time looking at the tangent point on the up-coming bend and it was shown that curves of gaze direction and steering wheel angle are almost identical. The implication is that this angle, which is equal to the eye-in-head plus the head-in-body angle when the driver is looking at the tangent point, is translated more or less directly into the motor control signal for the arms.

Simulator studies showed that feed-forward information from the distant part of the road was not on its own sufficient to give good steering. In order to maintain good lane position it was required a view of the road only a few metres ahead and this region provided much of the feedback information identified by Donges. Interestingly this part of the road was rarely fixated compared with the more distant tangent point region, but it was certainly seen and used. The principal conclusion from these studies is that neither the far-road feed-forward input nor the near-road feedback input are sufficient on their own, but the combination of the two allows fast accurate driving [57]

Nowadays we know that the eyes are not absolutely glued to the tangent point, but can take time out to look at other things. These excursions are accomplished by gaze saccades and typically last between 0.5 and 1 s. The probability of these off-road glances occurring varies with the stage of the bend that the vehicle has reached, and they are least likely to occur around the time of entry into a new bend. At this point drivers fixated the tangent point 80% of the time. It seems that special attention is required at this time, presumably to get the initial estimate of the bend’s curvature correct [57].

Sometimes the eye must be used for two different functions at the same time, and as there is only one fovea and the off-axis vision is poor, the visual system has to resort to timesharing. A good example of this is when the driver is negotiating a bend and so needs to look at the tangent point, while passing a cyclist who needs to be checked on repeatedly. The record shows that the driver alternates gaze between tangent point and cyclist several times, spending half a second on each. Thus not only does gaze switch between tasks, so does the whole visual-motor control system. Presumably, whilst looking at the cyclist, the information from the tangent point is kept “on hold” at its previous value. In urban driving this is even more important as each traffic situation and road sign competes for attention. To the best of our knowledge, there has been no study of where drivers look in traffic, but it is believed that drivers foveate the places from which they need to obtain information: the car in front, the outer edges of obstacles, pedestrians and cyclists, road signs and traffic lights and so on. In general, speeds of 48.28 Km/h -30 mph- or less only require peripheral lane-edge information for adequate steering [57].

Until now it is the applications on driving simulators that seem to be more interesting for the researchers and thus the majority of studies didn’t take place in real life. Driving simulators have taken advantage of eye-tracking to monitor eye position and point of gaze for auto design, safety and instruction. For example Professor Schieber (University of South Dakota’s Heimstra Human Factors Labs’ Visual Performance Laboratory) is using ASL’s eye-tracking equipment in order to assess differences in the legibility and visibility of highway signs among adults. As a secondary field of research there is an attempt to characterize the differences in visual information available in nighttime versus daytime driving scenes. Additionally, the University of Massachusetts uses this kind of technology to develop manufacturing tutorials and pursue a broad range of visualization studies [67].

In addition, DENSO, one of the largest suppliers in the world of parts to the automotive industry, is trying to improve safety through enhanced driver visibility by doing research concerning the image recognition and the eye-tracking technique. In this approach DENSO, is combining eye-tracking technology, which can recognise what the driver has or hasn't seen, with image recognition, which extracts necessary information from images taken by a camera installed on the vehicle. There are various obvious applications, such as informing drivers about pedestrians in a blind spot or hidden behind another object, road signs which alert the driver of hazards, temporarily obstructed by larger vehicles, alerting the driver of an approaching vehicle which they might have missed [68]. Another, company that uses the eye-tracking technique both in a car simulator and in real-life condition, is Pertech. That company produces and sales innovative solution for behaviour analysis, especially concerning the ocular activity. The French car manufacturer RENAULT is an industrial partner of them in order to optimize the driver's cockpit [69].

Nakayasu et al in 2009, tried to use eye-tracking as a tool measuring the driver's skill of visual perceptions and motor behavior in various traffic situations. In that study they found that the visual stimuli in the periphery affect the eye movements during driving. Whenever there were not lots of objects needing attention, there were also less eye movements with bigger fixation times in comparison with situations there were lots of things to attract the attention. In general and in all situations it was found, as we would expect that the eye movements have a lot to do and depend on the driver's experience [70].

As this topic is rather "hot" nowadays, there are more companies developing their own solutions in order to analyze the behavior of someone while driving. A worldwide known company that is interested in this area of research is Toshiba which in 2009 showed off a new system that will not only let the driver control the A/C or radio with the glance of the eye, but also alert him if he happens to take his eyes off the road for too long. That will be possible by using a camera mounted above the steering wheel, in order to detect the driver's face, giving to the car's computer the chance to detect several clues of the driver's performance such as the head movement, the eye direction and eyelid blinks. According to Toshiba those measurements could eventually be used to alert drowsy drivers. Nevertheless, until now, Toshiba does not have any immediate plans to commercialize the technology [71].

A company that uses the eye-tracking technology combined with driving simulators is eye-com. At the moment, in the facilities of the company there is a driving simulator used in a variety of applications:

- Research (drowsiness detection, effects of inattention/distraction, etc.)
- Testing for fitness or ability to safely operate a vehicle
- Clinical rehabilitation (patients with mild dementia, stroke victims, etc.)
- Gathering of oculometric data when used in conjunction with the Eye-Com Biosensor, Communicator and Controller

This company conducted a research project in cooperation with the U.S. Department of Defense and the U.S. Department of Transportation named PERCLOS (PERcentage of eyelid CLOSure over the pupil over time). The findings of the PERCLOS study have been used in order to develop in the near future the first Composite Oculometric Fatigue Index (COFI) and some Safety Response (SAFE) algorithms, which would be used to predict driving accidents and perhaps even prevent them. Eyecom is also using this kind of technology in other military projects such as SBIR for the U.S Army (Soldier-Mounted Eye Monitoring System for the Measurement of Ocular Fatigue and Drowsiness-related Performance Failure). It seems that that specific company is a leader on this field as, in 2009, a Congressional Initiative grant was awarded to it in order to fund a series of complex eye-tracking challenges for the U.S. Department of Defense [72].

Furthermore, Noldus, a well-known software development company offers the Observer XT solution which analyses the data from eye-tracking devices and was acquired by Volvo, the Swedish car manufacturer, back in 1996, for its Cognitive Ergonomics laboratory in Göteborg. That laboratory offers a variety of research services to the Swedish company and as members of the staff, like the human factors engineer Trent Victor, noticed: "By accurately recording and analyzing the driver's actions in response to all sorts of signals coming from outside the car or from on-board information sources, we hope to improve the design of the human-machine interface" as their motto says: "Our ultimate goal is to design cars which are safer and easier to use." [73].

Another, solution which is already in use and mostly oriented to companies with fleets of trucks is the Seeing Machines' DSS (Driver State Sensor) suite.

Figure 3. 1: The Seeing Machines' DSS suite [74]

That suite is a robust, automatic platform that uses cutting edge face tracking techniques to deliver information on operator fatigue and operator distraction. The DSS has been specifically designed for straightforward deployment into vehicles & environments where fatigue and inattention need to be monitored and managed. The DSS-IVS (in vehicle system) measures the eyelid opening of the driver, and based on this data derives the drowsiness state. No sensors need to be worn by the driver; a remote sensor on the dashboard observes the face of the driver and measures eyelid closure.

Figure 3. 2: The DSS software [74]

With its dashboard mounted sensor, the DSS "sees" the driver's face and measures the amount of eyelid closure and orientation of the driver's head. Eyelid closure information is processed to determine the level of driver fatigue, and by tracking head movement, the DSS is able to detect driver distraction. Once installed in the vehicle, the DSS is fully automatic. There is no calibration procedure for new drivers, and slip-seat operations are handled transparently. Any driver can get behind the wheel without a time consuming setup process, annoying sensor attachment, or any special knowledge about the system. There is no disruption to work processes and no additional driver training required.

Seeing Machines has also produced a similar product named faceLAB.

Figure 3. 3: The Seeing Machines' faceLAB approach. [74]

The tracking technology behind faceLAB is deployed in vehicle fleets around the world to manage fatigue and operator attention on a large scale. faceLAB is uniquely capable of providing head and eye-tracking data for research into human behavior in real vehicles, cabins and cockpits, both indoors and outdoors. faceLAB has been successfully used in cars, trucks, trains, planes, submarines and other vehicles. Network remote-control allows seamless integration into your research vehicle. To capture subject interactions in large cockpits, additional cameras can easily be added to extend the tracking range. The additional cameras can also be used independently, as separate eye-trackers. Forward-facing cameras can be used to capture the subject's interactions with the dynamic scene in-front of the vehicle. Any number of these cameras can be used, together, to cover a large field of view, or independently. faceLAB tracks calibrated gaze fixations against any number of objects, such as cockpit controls or in-vehicle devices. The system provides instant feedback on gaze interactions with customizable regions of interest. At the moment Seeing Machines is working closely with simulator provider Realtime technologies, who now offer simulators with turn-key faceLAB integration. The simulator data analysis software automatically correlates the eye fixation data from faceLAB with the dynamic simulation scenario in real-time. All data is synchronized and available via the network as a single output stream.

faceLAB can also be used in collaboration with an extra component available by the same company. That component is called SceneCamera and allows recording and analysis of gaze intersections with objects in dynamic, moving environments objects in dynamic, moving environments.

In more details the SceneCamera provides a subject's-eye view of the experiment as the user can:

- Watch as the subjects drive, observing other vehicles, signposts and traffic signals.
- Study gaze behavior in two-person face to face interaction, watch eye contact and glances in a naturalistic setting.
- Provide objective feedback backed up by video evidence for training of drivers, pilots, or critical task operators [74].

Figure 3. 4: Fixations of a user while driving, using the faceLAB approach. [74]

Tobii, the Eye-tracking devices manufacturer, proposes, as additional results of that technology, the detection of the position of the driver, which will give the means to personal adjustments of the driver environment, such as individually tuned airbag deployment, rear mirror adjustment, seat to pedals, steering wheel adjustment and so on. By integrating a sort of "black box" data storage in the car which will include eye and face tracking it is possible to analyze the driver's actions before and during collisions. Thanks to those captures we will have the ability to replay the scene of an accident, fact which is vital in understanding how to improve traffic situations and develop road safety [75].

In general, there are lots of papers and presentations that have to do with driving safety and the variety of reasons and factors which can affect it. Even though some first steps have been made by big automobile industries it is clear that still lots of things need to happen and take place in order to have eye-tracking equipment as a standard tool in a car. Nevertheless, it seems that this is a hot topic for the industry and steadily the interest on this kind of technology is increasing. It is safe to assume that in the next years more and more companies are going to introduce new models of vehicles with their approach on eye-tracking tools that are going to improve the safety standards and, in the future, the convenience of the user during his driving activities. In addition, it is clear that up to now the vast majority of the approaches have to do with the use of eye-tracking technology concerning the detection of the driver's fatigue and drowsiness. We can assume that in the near future the uses of this kind of technology could be expanded and the fixations of the user would be used to perform specific tasks while driving. For example, a short of display panel can be projected on the lower part of the windscreen and the user might be able to change radio stations or change the airconditioning settings by fixating on that panel (in that case the eye fixations would be used instead of pressing buttons). In that case, the driver will be able to proceed in specific actions without having to move his eyes for a long distance and with having the ability to keep the road on the periphery of his visual field. Of course, when the eye-tracking technology will be integrated in the automobile industry the whole equipment used in that field will be optimized for that use and more and more applications, that now we can only imagine, might become available (e.g. imagine using a corner of the windscreen that isn’t that vital while driving-like an upper corner-to visualize a GPS application or a phone’s keypad).

Chapter 4

Presentation of the objective and initial algorithm design

4.1 Overview

This chapter introduces the approach of this thesis concerning the usage of the eye-tracking technology not only alone but also in combination with the GPS devices that are already widely used nowadays in vehicles. Those GPS devices can either be standalone devices planted on the car or the smartphone of the driver with GPS software installed. At the time being it seems that eye-tracking in driving is mostly used indoors with driving simulators and also in a small variety of things. Unfortunately, even though some experiments of this type do take place in real-life conditions, the results rarely get published in journals as, most probably, the knowledge acquired is confidential and for use only by the company conducting the experiment. It is safe to assume that, since nowadays the automobile companies seem to get more and more interested in that kind of applications, a variety of them will be developed and offered to the drivers. Eventually, this area of interest is going to be expanded gradually and perhaps in some years the eye-tracking technology will be commonly used in the driving industry.

4.2 Presentation of the objective of this study

The main objective of this thesis is to present the potential advantages, especially concerning safety, that the eye-tracking technology can offer when used while driving. As it was mentioned above, there already exist devices that constantly check the closure of the eye in order to make sure that the driver is not about to fall asleep, nevertheless, practically, in that case we have to do with eye-detection and not eye-tracking as the system is able to "tell" if the driver is about to close his eyes but there is no information about where he sees even when they are open. In our case, we are proposing the usage of equipment that will have all the original eye-tracking technology characteristics targeting at making the driving procedure easier and, more importantly, safer.

It is obvious that while driving we are surrounded by a variety of distractors such as cell phones, radios, and passengers requiring multitasking. Those distractors, no matter if they origin from the inside or the outside of the vehicle can potentially put in danger the driver's life and others. By using the eye-tracking technology, the system installed would be able to detect gazes of the driver on the phone or on another person triggering an alarm to remind where he should be focusing. It has been proposed also that in the near future, and after the required extensive research, eye-tracking systems could collect oculometric signs connected with intoxication not only helping the authorities detect non-sober drivers but also preventing the engine from starting. It is obvious that in that case, that application would prevent lots of accidents only by not allowing people drive when they are under the influence of intoxicating factors. [85]

As technology, advances rapidly there are thoughts about using Head-Up Display panels (HUDs) in vehicles to project vital information on the front window so that the driver has not to look around for them. Even though, this idea is not new, as it was firstly introduced by General Motors in 1988, with their Oldsmobile Cutlass Supreme model it is not widely used 25 years after that. Since then, other car manufacturers also adopted the technology (e.g. Lexus) on their sport or luxury vehicles and BMW was the first European one. [86] The latter, has launched a new HUD for its vehicles in April 2012 which thanks to modern technology is a full colour one and has an improved resolution (640x480) that is enough even for the moving map of a GPS application. [87]

In March 2012 General Motors has also offered some information concerning the HUDs some of its vehicles uses. According to the company the heads up display in the car is designed in such a way that the driver can keep the eyes on the road and still see important details such as the speed, and R.P.M of the vehicle.

Figure 4. 1: General Motor's recent HUD capture [88]

The technology will also indicate other things in the company's vehicles such as vehicle warnings, and radio information. The driver has the freedom to adjust the position of the display on the windshield, which can be dimmed or brightened.

That kind of displays is becoming increasingly available in production cars, and usually they offer speedometer, tachometer, and navigation system displays. Night vision information is also displayed via HUD on certain General Motors, Honda, Toyota and Lexus vehicles. Other manufactures such as Audi, Citroën, Saab, Nissan, and Kia currently offer some form of HUD system. [88]

By combining, the above mentioned HUDs with eye-tracking it could be possible to use applications, like a map, and collect data concerning our car without having to move the eyes from the road. Gazes and fixations can be used to control the radio, change the projected information according to the needs of the driver and in general as input data by using already existing and well-known application control algorithms that are used to help people with moving disabilities.

Tobii, one of the most key eye-tracking companies announced its own platform for drowsiness and distraction detection in driver safety systems [89] and since then it continued research on that area. In April 2012, they have published a video that actually uses eye-tracking and HUD technology in their approach of making a user-friendly interface that can make driving a safer procedure. [90]

Figure 4. 2: Capture from the Tobii's approach about eye-tracking and HUD [90]

In general, we can assume that the new technologies will make the already existing panel that gives us information about the car (e.g. velocity) and its condition (e.g. fuel, temperature etc.) obsolete and that all the information needed will be projected on the front window making easier for the driver to stay focused on the road. Yet, it is the eye-tracking technology that will also offer the ability not only to gather information but also interact with the new projected panel using only eye movements as an input.

In this thesis we propose also another new approach which can help drivers avoid crashes, caused by any kind of concentration or attention loss, by alerting them when a collision is imminent. For that purpose, the information taken in the car concerning the gaze of the driver is combined with the information taken for the vehicle thanks to the GPS device.

On this section we propose an algorithm that combines data gathered both from an eye-tracking and a GPS device in order to increase the safety standards during the driving procedure. The goal of the algorithm is to extract the location of a vehicle and the gaze of the driver relating with the information gathered by the system concerning vehicles that will come too close in the near future. The algorithm begins by tracking the gaze of the user as the vehicle's engine starts and continues by locating the exact location of the vehicle each specific moment. Then, as the eye-tracker continuously checks if the driver is looking at the specified areas of interest (through the front window or at the mirrors), the data concerning the vehicle (location, speed, direction) are sent to the server. At this stage the server, which gets GPS data from the other vehicles in the area also, computes the probability of a potential meeting with one of them and sends a feedback message to the vehicle's receiver in order to alert the driver. Finally, the system either keeps tracking the gaze of the driver and the location of the vehicle or, in case, according to the message, another vehicle is about to come too close it alerts the driver.

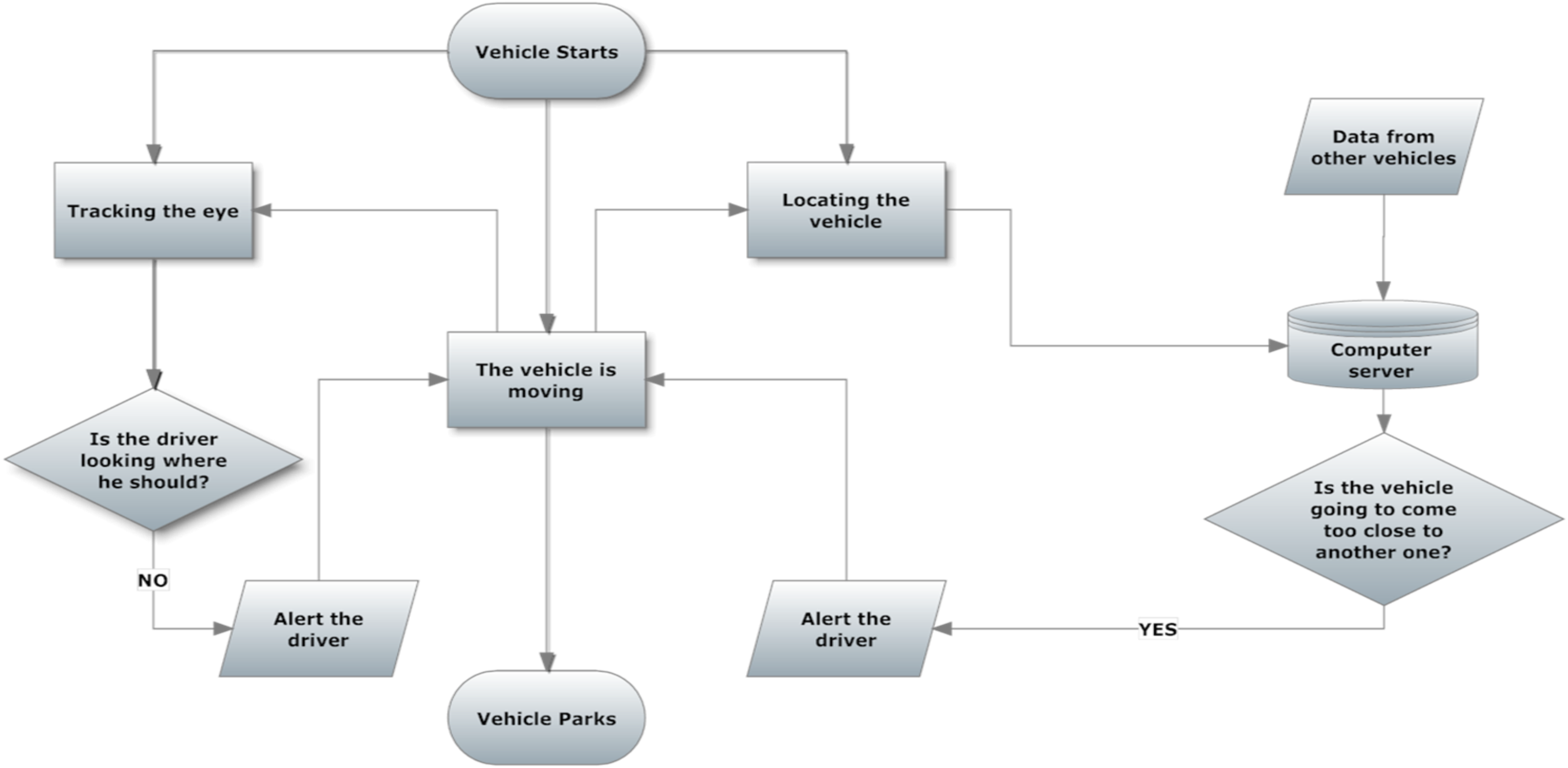


Figure 4. 3: Initial design of the algorithm concerning the problem proposed

4.3 Theoretical challenges in the design

Even though, the solution proposed in this thesis doesn't import any kind of new technology that did not exist before; it introduces a new, original approach concerning the combination of existing and well-known technologies which might significantly reduce accidents caused by the concentration loss of the drivers. Undoubtedly, there will be also problems if this approach ever comes in the real world as making totally different technologies work together can become a rather difficult task.

What is for sure is that eye-tracking in combination with the HUDs can make driving safer as by definition the driver won't have to move his eyes from the road to check the velocity, the map or change the radio volume. And, as the display cannot project everything a driver might want at the same time, the eye-tracking equipment, which constantly checks if he is not too fatigue and about to close his eyes, calculates the gaze and the fixations that would act as input data to make all the appropriate adjustments.

Practically, the main challenge into getting the HUD to work and look good is the seemingly simple task of making a square look square when it is projected onto a curved surface. Nevertheless, it seems that the recent approaches have managed to overcome that obstacle. General Motors in particular solved that problem thanks to a series of mirrors and a display screen projecting the HUD on the windshield.

One major question is how the GPS system in the car is going to send the vehicle's data to the computer server. Nevertheless, thanks to the recent advances on that area and the wide spread of GPS systems, that problem has been resolved by numerous ways. Actually, there are three basic ways to receive traffic information:

- **Bluetooth:** The navigation device uses the GPS in the car to receive the traffic information, which is beamed to the unit via Bluetooth technology. This method effectively requires from the user to have a data package, as a constant connection is needed to ensure continuous updates along the route.
- **FM:** Using the same FM Radio Data System (RDS) that delivers station IDs and song information to modern stereos, traffic updates are delivered over the airwaves. An external antenna may be optional to assist with reception.

A new twist on the traffic sources is MSN Direct from Microsoft. An FM-based service, MSN Direct is available on just a handful of models out of the box and several more are compatible with the purchase of an additional FM receiver. This service combines traffic information with weather, movie times, and gas prices.

Another solution which was given recently, at least in Greece, is the GPS device with a SIM card inside. Thanks to that SIM traffic data are received by the vehicle thanks to information taken from the Traffic Management Network which was built for the Olympic Games of Athens in 2004. Such, an approach can be used widely but in certain cases the data can be too many or interference problems may occur because of the large number of vehicles. A probable solution could be, copying the mobile phone networks, to divide the big urban areas into cells with a computer server in each cell. Those servers would, obviously be linked together, and the data computed would have to do only with a specific cell minimizing the workload. Finally, as at the moment there are services in which the driver can send an SMS and receive back a message concerning the traffic news we can expect, in the near future, those messages to be sent automatically to the GPS device and display the traffic data on the GPS' map.

The other important task that has to be resolved has to do mainly with the procedure of eye-tracking during driving. One of the biggest problems that exists, and has been in great percentage resolved is the one caused by the sunlight. As the eye-trackers are using infrared light to detect the pupil of the user the sunlight can cause extra corneal reflections and differences of the pupil's size making the tracking of the eye impossible. Another difficulty that may occur is when a driver is wearing prescription glasses. Even in that case though, significant progress has been achieved and in most cases the tracking of the eye can be done without any problem. Of course, if a driver is wearing sunglasses things become worse for the system to work effectively and a malfunction is almost certain. Finally, even the height of the driver might cause problems as the cameras have to be centered on the eyes of the user. Of course if there is only one person driving the vehicle then this centration must be done once. The majority of vehicles can be driven by a variety of people (members of the same family or company) and as a result the cameras installed in the camera should probably be on a motor-moving base which would focus on the eye(s) of the driver with the help of the software which will try to detect the face, and thus the eyes, of the user.

As a conclusion, a significant drawback can be the fact that a new hardware device has to be built which has to have a reasonable cost but in the same time to be able to detect the gaze of the driver, collect the vehicle's information and manipulate data of completely different nature. Nevertheless, as the optics are getting less expensive and the processors can compute a significant amount of data in less time such a challenge does not seem utopic and impossible to be achieved nowadays.

Chapter 5

Integrating an eye tracker to an automobile

5.1 Consideration in installation of an eye-tracker in a vehicle

As specified above, the vast majority of eye-tracking systems used in automobiles, have to do with the detection of the fatigue of the driver. Those products are mountable units that can be placed on any vehicle. The figure below depicts a standard driver fatigue monitor, which consists of a camera system comprising of camera and InfraRed (IR) sensors and a control unit. The control unit consists of the microcontroller (this is responsible for decision making) and the alert sounding system.

Figure 5. 1: Components of Driver Fatigue Detection systems [95]

Nevertheless, a fully working eye-tracking system would not be able to function the same way as it needs a computer analysing the data as long as it is operational. As a result, and since the modality proposed here has the communication between the eye-tracker and the vehicle as a prerequisite, the only viable solution seems to be an approach with a direct connection to the automobile's internal sensors and systems. However, there are several similarities on the way the camera has to be installed in an automobile, no matter if it is an eye-tracker or a simple driver fatigue detection system, in order to guarantee the systems' functionality the best possible way.

The camera unit is the main component of such a system. Hao Nai [95] describes the main requirements for placing the components in the vehicle:

- Monitor is best located within 20 degrees of the driver's normal sitting position.
- The camera lens must face the driver directly. This can be placed on the dashboard on various locations, as shown in the picture below

Figure 5. 2: Placement of camera unit [95]

- If the dashboard is too low (such as in a truck or in a bus) then the installation must be done at a higher position.

- If the camera is installed on the side front, then it should not be placed at an angle greater than 15 degrees (the angle can be set by rotating the foundation lens).

- The distance from the lens to the driver’s eyes should be between 600 mm to 700 mm for drivers with glasses or little eyes whereas the distance can be up to 900 mm for normal sized eyes.

- The control unit is typically placed on the dash, right to the steering wheel. It is small in size and light in weight [96]. The power to the circuit is provided by means of the 12V car adapter provided with the kit.

- In case of vehicles, where a driver fatigue detection system is inbuilt as part of the safety system, the dashboard consoles can be used as user interface.

Researchers have argued that there are many considerations that must be taken into account before installing in-vehicle technologies. For instance Hartley [97] argues various factors such as validity, reliability, generalisability, sensitivity and specificity must be addressed for eye-tracking systems in a vehicle. Also the system must not startle or confuse the driver thereby causing negative effects. These factors have a direct effect on the general design, installation of the system and its usage. The system must be assistive and not distracting. Hartley [97] suggests that many systems solely adopt a warning tone or signal, which can be either auditory or visual. Auditory signals can be relayed through the car's speaker system, but the effectiveness of alarm systems remain only up to a certain limit, after which the user can get prone to them. Visual signals can be displayed as a flashing light on the Dash board (similar to the check engine light) but again, these signals are often ignored by drivers. This reduces the system effectiveness. Unfortunately most of the available systems are portable, mountable products that can prove to be ineffective.

A more successful system can be obtained with technologies that include interface of in-vehicle Electronic Control Units (ECUs) that allow auto braking and relying of message to a centrally manned unit, that can contact the driver and make sure correct action has been taken. This forms a trade-off between effectiveness and complexity of interface with the car. With the increasing complexity of on-board electronics in vehicles, addition of further systems can cause issues in complying with safety features and on board electronics (as discussed in the section below).

5.2 Communication between the eye-tracker and vehicular electronics

As seen from the above examples, certain parameters are needed for an interface between an eye-tracker and the vehicle. These can range from simpler systems that are basically mountable products that require only power supply output from the car dashboard, to more complex systems that pass data over Controller Area Network (CAN) bus to communicate to other electronic units in the vehicle. In case of systems where an eye-tracker should be used as a channel that could transmit information back to the control unit about the driver's state (e.g. vigilance), further network communication might be required.

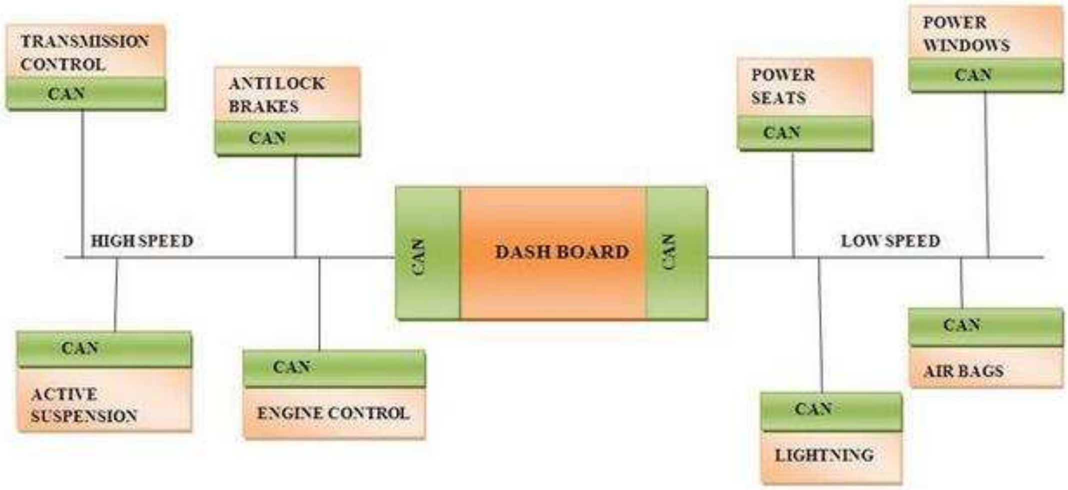


Figure 5. 3: Typical ECUs on CAN-bus [98]

The car consists of a multi-master CAN bus based system architecture as depicted above. Each of the ECUs has a specific function in the car and can communicate to each other over CAN. To the central dashboard (which acts as the main user interface), are connected all the other ECUs, via high speed links (with higher priority) and low speed links (with lower priority). The high speed links are ECUs pertaining to safety critical aspects of the car such as engine control, brakes control, transmission, gear shifts whereas the low speed links are for non-safety aspects such as infotainment systems, windows, doors and seat switches. This form of system is a distributed one, wherein, the control is not fixed to one centralized body, rather is distributed among several ECUs, each performing a dedicated task. This system ensures that the reliability of performance is not centred on one ECU alone, thereby ensuring that if one ECU fails, other components can function and provide the driver the ability to safely take a preventive action.

5.3 A more detailed view on the CAN bus system

The reason for using a CAN bus system in an automobile is the possibility to create a network connecting several modules and sensors that are used in it. That approach makes the CAN bus a sort of "data highway" and it was chosen because it offers several advantages [99]:

- ✓ Concerning a more technical point of view, the CAN bus is reasonably flexible as there is the ability to use a variety of media to transmit information (e.g. copper cable, optical fibres). In parallel, it is possible to perform cross-system diagnostic tests across a number of control units
- ✓ It is a uniform platform used to exchange data among the separate control units simplifying the whole procedure
- ✓ The automobiles systems and subsystems can be expanded and installed more easily. This is extremely helpful, especially in cases that systems involving more than one control units (e.g. ESP) have to be implemented
- ✓ The CAN bus can also be used to perform diagnosis inside the car (e.g. airbag control unit). In that case, it is referred as "K-wire" or "virtual K-wire"

Since there is a variety of modules connected on a CAN bus its design has to follow certain requirements to ensure it is functioning properly [99]:

1. High degree of certainty concerning the ability to detect errors caused by transmission interferences (either internal or external)

2. In case a control unit fails for any reason, the rest of the systems have to keep being operational and exchange information (as far it is possible)
3. To make sure that the system meets the requirements of real-time driving conditions there has to be a high data transmission rate
4. There has to be a high data density, meaning all the connected nodes must have the same information status. That way, if an error is detected all the users will be informed simultaneously

Nowadays, copper wires are used to transmit the information over a CAN bus. This transmission has a secure maximum rate of 1 Mbps (1000 Kbps). According to the time requirements, the signal repetition rate and the data volume of each unit, the CAN bus is divided in 3 subsystems. Firstly, there is the drive train CAN bus at 500 Kbps that has practically real time requirements. Then, there is the infotainment CAN bus and the convenience CAN bus both at 100 Kbps and with low time requirements.

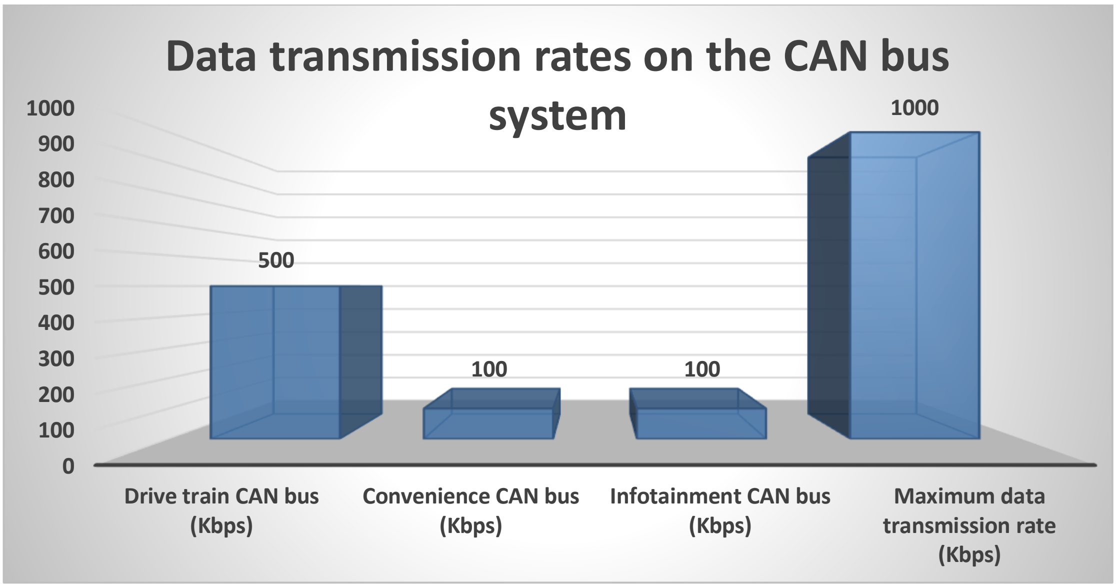


Figure 5. 4: Data transmission rates on the CAN bus system

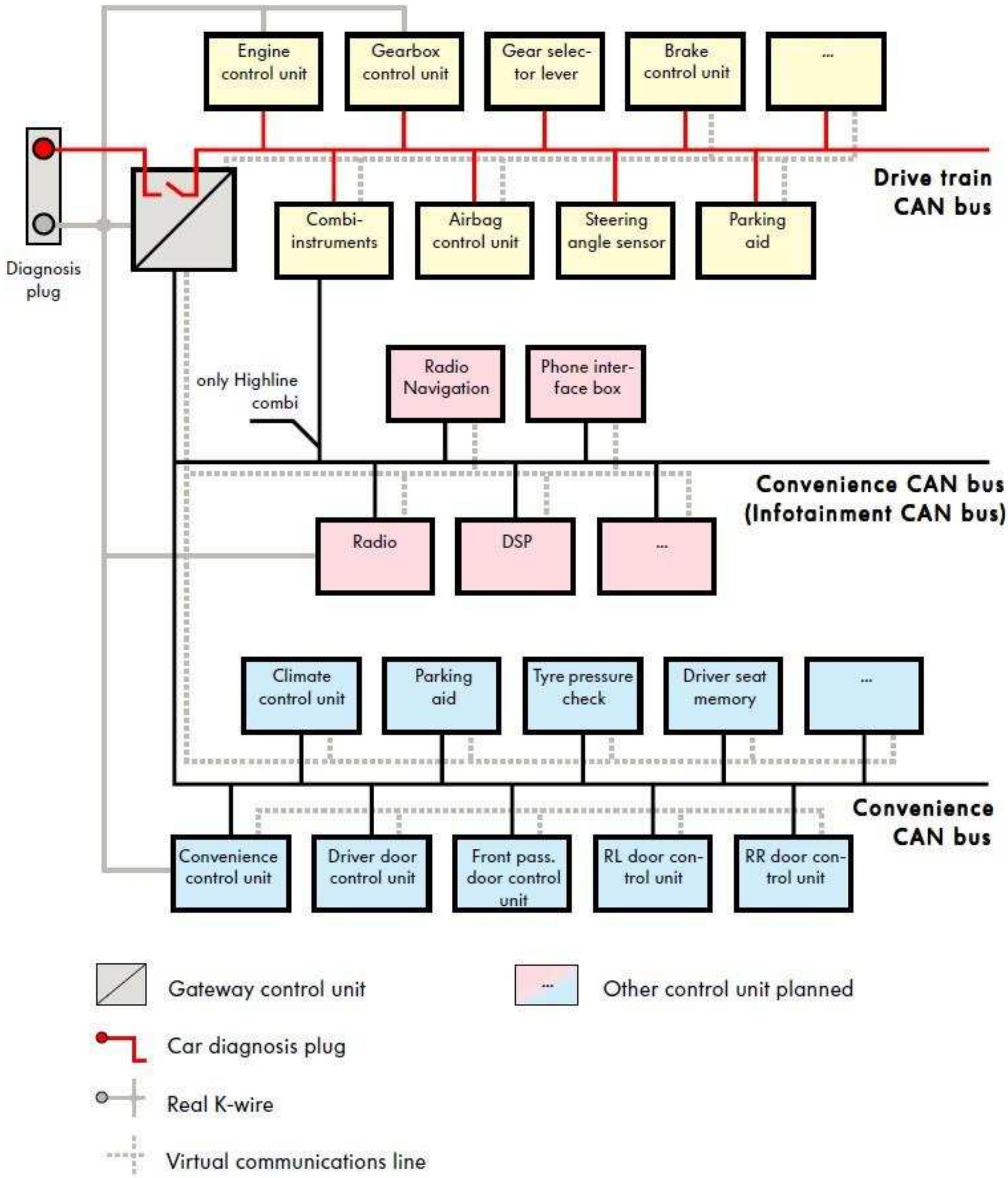


Figure 5. 5: A typical CAN bus system [99]

(The figure above depicts the CAN bus system of a VW Polo)

As in all the network systems the information is exchanged using packets known as messages. Any of the connected control units is part of the network and it can receive and send messages. Such messages transmit values such as the fuel level. This information (fuel level), is represented as a binary value (a string of ones and zeroes) which is converted into a serial bit stream [99]. In that time, the new bit stream is sent over the transmission line to the transceiver (that acts as an amplifier). The transceiver converts again the bit stream, this time into voltage values, which are then sent –one by one-over the bus line. At the time the message reaches the receiver the voltage values are converted back into a bit stream by the transceiver and sent over the reception line to the control units. Finally, the control units convert the binary values into messages [99]. As it was noted above, a sent message can be received by any of the existing control units. The whole procedure is similar to the way a radio station operates, so in our case a message is broadcasted and any unit can receive it. That approach ensures that the last requirement mentioned before concerning the design issues, meaning that all control units connected to the bus have the same information status, is met [99].

5.3.1 CAN bus Architecture

Because of the augmentation of the control units used in a vehicle there had to be developed a technology allowing their intercommunication. As a result, the CAN bus was developed. The following figure depicts a typical example of a CAN bus used in a vehicle.

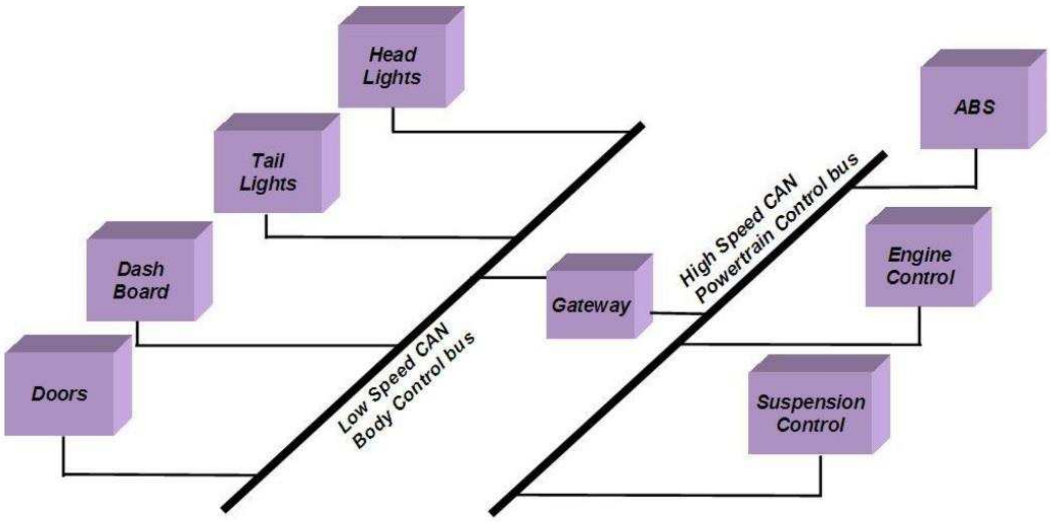


Figure 5. 6: Typical CAN bus architecture [100]

Each control unit, before sending a message has to see that the bus is not currently busy using the same media access control protocol used Ethernet LANs which is known as CSMA/CD (Carrier Sense Multiple Access, Collision Detection). In case the bus is busy, then the control unit holds the data until the bus is free, and then transmits them.

The Collision Detection part is used when more than one control units are about to transmit a message at the exact same time after sensing that the bus is free. In that case, the data from the different control units would collide and get corrupted [100]. Similarly to computer networks, in order to avoid such an incident, CAN prioritizes the information and allows the transmission of high priority (e.g. speed) delaying information of lower importance (e.g. air condition malfunction) [100].

Moreover, to avoid the risk of delay of critical data the control units of low importance are connected to an independent CAN bus which is further connected to the high importance CAN using a gateway that filters the data to be exchanged between the two CAN buses [100].

5.3.2 ECU CAN Interface

Nowadays, almost all the in-vehicle control units are interconnected using CAN bus. As it shown in the figure below a control unit contains a microcontroller and an interface (CAN controller and CAN transceiver). The microcontroller contains the corresponding program in each case and the CAN controller (network interface) extracts and puts into frames data from it in order to get them transferred to other ECUs. It is the CAN transceiver that sets up the electrical signalling needed for the transmission [100].

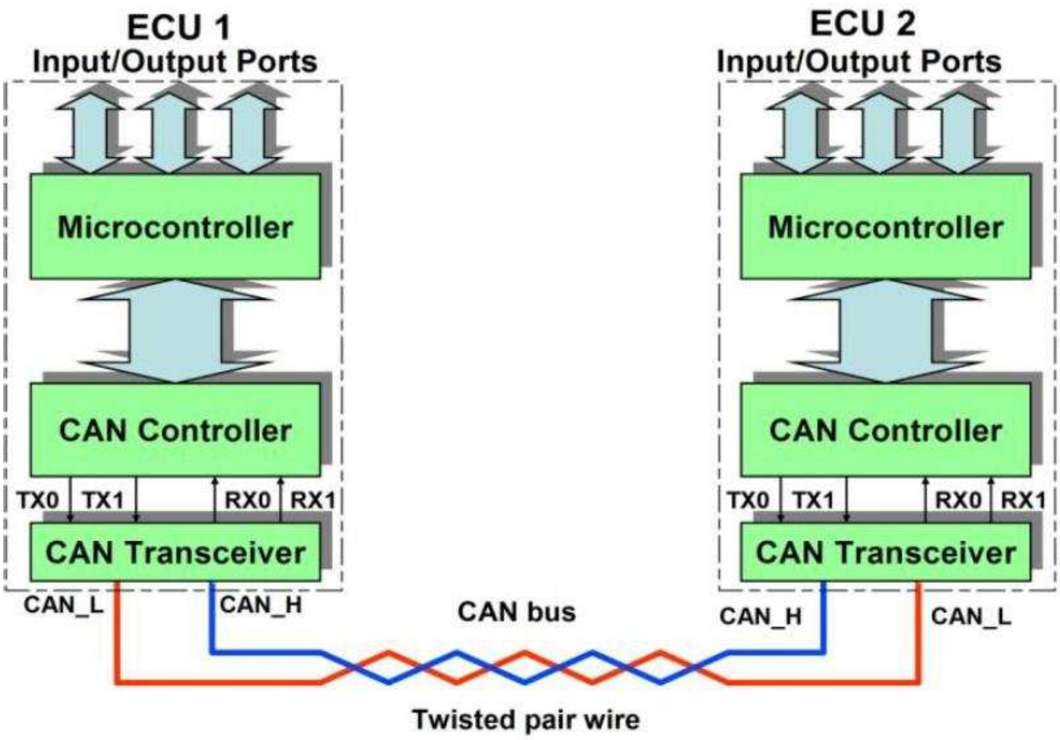


Figure 5. 7: ECU CAN structure [100]

5.3.3 CAN Bus protocol

A message containing data has to have a specific format following several specifications in order to be transmitted over the CAN bus [101]:

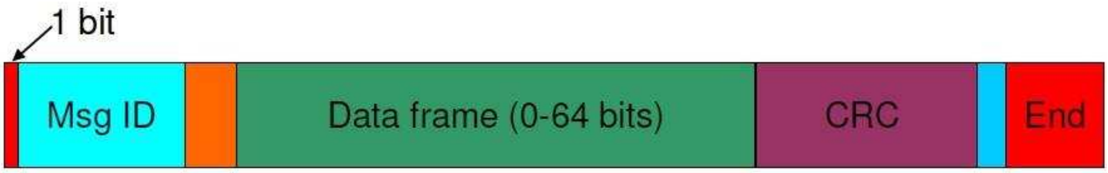


Figure 5. 8: CAN message layout [101]

1. Start of frame – Contains a dominant bit, which is used to synchronize all units on the bus.
2. Arbitration field – Contains a message identifier
3. Control field – Contains information on how big the message's data field is.
4. Data field – Contains the actual information that is being transmitted. Can range from 0 to 64 bits (with 8 bit increments).
5. CRC field – Contains information that allows the receiver to determine if an error has occurred in the transmission
6. ACK field – Contains a gap in the message where the receiving units can send a dominant bit if they have detected an error in transmission.
7. End of frame – Contains seven recessive bits, this effectively works as gap before the next message.

Here follow some more details on the most important parts of such a message:

5.3.3.1 Start of frame

To begin with all CAN units must be synchronised. The start of frame bit is used by all the units on the network to get synchronised. After being synchronised, every control unit knows when a bit is sent and can of course it can read it. Since, the majority of clocks in the ECUs is quartz based it is also temperature dependent and as a result they have to be frequently re-synchronised. This is done using a technique named bit stuffing and is performed in the 5 first fields in the message (Start of frame, Arbitration field, Control field, Data field and CRC field). In the bit stuffing technique, an opposite bit (0 or 1) is added after 5 identical bits while the message is transmitted (so 000000 becomes 000001 and 11111 becomes 111110). Practically, the ECUs can resynchronise themselves using any or every change of the bus state (0 to 1 or 1 to 0). The bit that was added at the end is removed by all the units and as a result the sent date are not compromised or affected [101].

5.3.3.2 Arbitration field

The arbitration field (known also as message ID) is sent in the beginning of a message. This field is consisted of 11 or 29 bits depending on the CAN protocol used. There is CAN 2.0A and CAN 2.0B, the first one has an 11 bit and the second one a 29 bit arbitration field with the most significant bit first (Figure 5.9). It is the message with the smallest value that has the highest priority [101].

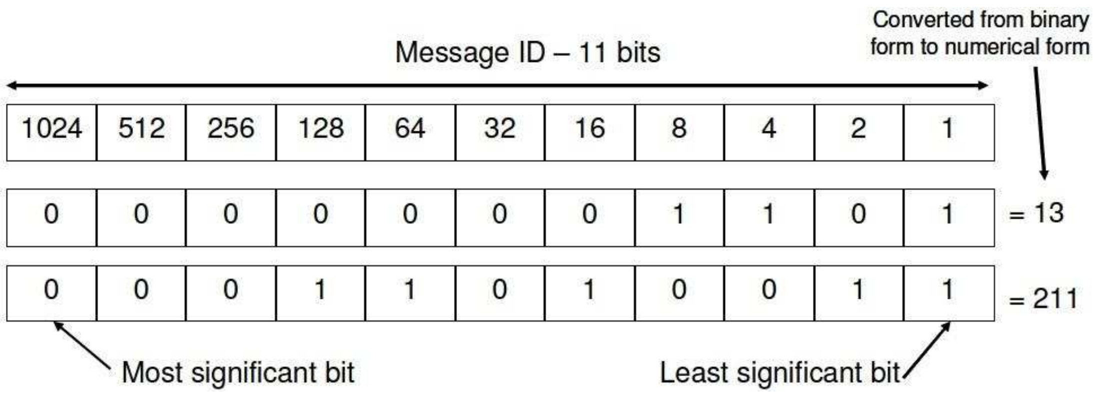


Figure 5. 9: CAN arbitration field. The message ID is sent in binary form. A message ID equal to 211 is transmitted by sending bits for 128+64+16+2+1= 211 [101]

5.3.3.3 Data field

In this field, the really useful information is included. The length of the information can vary everywhere up to 64 bits. Nevertheless, usually a 0 bit data field is used to inform about the status of an ECU (e.g. 0 bit data might mean ECU's failure) [101].

5.4 Fusing an eye-tracker in a vehicle

From the above sections, it is obvious that the eye-trackers to be installed in a vehicle could be categorised into two types; off the shelf products that can be installed in any car and inbuilt systems found as part of the safety features in a car (provided by the car manufacturer). The former are self-contained products, which are readily available in the market and do not have to join the CAN bus in order to function. As discussed before, they are basically mounted on the dashboard and interfaced with the power supply output. The second type of those systems is the one of interest in our case here. They could provide more elaborate features as they would be part of the vehicular electronics and information would be shared over the CAN bus with other modules. In such systems, the eye tracker could provide information such as gaze and head movement to the main safety system which could combine it with other safety assessment parameters such as steering wheel information, braking information, accelerometer information, lane tracking etc. This combination of inputs will allow the safety system to decide whether the driver is inattentive and whether any action needs be taken or not. All this information is passed over CAN-bus. Thus an eye-tracker could potentially become part of the safety features of the car.

Inagaki [102] has described a method of sensor fusion of various inputs to create a driver assistive system that can provide safety functions. Sensory information is received as depicted in the figure below:

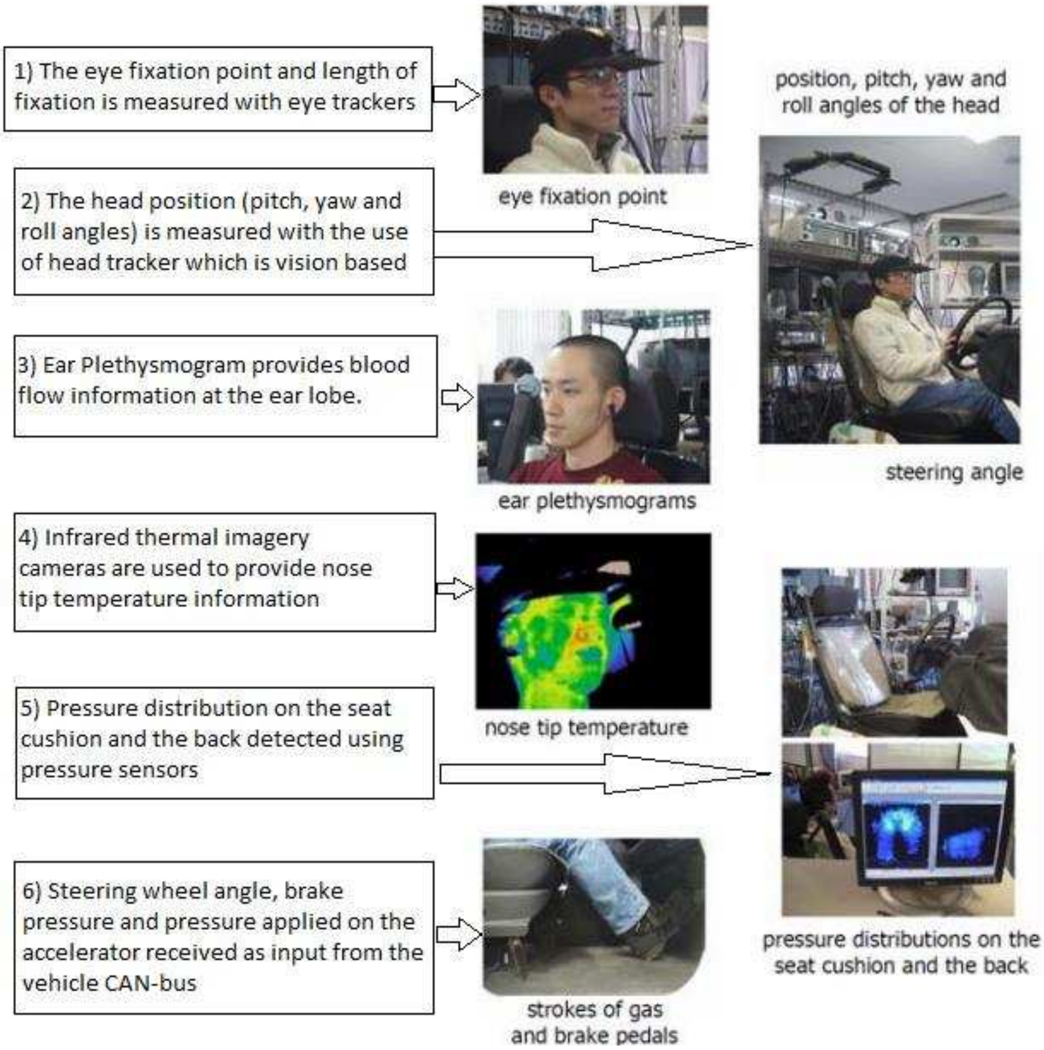


Figure 5. 10: Various sensory inputs to Driver assistive system [102]

(1) & (2) provide information about driver gaze, attention, blink rate and direction in which the driver is facing. (3) & (4) provide information about driver's mental workload. (5) provides information about the driver's body posture and (6) provides information about the reaction from the driver.

These sensory inputs, or some of them, could be used in algorithms in order to decide the state of the driver and what preventive/corrective actions should be taken by the vehicle or proposed to the driver.

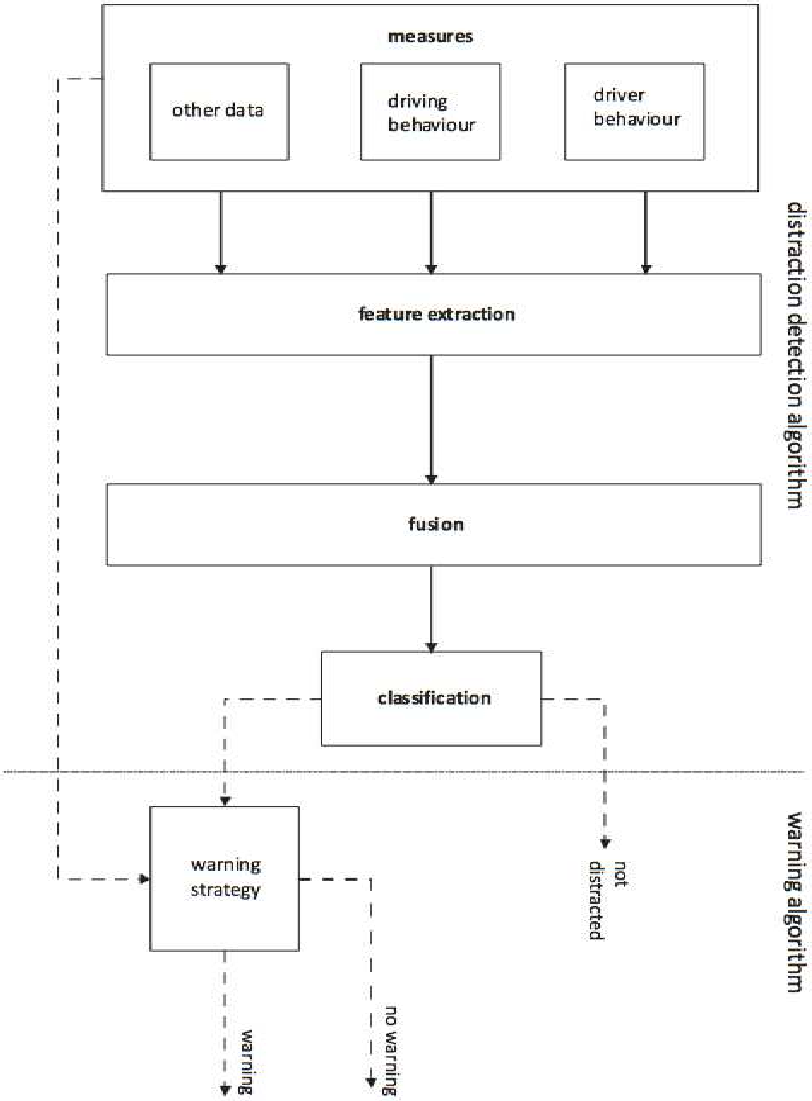


Figure 5. 11: Algorithm for Driver Fatigue Detection Systems [103]

Above, there is a standard schematic representing most driver detection algorithms and mitigation systems. The idea is based on getting information from the various sensors and feed them into a decision making algorithm that processes the information to produce the desired output in the form of the warning level. Until now, the main aim of these inbuilt systems is to alert the driver in time so that accidents can be avoided.

5.5 Further recommendation for an improved system

As seen until now, eye-tracking systems are practically used just as driver fatigue detection systems. In our case here, in which we are examining the possibility to take advantage more of the potentials this technology can offer, vehicular data fusion is needed in order to create safety systems that can help in accident avoidance. The efficiency of such system is dependent on the algorithm used and the end decision taken by the system. In order to make the system efficient, only audio alerts are not enough, since these can be ignored by the driver. A more efficient system could aim to take preventive action on its own, in case the driver is not vigilant. This poses many challenges, such as complexity of circuit/algorithm, issue with data interpretation etc. In addition, external factors such as environment variation, different driver profiles, different vehicle architectures and processing capabilities pose a major challenge in making a robust system of this kind.

Current implementations classify the scene as either dangerous or not, but in reality scene is a dynamic quantity, that is constantly changing and therefore the corrective action also needs a constant update. A further recommendation to improve the system may be to provide an algorithm that is more definitive in accident avoidance. For this, the analysed scene can be categorised into different levels of severity and corrective action can be taken based on the category to which the scene belongs. This categorisation can be updated constantly based on a feedback mechanism, to replicate the dynamic scene.

The central processing unit of the system gets useful input from the sensor technology such as gaze intent, head movement, number of blinks of the eyes, GPS location, traffic information etc. that are provided by various peripheral inputs installed in the car. The central unit can now make an informed decision as to how severe the warning level should be based on the danger posed at the scene. A decision matrix to aid such a decision can be based on the combination of inputs received as follows:

Inputs	Gaze intent	Head movement	Number of blinks	Lane tracking	Ultrasonic distance sensing
Gaze intent	1	1	2	3	4
Head movement	1	1	2	3	4
Number of blinks	2	2	2	3	4
Lane tracking	3	3	3	3	4
Ultrasonic distance sensing	4	4	4	4	4

Table 5. 1: Mitigation Strategy based on severity of input

Severity levels and suggested outcomes

Table 5.1 represents a dynamic learning algorithm that aims to categorise the warning levels. The interpretation of inputs is based on the studies described in the end of section 5.4 (Inagaki [102] (Figure 5.10)).

The scenes can be described as follows:

1) If the driver looks away from the road, such as while using the mobile phone or infotainment unit or distracted by a fellow passenger for a long duration (eyes off road scenario), then the severity level is low. In such a scenario, the alert can be in terms of

audio signalling alone, as no immediate threat of accident is detected. Therefore merely reminding the driver to remain focussed should be enough.

2) If the number of blinks has reduced (as the driver's eyes are shut) and the head movement is either still or in nodding motion, then the system can interpret that the driver has fallen asleep. Severity level 2 outcome can in terms of audio alert accompanied by seat vibration to wake up the driver. Here also, the system has not yet discovered a big threat of accident, but it is imminent that the driver remains vigilant at all times during driving.

3) If the driver is not driving in his lane, then the system can interpret that either the driver is distracted or asleep. This requires more warning, since it can be dangerous for the other cars on road. Severity level 3 outcome can be in terms of steering wheel correction along with level 2 outcome.

4) This is the most serious case and it will be needed in case the distance sensors detect an immediate warning level as well as the system detects that the driver is not attentive. Severity level 4 outcome must take control of the braking system and work in a constant feedback loop, to ensure that the controls are passed back to the driver once he becomes vigilant again.

Classification of warning levels and constant feedback from the scene ensure that the system caters to the dynamic features of the changing scene, whilst driving.

Chapter 6

Conclusion

As computers have become faster, the way we apply them becomes increasingly complex. This opens a wide range of possibilities, for using computers as a tool for enhancing the quality of life, learning human behaviour, and increasing the general safety. In reality, even today, eye-tracking is a technology with lots of its aspects in the making, and a bright new world is opening in front of us. To ensure the success of eye-tracking applications, wide accessibility is required. It was shown on this thesis that slowly but gradually, the eye-tracking technology is being part of the areas of interest of automobile companies.

Numerous approaches have been developed or proposed in order to introduce new safety standards concerning the driving procedure and it seems that in the near future the vehicles while have integrated eye-tracking equipment specialized and optimized for this kind of use. In this thesis an innovative approach is proposed based not only on the abovementioned technology but also on HUDs and the GPS devices which are already widely spread and their ability to send and receive data concerning the vehicle's movement. Thanks to the combination of those technologies we can achieve much better security during driving as the driver would be focused on the road easier and also notified in case another vehicle is going to come dangerously close in a short period of time. In the same time his concentration and focus would be continuously inspected alerting him in case he is not paying enough attention.

Nevertheless, we cannot expect this to happen without any problems. Firstly, the right hardware has to be found and optimized in order to be used for eye-tracking during driving. HUDs technology seems to be already adequate enough to perform all the tasks needed during driving but future advances might positively surprise us. In addition, the nature of the computer server that is going to compute the traffic has to be decided to make this approach possible. In addition, as the connected car is coming closer to reality security issues have to be taken into account. Those vulnerabilities and concerns were mentioned and examined in a study conducted by the Interactive Adverstising Bureau (IAB) of Spain with Applicantes, Motor.com and Kaspersky Lab [104]. In this study, which hopes to be the first step for implementing a minimum of unity to the “highly fragmented software ecosystem currently offered by manufacturers”, it is mentioned that on the one side new technologies offer to the drivers access to social networks, route calculation, in-car apps, etc. but on the other hand safety concerns about the communications and Internet services should not be ignored. These technologies offer lots of advantages to their users, but, in the same time, they could become a liability as privacy, software updates and applications focuses developed for usage in a car could become target of cybercriminals. Of course, all those hardware characteristics must lead to a new device with special specifications and designing it and making it work might be proved more difficult than expected. Even though there are still several issues to be resolved, the approach proposed here seems to be viable and applicable making driving a lot safer than before and opening a new era of eye-tracking in vehicular applications.

Bibliography

01. Delabarre, E., 1898. A method of recording eye-movements. *American Journal of Psychology,* p. 9(4):572–574.

02. R. Dodge and T.S. Cline, 1901. The angle velocity of eye movements. *Psychological Review,* p. 8(2):145–157.

03. Spivey, D. C. R. a. M. J., 2004. *Eye-tracking: Characteristics and methods (part 1); eye-tracking: Research areas and applications (part 2).* s.l.:Wnek. G. and Bowlin, G. (Eds.).

04. R. Argue, M. B. J. D. a. G. H., December 2004. *Building a low-cost device to track eye movement. Technical report report, Faculty of Computer Science,,* Dalhousie University: s.n.

05. M. S. Atkins, A. M. a. R. R., Jan 2007. *Understanding search and errors in studies. Technical Report. TR-2007,,* Simon Fraser University School of Computing Science: s.n.

06. H. B. Barlow, 1952. Eye movements during fixation. *J. Physiol.,* p. 116:290–306.

07. T. N. Cornsweet and H. D. Crane, 1973. Accurate two-dimensional eye tracker using first and fourth Purkinje images. *J. Opt. Soc. Am.,* p. 63(8):921–928.

08. Austin Roorda, M. C. B. B., n.d. [Online]
Available at: http://vision.berkeley.edu/roordalab/cslo.htm

09. M. Dorr, M. B. T. M. K. R. G. a. E. B., 2005. Eye movements on a display with gaze-contingent temporal resolution. *Perception ECVP 2005 Supplement,* p. 34:50.

10. S. Stevenson, G. K. a. A. R., 2007. Psychophysical and oculomotor reference points for visual direction measured with the adaptive optics scanning laser ophthalmoscope. *J. Vis.,* p. 7(9):137–137.

11. Rick Demeis, n.d. [Online]
Available at:
http://www.laserfocusworld.com/display_article/214660/12/none/none/News/Vision-systems-improve-road-safety

12. N. Edenborough, R. I. H. A. H. a. t. o., June 27–July 2 2004. *Drowsy driver monitor from delphi..* Washington, DC, USA, In Proc. IEEE Conference on Computer Vision and .

13. R. I. Hammoud, A. W. P. M. a. G. J. W., June 2005. *Efficient real-time algorithms for eye state and head pose tracking in advanced driver support systems..* San Diego, CA, USA, In IEEE Computer Vision and Pattern Recognition (CVPR), p. 1181.

14. Alper Yilmaz, O. J. M. S., 2006. Object Tracking. *ACM Computing Surveys.*

15. Xindian Long, O. K. T. a. A. K., Neuro Kinetics, Inc.. A High Speed Eye Tracking System with Robust Pupil Center Estimation Algorithm. *Department of Electrical and Computer Engineering, Carnegie Mellon University.*

16., 2009. [Online]
Available at: http://en.wikipedia.org/wiki/Eye_tracking

17. Jacob, R. J. K. &. K. K. S., 2003. Eye tracking in Human-Computer Interaction and usability research: Ready to deliver the promises. In: *The mind's eye: Cognitive and applied aspects of eye movement research.* Amsterdam: Elsevier, pp. 573-605.

18. Just, M. A. &. C. P. A., 1976. Eye fixations and cognitive processes. *Cognitive Psychology,* pp. vol: 8, pp: 441-480.

19. Goldberg, H. J. &. K. X. P., 1999. Computer interface evaluation using eye movements: Methods and constructs. *International Journal of Industrial Ergonomics,* pp. vol: 24, pp: 631-645..

20. Rayner, K. &. P. A., 1989. *The psychology of reading.* Englewood Cliffs, NJ: Prentice Hall.

21. Bruneau, D. S. M. A. &. M. J. D., 2002. *The eyes never lie: The use of eye tracking data in HCI research.* NY, ACM Press.

22. Brookings, J. B. W. G. F. &. S. C. R., 1996. Psychophysiological responses to changes in workload during simulated air traffic control.. *Biological Psychology,* pp. vol: 42, pp: 361-377.

23. Marshall, S. M. a. a. f. e. t. a. m. p. d. t. e. c. a., 2000. U.S.A, Patent No. 6,090,051.

24. Pomplun, M. &. S. S., 2003. *Pupil dilation as an indicator of cognitive workload in Human-Computer Interaction.* Mahwah, NJ, Lawrence Erlbaum Associates, pp. vol:3, pp: 542-546.

25. Goldberg, H. J. &. W. A. M., 2003. Eye tracking in usability evaluation: A practitioner's guide. In: *The mind's eye: Cognitive and applied aspects of eye movement research.* Amsterdam: Elsevier, pp. 493-516.

26. Poole A., B. L., 2006. Eye Tracking in HCI and Usability Research. In: *Encyclopedia of Human Computer Interaction.* London: Idea Group Inc., pp. 213-215.

27. Yu, L. E. M., 2004. *A New Methodology for Determining Point-of-Gaze in Head-Mounted Eye Tracking Systems.* s.l., s.n., pp. vol: 51, pp: 1765-1773.

28. T. Ohno, N. M. a. A. Y., March 2002. *Freegaze: a gaze tracking system for everyday gaze interaction..* s.l., In Eye Tracking Research and Applications Symposium, pp. 15-22.

29. X. Brolly and J. B. Mulligan, 2004. *Implicit calibration of a remote gaze tracker..* s.l., In IEEE Conference on CVPR Workshop on Object Tracking Beyond the Visible Spectrum.

30. J. Zhu and J. Yang, May 2002. *Subpixel eye gaze tracking.* s.l., In IEEE Conference on Automatic Face and Gesture Recognition, pp. 124-129.

31. J. Wang and E. Sung, 2002. *Study on eye gaze estimation, Man and Cybernetics.* s.l., IEEE Transactions on Systems,, p. 32(3):332{350.

32. D. Zhu, S. M. a. T. R., 1999. Robust pupil center detection using a curvature. *Computer Methods and Programs in Biomedicine,* p. 59(3):145{157.

33. Y. Matsumoto and A. Zelinsky, March 2000. *An algorithm for real-time stereo vision implementation of head pose and gaze direction measurement.* Grenoble, France, In IEEE Conference on Automatic Face and Gesture Recognition, pp. 499-504.

34. Dongheng Li, 2006. *Low-cost eye-tracking for human computer interaction,* Ames, Iowa: Iowa State University.

35. Dongheng Li, D. J. P., 2005. Starburst: A robust algorithm for video-based eye tracking. *Preprint submitted to Elsevier Science.*

36. J. Daugman, 1993. *High confidence visual recognition of persons by a test of statistical independence.* s.l., IEEE Transactions on Pattern Analysis and Machine Intellegence, pp. 15(11):1148-1161.

37. K. Nishino and S. Nayar, 2004. Eyes for relighting. *ACM SIGGRAPH 2004,* p. 23(3):704{711.

38. P. Burt and E. Adelson, 1983. A multiresolution spline with application to image mosaics. *ACM Transactions on Graphics,* pp. 2(4):217-236.

39. D. Hansen and A. Pece, 2005. Eye tracking in the wild. *Computer Vision and Image Understanding,* pp. 98(1):155-181.

40. Kobi Levi and Yair Weiss, 27 June-2 July 2004. *Learning Object Detection from a Small Number of Examples: the Importance of Good Features.* Washington, DC, USA, IEEE, Computer Society Conference on Computer Vision and Pattern Recognition, pp. II-53- II-60 Vol.2.

41. Rong Xiao, L. Z. H.-J. Z., 13-16 Oct. 2003. *Boosting Chain Learning for Object Detection.* Nice, France, IEEE, International Conference on Computer Vision, pp. 709-715 vol.1.

42. Yu-Ting Chen, C.-S. C., 26-28 Nov. 2007. *Cascading Rectangle and Edge Orientation Features for Fast Pedestrian Detection.* Washington DC, USA, IEEE, International Information Hiding and Multimedia Signal Processing, pp. 407 - 410, Volume 2.

43. M. Fischler and R. Bolles, 1981. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM,* pp. 24(6):381-395.

44. R. Hartley and A. Zisserman, 2000. *Multiple view geometry in computer vision.* Cambridge, UK: Cambridge University Press.

45. D. M. Stampe, 1993. Heuristic filtering and reliable calibration methods for video-based pupil-tracking systems. *Behavior Research Methods, Instruments, and Computers,* pp. 25(2):137-142.

46. Chen Shoushun, A. B. W. Y. a. D. M., JANUARY 2007. Adaptive-Quantization Digital Image Sensor for Low-Power Image Compression. *IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS,* pp. VOL. 54, pp 13-25.

47. Wang Xue, W. S. M. J., June 21 - 23, 2006,. *Multi-View Tracking in Wireless Sensor Networks.* Dalian, China, Proceedings of the 6th World Congress on Intelligent Control and Automation, pp. 5115-5119.

48. L. M. Bergasa, J. N. M. A. S. R. B. a. M. E. L., 2006. Realtime system for monitoring driver vigilance. *IEEE Transactions on Intelligent Transportation Systems,* p. 7(1):63–77.

49. Riad Ibrahim Hammoud, 2008. *Passive Eye Monitoring: Algorithms, Applications and Experiments.* s.l.:Springer.

50. J. B. Mulligan, 1997. Image processing for improved eye tracking accuracy. *Behavior Research Methods Instruments and Computers,* p. 29:54–65.

51. Y. Ebisawa and S. Satoh, 1993. *Effectiveness of pupil area detection technique using two light sources and the image difference method.* s.l., Proceedings of the 15th Annual International Conference of the IEEE, p. 1268–1269.

52. C. H. Morimoto, D. K. A. A. M. F. a. S. Z., October, 1999. *Keeping an eye for hci.* CAMPINAS, BRAZIL, XII BRAZILIAN SYMPOSIUM IN COMPUTER GRAPHICS AND IMAGE PROC., p. 171–176.

53. R. E. Haven, D. J. A. J. E. F. a. J. S. W., 2007. U.S.A/California, Patent No. 7280678.

54. Lund University, n.d. [Online]
Available at:
http://www.sol.lu.se/humlab/research/humlabResearch.html?fileName=et_sv.html&language=EN

55. LS460 achieves a world-first in preventative safety, 2006. *www.newcarnet.co.uk.* [Online]
Available at: http://www.newcarnet.co.uk/Lexus_news.html?id=5787

56. Hewitt D. Crane and Carroll M. Steele, 1985. Generation-V dual-Purkinje-image eyetracker. *Applied Optics,* pp. 24, 527-537.

57. Roger P. G. van Gompel, M. H. F. W. S. M. R. L. H., 2007. *EYE MOVEMENTS: A WINDOW ON MIND AND BRAIN.* Amsterdam: Elsevier.

58. Duchowski, 2002. A breadth first survey of eye-tracking applications.. *Behavior Research Methods, Instruments, and Computers.*

59. Edmund Donges, 1978. A two-level model of driver steering behavior. *Human Factors, 20,* p. 691–707.

60. Land M. F. & Lee D. N., 1994. Where we look when we steer. *Nature 369,,* pp. 742-744.

61. Michael Land & Julia Horwood, 1995. Which parts of the road guide steering?. *Nature 377,* pp. 339-340.

62. Zwahlen, H. T., 1993. Eye scanning rules for drivers: how do they compare with actual observed eye-scanning behavior?. *Transportation Research Record, 1403,,* p. 14–22.

63. Underwood, G. C. P. C. D. C. S. &. W. R., 1999. The visual control of steering and driving: where do drivers look when negotiating curves. In: *Vision in Vehicles VII.* Amsterdam: Elsvier, p. 245–252.

64. Land, M. F., 1998. The visual control of steering. In: *Vision and action.* s.l.:Cambridge University Press, pp. 163-180.

65. Summala, H. N. T. &. P. M., 1996. Maintaining lane position with peripheral vision during in-vehicle tasks. *Human Factors, 38,* p. 442–451.

66. Mourant, R. R. &. R. T. H., 1970. Mapping eye-movement patterns to the visual scene in driving: An exploratory study. *Human Factors, 12,* pp. 81-87.

67. Tech Ascension, n.d. [Online]
Available at: http://www.ascension-tech.com/docs/pdf/asl_simulator.pdf

68. DENSO-Europe, n.d. [Online]
Available at: http://www.denso-europe.com/Image-Recognition-and-Eye-Tracking--1014050000000001.aspx

69. Pertech, n.d. [Online]
Available at: http://en.pertech.fr/

70. Nakayasu H., S. Y. Y. T. M. T., 2009. Visual perception and response behavior by driving simulator and eye tracking system. *Advances in Transportation Studies an international Journal Section B 19.*

71. Donald Melanson, n.d. [Online]
Available at: http://www.engadget.com/2009/05/25/toshiba-shows-off-in-car-facial-recognition-system/

72. eyecomcorp.com, n.d. [Online]
Available at: http://eyecomcorp.com/military/

73. Noldus, n.d. [Online]
Available at: http://www.noldus.com/human-behavior-research/products/the-observer-xt

74. Seeing Machines, n.d. [Online]
Available at: http://www.seeingmachines.com/

75. Tobii, n.d. [Online]
Available at: http://www.tobii.com/oem_technology/automotive.aspx

76. Zamora Nicholas, K. J.-C. M. R., Carnegie Mellon University. Distributed Power-Management Techniques for Wireless Network Video Systems. *Department of Electrical and Computer Engineering.*

77. Bulling, A. & Roggen, D. a. T. G., 2009. Wearable EOG goggles: Seamless sensing and context-awareness in everyday environments. *Journal of Ambient Intelligence and Smart Environments (JAISE),* pp. 157-171.

78. Charles Strangor, Beginning Psychology, 2012.

79. D. A. Huffman, 1952. *A method for the construction of minimum-redundancy codes.* s.l., s.n., pp. 1098-1101.

80. T. H. Cormen, C. E. L. a. R. L. R., 1990. *Introduction to Algorithms.* s.l.:MIT Press.

81. A. S. Tanenbaum, 2001. *Modern Operating Systems.* s.l.:Prentice-Hall.

82. D. Le Gall, 1991. *MPEG: a video compression standard for multimedia applications, 34(4).* s.l., s.n., p. 46–58.

83. G. Buchsbaum, 1980. *An analytical derivation of visual nonlinearity.* s.l., s.n., pp. 237-242.

84. Francesco Camastra, A. V., 2007. Image and Video Acquisition, Representation. In: *Machine Learning for Audio, Image and Video Analysis: Theory and Applications.* s.l.:Springer, pp. 51-80.

85. http://eyecomcorp.com, n.d. *3 Ways Eye Tracking Can Save Lives Behind the Wheel.* [Online]
Available at: http://eyecomcorp.com/eye-tracking/3-ways-eye-tracking-save-lives-wheel/
[Accessed 30 August 2012].

86. Alexandru Radu Chis, 2010. [Online]
Available at: http://www.autoevolution.com/news/gms-full-windshield-hud-technology-explained-18454.html
[Accessed 30 August 2012].

87. Bill Howard, 2012. [Online]
Available at: http://www.extremetech.com/extreme/124847-bmws-full-color-hud-distraction-minimizer-of-distractions-or-costly-tech-gadget/2
[Accessed 23 September 2012].

88. Shane McGlaun, 2012. [Online]
Available at: http://www.slashgear.com/gmc-talks-heads-up-display-technology-for-vehicles-30220691/
[Accessed 28 September 2012].

89., n.d. [Online]
Available at: http://www.theautochannel.com/news/2011/11/29/016446-tobii-paves-way-

for-use-eye-tracking-driver-safety.html
[Accessed 29 August 2012].

90. Tobii, 2012. [Online]
Available at: http://www.tobii.com/driver
[Accessed 24 September 2012].

91. Ethem Alpaydin, 2010. Machine Learning 2nd edition. In: Cambridge, Massachusetts: The MIT Press.

92. D.J. Mostow, 1983. Machine transformation of advice into a heuristic search procedure. In: *Machine Learning, An Artificial Intelligence Approach.* Berlin: Springer, pp. 367-404.

93. J.G. Carbonell, 1983. Learning by analogy: Formulating and generalizing plans from past experience. In: *Machine Learning, An Artificial Intelligence Approach.* Berlin: Springer, pp. 137-162.

94. Sabri Bayoudh, L. M. A. D., 2007. *Learning by analogy: a classification rule for binary and nominal data.* Hyderabad, India, Morgan Kaufmann Publishers Inc. San Francisco, CA, USA, pp. 678-383.

95. Hao Nai Industrial Co., L., n.d. [Online]
Available at: http://www.care-drive.com/theme_detail.php?id=8
[Accessed 20 February 2014].

96. Attention Technologies, n.d. [Online]
Available at: http://www.phdassessments.com/en/docs/DriverFatigueMonitor.pdf
[Accessed 16 February 2014].

97. Laurence Hartley, T. H. &. N. M., 2000. *REVIEW OF FATIGUE DETECTION AND PREDICTION TECHNOLOGIES,* Virginia, USA: National Road Transport Commission: Institute for Research in Safety & Transport.

98. Bijal Parikh, n.d. [Online]
Available at: http://www.engineersgarage.com/article/what-is-controller-area-network
[Accessed 28 February 2014].

99. Volkswagen-Audi, n.d. [Online]
Available at: http://www.volkspage.net/technik/ssp/ssp/SSP_238.pdf

100. Richard McLaughlin, n.d. [Online]
Available at: http://www.warwickcontrol.com/ata/Infobackup_and_library/Basicbus.pdf

101. Henning Olsson, n.d. [Online]
Available at: http://www.optimumg.com/docs/CANTechTip.pdf

102. T. Inagaki, 2008. *Smart collaboration between humans and machines based on mutual understanding.* Tsukuba, s.n.

103. Ahlström C. & Kircher K., 2010. *Review of Real-time Visual Driver Distraction Detection Algorithms.* Eindhoven, s.n.

104. Kaspersky Lab [Online]
Available at: http://www.kaspersky.com/about/news/events/2014/Connected-cars-are-now-a-reality-but-are-they-secure
[Accessed 20 July 2014].

105. C. Cortes and V. Vapnik., *Support-vector networks. Machine Learning, 20:273–297, 1995.*

106. Zhiwei Zhu, Qiang Ji., 2005. *Robust real-time eye detection and tracking under variable lighting conditions and various face orientations. Computer Vision and Image Understanding - Special issue on eye detection and tracking, 98: 124-154.*

107. Ovidiu Ivanciuc. A*pplications of Support Vector Machines in Chemistry. In: Reviews in Computational Chemistry, Volume 23, Eds.: K. B. Lipkowitz and T. R. Cundari. Wiley-VCH, Weinheim, 2007, pp. 291–400.*

108. J. Huang, D. Li, X. Shao, and H. Wechsler., 1998*. Pose discrimination and eye detection using support vector machines (SVMs). In Proceeding of NATO-ASI on Face Recognition: From Theory to Applications, pp 528–536.*

109. Su Yeong Gwon, Chul Woo Cho, Hyeon Chang Lee, Won Oh Lee and Kang Ryoung Park., *2013*, *Robust Eye and Pupil Detection Method for Gaze Tracking*. Int J Adv Robot Syst.

110. Veenman, C., Reinders, M., and Backer, E. 2001. Resolving motion correspondence for densely moving points. IEEE Trans. Patt. Analy. Mach. Intell. 23, 1, 54–72.

111. Comaniciu, D., Ramesh, V., and Meer, P. 2003. Kernel-based object tracking. IEEE Trans. Patt. Analy. Mach. Intell. 25, 564–575.

112. Yilmaz, A., Li, X., and Shah, M. 2004. Contour based object tracking with occlusion handling in video acquired using mobile cameras. IEEE Trans. Patt. Analy. Mach. Intell. 26, 11, 1531–1536.

113. Ali, A. and Aggarwal, J. 2001. Segmentation and recognition of continuous human activity. In IEEE Workshop on Detection and Recognition of Events in Video. 28–35.

114. Cootes, T., Edwards, G., and Taylor, C. 2001. Active Appearance Models. IEEE Trans. Patt. Analy. Mach. Intell. 23, 6, 681–685.

115. Greenspan, H., Belongie, S., Goodman, R., Perona, P., Rakshit, S., and Anderson, C. 1994. Overcomplete steerable pyramid filters and rotation invariance. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 222–228.

116. Bowyer, K., Kranenburg, C., and Dougherty, S. 2001. Edge detector evaluation using empirical roc curve. Comput. Vision Image Understand. 10, 77–103.

117. Barron, J., Fleet, D., and Beauchemin, S. 1994. Performance of optical flow techniques. Int. J. Comput. Vision 12, 43–77.

118. Blum, A. L. and Langley, P. 1997. Selection of relevant features and examples in machine learning. Artific. Intell. 97, 1-2, 245–271.