Eye Gaze Tracking for Tracking Reading Progress

AALBORG UNIVERSITY
DENMARK

Department of Electronic Systems
Vision, Graphics and Interactive Systems
Master Thesis

Spring 2013
Julia Alexandra Vigo

Department of Electronic Systems
Fredrik Bajers Vej 7
DK-9220 Aalborg Ø
http://es.aau.dk

Title:
Eye Gaze Tracking for Tracking Reading Progress

Theme:
Vision, Graphics and Interactive Systems

Project Period:
Spring Semester 2013

Project Group:
13gr1050

Participant(s):
Julia Alexandra Vigo

Supervisor(s):
Zheng-Hua Tan
Morten Højfeldt Rasmussen

Copies: 3

Page Numbers: 100

Date of Completion:
June 3, 2013

Abstract:

The purpose of this project is to follow the reading process of subjects thanks to an eye gaze tracker. To do that, tests have been conducted on dyslexic persons. For these tests, we use an eye gaze tracker with an already implemented reading tutor using speech recognition. The baseline of this project is to map gaze data points to the corresponding words in the text. Algorithms are implemented in order to compensate issues that are inherent when using eye gaze tracking for reading process.
To evaluate the accuracy of the system, the tracking error rate is calculated. We use as references the manual transcriptions of the speech recorded during the experiments and as hypothesis the gaze data points mapped.
The resulting tracking error rate is 114.6% which is really high. But when using speech recognition fused with eye gaze tracking for the reading tutor, the accuracy is better. Eye gaze tracking actually improves the already implemented reading tutor using only speech recognition.

Preface

This report documents the master thesis project entitled *Eye Gaze Tracking for Tracking Reading Progress*. The project was carried out during the 10th semester of specialization *Vision, Graphics and Interactive Systems* under the Department of Electronic Systems at Aalborg University in Spring 2013.

The report is divided into five parts plus appendices: *Introduction*, *Analysis*, *Design and Implementation*, *Evaluation* and *Conclusion*. The first part introduces the project. Analysis of eye gaze tracking and reading is done in the following part. A description of the design of the project and its implementation is done in the third part. The fourth part evaluates the performance and accuracy of the system and the last part concludes the whole project.

The syntax [number] indicates references to secondary literature sources. The numbers refer to the alphabetically sorted bibliography found at the end of the report, just before the appendices.

I would like to thank my supervisor at Aalborg University Zheng-Hua Tan for supporting me in this challenging project, as well as the PhD student and my other supervisor, Morten Højfeldt Rasmussen, for helping me through this master thesis.

A CD is attached to this report which includes:

- Source code of the developed program.
- PDF file of this report.

Aalborg University, June 3, 2013

Julia Alexandra Vigo
<jvigo12@es.aau.dk>

Abbreviations

API	Application Programming Interface
ART	Automatic Reading Tutor
ASR	Automatic Speech Recognition
ECS	Eye Contact Sensor
EOG	Electrooculography
FSMs	Finite State Machines
GUI	Graphical User Interface
HCI	Human Computer Interface
RSI	Repetitive Strain Injury
RFLs	Required Fixation Locations
SDK	Software Development Kit

Contents

Part I

Introduction

Contents

The main motive of this project is to use eye gaze tracking to follow reading processes. First, the motivations behind this project will be explained. Then the problem formulation of the project will be done.

Chapter 1

Motivations

In the last decade, we have witnessed a real *boom* of the computer devices and their use. It is now common to find at least one computer per household, plus other kinds of devices such as smartphones, tablet computers and others. These technologies become more and more affordable for the average consumer.

Moreover the computer performances have been increased exponentially along the years. Systems such as eye gaze tracking that were only able to run on high performance computers can now run under any computer. An average person with a webcam and a computer is able to develop an eye gaze tracking system on its own. This is why a lot of eye gaze tracking systems can be found on the internet like in [17], [18] and [30].

This increase in the number of computers by person pulls the increase of time interacting with such devices along. With the quite infinite possibilities of the computers, the interaction between humans and computers is a limiting factor. Computers need to understand what humans want in order to have the best interaction possible. People started searching for new ways to interact with computer; more efficient, more intuitive and easier. Nowadays, computers are equipped with webcams, microphones, speakers, high resolution displays and touch sensitive surfaces. All these inputs lead to new interactions and one can be eye gaze tracking. Most of the time eye gaze is the only input but it can be combined with other modalities [6].

Interaction with computer can be done all day long at work or at home for leisure activities. And interaction is mostly done with keyboard and mouse. But at the end of the day, some people can suffer from over stressing hands or part of it. Eyes are indeed a good solution; they allow to free the hands when interacting with them and they move anyway in all situations [6]. Furthermore, eyes are really important in communication (it will be more detailed in Chapter 3).

Combining eye gaze tracking with reading can produce tools helping people. We would be able to detect people having reading troubles, to help children increasing their skills in reading, to help disabled people, to help people with other languages and much more. In order to do that, reading paths must be studied and the eye gaze tracking system must have a great accuracy.

For this project, we want to use eye gaze tracking to follow reading process for an automatic reading tutor application. This application is predetermined for dyslexic persons; who have by definition reading issues. It is detailed in the Chapter 2.

Chapter 2

Problem formulation

Eye gaze tracking will be used with an already implemented application, an Automatic Reading Tutor (ART), in order to improve the actual accuracy of the application.

The Automatic Reading Tutor application has been realized by Jakob Schou Pedersen, in [19]. It is based on Automatic Speech Recognition (ASR). The purpose is to realize a tool to help dyslexic persons with their reading.

The application displays some sentences to read for the dyslexic persons and when they are reading these sentences, the ASR detect the words correctly read along with the one misread. This way, a feedback is given and the dyslexic persons are able to see their mistakes and to train in order to get better.

The accuracy with only the ASR can be improved and the fusion of eye gaze tracking with ASR is supposed to give better results. The purpose of using eye gaze tracking is to estimate the eye gaze positions and to map these positions to the closest words. This way, the system outputs the words the person is looking at. Tests will be conducted on dyslexic persons with ASR and eye gaze tracking to have data to work on and to calculate the accuracy of the system.

The eye gaze modality in itself should be reliable in order to improve the whole application. The whole purpose of this project is to understand reading process and to implement algorithms that are able to give reliable eye gaze data. Some algorithms have already been implemented but they may not be enough to get reliable data. The real challenge will be to create and implement algorithms able to improve the accuracy of the eye gaze tracking.

The main issue that we will encounter while working with eye gaze tracking and reading activity is the following: the accuracy of the eye gaze tracker. This issue is detailed in the next section.

Eye gaze tracker accuracy

The main problem with eye gaze trackers is their accuracy and their reliability. Depending on the kind of system used, the accuracy changes. What we want is the position of our gaze to distinguish the words we are looking at, so it is mandatory to have a good reliability. The way of detecting the eyes, calculating the gaze position, calibrating and the environment have an influence on the accuracy of the tracker.

Using the eye gaze tracker to study reading activities requires having an even better accuracy because the system should be able to detect which line and which word on this line the user is looking at; even when the fonts are small.

The aim of this project is to follow the reading paths of subjects and to identify which word is being look at with the best accuracy possible.

Part II

Analysis

Contents

Firstly, an analysis of eye gaze tracking systems and a survey of the work done in the area of eye gaze tracking will be done. Secondly, the process of reading will be analyzed and then the Dyslexia condition will be explained. Then a state-of-the-art of eye gaze tracking used for reading process will be detailed. Issues coming along with such systems will be described. Finally, the specification of the requirements will be done.

Chapter 3

Eye Gaze Tracking Analysis

A review of the importance of eye gaze tracking systems in different areas will be done. A review of the technology behind eye gaze tracking systems will also be done, along with a state-of-the-art of already existing technologies.

3.1 Eye Gaze Tracking

Eye gaze tracking means the tracking of the approximation of the eye gaze direction of a user. Most of the time, the approximation of the eye gaze direction of a user means that we try to identify which object a user is looking at. In this case, with a computer screen, we try to find the coordinates of the screen the user is looking at.

Depending on the method used to track the eye gaze, the device has different degrees of freedom. The most simple eye tracking systems give the direction of the eyes based on the head (mounted head systems) or based on a fixed position of the eyeball (systems with head fixation). More complicated systems allow the user to move their head in front of a fixed device. These systems implicitly track the head. The eye gaze trackers that are wearable in order to be used in 3D world, have to track the direction of the gaze not only based on the head but also in space. It would be more suitable to use the term *eye and head tracker*, than using the term *eye gaze tracker* in this case [6].

Working with video based eye gaze trackers allows to get the gaze direction but also the size of the pupil. When the pupil becomes wider or when the pupil narrows, it indicates an emotional response to the scene being looked at. It is quite interesting for research to look into that; but the change of the size of the pupil can be also caused by a change in lightning. If a large amount of light enters the pupil, then the pupil narrows and on the contrary if the pupil does not receive enough light, it becomes wider. This kind of work focuses more on the size of the pupil than on the gaze direction; it is another interpretation of the term *eye gaze tracking* [6].

The term *eye gaze tracking* includes also some works that track the whole eye, focus on the shape of the eye (even when it is closed) and contains the eyebrows in the eye tracking. Consequently, the term *eye gaze tracking* represents a lot of different types of eye tracking [6].

3.1.1 History

This section presents an overview of researches that have been done on eye gaze tracking; but presents also the first time eye gaze tracking was used.

One of the first time eye gaze tracking has been used was in 1947 by Fitts et al. [9], after the second world war. In technical reports published in the late 1940s, they did some researches and were considered as pioneers in the field of visual sampling. During these researches, the largest collection of eye movement data was collected for a visual monitoring task. The data collected represent 500,000 frames of video sequences of more than 40 pilots while flying with different flight conditions [16].

A general conclusion is made in [9] : *"It is reasonable to assume that frequency of eye fixations is an indication of the relative importance of that instrument. The length of fixations, on the contrary, may more properly be considered as an indication of the relative difficulty of checking and interpreting particular instruments. [...] If we know where a pilot is looking, we do not necessarily know what he is thinking, but we know something of what he is thinking about."*

With this work done, Fitts et al. offered a better disposition of instruments and were able to find the instruments that were too difficult to read, so that the instrument can be redesigned. For the first time, there was an interaction between the airplane cockpit (an application) and an eye gaze tracking system (which was manual). Video was also used for the first time to realize measures. The method used to perform these researches is based on electrodes placed around the eyes to register the movements of the eyeball. It is a medical technique. This kind of technique requires the head to be motionless while eyes are being tracked and the devices used are obtrusive [16].

The invention of head mounted eye trackers is one of the major innovation in the field of eye gaze tracking. The head mounted eye tracker is a technique still greatly used [16]. The Figure 3.1 shows a typical head mounted eye tracker.

Another precursor work in the area of eye gaze tracking has been done by Yarbus, a Russian psychologist. This psychologist studied in the 1950s and the 1960s the movements of the eyes as well as the saccadic exploration of complex images. The movements of the eyes were recorded while observers viewed ordinary scenes and objects. This researches show once again that gaze direction is fundamental in interactivity. Yarbus actually demonstrated that depending on the task that is asked to the observers, the gaze trajectories vary. The eyes focus on the part of the images

Figure 3.1: ASL 501 Head Mounted Optics from Applied Science Laboratories [26]

that are relevant to the questions asked [16]. For example, in the Figure 3.2, the image A can be seen in two ways, it can be seen as a musician or as a woman. If we ask the user to look at the musician, the eye gaze trajectories will be as in the image B. However, if we ask the user to look at the woman, the eye gaze trajectories will be as in the image C.

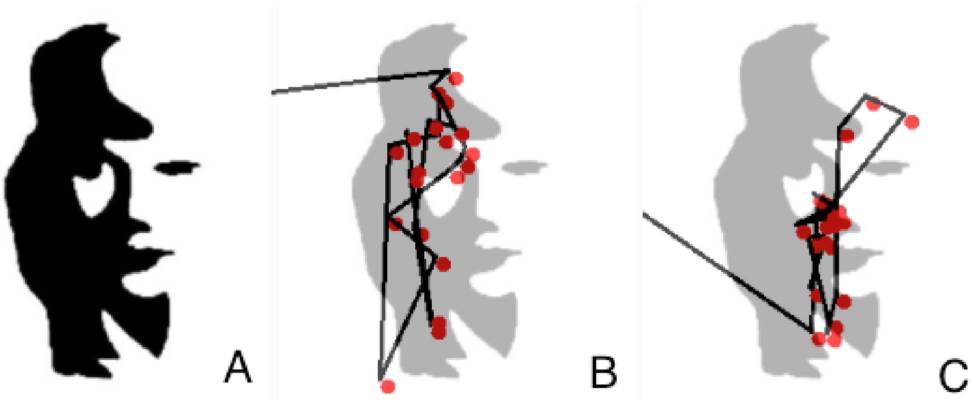

Figure 3.2: Eye gaze trajectory variations

In the 1970s, all the relevant work done in the field of eye gaze tracking was about getting a better accuracy thanks to technical improvements and getting less intrusive devices. Since this period, all that had been done was extremely correlated to the computer performances. The more the computer is powerful, the more it is possible to do complicated tasks as real time tasks or complex applications. Nowadays, it is possible to realize an application using eye gaze tracking for HCI, for example; it was impossible before [16].

In the 1980s, gaze tracking still aroused interest; the amazing *boom* of personal computers makes it possible to re-think our relation with computer but also to design new interfaces. A precursor in the field of interactive applications is Dr. Levine. He was one of the first one to see the possibilities of eye gaze tracking for interactive applications [16].

For 15 years, a lot of manufacturers have created and developed products in the area of eye gaze tracking because it has become an industrial stake. Logically, the number of papers about this subject has incredibly increased and it is complicated to do an exhaustive overview of all that has been done. Actually, many conferences, journals and publications have been created about eye gaze tracking. In the present days, a lot of systems (obtrusive or not), make it able to measure, track, analyze many data that are coming from the video cameras and this in real time. From advertisement to military analysis and via medical applications also, the possibilities are numerous [16].

3.1.2 Eye Gaze and Communication

Gaze is one of the main keys to communication. Most of the time, when we are asking something, the object of the question does not need to be mentioned if we are looking at it while asking the question. For example, if we ask *what is that ?* while looking at a specific object, there is no need to specify what is the object; assuming the person asked is aware of what we are looking at [6].

A pioneer in research about gaze and interaction with gaze, Bolt, wrote in [2] about the importance of the communication through gaze. He explained it with this example: *"Consider the case of asking the question "What is your favorite sport?" in the presence of several people, but looking at Mary, say, not at Frank, Judy, or Dave. Then you utter the selfsame words, but now looking at Dave. The question is a different question; the difference lies not in its verbal component, but in its intended addressee given by eyes."*

In this example, we can observe that by simply looking at a person, communication is already established. Most of the time, people start by looking at a person to address him or her. Another example is that if we stare at a person for too long and for no reason, this is not polite. These examples show that our eyes are not here only for vision but also for communication and it is our responsibility to handle what we are looking at [6].

If we compare human eyes with animals eyes, the main difference is the absence of white around the pupil in the animal eyes (see Figure 3.3). No white eyeball is visible in animal eyes, in particular concerning mammals, even if the eyes for both human and mammals work in a similar way. Because of the absence of white in the animal eyes, it becomes more difficult to find out what animals are looking at. Based on that, we can wonder if knowing gaze direction have been helpful in the development and evolution of the humans [6].

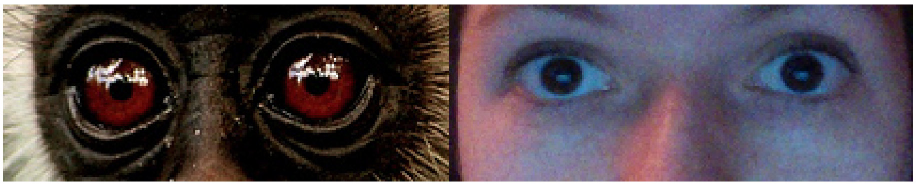

Figure 3.3: Comparison of Animal eyes (left) and Human eyes (right)

Studies have been made, and in one of them, Vertegal et al. explained that there is a high probability that in a discussion, the person listening ($p = 88\%$) or the person spoken to ($p = 77\%$) is also the person that is looked at [29].

In [8], Eibl-Eibesfeldt notices that a communication channel can be created just by looking at someone; but depending on our culture, this can be interpreted as an act of aggression. The cultural background also seems to play a role in the movement of the eyes. In [3], a study is conducted to find the differences in scene perception between Americans and Chineses. The result of the study is that the Americans are looking at the foreground objects longer and sooner while the Chinese are more looking at the background. The formation of a person is also taken into account for the movement of the eyes. People reading different texts in different languages have a different eye response: it depends if the person speaks the language that they are reading, if the person is learning it or if they do not speak the language at all. Eye motions imply high levels of cognition and are based on the intent of a person. Even if there are universal aspects as biology, eye motions rely a lot on personal characteristics. *Every person sees the world with their own eyes* [6].

3.1.3 Eye Gaze and Human Computer Interface

Using eye gaze interface is not so much spread unless it is really necessary; the only users of this technology are disabled people who cannot move anything else than their eyes. These interfaces are realized to control the computer only with gaze. And these systems work very fine. But for a person with no disability, it seems that using eye gaze for interaction is less efficient than other modalities (keyboard, mouse, speech interaction), even if eye motions are fast and if looking is an easy task [6]. The following two sections present a list of advantages and disadvantages of eye gaze interfaces.

Advantages

This section presents the advantages of using eye gaze tracking for HCI.

Ease of use Using eye gaze interaction would allow to free the hands and by consequence not to overstress the hand muscles. Plus it would not add some load to the eye muscles because eyes move anyway without any interaction constraint. A simple example shows that when using the mouse to click on a button, the eyes follow the movement of the mouse on the screen in any case [6].

Faster interaction Eye movements are fast and using them to interact would be fast as well. But nowadays most of the interactions with eye gaze, as eye typing systems for instance, is slower than with ordinary inputs (keyboard for example). Combining eye gaze and another modality would allow to speed-up the interaction [6].

No maintenance The eye tracking systems based on video do not need interaction or contact: that is why maintenance is not mandatory. Unlike mice and keyboards, the eye tracking devices do not need to be cleaned, which becomes a real problem for mice and keyboards. For example, a glass or a strong transparent material can be placed in front of the camera and this way, it becomes vandalism-proofed; which is not possible for keyboards and mice [6].

Hygienic interaction Another advantage of eye gaze tracking systems is their hygienic interface. Because of the absence of contact needed with such systems, it would fit perfectly in environments with high hygienic requirements. For example, in an operation room for surgery, eye gaze systems would fill the requirements; there is indeed no need of touch interaction with the systems. It can also be useful for public interfaces when there is epidemic menace and when there is a need for hygienic interaction [6].

Remote control Nowadays, with the technology, it is possible to have remote control with eye gaze tracking systems. Detecting the eyes over meters of distance is now possible with the lenses of the camera and its high resolution. It is even possible to detect the eyes at one meter with low cost eye gaze tracking systems [6].

Interaction certified Using eye gaze interaction certifies the presence of a user in front of the webcam as well as the attention of this user. For example, if we use a cellphone with eye gaze interaction, we do not fear to call someone by accident while the cellphone is in our pocket. Eye gaze interaction can require some specific behaviour of the user; for instance, for the user to go on with further features, it can be asked to read a warning text [6].

Users activity detailed The eyes reveal a lot about someone activities; tracking them gives useful information about what the user is doing. Without further data analysis, an eye gaze tracking system provides information about what the eyes are looking at, and this is already of big potential for context-aware systems. With simple data analysis, it is possible to detect whether a user is reading or doing other activities for example. With further data analysis, it is possible to detect emotional or physical condition of the user, as their level of literacy and their age [6].

Disadvantages

This section presents the disadvantages of using eye gaze tracking for HCI.

Eyes control Because eyes are what we use for vision, it is common that eyes do unconscious movements. This can mess the system that uses eyes as input. Controlling our eye motions and at which level we are able to do it are two components not well-known. By controlling the eye motions, we mean to remove the unconscious movements of the eyes to only keep the intended movements. The only thing we know is that we are able to control where we look because it is a requirement for our social life. The unknown variable is whether we are able to train our eye muscles as well as we train our fingers in order to play guitar for example [6].

Midas Touch Because once again we use our eyes for vision, using them for computer interaction can cause conflicts. This conflict is called the Midas Touch problem. The main problem is to know if our eyes are just looking around to gather information on our environment or if they are invoking an action for interaction. If we cannot distinguish these two kinds of action; it can cause unwanted action for interaction. The interface has to distinguish the intentional eye movements and the natural ones. There is also the question of the blinking objects and the moving objects that can result in conflicts as well. For example, a blinking pop-up on a web page can disturb our eye movements. It is still a topic for researchers [6].

Eyes muscles overstressed Using to such an extent the muscles for specific actions causes damage and it is known as repetitive strain injury (RSI). The same thing happening to the eye muscles is a real fear. This fear is justified, but because the eyes move without any break, even in our sleep, it seems that it is not a problem [6].

3.1.4 Eye gaze and other Applications

Historically, eye gaze tracking was used for a interface design application (as seen previously). The knowledge of what a person is looking at is very important to design cars, devices, cockpits and others. Studies about these fields are based on the hypothesis that what a user is looking at is supposed to show the thought that is on the top of the stack of cognitive processes. It is very useful for designing and improving interfaces because the meaningfulness, visibility and placement of the elements of the interface can be estimated. Eye gaze tracking is used a lot for HCI researches but only some applications have been made for consumers; because of the midas touch problem. The application is supposed to react only at the right time in appropriate situations and not every time the direction of our gaze changes [16].

Disabilities

Even if the commercial applications with eye gaze tracking are not common, a field where eye gaze tracking is commonly used is the field of people with severe physical disabilities. These applications help them communicate and interact with computers and people. Persons who are quadriplegics (paralyzed of all four limbs) or who have diseases that cause a loss of control of the muscles can interact again with the world. These systems offer eye-typing application with speech synthesis; but other applications can be added as playing games, control of the wheel chair, switching the TV on and others. Eye gaze tracking systems increase the quality of life of the disabled people, but also of their family by allowing them to communicate [16].

Marketing and advertising

The biggest domain where eye gaze tracking is used in terms of money is the domain of marketing and advertising research. When an advertising campaign is realized, the marketing department wants to know some things about the products as *Is the product looking at ?* or as *Has the client looked at the company's logo for a long time* and others. Tests are for example realized in supermarkets, with a portable eye gaze tracking device. A consumer is sent to the supermarket and the marketing department studies which product the consumer notices and how much the shape and the color of the product influence the consumer's gaze. This way, market researchers are able to know what attracts the consumer's attention and advertisers are able to know if the potential consumer is looking at the right thing in their advertisements [6].

Usability

Usability testing is another field of commercial interest for eye gaze tracking. As seen previously, the first use of eye gaze trackers has been for military research to find out the best location of instruments in aircraft cockpits. It is an advantage to be able to track the direction of the eye when introducing a new device to someone. This way, it is possible to see where the user is looking when searching for the control to solve a specific task. Nowadays, internet is also a commercial platform and it is important to find out the usability of web pages as in [1]. Indeed, the web page is the only link between the client and the company behind it. And it is important for the sales of the company that the user clicks on the button *order now* [6].

Psychology and vision

Eye gaze tracking is a great tool for researches in the fields of psychology, vision, perception and cognition. It is used for therapy like behavioral and cognitive therapy which is for the treatment of anxiety disorders as phobias for example. Psychologists also studied the process of reading thanks to eye gaze tracking [6].

Medical research

Eye gaze tracking is also very valuable to examine and diagnose the eye and its function. It is especially useful for people who suffered partial loss of vision because of eye injuries or because of brain damage. For these kinds of injuries, eye gaze tracking is used to follow the progress made in physical rehabilitation [6].

Car assistant system

Eye gaze tracking is a field of research for car industry. The goal is to develop assistant systems for cars. A possible application would be to warn the driver when he falls asleep while driving his car in order to avoid an accident. These kinds of applications are still in the domain of research and are not yet available for consumers [6].

Military research

Eye gaze tracking is also used in military research and more specifically in the field of military weapon control and remote robotics. Militaries were the first one to research in the field of eye gaze tracking in order to help the pilots with the instruments in the cockpit. Now, they investigate in helping the pilots who are busy flying, controlling their weapons. Eye gaze trackers allow the pilot to use his eyes to observe and select the weapons while flying and then to fire them with his hands [16].

Video games

Surprisingly, eye gaze tracking is a technology which is not really common in the field of video games; either in entertainment, educative or rehabilitation context. Few information are found on the subject of video games and eye gaze tracking. Some researches have been made about eye behaviour in game environment in order to be able to find the foundations of the behaviour of the player eyes. It seems that after research, using eye gaze interaction in video games would help rehabilitating people with eye trouble but also would improve visual attention [16].

Studies show that a player feels more immersed in a game and enjoyed it more when using an eye gaze tracking technology. There is also the need to leave the real world in order to enter a virtual one thanks to the video games; that is why the market tends to develop controls easier to use, especially eye gaze controls. There is a big potential behind including such technology in video games; particularly since the video game industry is bigger than the film industry. Some recent video games include some kinds of eye gaze interaction but it is not a real input. For example, in the video game called *The Legend of Zelda: The Wind Waker*, the player can inspect an object that is interesting by looking at it [16].

3.1.5 Methods

Nowadays, most of the eye gaze tracking systems are video-based. But other methods exist for eye gaze trackers. And this section is a survey of all the methods existing for tracking the eye direction.

Contact lenses method

A more direct way than contact lenses is to fix a sensor directly to the eye. But this method is not recommended because fixing small levers to the eyeball can badly injure the eye. That is why contact lenses are used; it is a direct method but it is safer, without the high risk of injury [6].

There are two kinds of contact lenses that can be used for eye gaze tracking. The first kind is contact lenses with an integrated mirror that make it possible to measure the reflection of the light [6].

The other kind of contact lenses is the magnetic one. The contact lenses have a coal that is integrated in it. This coal allows to detect the change in the magnetic field. It is very useful because when eyes rotate, an electric field is produced. But to get the data from the lenses, there is a thin wire connected to the measuring device that is uncomfortable for the user. This method is used anyway because it has a really high accuracy and a nearly unlimited resolution. Magnetic contact lenses are found in the field of psychological and medical researches [6]. The Figure 3.4 shows magnetic contact lenses.

Electrooculography (EOG) method

Another intrusive method is electrooculography. Sensors are placed on the skin around the eyes and it measures an electric field. At the beginning, researchers thought that the electric field measured was the electric potential of the muscles of the eyes. In fact, it is the electric field of the eyes that is measured because the eyes are electric dipoles. There is a disadvantage for this method: the electrodes are sensitive to electro-magnetic interferences. But this method has now existed for a long time and the technology that is at the basis of this method is advanced. On the contrary there is an advantage to this method; it is possible to detect movements of

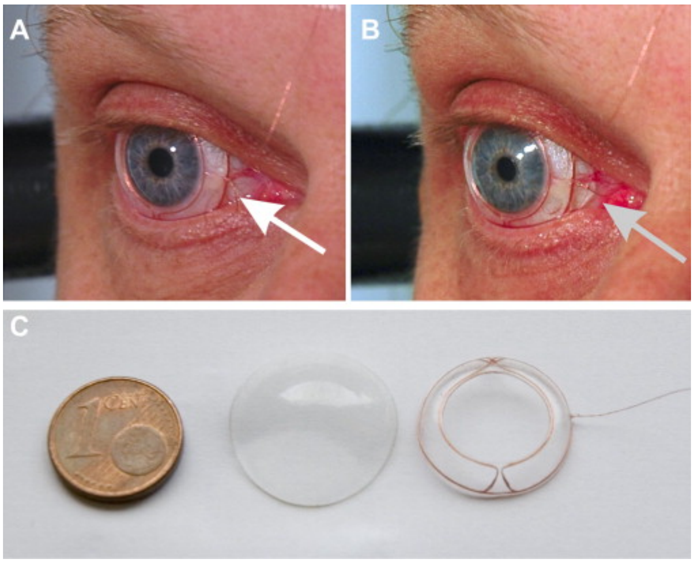

Figure 3.4: Scleral search coil [25]. Images A and B show two ways to exit the connecting wire. Image C shows the ordinary size of a scleral search coil.

the eyes even when the eyes are closed. It is thus possible to detect the movements of the eyes of a user that is asleep for example; which can be really interesting in the field of medical research [6]. The Figure 3.5 shows a typical electrooculographic system.

Figure 3.5: Electrooculography with sensors placed around the eyes on the skin [15]

The contact lenses method as well as the EOG method are intrusive and both do not fit well gaze interaction. The next subsection presents an unobtrusive method that is video-based eye gaze tracking. This method is the one that is the most used nowadays.

Video-based eye gaze tracking method

The principle of video-based eye gaze tracking is to detect the eye and the pupil from video sequences to find out the gaze direction. These video sequences are obtained thanks to a video camera connected to a computer to get real time image processing. Because the method is unobtrusive, it is the one that is the most used.

The purpose of video-based eye gaze tracking is to estimate the gaze direction based on video sequences. One way of doing this is to detect the iris using the contrast between the white of the eyeball and the dark of the iris. Concerning the horizontal estimation of the gaze direction, the results obtained are good but concerning the vertical estimation, the results are bad because the upper and lower part of the iris are covered by the eyelid. That is why it is preferred to work with the pupil for this kind of method [6].

All the video-based methods require the detection of the pupil. To do that, image processing is mandatory. In image processing, what is called edge detection is typically what is needed to detect the elliptical contour of the pupil. A method is to calculate the characteristics of the ellipse formed by the shape of the pupil to estimate the gaze direction. Another method tracks the rotational movements of the eyes but this kind of systems are not very spread. It is mostly used for controlling a head camera by the movements of the eyes [6].

It exists two methods to detect the pupil based on the illumination: the bright and the dark pupil methods. The principle of the dark pupil method is to detect the position of the black pupil in the image sequences. But for people having brown eyes, it is complicated to detect the pupil because of the absence of contrast between the iris and the pupil. The other method, the one of the bright pupil, uses infrared light that are reflected by the retina. This reflected light makes the pupil appear white which gives a good contrast between the pupil and the iris. It is then easier to detect the pupil with image processing. It is the same process than the *red eye* when someone is photographed with the flash. To use this method, the eye gaze tracking device needs to have infrared lights embedded in it. These lights have to come from the same direction that the view of the camera. Embedding these infrared lights with the camera requires mechanical effort. This technique is also called corneal reflection [6].

Corneal reflection method The method that is the most used by the eye gaze trackers is the one of the corneal reflection. It is based on the fact that the cornea has almost a perfect sphere shape. Thanks to that property, a glint will stay in the same position while the pupil moves whatever the gaze direction is. But because of the physical characteristics of each user, the diameter of the cornea is different for each one of them. Calibration is then required each time the user changes [6].

It exists some devices based on corneal reflection that are more sophisticated and that used 2 cameras, one for each eye, with multiple infrared lights. Free head movements are then allowed in front of such devices [6]. These properties are very interesting for the field of HCI. This is this kind of devices that is used for this project.

The method of the corneal reflection can be used to build very simple attention sensor. Vertegaal et al. call it Eye Contact Sensor (ECS). The device is composed of a camera and an infrared light. When looking straight at the device, the glint is located inside the pupil. The principle is to detect whether the glint is located inside the pupil; this allows to know whether an eye is looking at the device. Consequently, it allows to detect the attention of the user. This kind of systems does not need any calibration [5]. The Figure 3.6 shows for image A, a random corneal reflection and for image B, a corneal reflection inside the pupil which means that the user looks directly at the device.

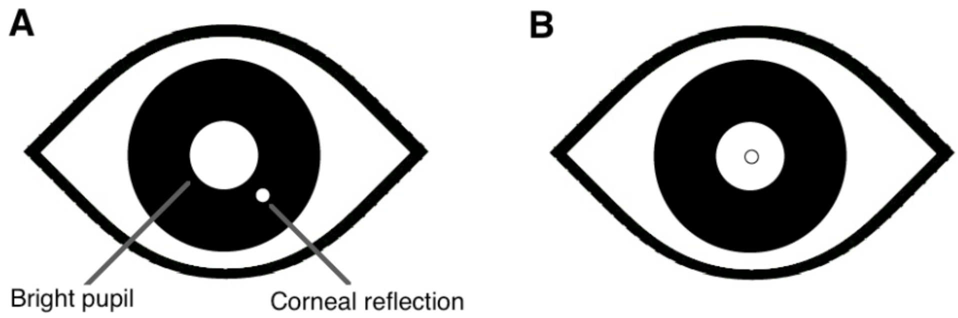

Figure 3.6: Corneal reflection

The principle of the corneal reflection method is to detect the reflection caused by the infrared lights on the cornea. This was called Purkinje image because of the Czech anatomist Jan Evangelista Purkinje who studied them. Because of the nearly perfect shape of the cornea, the glint always stays at the same position on the eye, whatever the direction of the gaze is. This method does not work for person with a deformation of the eyeball. People wearing lenses can also have some trouble with this method. On the contrary, glasses do not disturb this system; the glint can be in a different position because of the glasses, but it will remain at this position. Using the calibration step allows to compensate this change for people wearing glasses. It happens that a reflection on the glasses can fool the system which misinterprets the glasses reflection with the cornea reflection but it is really rare [6]. The Figure 3.7

shows the corneal reflection when the user looks at the four corners of the screen. The glint stays at the same place. The image A shows the corneal reflection when the user looks at the top right corner of the screen; for image B, it is the top left corner; for image C, it is the bottom right corner and for image D, it is the bottom left corner.

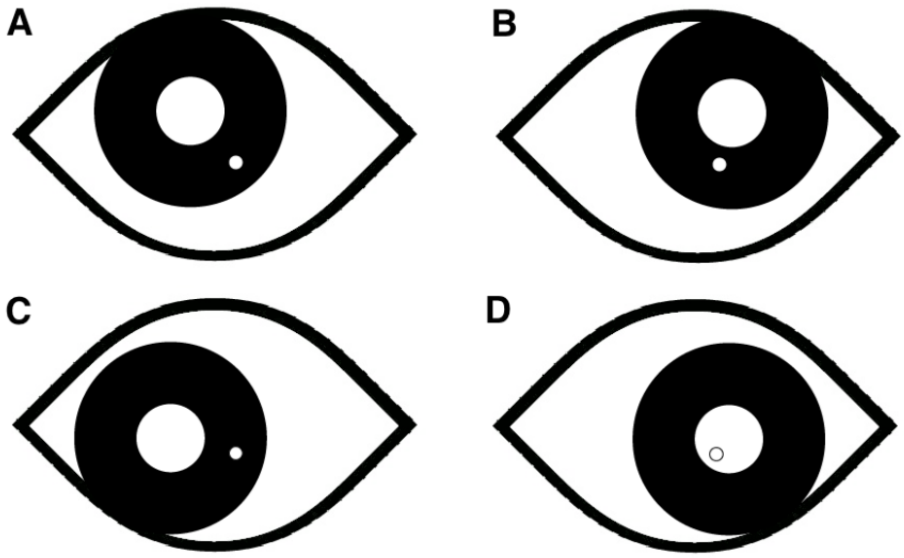

Figure 3.7: Corneal reflection for a user looking at the four corners of the screen

Image processing is used to detect the reflection and also the center of the pupil. Then, a vector from the reflection to the center of the pupil is calculated. This is the basis for the estimation of the gaze direction and the position of the gaze on the screen. This calculation requires different parameters: the display parameters, the spatial geometry of the device and the diameter of the eyeball. Because the diameter of the pupil is specific to each user, that is why calibration is needed. The calibration step estimates the parameters that allows to map the vector to the positions of the screen [6].

A simple way to calibrate is to make the user look at the four corners of the screen. The four lines of sight that start from the eye to the screen form a pyramid. The four lines intersect the plane of the screen and form then a trapeze representing the four pupil centers looking at the screen display corners. Calibration usually uses more points on the screen to achieve better results [6]. The Figure 3.8 shows the vector from the reflection to the center of the pupil and the the four lines of sight for the four calibration points.

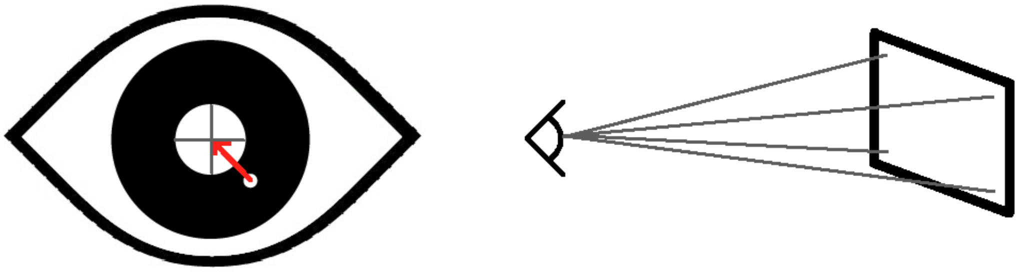

Figure 3.8: Calibration vector and the four lines of sight

Types of video-based eye gaze trackers Most common video-based eye gaze trackers are the stationary one. These devices are commercially available for laboratory and are most of the time used for medical research and marketing research. These systems are embedded directly in a desktop computer and are given with a software which analyzes and visualizes the gaze data. For medical research, most of the time these systems use a chin rest so that the head does not move in order to have a great accuracy. On the contrary, for medical research, the systems used have additional head tracking system so the users are free to move the head. Most of these systems has two cameras for stereoscopic vision but it is not possible to know exactly how these systems are designed because it is business secret of the company [6].

Systems for disabled typically use a tablet computer with an external camera so that both devices can be mounted on a wheel chair. These systems are given with an eye typing application which synthesizes speech [6].

There are also systems that are head mounted. A laptop is inside a backpack to record the data and this way the user can move freely. A camera observes the eyes to get an estimation of the gaze direction and sometimes an additional camera is used to record the field of view [6].

Research eye gaze trackers are abundant but they are not commercially available. These systems are built for research only and for special purposes [6].

3.2 Tobii Eye Gaze Tracker

Nowadays, the number of available eye gaze trackers does not stop increasing. The device presented in this part is the Tobii eye gaze tracker. *Tobii Technology* is probably the most popular eye gaze tracker manufacturer. The Tobii devices allow to move the head freely in front of the devices. Most of the time, these devices are used in marketing research or in psychology department. The company keeps the technology behind their trackers very secret. Information about the trackers cannot be found anywhere else than in their manuals. Consequently, all the following information are got from [28] and [27].

3.2.1 Overview

The *Tobii Technology* company is the leading company in the world selling eye gaze trackers and interactive gaze technologies. The company gives a lot of solution in numerous domains as in communication aids, interfaces and research.

The Tobii technologies have revolutionized research in various domains but also helped people with communication and speech disabilities. The goal of the *Tobii Technology* company is to make eye gaze trackers be broader used. They already succeeded because they are the leading technology on the market in fields such as video games, computer manufacturing, vehicle security, hospitals and diagnostics.

The eye gaze trackers developed by the *Tobii Technology* company work for any kind of user in any kind of environment. It is easy to observe the eyes and their movements. It is known that the eyes give information about what we pay attention to and about the processing of information by the brain. These information can help researches in many fields.

Research

Tobii eye gaze trackers are the favorite eye gaze trackers for research in all the leading universities. Market research companies and web companies use also Tobii eye gaze trackers to research in advertising and test usability.

Diagnostics

Tobii eye gaze trackers have an immense potential in clinical diagnostics. Tobii trackers are already used in order to rehabilitate persons who are in intensive care, to measure acuity of toddlers and to measure the cognitive level of non-speaking persons.

Process control

Tobii eye gaze trackers are able to detect if an operator does not pay attention to important events and then highlight these specific events.

Somnolence detection

Eye gaze information are highly reliable. Tobii eye gaze trackers can give accurate indicators in order to prevent a driver from falling asleep or from being distracted.

Ordinary computers

Tobii technology can add features based on our gaze direction as zooming, selecting and scrolling. All these features are intuitive features that can complement the traditional interfaces. Other features can also be added as power saving, enhancing security or easier computer login.

Sterile environments

For example, in operating rooms, surgeons would be able to control computer interfaces without touching it thanks to Tobii technology; which means the environment will remain sterile. But it can also be used for public information displays.

Disabilities

For persons that cannot use their hands because of disabilities such as quadriplegia or other types of physical restraints, they have another way to keep up with activities on computer and with communication.

3.2.2 Technique

In order to protect their unique technology, the exact technique used for their eye gaze trackers are not revealed. Only the major steps are given.

The Tobii eye gaze trackers are based on the technique of the corneal reflection. Multiple infrared diodes, not visible to the human eyes, are used to create a reflection on the cornea of the eyes. These reflections are get by the image sensor as well as other visual data about the user. Image processing is the next step. Firstly, the eyes are detected, then the exact position of the pupil is detected, finally the right reflections from the diodes and their exact positions are identified. A complex mathematical model is then used to represent the eyes in 3-dimensions and to calculate the point of gaze; or in other words, where the user is looking.

3.2.3 Tobii X120

The Tobii X120 is the model that is used for this project. The X120 model gives twice as accurate eye gaze information than the previous models. Because of their plug-and-play characteristics, the trackers can be installed in a few minutes. And because the trackers are time and cost efficient, the eye gaze tracking can be used in much wider fields for application than has never been done before.

Figure 3.9: Tobii X120 unit mounted on a computer screen

As it can be seen in the Figure 3.9, the Tobii X120 tracker is a standalone eye gaze tracking unit that can be mounted on computer screen or TV. These trackers are designed for research on any kind of surface. Various stimuli can be used such as computer screen, TV, projection screen or any physical scene. This model is the most flexible one. It is particularly suitable for research requiring a specific setup.

The Tobii X120 eye gaze trackers have a lot of advantages that are explained in the following sections.

Plug-and-play

The first advantage described in the next sections is the plug-and-play.

Automatic tracking It is really quick and easy to setup the Tobii eye gaze trackers. There is no need of any manual adjustments and additional software. As a beginner, it is possible to start using the tracker without the need of a professional.

Compatible with ordinary computers Because of the absence of need for a specialized hardware, ordinary computers and laptops can use the trackers.

Automatic calibration procedure The calibration process displays points on the screen. The calibration can only use two points and still have a high accuracy. Some filters are used to keep only the good data points.

Application software The Tobii X120 provides programming interfaces that are easy to use and powerful: it is the Tobii software development kit (SDK). This SDK offers high level interfaces that allow to develop custom applications.

Tracking quality

The second advantage described in the next sections is the tracking quality of the Tobii trackers.

Good tracking ability The Tobii eye gaze trackers are able to track the gaze of any person, regardless of their ethnicity, the wear of glasses or contact lenses, their age. There is also a good tolerance to lightning; the trackers are able to work even with large differences in light conditions.

Tracking optimization The methods of dark and bright pupil are both used while tracking. The most suitable method is automatically calculated and used. Both methods are optimized; it is thus possible to get better tracking quality and better ability to track a wider range of population.

High accuracy High accuracy of the trackers provides precise and reliable data about the position of the gaze of the user. The ability to move the head and the possibility of having changes in lightning allow to have a natural environment for the user.

Binocular tracking The Tobii eye gaze trackers track both eyes at the same time. The left eye and the right eye are determined even when the eyes blink or if the head moves. Using binocular tracking allows more head movements because even if one eye is hidden, the other is still tracked.

Validity measures The Tobii eye gaze trackers provide a tracking status meter and numerical validity measures for each data point, that are built-in and in real time. Thanks to it, it is possible to specify the correctness of the data that are recorded. With these validity information, it is possible to filter the data recorded in order to remove the corrupted one and to have a better data quality. Because the Tobii X120 model can have a data rate of $120Hz$, it is possible to use it for research in the field of neurological processes where it is valuable to have a higher quantity of detailed measures.

Natural User Environment

The third advantage described in the next sections is the natural user environment offered by the Tobii trackers.

Free head movements The Tobii eye gaze trackers allow rapid and large head movements; this way, the users are not restrained in their movements. All the movements that are in the field of tracking are nearly compensated for. Each time the user leaves the field of tracking, tracking stops; but as soon as the user re-enters in the field of tracking, the tracking resumes immediately. It allows the users to be free of their movements and to be natural in the environment. It is an advantage because restraining the head movements for a long time is difficult and not comfortable. With the compensation of the head movements, the quality of the tracking over long recording sessions is still excellent.

Long-lasting calibration The user needs only to calibrate the Tobii tracker once. Then this calibration can be re-used for the same user over and over again. This way, it allows the user to be in a natural environment; they can move, leave the recording room, come back and take the tracking up again without any problem. There is no need to recalibrate the tracking device; unlike other trackers that need to be recalibrated constantly because of the deterioration of the tracking quality.

Technical specifications

The physical specifications of the Tobii X120 model are the following: the device has a width of 320 millimeters and from profile, the device has a width of 163 millimeters and a height of 85 millimeters; as it is shown in the Figure 3.10.

Figure 3.10: physical specifications of the Tobii X120 model

The data rate of the Tobii X120 tracker can be of $60Hz$ or $120Hz$. The data rate is the number of gaze points recorded per second. With this model, there are 60 or 120 data points that are collected per second and for each one of the eyes. In other word, data are collected each $16,7$ milliseconds or each $8,3$ milliseconds.

The accuracy of the Tobii X120 tracker is typically of $\pm 0.5\,°$ which is the best accuracy possible because of the form of the eyeball. This number represents the difference between the actual direction of the gaze and the measured direction of the gaze. This difference is measured for a user located in the center of the field of tracking. This calculation does not include the compensation errors from head movements and the effect of the changes in lightning. This accuracy can slightly change because of changes in the lightning, the calibration quality and the characteristics of the eyes of each user.

The tracking distance for the X120 model is between 50 and 80 centimeters. The tracking distance represents the distance to the sensor within which eyes can be detected and tracked.

The latency is of a maximum of 33 milliseconds. The latency represents the time between the moment when a image was taken and the moment when a gaze data point is given. This number is affected by the computation time of the eye gaze algorithms and by the tracker hardware in itself.

The time to recover the tracking after blinking is of 8 milliseconds. Blinking is the non voluntary act to close and open the eyelids. Each time a user blinks, the pupil and the cornea are covered by the eyelid and the tracker cannot get gaze data. This time of 8 milliseconds represents the time that the tracker takes to track again when the user opens his eyes again. Resuming the tracking is almost instantaneous with the Tobii eye gaze tracker.

The time to recover the tracking after losing track of the eyes is typically of 300 milliseconds. This can happen in case such as the user completely turning away from the tracker. This number represents the time that the tracker takes to recover the tracking after completely loosing it.

The eye gaze techniques used are the technique of the bright pupil and the technique of the dark pupil. The best technique is automatically chosen by the algorithm.

The eye gaze tracking server is embedded inside the tracker hardware. The calculation of the eye gaze data are done by firmware embedded inside the eye gaze tracker. It is possible to connect various applications as clients to the eye gaze tracker with a LAN wire and this way to get data in real time.

Physical setups

The Tobii eye gaze tracker X120 makes it possible to use different stimuli as computer screen, TV and projection screen. In the following figures, a computer screen is used but it can be replaced by any other displays.

The Figure 3.11 represents the basic configuration of the Tobii eye gaze tracker X120. The tracker is mounted on a computer screen. This is mostly used for research on eyes and reading. This configuration is the one used for this project.

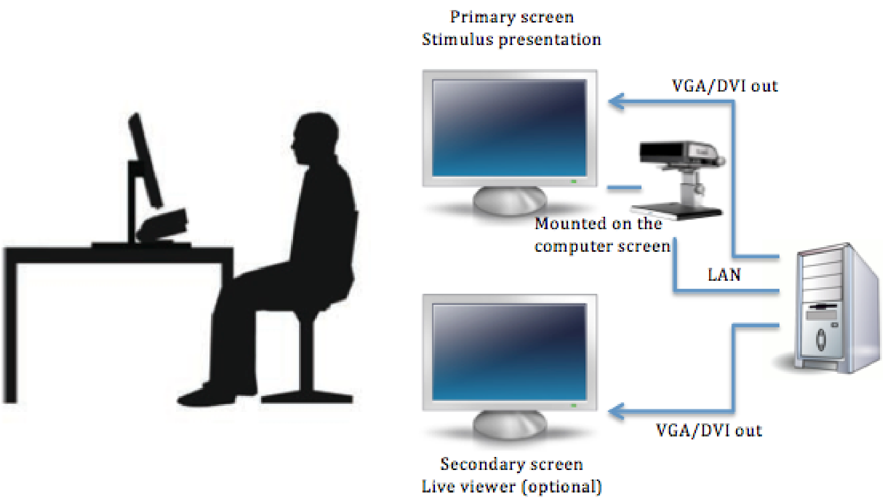

Figure 3.11: Tobii X120 in a basic setup

The Figure 3.12 represents a configuration with a projection screen. The user is standing in front of the tracker. This configuration is traditionally used for marketing and advertising research; it allows the user to stand in front of different products as on the supermarket shelves.

3.2.4 Softwares and Implementation

The Tobii eye gaze trackers are given with some tools in order to setup the tracker and to be able to use it right away.

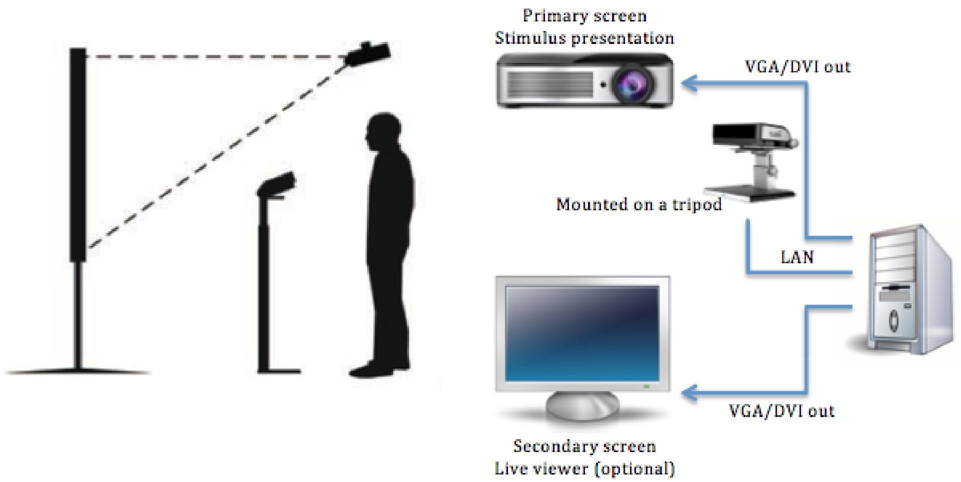

Figure 3.12: Tobii X120 in a setup with a projector screen

Tobii X Configuration

The first important tool is the *Tobii X Configuration*. It allows the user to enter the physical specifications of the Tobii eye gaze tracker setup in order to have the best calibration possible.

Tobii Studio Gaze Analysis Software

Another tool is the *Tobii Studio Gaze Analysis Software*. It is a software that provides sophisticated analysis tools that can be used in many fields as usability tests, marketing researches or psychological researches. This software makes it possible to record and analyze eye gaze data in order to make the interpretation of the eye behaviour easier. In order to record data for a series of tests for this project, this software was used. It is possible to choose among a wide range of options and to record the exact data needed for the user; it is possible to follow the recording in a secondary screen to have a live view.

Tobii Software Development Kit (SDK)

With the Tobii tracker, the Tobii SDK is provided. It makes it possible to develop applications that use, control and retrieve data from the Tobii eye gaze tracker. This is really useful for customized application based on eye gaze tracking.

The SDK provides two kinds of interface (API): a low level interface that allows high customization and a high level interface that does not need a lot of efforts into programming. The SDK also provides some code samples in order to get to know better the SDK and to demonstrate the possibilities of their APIs.

The low level API is composed of basic libraries that makes it possible for the user to develop applications from scratch. On the contrary, the high level API is a Component Object Model (COM) implementation of the low level interface; this way, the user does not have to program each time the same tools. These APIs have everything required to implement an application using the tracker in real time and to get the gaze information of high precision from the eye gaze tracker. These APIs include also GUI tools allowing to display the tracking and to calibrate the tracker.

Chapter 4

Reading Analysis

A review of the process of reading will be done. Then the process of saccades and fixations will be studied along with their signification. Then the Dyslexia condition will be explained and the problems caused by it relative to reading will be detailed.

4.1 Reading

In psychology, the eye gaze trackers have been first used to study human visual attention while reading experiments. For this project, the reading process needs to be understood. In the following sections, the main processes that are at the basis and that constitute reading are detailed.

4.1.1 Overview

While reading, the eyes follow some specific patterns known as saccades and fixations. Saccadic eye movements separate some fixational pauses. In many studies, the eye movements are considered as a reflection of the cognitive process. In general, the interest and the attention of a reader are revealed by a fixation. The retrieval of information in reading is done during the fixations; no information is obtained during a saccade (between two fixations). The average values, when reading english are of 200-250 milliseconds for eye fixations and of 7-9 letter spaces (about $\pm 2^\circ$ of visual angle) for saccades [22]. But a lot of factors can make these values change.

The single fact that a different user is reading causes changes in the eye behaviour characteristics. The saccade size can vary from 1 character space to 20 character spaces and the fixation time can vary from 100 milliseconds to 500 milliseconds [22]. The eye movements can also be influenced by the aspect of the text: the quality of the print of the text, the font, the letter spacing and the line length. The more difficult the text is to read, the more the fixation duration increases and the more the saccade length decreases. The eye movements can also be influenced by the fact that the reader is reading aloud or silently. While reading aloud or while listening to someone reading the same text than in silent reading, the fixation duration lasts longer [7]

In [23], it was found that there is no relationship between the fixation duration and the saccade length. These two processes are not influenced by only one mechanism. It is thus a real important objective for researchers to be able to determine the processes that are behind the eye movements while reading.

4.1.2 Reading process

Two main events happen when reading: the saccades and the fixations. Both are described in the following sections.

Fixations

The context makes the fixations be interpreted in different ways. For example, while browsing a webpage, a higher rate of fixations on a specific area can show a greater interest in the area fixed on; it can also be an indication that the area is in some way complex. These interpretations can be different if the fixations are observed in a search task for example; a higher rate of fixations indicates a greater uncertainty when trying to recognize an item [20]. The Table 4.1 describes what measures some eye movement metrics.

Table 4.1: Eye movement metrics [20]

Eye movement metric	What is measured
Fixation per area of interest	When reading a text, the number of fixations should be divided by the number of words in the text. This way a difference can be made between a higher number of fixations because of the length of the text or because a word is difficult to understand.
Fixation duration	If the fixation lasts longer, it is an indication that the reader has some trouble extracting the information from a word.
Repeated fixations	If a target has been fixated once and then there is still a high number of fixations on the target, it gives an indication that the target lacks of visibility or meaningfulness.

In reading process, a strong correlation between the number of fixations and the number of words in a sentence can be observed. For example, a good reader is likely to make a fixation on each word in a sentence. Most of the time, the short common words are skipped and the long words get two fixations [13].

Usually, the fixations are approximately on the middle of the word for a normal word. For a long word, multiple fixations can happen on the same word. What guides the saccades is the field of vision that is parafoveal. The attention can be on the next word without having the eyes moving and without having a fixation on it. Before a saccade happens, the shape and the length of the next word can be evaluated. In order to sharpen the landing of the saccades, some micro saccades can happen within a word [13].

The difficulty of a word is expressed by the duration of the fixations. When a reader encounters a word that is unfamiliar, he fixates the word longer. On the contrary, if the word has a high frequency and if it is familiar, the fixation is only a short one on the word [13].

At the beginning of the researches on eye movements and reading analysis, two hypothesis have been made about the fixation duration. The first hypothesis is that the fixations are too short and the eye movements too quick that the processing of the text has to lag behind the perceptual input of the text. In other words, our comprehension of the text lags behind the part of the text just read. The second hypothesis is that the fixation duration and the movements of the eyes are both affected by the cognitive process that occurs during the fixation. As a consequence, the more the text or the words are difficult, the more the fixations last. There were some attempts to demonstrate either one of the hypothesis but it was not conclusive [22].

Saccades

During saccades, no encoding event takes place. Because of that, no information can be retrieved from a saccade. However, we can get some information from saccades; when a regressive saccade happens, it is usually a measure of processing difficulty when decoding a text [20].

Regressions are a good way to estimate comprehension failure. In order to re-read a sentence, the reader has to go back to its beginning. So if a reader did not understand what he just read, he needs to go back to re-read it so that he can try to understand it. For example, it is typical that poorly written texts or difficult texts cause a lot of regressions and consequently require for the reader a lot of time to process the text. Each time the reader has trouble with the text, it results either in a longer fixation, either in a regression depending on whether the problem is on the word being read or on a word already read.

Most of the time, a regressive saccade (or a *regression*) is very small; it goes backward of two or three letters. When a regressive saccade goes way back in the text, it usually indicates a confusion in a higher level of processing of the text [20].

But regressions are not always an indication of difficulties while processing a text. It can also be an indication of a good reader; the reader goes back to previous parts of the text to make cross-references with what he just read. In this case, the regressions are most of the time huge. The reader can go back to previous paragraphs. However, when regressions are an indication of difficulty, the regressions are often smaller; the reader usually goes back to previous words.

4.1.3 Reading patterns

In [13], four kinds of reading patterns for a user processing a text are described: the *reading* pattern, the *scanning* pattern, the *dormant gazing* pattern and the *encountering difficulties* pattern. Changes in the regression structures, in the fixations and in the scan paths are assumed to be detectable in order to identify one of the four patterns. But it may not be an easy task to classify the information obtained to match one or another pattern. Some patterns can lead to the same observations; for example, re-reading the text in order to have a good comprehension of the content of the text, and stopping at a complex word and searching through the text lead to the same behavior that is a regression for this case. In order to make a difference between these behaviours, the user's reading style must be observed and studied.

Reading

The attention of the reader is focused on the text being read. The eye movements follow a specific pattern; they go from left to right and follow the text on the screen. The fixation duration falls into threshold values defined previously. This is this kind of pattern that is interesting for this project. For the tests, the subjects are asked to read aloud some sentences. They are *forced* to have a reading style.

Scanning

The reader glances or browses through the text. The intention of the reader can be to search something in the text or just to speed-read the text. The number of fixations is too low for the user to actually be able to read the text. Unlike the reading pattern, here, the eye movements do not follow the text from left to right; the movements are not predefined.

Dormant gazing

For this pattern, the user does not read or scan the text. The user may just look at the text but without getting any information from it, there is no cognitive process; he can just be thinking about what he just read. When this happens, the attention of the reader is introverted. There is no predefined pattern for the eye movements; the eyes may move or not. The term *blank eyes* is the term used to describe the eyes with which the user is looking at the text.

Encountering difficulties

While reading, the user may have encounter a word or a syntax of a sentence that is problematic for him. When this happens, the number of fixations around this word and the duration of the fixations increase, while the reader tries to understand it. This pattern is also interesting for this project especially since dyslexic persons are the subjects. As it will be explained in the next section, dyslexic persons have trouble reading and they encounter more difficulties than normal persons.

Analysis

In order to analyze reading patterns, a difference must be made between a fluent and uninterrupted reading and difficult reading. In a case of fluent and uninterrupted reading, the reading is from left to right with no interruption, regression or specific fixation. In the case where the reader has trouble while reading, it is possible to observe multiple fixations on a difficult word and some regressions. The reading is not fluent. For this project, the reading is likely to be not fluent, but interrupted and with regressions because the analysis is based on dyslexic subjects.

In addition to the reading analysis, it may happen that the reader stops his cognitive process by looking at the text with *blank eyes* maybe in order to think about what he just read. In this case, it is not possible to assume that the reader is encountering difficulties if the fluent reading is interrupted. Sometimes, the eyes just wander around the screen with no specific purpose. The pattern is thus the *dormant gazing* pattern.

It may also happen that the reader is searching through the text some information without actually reading it. It may happen as well that the reader re-read some parts of a sentence not because he did not understand it but because he wants to confirm some information or to make cross-references. The pattern is thus the *searching* pattern.

Because of all of this, it is really difficult to tell with certainty what the user is thinking and what is behind his behaviour. The following question can be asked: *How to follow the reading path?*. The algorithms used to follow the reading process are described in Chapter 7 and in Chapter 8.

4.2 Dyslexia

Dyslexia is a disability. It affects the people having difficulties while reading; in particular difficulties in learning to read and in understanding what was read. There is a large amount of possible reading deficits; dyslexia is indeed a definition that gathers together all these deficits.

4.2.1 Overview

If we look at the origins of the word, from Greek, *dys* means *difficulty* and *lexia* means *word*. Dyslexia thus means *difficulty with words*. This condition has been studied for a long time now and has been defined in many ways [19]. The most common and accepted definition is the one in [4]. The definition is the following *Dyslexia is a disorder manifested by difficulty in learning to read, despite conventional instruction, adequate intelligence and sociocultural opportunity.*

A lot of hypothesis have been made about the cause of the reading difficulties for dyslexic people. The one called the *phonological awareness* seems to be the most accepted. The reading process can be divided into two parts that are the *decoding* part and the *understanding* part. The *understanding* part is based on the personal experiences and on the education of the person [19].

Concerning the *decoding* part, there are two kinds of processes: the *phonological* and the *orthographic* decoding. The *orthographic* decoding consists of decoding a word thanks to its graphical representation in order to get its meaning. The reader must have an orthographic footprint of this word in their memory [19].

The *phonological* decoding consists of decoding a word by splitting it up into smaller parts. These parts are decoding phonologically; it means that the parts are decoding by their sounding. Then these parts are combined together to lead to the decoding of the word. This decoding is useful for decoding unfamiliar words unlike the *orthographic* decoding [19].

The reader can also rely on the context of the text. For example, they can rely on semantics which are the clues in the text that can help describing a word; and there is also the syntax which represents the rules that can help identifying the type of a word [19].

For a non dyslexic person, these decoding processes are automatic and the reading is not a problem. It is not the case for dyslexic persons; the main problem can be narrowed down to the problems explained previously: the decoding processes. In addition, the dyslexic persons seem to have poor phonological awareness [19].

4.2.2 Characteristics

Dyslexic persons show typical characteristics that are the following [19]:

Difficulties to understand the text just read To understand the text being read, the decoding process must be well performed. Because dyslexic persons have trouble decoding the text they are reading, they use all their cognitive resources on decoding the text and not on understanding it. Moreover, words are not well decoded which leads to more difficulties understanding the text. It happens that sometimes a word is replaced by another while reading and the dyslexic reader does not correct it; so their comprehension is based on wrong words.

Guesses based on context Because dyslexic persons have trouble decoding the text, they often try to guess the phonology or the orthography of a unknown word by looking at the context of the text.

No regular eye movements While reading, dyslexic persons do more regressions than normal persons because of their troubles decoding the text. It can be observed that while reading, dyslexic persons misread some words and then in order to correct them, they go backward. It also happens that they have trouble understanding the text so they pause and go back in order to have a better comprehension.

Hasty reading Dyslexic persons have trouble following the rhythm of a sentence and because they try to follow it, they often change times and conjugations of verbs but also replace words with synonyms and omit other words. Sometimes, they want to follow the rhythm of a sentence so bad that even if they replace words or omit some, they continue reading without correcting their mistakes.

4.2.3 Dyslexic reading behaviour

While conducting tests with dyslexic persons, it was possible to divide them into two groups. There were the ones that, when making a mistake while reading, correct themselves. For example, if a word is misread, a regression occurs immediately in order to correct the mistake. This causes a higher number of regressions but also a slower reading path. The dyslexic persons want to understand what they read.

The other group gathers dyslexic persons that do not correct themselves when they misread some words. For example, if one misread a word or omit one, they do not correct the error made; even if the sentence does not mean anything afterward. This causes a lower regression frequency but also a faster reading path. This group of readers seems to read just to read and not to understand what they read.

In [19], the first group is called the *analytical* readers and the second group is called the *mechanical* readers.

The reading path of dyslexic persons (both *mechanical* readers and *analytic* readers) has a similar behaviour that can be observed. This behaviour is different from the non dyslexic persons reading behaviour. The typical characteristics of this behaviour are the following [19]:

Frequent regressions Most of the time, for dyslexic persons, the eye movements are not regular which lead to a high frequency of regressions. Whether they try to correct a misreading or to understand what was just read, they go backward to collect the necessary information; which leads to more regressions.

Mixing up similar words or similar letters Dyslexic persons can have trouble perceiving and remembering letters; that is why they may mix these up. It is the same for words that are similar.

Difficulties reading complex or long texts For some dyslexics persons, the ones with short term memory impairments, it can be difficult to read a long or a complex text.

Slow decoding Dyslexic persons have the tendency to have a slow reading path. Indeed, because they have trouble reading and decoding the information from the text, they spend more time on the text. This results also in higher fixation duration.

Addition and omission of words While reading a text, it is frequent that dyslexic persons add or omit words.

Non frequent words Dyslexic persons show some difficulties when they encounter a non frequent word; words that are frequently used are more often read with less difficulties. In order to decode a word, dyslexic persons have to have an orthographic footprint of the word as explained before; but if it is the first time they encounter a word, they are more prone to have difficulties decoding it.

Pauses It happens often that dyslexic persons make pauses of abnormal duration between words. Most of the time, these pauses occur with *analytical* readers, it can occur with *mechanical* readers that encounter difficult words. These long pauses are a direct indication of a reading mistake to come with the words following the pauses. It can also happen that the pauses happen in order to have time to go backward to understand what was read. For both kinds of readers, it often happens that they use words such as *hmmm* or *ehm* in order to fill the pauses.

Difficulties with long words It is frequent that when a word is particularly long, dyslexic persons have trouble reading it. The relationship between the frequency of a misread word and the length of the word appears to be proportional. However, the composition of the word in vowels or in consonants does not influence this relationship. It is observable also that the initial syllables are correctly read and only the end of the word is misread.

Chapter 5

Eye Gaze Tracking and Reading

For this project, we are focusing on eye gaze tracking used to follow reading process. This part describes what has already been done on the subject and what are the issues that we will encounter when using eye gaze tracker for reading process. It also describes algorithms used to follow reading paths.

5.1 State-of-the-art

Researchers have been studying eye gaze tracking and have been using these studies to detect reading or to analyze reading. Some papers have been written about eye gaze tracking combined with reading. The following sections are an overview of the state-of-the-art of eye gaze tracking and reading.

5.1.1 How Eye Gaze Feedback Changes Parent-child Joint Attention in Shared Storybook Reading?

In [11], a study is made on measuring and improving the parent-child joint attention. It is really important for social learning activities. The study is made on one of these social learning activities that is storybook reading. The point of this study is to give to the mother a feedback of where the child looks and to give to the child a feedback of where the mother looks. Most of the time, when reading storybook, the child and the mother are not paying attention to the same things at the same time. The mother is reading the text and the child is mostly looking at images. Learning print-related skills is done when there is a real joint attention. Giving them real-time feedbacks allows the mother to know what the child is looking at and to adapt her reading; it also allows the child to know which word his mother is reading and the pronunciation of the word.

5.1.2 Eye Gaze Based Reading Detection

In [14], the purpose of the study is to determine whether the user is engaged in reading activity. This is done by using ordinary video camera. The real challenge in this research is to find out whether the user is reading or doing some other activities on the computer and this for each user, in spite of the differences between each user. To determine reading activities, the researchers use finite state machines (FSMs) and model the changes of direction taking place in reading. The system is however user dependent and has to be recalibrated before each user.

5.1.3 The Reading Assistant: Eye Gaze Triggered Auditory Prompting for Reading Remediation

In [24], the study is about helping users that are reading by prompting audio information. If the user is having trouble reading a word, the system reads the word aloud in order to help with the pronunciation and the recognition of the word. The purpose is to create a system unobtrusive with an ordinary video camera and information prompted automatically when the user has trouble reading, unlike mouse activated prompting.

5.1.4 Design Issues of iDict: A Gaze-Assisted Translation Aid

In [13], the purpose of the study is similar to the one before but for foreign languages. Help is prompted when a user has trouble reading a word from a text in a foreign language. This way the user can learn the pronunciation of the words. This system helps with translation and is based on information got from reading researches, syntax analysis, language model and on the user profile.

5.1.5 Reading Tutor

In the already-implemented application used for this project, speech recognition is used to detect miscues in reading of dyslexic persons. This will be more detailed in Chpater 9.

5.2 Issues when using an eye gaze tracker for reading process

The main issue that is encountered when working with an eye gaze tracker on reading paths is the lack of accuracy of the eye gaze tracker. As said previously, this inaccuracy can be caused by one of the three following sources:

Inaccuracy This element varies depending on the type of devices that is used to track the eye gaze and on the quality of the calibration. Head mounted systems are more accurate than remote trackers. Mapping the gaze data points to words in the text is straightforward with head mounted systems. With remote system, like the Tobii tracker used in this project, mapping the gaze data points to words in the text is not so clear. Using a remote tracker induces noise and drift in the gaze fixations.

Calibration drift This represents the decreasing quality of the calibration even if at the beginning the calibration was great. It can be caused by the inexact compensation of the head movements or by the change in the characteristics of the eye (for example, a change in the size of the pupil being tracked).

Physical characteristics of the eye Because of the shape of the eye, there is an inaccuracy in the measured gaze position. The reader, without any movement of the eyes, can focus their visual attention.

With the development of the eye gaze trackers and their technology, the problem of accuracy caused by the first reason and the second reason presented above can be decreased. But the physical characteristic of the eye cannot be changed and will represent a permanent source of error for the eye gaze trackers.

The goal when extracting the data collected during the experiments, is to map the gaze points obtained into the words of the sentence being read and to compare with the words identified by a manual transcription. Because of the inaccuracy of the eye gaze tracker, it is a challenge to have reliable results. In [12], a solution is approached: using what has been called the *implicitly Required Fixation Locations* (RFLs) in order to make up for the errors in the measurements of the eye gaze tracker. The principle is the following: when tracking the eye gaze, if at a moment we can know for sure what the user is looking at, the information of the object looking at and of the measured gaze points can be used to estimate the error and to correct it.

5.2.1 Offset

When working with the eye gaze tracker and testing it, some observations can be made. First, when trying to map the gaze point coordinates into a word, the whole context of the text must be taken into account. Indeed, there is a need to analyze the whole reading path instead of only some parts of it. Because of the inaccuracy of the tracker, there can be an offset in the estimation of the gaze point coordinates. Sometimes when looking at the first line of a text, the gaze points are located above the first line and the gaze points corresponding to the second line are located on the first line. If we only analyze the part corresponding to the second line, then the gaze points will be mapped into the first line. With the whole context, it reveals that the gaze points are belonging to the second line of the text. For example, in the Figure 5.1, the gaze points belonging to the stroke of the reading path with the number 1 would be mapped into the second line. However, by looking at the context, we know that they belong to the first line. It is the same for the gaze points belonging to the stroke of the reading path with the number 2; they belong to the second line and not the third line.

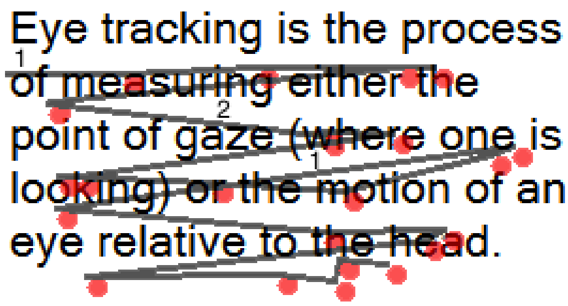

Figure 5.1: Reading path with offset

5.2.2 Ascending and Descending Paths

Other observations can be made. It is very common to observe an ascending reading path. At the beginning of a line, the gaze point positions are quite accurate and fall on the first words. But then, as the reader progresses in their reading, the gaze point positions rise. At the end of the line, the gaze points are located on the previous line. Once again, if the gaze point positions are mapped without looking at the whole context, the positions would be mapped onto the previous line and not onto the right line. For example, in the Figure 5.2, the stroke of the reading path with the number 1 is ascending at the end of the line; here, the gaze points do not fall right on the beginning but fall right at the end of the line. The beginning of the stroke would be mapped into the second line if we look at the context but the end would be mapped into the first line while it still belongs to the second line. The stroke of the reading path with the number 2 is descending; here, the gaze points do not fall right on the beginning once again, but fall right at the end of the line. The beginning of the stroke would be mapped into the third line if we look at the context but the end would be mapped into the second line while it still belongs to the third line.

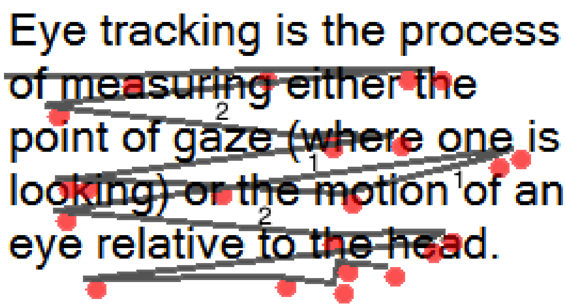

Figure 5.2: Ascending and descending reading paths

Globally, the vertical accuracy is the biggest problem. There is indeed a vertical inaccuracy in the estimation of the eye gaze positions. But this inaccuracy is not equal for every point of the screen; which makes it difficult to compensate. It is however more difficult to detect the horizontal inaccuracy but it is less often encountered. This problem is specific to the task that the tracker performs; for the reading task, when reading a sentence, the visual angle is much larger horizontally than vertically [12].

The drifts in the gaze point positions caused by the inaccuracy of the tracker are not equal for every pixel of the screen, so this inaccuracy cannot be compensated in an equal way for all the fixations as seen previously.

5.3 Algorithms

Two algorithms for improving the accuracy have been described in [12].

5.3.1 Sticky lines

This algorithm has been implemented in order to correct the problem of ascending and descending reading paths. This way, we follow the line instead of the real reading path that is ascending or descending. The reading path is observed across the entire line being read in order to compensate this error. The reading path is thus tracked and as long as the reading of this line seems to be continuing, the gaze point positions are mapped onto this line [12].

5.3.2 Magnetic lines

This algorithm has been implemented in order to have the right vertical position when starting a line. This way the gaze point positions are mapped into the right words on the right lines. The algorithm detects the end of a line and the transfer to the beginning of the next line. It adapts the mapping by taking into account the current line and some of the previous and of the following lines. The reading path is tracked and the algorithm detects specific events that are the transitions to next lines. These events are called *next_line_events* [12].

In [1], Beymer et al. map data obtained from a Tobii tracker to words in a text on a webpage. They decided to only look at their data points and not to use the RFLs to improve their accuracy. An algorithm is presented to overcome the vertical inaccuracy.

5.3.3 Matching groups of fixations to lines

The baseline of their algorithm is to match gaze data points to the text in groups. When using a text with multiple lines, a natural grouping happens: horizontal lines. Because the vertical inaccuracy can affect the whole line (ascending/descending paths, offset), identifying groups of fixations corresponding to a line is a good way to overcome this inaccuracy [1].

The more the paragraph is long, the higher the ambiguity in matching groups of gaze points to lines in the text is. When the text is a paragraph with only a few lines, it is easier to match groups of gaze points to lines [1].

In [1], the fixations are detected thanks to the dispersion-based method. This method defines a fixation as a time window of gaze points where the spatial dispersion of the

X and Y coordinates falls below a specific threshold. Here, the time window has a length of 100 milliseconds minimum. Next gaze points are included while the spatial dispersion remains below the threshold. The fixation is then represented by the average of the X coordinates and of the Y coordinates and by their start and stop times.

The second step is to group these fixations into *gaze lines*. The method used to detect a group looks like the growing technique used for detecting the fixations. A line is fitted to the fixation centers, and fixations are added to the line as long as the spatial dispersion remains under a specific threshold. Then, the fixation groups are matched to lines in the text [1].

Chapter 6

Specification of requirements

The requirements specific to this project are the ones described in the following paragraphs.

Tracking

The system should be able to track the reading process of the subjects. The system should follow the reading path and map the gaze point locations to the corresponding words. In other words, the system should be able to identify which word is being looked at.

Detection of transitions to the next line

It has been seen previously that the eye gaze trackers when used for reading process are bad regarding the vertical accuracy. Because of this vertical inaccuracy, detecting on which line are the gaze points can be tricky. It would be more efficient to start on the first line and then to move to the next line each time we detect a transition to the next line. This way, the Y coordinates is not taken into account in our calculation. The vertical inaccuracy cannot induce error anymore.

Drift correction

As explained previously, the vertical accuracy of trackers are quite bad. It is common that a gaze point is well located on the screen but with time, vertical drifts happen. When it happens, most of the time it results in ascending and descending paths. The system should be able to correct these drifts and to improve the vertical accuracy.

Offset correction

Once again, because of the vertical inaccuracy of the eye gaze trackers used in reading process, it also happens that the vertical coordinates of the gaze point have an offset. This is caused by the calibration step. The better the calibration is, the better will the vertical accuracy be. But even with a very good calibration, an offset is added. Moreover, as explained before, the vertical inaccuracy is not the same in every part of the screen. So the offset is not the same in every part of the screen. The system should be able to compensate this offset so that the resulting accuracy would be good.

Correction of the erratic movements of the eyes

Micro saccades can happen while reading. Sometimes, the eyes can be attracted by anything on the screen while reading and a micro saccade happens. These gaze points are considered as errors; they do not mean anything. It corresponds to an erratic movement of the eye and it stands out of the other gaze points. The system should be able to filter these gaze points in order to have a good accuracy.

Part III

Design and Implementation

Contents

In the previous parts, how eye gaze trackers and more particularly Tobii eye gaze tracker work, and how reading can be analyzed were studied. This part will describe the design of the project and the algorithms that were actually implemented.

Chapter 7

Design

Two algorithms are designed to improve the accuracy of the data collected thanks to the eye gaze tracker. One of these algorithms is based on the *Sticky lines* and *Magnetic lines* algorithms described in the Section 5.3 and its purpose is to compensate the vertical inaccuracy of the tracker. The second algorithm is designed to compensate the little *jumps* that the eyes are doing sometimes; it is an algorithm to *smooth* the data.

Baseline algorithm

Based on the *Sticky lines* and *Magnetic lines* algorithms described in the Section 5.3, an algorithm is designed. In order to avoid the problem of vertical inaccuracy, the algorithm designed does not take into account the Y variable of the gaze point coordinates.

The purpose of the algorithm is to follow the lines of the text and to detect what is called a *next_line_event*; when the user goes to the next line.

At the beginning all the gaze point positions are mapped into the words of the first line. When the algorithm detects a transition to the next line, then the gaze point positions are mapped into the words of the second line. And it goes on as many as there are lines.

Algorithm to smooth the data

This algorithm is designed to compensate the inherent movements of the eyes that are erratic. During some experiments, it was possible to observe some *jumps* in the movements of the eyes. It can happen that sometimes the eye is attracted by something on the screen and does a fast saccade and then goes back to where it was looking at. In order to avoid these *jumps*, this algorithm is designed.

In the Section 4.1.1, it was explained that the average time for a fixation can go from 200 milliseconds to 500 milliseconds in function of the person. It was also detailed that for a dyslexic person, the average time for a fixation is longer than for a normal person.

Based on this knowledge, we look at the word that the gaze point position where mapped into; if the previous word is different from the actual word, if the following word is different from the actual word, if the previous word is the same that the following and if the time passed from the previous word to the following word is lower than 200 milliseconds, then the word being look at is consider as an erratic movement of the eye.

If the time passed is lower than 200 milliseconds, it is not considered as a fixation indeed but as a saccade. This way, the *jumps* that the eye can make when looking at a word are removed and the data is more coherent.

Inputs

The data that we need in order to be able to implement the algorithms are the one presented in the Table 7.1. The data are what the algorithms need in input in order to be able to treat the data and to get reliable information to estimate the accuracy of the eye gaze tracking.

Table 7.1: Required input data

Word information	We need the word information relative to the screen: the X coordinate, the Y coordinate, the *width* of the word and the *height* of the word. With this information, we will be able to map a gaze point position into a word.
Time information	We need the time information relative to the word; when the word was started to be said, when it was finished. This way, we can match the time of a word spoken and the time of a given gaze point.
Sentence information	We need to have information about the sentences and the words in each sentence; the number of sentences, the number of words in each sentence, the word Id in a sentence (the place of a word in a sentence).
Gaze information	We need to obtain information on the gaze from the eye gaze tracker; the X coordinate of the gaze, the Y coordinate of the gaze, the time at which the gaze point was recorded.

Tracking error rate

The tracking error rate is calculated as the number of times the system detects a change in the word being looked at, when the change is not correct, plus the number of times the system detects a change missing, divided by the number of actual word changes. In other words, we divide the number of non correct hypothesis changes and the number of missing hypothesis changes by the number of reference changes.

The reference data are obtained from manual transcription. The reference data give information on the words said by the subject, the time at which the subject starts saying the word and the time at which the subject has finished saying the word. We only consider the words said correctly as well as the words said with miscues. These words are the references used for our tracking error rate calculation. We consider the time at which the subject starts saying the word for the calculation of the tracking error rate.

A tolerance of ±250 milliseconds is used for this system as in [21]. In order to have coherent results, the tracking error rate must be as low as possible.

Chapter 8

Implementation

The previous chapter described the design of this project; this chapter describes the algorithms that were actually implemented. The system has been implemented offline but a live setting would not require a new implementation since the algorithms that follow the reading process would work in the same way with a live setting.

The eye gaze tracker outputs files containing all the data collected during the tests. In these files, for each gaze point recorded, the tracker specifies whether the gaze point belongs to a fixation. It also gives an index for each fixation. In order to have consistent data, we calculate the average of the X coordinates and of the Y coordinates of all the gaze points belonging to the same fixation.

For the Tobii trackers, the fixation filter's duration is set to 60 milliseconds. A cluster of gaze data points is detected as a fixation, and not as noise, if gaze data points belonging to the same cluster are still recorded 60 milliseconds after the recording of the first gaze data point of the cluster. As explained in Chapter 4, a fixation duration can vary from 100 milliseconds to 500 milliseconds. Tobii sets the threshold for detecting a fixation at 60 milliseconds in order to have more false positives than false negatives.

Implementation with no processing

For this implementation, no algorithm is applied to the data. The tracking error rate is calculated directly from the raw data collected during the tests. The implementation described in this section is the implementation that maps the gaze points into words.

To do that, information have been extracted from the files containing all the data obtained during the tests. The gaze information are extracted and stocked into a list for each sentence. It is the same for the information collected from the manual transcription. There are also the information got from the graphical interface about the word coordinates that are stocked into a list for each sentence.

For each sentence, we look at the first gaze point information in the list; particularly at the X coordinates and the Y coordinates. Then we first check if the gaze point belongs to one of the word of the sentence (a rectangle represents each word with a X coordinate, a Y coordinate, a width and a height). If the gaze point coordinates do not fall directly into a word, we calculate the distance to each word and find the shortest; then the gaze point is mapped into the word that is the closest. Otherwise, if the gaze point coordinates do fall directly into a word, the gaze point is mapped into this word.

The Figure 8.1 shows the flowchart of the implementation realized on the raw data.

Implementation
  for $k = 0 \rightarrow Ksentences$ **do**
    for $m = 0 \rightarrow Mgaze\_data\_points$ **do**
      for $n = 0 \rightarrow Nwords$ in *sentence* k **do**
        if *gaze_data_point* m falls into the rectangle containing the *word* n) **then**
          gaze_data_point m is mapped into the corresponding *word* n
          word_Id of *gaze_data_point* $m \leftarrow Id$ of *word* n
        else
          word_Id of *gaze_data_point* $m \leftarrow 0$
        end if
      end for
      if *word_Id* of *gaze_data_point* $m == 0$ **then**
        for $n = 0 \rightarrow Nwords$ in *sentence* k **do**
          Calculation of the distance between the *gaze_data_point* m and the *word* n
          if *distance_calculated* == shortest distance **then**
            word_Id of *gaze_data_point* $m \leftarrow Id$ of *word* n
          end if
        end for
      end if
    end for
  end for

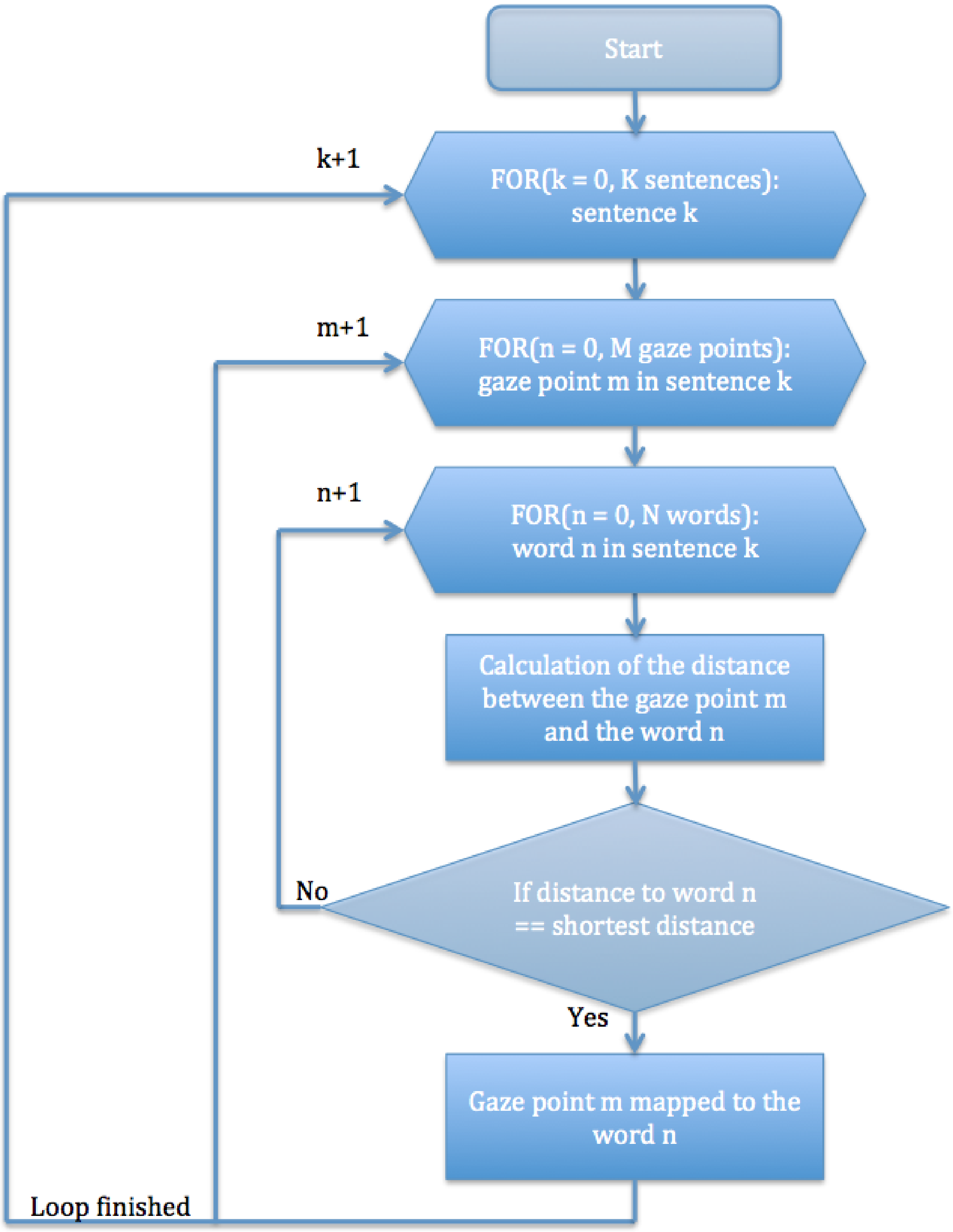

Figure 8.1: Implementation flowchart

Baseline implementation

The baseline algorithm described in Chapter 7 is the one that has been implemented firstly for this project.

As described in the previous section, there are three lists containing respectively the gaze information, the speech information and the graphical information. The goal of this algorithm is to follow the line and to detect the next_line event in order to avoid the vertical inaccuracy.

To do that, we start on the first line. For each gaze point, we look if the line is started; it means that we look at the gaze point to see if it is mapped onto one of the first words of the line. If it is, we consider the line started and we can follow the reading path to detect the next_line event. Then we map the gaze points into the corresponding words on the first line. It uses the same principle that in the Chapter 7. First we check if the gaze point falls directly onto a word and if not, we calculate the nearest distance to the word. The gaze point is mapped only if the distance is less than 200 pixels.

The distance of 200 pixels has been chosen in order to exclude the gaze points too far away from any word. However, it has also been chosen in order to keep the gaze points far from the word because of offset or drifts.

If a gaze point is located into the last 30% of the line, we check if the next gaze point goes backward. If the distance between the actual gaze point and the next gaze point is over 70% of the line; it means that a next_line event has happened. Then we are on the second line. And it goes on like this until all the gaze points are looked at.

The value of 30% has been chosen in order to have a certain tolerance. It is possible that the last word of the line is quite long and so the gaze points won't be at the very end of the line. This tolerance value is also why we chose the 70% value to detect a transition to the next line. We do not expect the subjects to make a 100% go back. Because their last gaze point is not at the very end of the line and the next gaze point on the next line is not at the very beginning of the line.

The Figure 8.2 shows the flowchart of the algorithm used to compensate the vertical inaccuracy.

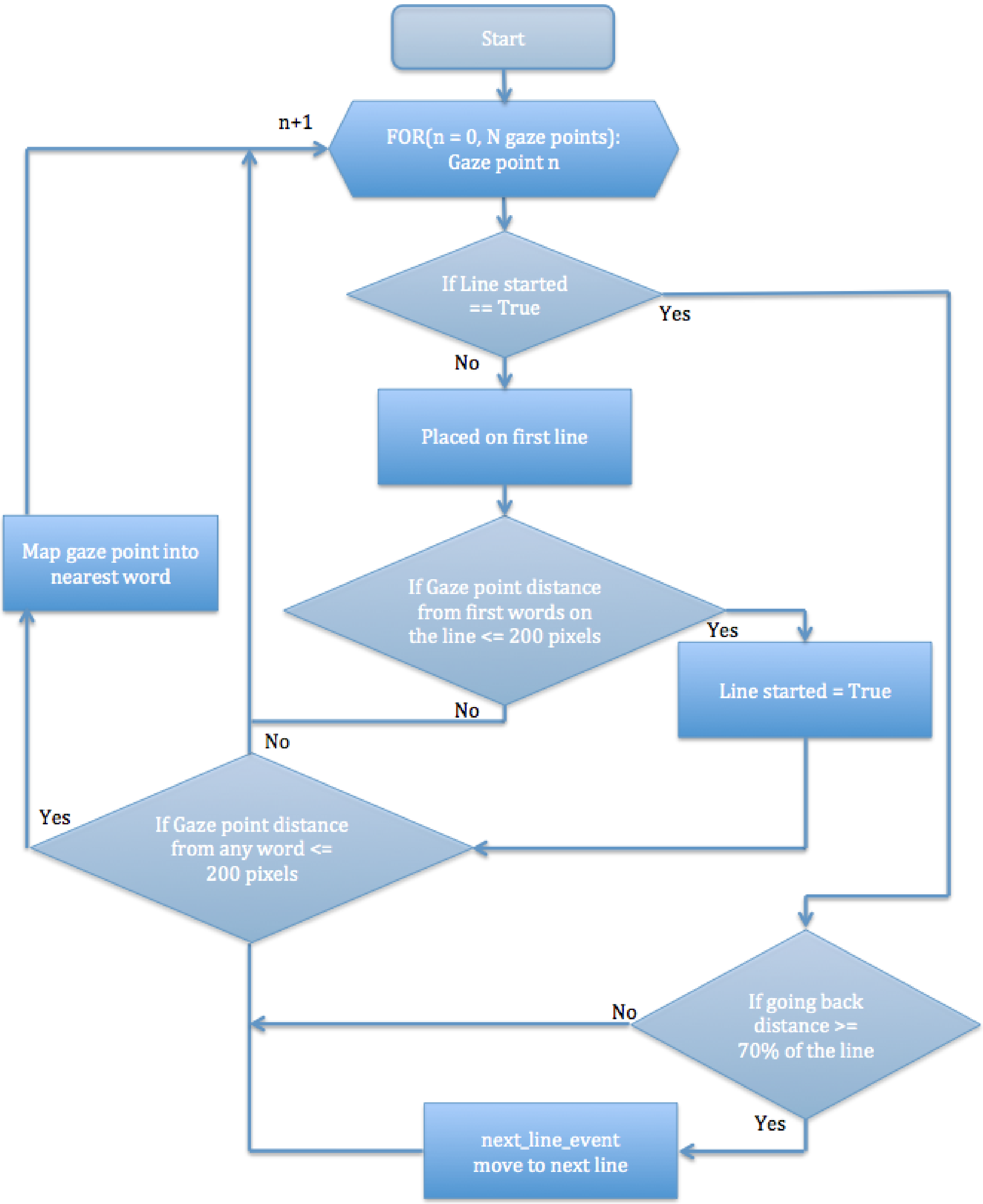

Figure 8.2: Baseline algorithm flowchart

Baseline algorithm

```
line = 0
line_started = false
for n = 0 → Ngaze_data_points do
    if line_started == True then
        if (the x_coordinate of the previous gaze_data_point n−1 − the x_coordinate of the actual gaze_data_point n ≤ 70% of the line_length then
            next_line_event
            line ← line + 1
        end if
        if the distance of the gaze_data_point n to a word ≤ 200 pixels then
            the x_coordinate of the gaze_data_point n is mapped into the word m
            word_Id of gaze_data_point n ← Id of word m
        else
            word_Id of gaze_data_point n ← 0
        end if
    else
        line ← 0
        if the distance of the gaze_data_point n to a word ≤ 200 pixels AND word_mapped ≤ 5th_word of the sentence then
            line_started ← True
            the x_coordinate of the gaze_data_point n is mapped into the word m
            word_Id of gaze_data_point n ← Id of word m
        else
            line_started ← False
            word_Id of gaze_data_point n ← 0
        end if
    end if
end for
```

Implementation to smooth the data

The algorithm for smoothing the data described in Chapter 7 is the one that has been implemented secondly for this project.

Once again, as described in the previous section, there are three lists containing respectively the gaze information, the speech information and the graphical information. The goal of this algorithm is to remove the gaze points that correspond to the erratic movements of the eyes. The gaze points do not represent a fixation so it is not interesting for us.

To do that, we look at the gaze points in the list containing the gaze information. And we check if for a gaze point, the previous gaze point is equal to the following and if the time elapsed is less than 200 milliseconds. If for a gaze point all the conditions are gathered, the gaze point is removed from the list. And it goes to the following one.

The Figure 8.3 shows the flowchart of the algorithm used to compensate the erratic movements of the eyes.

Smoothing algorithm

for $k = 0 \rightarrow Ksentences$ **do**
  for $n = 0 \rightarrow Ngaze\_data\_points$ **do**
    if n of *gaze_data_point* $n \neq 0$ AND n of *gaze_data_point* $n \leq (Ngaze\_data\_points\ -1)$ **then**
      if the *word_Id* of the previous *gaze_data_point* $n - 1 ==$ the *word_Id* of the following *gaze_data_point* $n + 1$ AND the *time* of the following *gaze_data_point* $n + 1$ $-$ the *time* of the previous *gaze_data_point* $n - 1 \leq 200$ milliseconds **then**
        remove *gaze_data_point* n from the list
      end if
    end if
  end for
end for

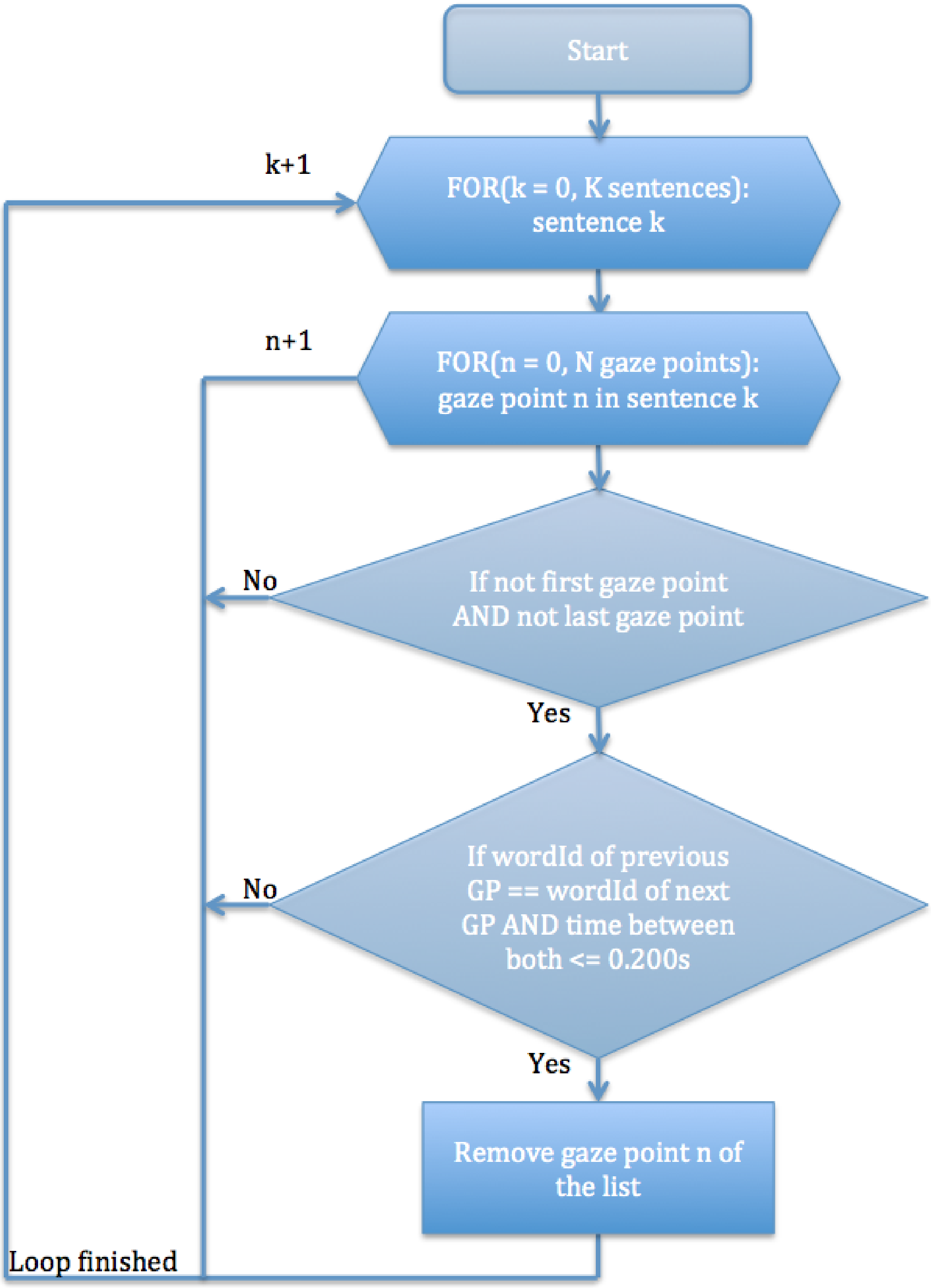

Figure 8.3: smoothing algorithm flowchart

Part IV

Evaluation

Contents

This part will firstly describe the tests that were conducted. Then the results obtained after processing the data collected during these tests will be given. In the last chapter, a discussion will be made on the results obtained.

Chapter 9

Tests Conducted

Some tests have been conducted with the *Reading Tutors* application and eye gaze tracking. As explained in Chapter 2, the purpose of the *Reading Tutors* application is to analyze dyslexia in reading of dyslexic subjects and to help these subjects improving their reading. The basis of the application is speech analysis and in order to have a better accuracy in the word recognition, the eye gaze tracking modality is added.

Tests setup

Two schools having dyslexic students have been contacted and some of these students have been volunteers to be test subjects for the application. In total, there were 7 volunteers for the experiment. Because of the quasi-portability of the eye gaze tracker, we have been able to setup the experiments in both schools. The Figure 9.1 shows the setup in one of the schools with one of the subjects. On the image A, the setup is shown in the empty room; on the image B, the experimenter and the participant are present. For each experiment, only the participant, the experimenter and the second experimenter were present.

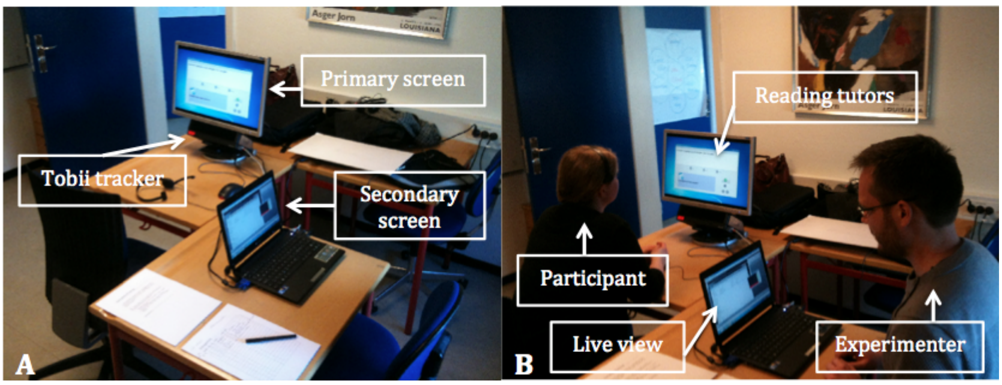

Figure 9.1: Experiment room and setup

The physical setup of the computer and of the Tobii tracker was as it is shown in the Figure 9.2. These measurements are used to set up the tracker with the *Tobii X Configuration tool* in order to have the best calibration possible.

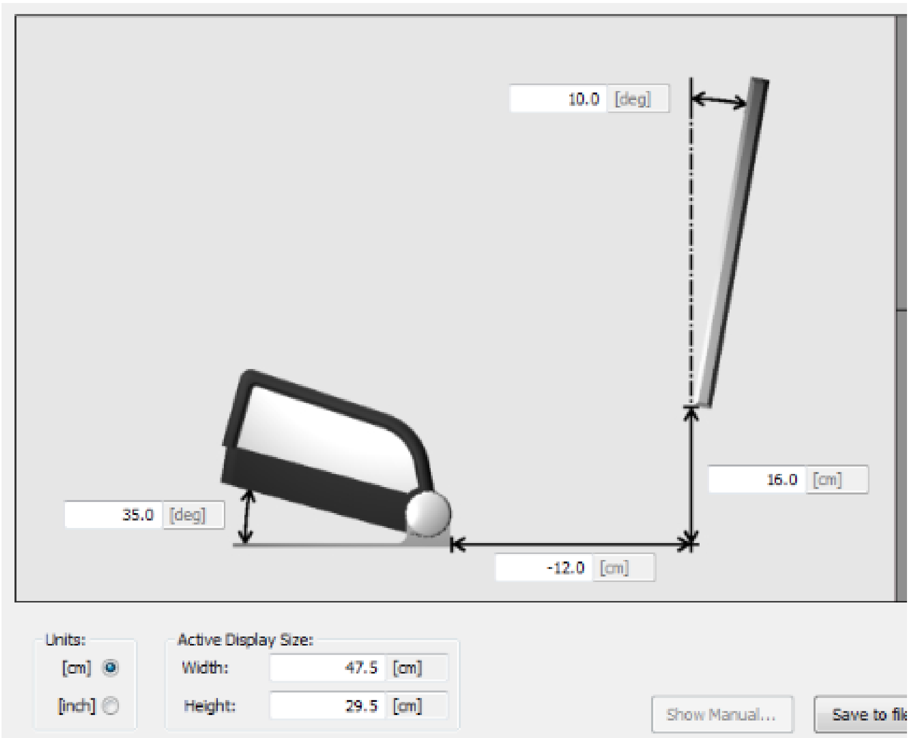

Figure 9.2: Computer and Tobii tracker physical setups

The process of the experiment is the same for all the participants. First, the experimenter explains the purpose of the experiment to the participant and collect information about them. Then he asks the participant to sign a form so that their data can be collected and reused. After this, the experimenter explains how the experiment will take place; the GUI of the application is presented (the Figure 9.3 shows it) as well as the action a click on a button will lead to.

The participant proceeds then to the calibration step. The calibration is done thanks to the *Tobii Studio Gaze Analysis Software*. The result of the calibration is like in the Figure 9.4. The calibration is good when the lines starting from the circles are as short as possible. The left part with the red lines represents the calibration for the left eye and the right part with the green lines represents the calibration for the right eye. This calibration is really good, for example.

After that, the experimenter presents what the participant is supposed to do: the

Figure 9.3: GUI of the Automatic Reading Tutor application

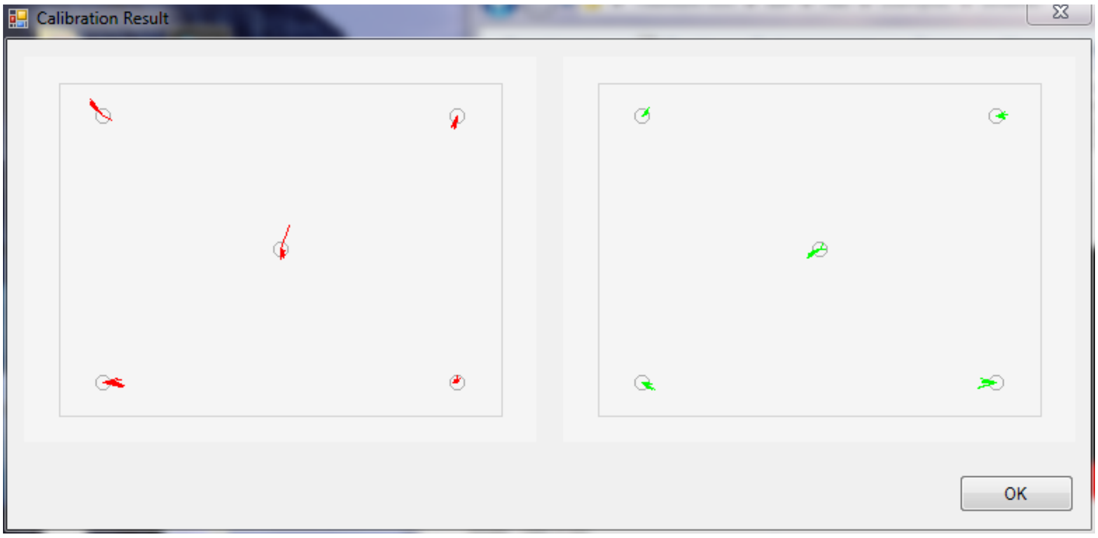

Figure 9.4: Calibration result

participant clicks on the *Læs* button to listen to the sentence first. Then they click on the *Ja* button to read the sentence aloud. The speech recording and the speech recognition start at this moment. Once the participant has finished to read the sentence, they click on the *Stop* button. The recording stops. Then, they click on the *Next* button to proceed to the next sentence. In total, there are 23 sentences of different lengths. The eye gaze tracker starts collecting data before the calibration step and stops after the last sentence.

Outputs

The data collected during these experiments are the one presented in the Table 9.1. It is raw data and these data needs to be analyzed in order to have the tracking error rate for the eye gaze tracking.

Table 9.1: Data collected during the experiments

X coordinate	This is the X coordinate for a word in a sentence. It represents the X coordinate of the pixel at the top left corner of the rectangle containing the word.
Y coordinate	This is the Y coordinate for a word in a sentence. It represents the Y coordinate of the pixel at the top left corner of the rectangle containing the word.
Width	This is the length in pixels of the rectangle containing the word.
Height	This is the height in pixels of the rectangle containing the word.
Duration time	This is the time that it took for the participant to read a sentence. It is under the 00:00:000 form (minutes:seconds:milliseconds).
Start time	This is the time at which the participant starts saying the word in seconds. It starts at 0 when the recording starts and the times are added up until the end of the session of the participant. It is under the 0.0000000000000000 form (seconds.milliseconds).
End time	This is the time at which the participant stops saying the word in seconds. It starts at 0 when the recording starts and the times are added up until the end of the session of the participant. It is under the 0.0000000000000000 form (seconds.milliseconds).
Word Id	This is the Id for a word in a sentence. It starts from 1 at each sentence.
Miscue type	This is the number qualifying a miscue done by a participant. It goes from *none* to 3 in function of the importance of the miscue.
Click time	This is the time at which the participant clicks on the *Ja* button to start reading the sentence. It is under the 00:00:00.000 form (hours:minutes:seconds.milliseconds). This is the actual time of the computer.
Gaze time	This is the time at which a gaze point data is collected. It is under the 00:00:00.000 form (hours:minutes:seconds.milliseconds). This is the actual time of the computer.
Gaze point X	It is the X coordinate in pixels of the gaze direction of the participant at a certain time. It is collected from the Tobii tracker.
Gaze point Y	It is the Y coordinate in pixels of the gaze direction of the participant at a certain time. It is collected from the Tobii tracker.

Chapter 10

Results

In order to evaluate the processing done on the raw data that were collected during the tests, the tracking error rate is calculated. As explained in Chapter 7, it is calculated by dividing the number of word changes when it is incorrect plus the number of missing word changes, by the reference number of actual word changes. An evaluation is made for the different types of processing realized: no processing, baseline processing and smoothing processing. The results are given in the following sections.

Results with no processing

The results obtained in this section are obtained directly from the raw data. The tracking error rate is calculated directly from the data with no processing and algorithms. These results are presented in the Table 10.1.

Table 10.1: No processing results

Subjects	Tracking error rate
Subject 01	1.288
Subject 02	1.387
Subject 03	1.383
Subject 04	1.522
Average	**1.395**

Results with the baseline processing

The results obtained in this section are obtained by extracting information from the raw data. The tracking error rate is calculated on the processed data. The baseline algorithm presented in Chapter 7 and in Chapter 8 is applied. These results are presented in the Table 10.2.

Table 10.2: Baseline processing results

Subjects	Tracking error rate	Improvement
Subject 01	1.317	−2.21%
Subject 02	1.246	10.15%
Subject 03	1.233	10.83%
Subject 04	0.935	38.57%
Average	**1.183**	**14.33 %**

Results with the processing to smooth the data

The results obtained in this section are obtained by applying the smoothing algorithm to the previous processed data. The tracking error rate is calculated on this new processed data. The smoothing algorithm presented in Chapter 7 and in Chapter 8 is applied. These results are presented in the Table 10.3.

Table 10.3: Smoothing processing results

Subjects	Tracking error rate	Improvement (basic processing)	Improvement (raw data)
Subject 01	1.260	4.48%	2.21%
Subject 02	1.211	2.76%	12.69%
Subject 03	1.195	3.40%	13.60%
Subject 04	0.918	3.72%	39.69%
Average	**1.146**	**3.02 %**	**17.05 %**

We conducted a matched-pairs test (as described in [10]) of the tracking error rate values obtained with no processing and of the tracking error rate values obtained with both algorithms applied. This test evaluates the probability that between to sets of measurements, there is a significant difference. Consequently, the efficiency of the algorithm can be evaluated. The p-value obtained is 0.044. The improvement realized thanks to the algorithms is thus significant.

We also conducted a matched-pairs test of the tracking error rate values obtained with the baseline processing and of the tracking error rate values obtained with both algorithms applied. The p-value obtained is 0.052. The improvement realized thanks to the smoothing algorithm is thus significant.

The tracking errors have been clustered based on the type of the fixations of the errors. Three groups have been made: normal fixation, regression and skip fixation (between the previous fixation and the actual fixation, one or more words have been skipped). The corresponding percentages are given in the Table 10.4.

Table 10.4: Errors cluster percentages

Subjects	normal fixation	regression	skip fixation
Subject 01	23.60%	11.80%	64.60%
Subject 02	11.41%	7.51%	81.08%
Subject 03	15.90%	10.70%	73.39%
Subject 04	20.89%	16.03%	62.98%
Average	**17.98 %**	**11.51 %**	**70.51 %**

The main source of errors is the skip fixation. Most of the time, we do not find hypothesis changes that correspond to reference changes. That is why this cluster has a high percentage. The percentage of the errors caused by the regressions is quite high. When reading, regressions happen but at a low frequency. Here, the percentage is close to the percentage of the errors caused by normal fixations, fixations which happen at a higher frequency. Detecting the regressions may help reducing the tracking error rate.

Line tracking error rate

We calculated the line tracking error rate in order to evaluate how well the baseline algorithm performs. We obtain the line tracking error rate by dividing the actual number of *next_line_events* by the reference number of *next_line_events* and then by subtracting this number to 100. The value found is 9.72% as it can be seen in the Table 10.5. We can observe that the algorithm is efficient. It detects 90.28% of the reference *next_line_events*.

Table 10.5: Line tracking error rate

Subjects	Line tracking error rate
Subject 01	22.22
Subject 02	5.56
Subject 03	11.11
Subject 04	0.00
Average	**9.72**

This line tracking error rate is important. For the ASR system, knowing on which line is the user reading gives a valuable information; sometimes as valuable as the word the subject is looking at.

Chapter 11

Discussion

For this system, all the algorithms used give a final tracking error rate value of 114.6%. As explained in Chapter 7, the tracking error rate is calculated based on wrong changing points and missed changing points. In [21], a previous implementation on the same data was done and the final tracking error rate value was of 126%.

The tracking error rate value seems to be abnormally high but this value is actually useful as described in [21]. As explained in Chapter 5, there are three main origins of the inaccuracy of the eye gaze trackers; and this inaccuracy is displayed by the tracking error rate value obtained. The three main sources of errors are the calibration drift, the quality of the setup and the physical characteristics of the eyes. Even with a good calibration, after a certain time, the accuracy decreases during a session. The quality of the setup depends on the eye gaze tracker used and on the quality of the calibration. The physical characteristics of the eyes induce an uncertainty of $\pm 0.5\,^{\circ}$ of visual field of focus. The result of these errors on the gaze points is an offset with some noise.

The inaccuracy of the eye gaze trackers while used for reading process is mainly a vertical inaccuracy as explained in Chapter 5. Horizontal inaccuracy is much more less pronounced; which is positive because it will be harder to track the horizontal inaccuracy.

For the tests, the graphical interface use characters with an average height of 24 pixels for the text. By considering that the subject was approximately at 60 centimeters of the tracker. The character on the screen represents an angle of $\pm 0.88\,^{\circ}$; which is just above the threshold of $\pm 0.5\,^{\circ}$ imposed by the physical characteristics of the eyeball. This represents one cause of the vertical inaccuracy and also induces tracking errors.

The Figure 11.1 shows plots of the gaze data where we can find these errors. For both image A and B, the information plotted are obtained from the 11^{th} sentence of the 1^{st} subject. Coordinates in pixels are plotted depending on the time in seconds. The blue points represent the raw gaze data points obtained from the eye gaze tracker. The green line represents the reference change points and the red line represents the hypothesis (or gaze) change points. The image A represents the plot of the X coordinate in function of the time and the image B represents the plot of the Y coordinate in function of time.

The offset we just broached is noticeable in the image B; the raw gaze points are all located at a minimum of approximately 30 pixels above the real Y coordinate of the words. The baseline algorithm described in Chapter 7 makes it possible to compensate this offset. In the image B, we can observe that the green line follows mostly the red line and not the raw gaze points.

The problem of the ascending and descending paths has been described in Chapter 5. In the image B, it is easy to spot these ascending and descending paths. There are roughly three groups of raw gaze points that should normally be well differentiate on three different stages. It is possible to differentiate the three groups but for each group, the raw gaze points are not equally spread on the same Y coordinate. It is easily noticeable that the Y coordinates of the raw gaze points within a group increase. For this sentence, we only observe ascending paths. By looking at the hypothesis change points, we see that the baseline algorithm compensates these ascending and descending paths; and does not follow the raw gaze points.

In the image A, we can easily spot the *next_line_events* and the baseline algorithm is able to follow these events too. There are three groups of raw gaze points increasing when looking at the X axis. Then, there is a huge gap and the raw gaze points are back at the bottom of the scheme.

The Figure 11.2 shows plots realized in the same way as Figure 11.1. For both images A and B, the information plotted are obtained from the 14^{th} sentence of the 3^{rd} user.

One cause of the high tracking error rate can be observable in the Figure 11.2. Between approximately 2 seconds and 8 seconds, the eye gaze tracker did not acquire reliable data from the user. We have about 6 seconds of missing data. As a result, the first *next_line_event* is not detected, which induces errors in the hypothesis change points. This can be caused by a bad detection of the eyes because of the tracker or by a user who was not in the detection field of the tracker at a certain moment.

Another source of error is the lack of real reference data for the calculation of the tracking error rate. It is not possible to know exactly what a person is looking at. In order to have the best reference data possible, we use the modality that gives the best accuracy which is the speech. Using the speech as a reference for the gaze points induces a specific difficulty: the *eye-voice-band*. It it the difference between the eye getting the information and the voice saying the information. We have to get the information from the eyes about the word first in order to be able to read it aloud after. When calculating the tracking error rate, we compare the time of the reference with the time of the hypothesis to consider a hypothesis change as an error.

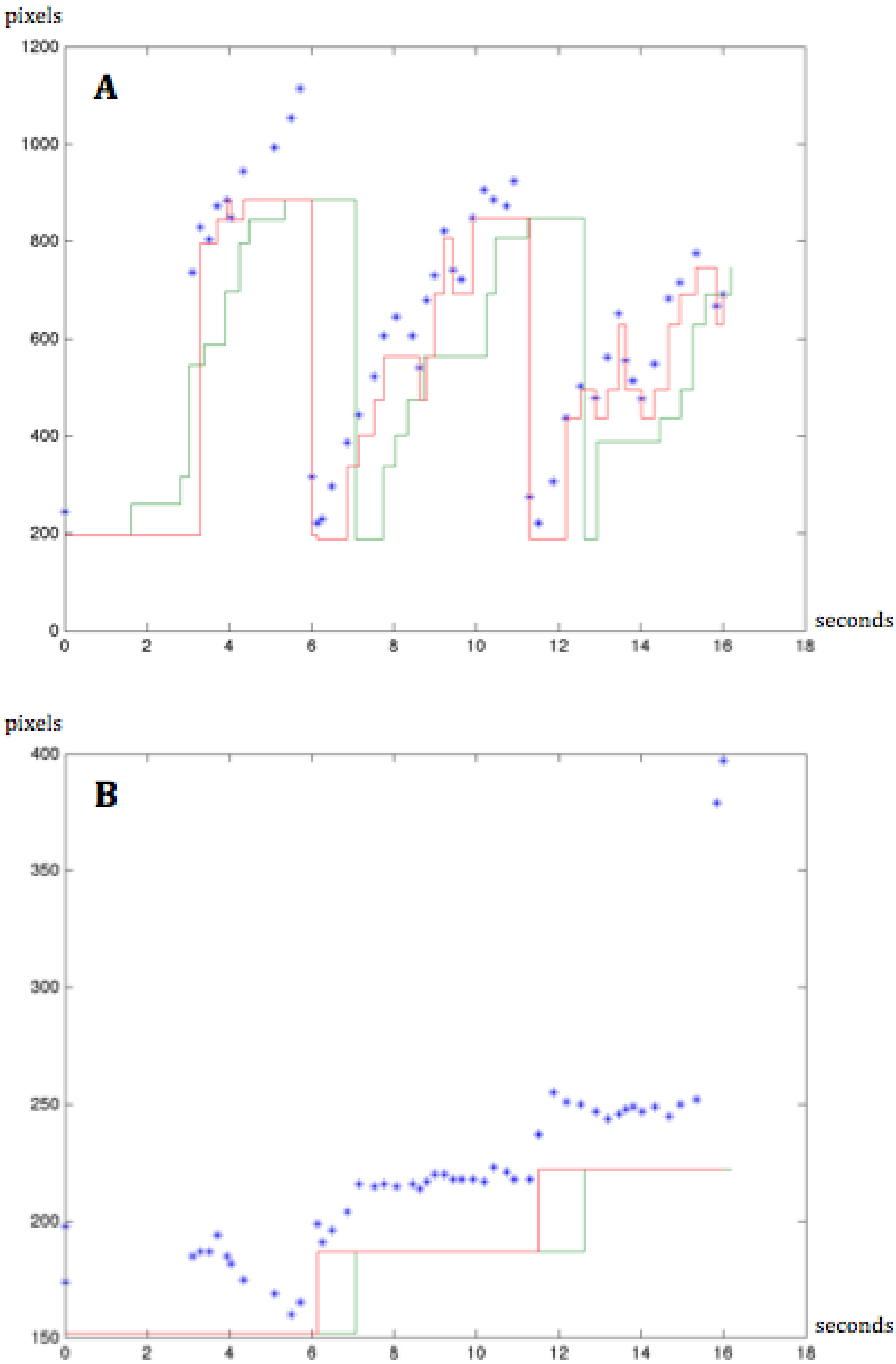

Figure 11.1: Gaze data plots for sentence 11 of Subject 01

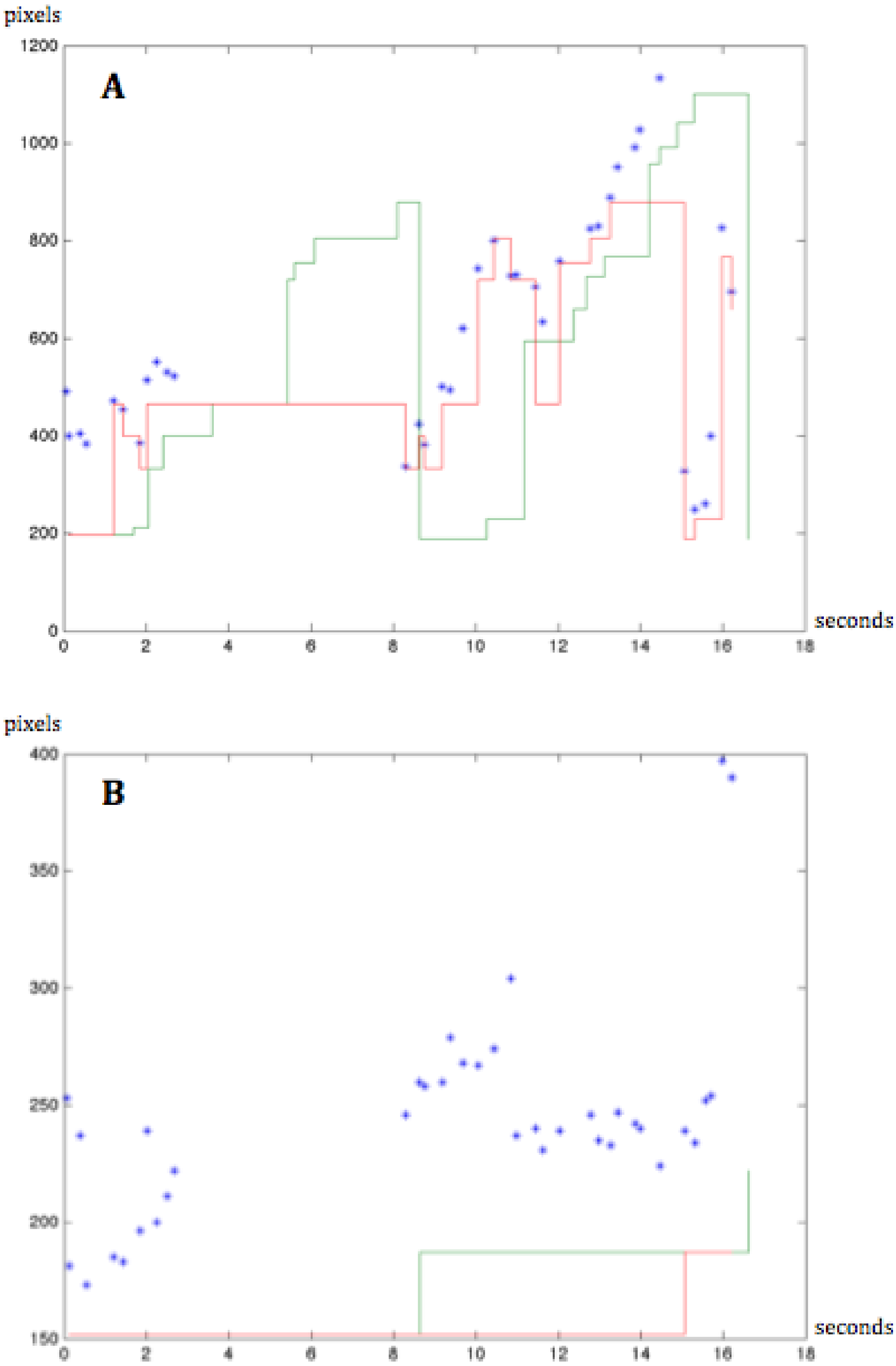

Figure 11.2: Gaze data plots for sentence 14 of Subject 03

With this eye-voice-band, it induces an offset as well as errors in the calculation of the tracking error rate.

For both Figure 11.1 and Figure 11.2, it is noticeable that the hypothesis change points happened before the corresponding reference change points in a constant way. It is consistent because we need to look at the word before to be able to read it aloud as explained before. We can try to identify this constant and to determine the constant value empirically by looking at the plots. If we transpose the hypothesis change points about 410 milliseconds later, the points will fall on the reference change points. By implementing the transposition, the tracking error rate can be lowered of approximately 26%.

In Chapter 4, It was described that dyslexic persons have irregular eye movements because of their difficulties while reading. It results in a higher fixation frequency and a higher regression frequency. In both Figure 11.1 and Figure 11.2, for the X coordinate, the line corresponding to the hypothesis change points appears to be irregular with a lot of descending peaks. These descending peaks represent small regressions.These small regressions can mean either the dyslexic person has misread a word and goes back to correct their error, either the word was well read but the dyslexic person goes back to be sure of what they just read. But the dyslexic person hardly ever says the correction aloud; what causes tracking errors. The subject goes back to correct an error but does not say the error aloud; the speech does not match the eye gaze tracking anymore.

A solution to compensate these small regressions would be to consider only the gaze points that go forward and only the *next_line_events*. But in this case, if a regression is made and the correction is said aloud, then it also causes tracking errors.

Because of all these irregularities, especially while dealing with dyslexic persons, the tracking error rate is high. Even with the algorithms used to reduce the vertical inaccuracy and to smooth the data that lower the tracking error rate, it does not drop below 100%

In the appendices, the plots of all the four subjects for the sentences 11 and 14 are given.

Part V

Conclusion

Contents

This part concludes the whole project. Firstly, an overview of what was wanted to be done and of what was done is detailed. Then possible improvements are broached.

Chapter 12

Conclusion

The purpose of this project is to understand reading process in order to be able to follow this process with an eye gaze tracker. This way, mapping gaze points into corresponding words would be possible without too many errors. As a consequence, the fusion of eye gaze tracking and speech recognition would give better results than only speech recognition.

During tests carried out to collect the data, an eye gaze tracker was used with different subjects. While recording the subjects reading the texts, the eye gaze tracker recorded the gaze points location of the subjects. A lot of data were collected during this experiment and we used an offline implementation to process all the data. The information obtained from manual transcription are used as reference and the information from the eye gaze tracker are the hypothesis. The evaluation is performed by calculating the tracking error rate. The goal is to have the lowest tracking error rate while using only the hypothesis data.

In order to realize that, we should have a good understanding of the reading process and of the eye gaze tracking systems; as well as a good understanding of the issues existing while using eye gaze tracking system for reading process. Some issues are very common and algorithms exist to compensate these issues. Some other issues are more specific to our system.

To obtain the best results possible, two mains algorithm have been implemented: the baseline algorithm and the algorithm to smooth the data.

The baseline algorithm follows the reading process line by line. The tracking of the reading process waits to detect a *next_line_event*. The algorithm maps the gaze points into the words of the corresponding line. With this algorithm the vertical inaccuracy is compensated; the gaze points are only mapped into the words of the corresponding line. And the gaze points too far away are disregarded.

The algorithm implemented to smooth the data corrects the erratic movements of the eyes. The algorithm looks at the gaze data points and detects an erratic gaze data point, which is removed. The gaze data point removed is not considered as a fixation below the threshold of 200 milliseconds.

With these two algorithms, the final tracking error rate is 114.6%. The previous tracking error rate found in [21], is 126%. With a modality such as eye gaze tracking, it is difficult to have a really low error rate. But when combined with ASR, an improvement is realized as in [21]. Even with a tracking error rate this high, the system is still useful. Moreover, there is a lot of parameters that can be tweaked.

For example, we observed that the subject made a lot of small regressions to check their reading and to correct for themselves misreading. It would be possible to only consider the gaze points that go forward and to ignore these small regressions.

We also observe a constant offset between the time when a gaze point is recorded and the time when the word corresponding to the gaze point is read aloud. It would be possible to estimate this offset thanks to an algorithm and to transpose either the speech to the gaze or the gaze to the speech.

Bibliography

[1] David Beymer and Daniel M. Russell. Webgazeanalyzer: a system for capturing and analyzing web reading behavior using eye gaze. In *CHI '05 Extended Abstracts on Human Factors in Computing Systems*, pages 1913–1916, 2005.

[2] Richard A. Bolt. Eyes at the interface. *Proceedings of the 1982 Conference on Human Factors in Computing Systems*, pages 360–362, 1982.

[3] Hannah Faye Chua, Julie E. Boland, and Richard E. Nisbett. Cultural variation in eye movements during scene perception. *Proceedings of the National Academy of Sciences, PNAS*, 102(35):12629–12633, 2005.

[4] Macdonald Critchley. *The dyslexic child.* Heinemann Medical, 2. ed. edition, 1970.

[5] Connor Dickie, Roel Vertegaal, Changuk Sohn, and Daniel Cheng. eyelook: Using attention to facilitate mobile media consumption. *Proceedings of the 18th annual ACM symposium on User interface software and technology*, pages 103–106, 2005.

[6] Heiko Drewes. *Eye Gaze Tracking for Human Computer Interaction.* Ludwig-Maximilians-Universität München, 2010.

[7] Andrew T. Duchowski. A breadth-first survey of eye-tracking applications. *Behavior Research Methods, Instruments, & Computers*, 34(4):455–470, 2002.

[8] Irenaus Eibl-Eibesfeldt. *Die Biologie des menschlichen Verhaltens: Grundriss der Humanethologie.* Piper, 5. ed. edition, 2004.

[9] Paul M. Fitts, R. E. Jones, and J. L. Milton. Eye movements of aircraft pilots during instrument-landing approaches. *Aeronautical Engineering Review*, 9(2):24–29, 1950.

[10] L. Gillick and S.J. Cox. Some statistical issues in the comparison of speech recognition algorithms. In *Acoustics, Speech, and Signal Processing, 1989. ICASSP-89., 1989 International Conference on*, pages 532–535 vol.1, 1989.

[11] Jia Guo and Gary Feng. How eye gaze feedback changes parent-child joint attention in shared storybook reading? an eye-tracking intervention study. *Eye Gaze in Intelligent Human Machine Interaction*, pages 1–8, 2011.

[12] Aulikki Hyrskykari. Utilizing eye movements: Overcoming inaccuracy while tracking the focus of attention during reading. *Computers in Human Behavior*, 22(4):657–671, 2006.

[13] Aulikki Hyrskykari, Paivi Majaranta, Antti Aaltonen, and Kari-Jouko Raiha. Design issues of idict: a gaze-assisted translation aid. *Proceedings of the 2000 symposium on Eye tracking research & applications*, pages 9–14, 2000.

[14] Foo Tun Keat, Surendra Ranganath, and Yedatore. V. Venkatesh. Eye gaze based reading detection. *Conference on Convergent Technologies for the Asia-Pacific Region*, 2:825–828, 2003.

[15] Metrovision. Instruments for measuring eye movements. `http://www.metrovision.fr/mv-po-notice-us.html`.

[16] Abdallahi Ould Mohamed, Matthieu Perreira Da Silva, and Vincent Courboulay. A history of eye gaze tracking, 2007.

[17] IT University of Copenhagen. Itu gaze tracker. `http://www.gazegroup.org/downloads/23-gazetracker`.

[18] OpenCV. Eye detection and tracking. `http://opencv-code.com/tutorials/eye-detection-and-tracking/`.

[19] Jakob Schou Pedersen. User centred design of a multimodal reading training system for dyslexics, 2008.

[20] Alex Poole and Linden J. Ball. Eye tracking in human-computer interaction and usability research: Current status and future. In *Encyclopedia of Human-Computer Interaction*, 2005.

[21] Morten Hojfeldt Rasmussen and Zheng-Hua Tan. Fusing eye-gaze and speech recognition for tracking in an automatic reading tutor - a step in the right direction? *submitted to SLATE 2013, France*, 2013.

[22] Keith Rayner. Visual attention in reading: Eye movements reflect cognitive processes. *Memory & Cognition*, 5(4):443–448, 1977.

[23] Keith Rayner and George W. McConkie. What guides a reader's eye movements? *Vision Research*, 16(8):829–837, 1976.

[24] John L. Sibert, Mehmet Gokturk, and Robert A. Lavine. The reading assistant: Eye gaze triggered auditory prompting for reading remediation. *Proc. ACM Symposium on User Interface Software and Technology*, pages 101–107, 2000.

[25] Andreas Sprenger, Birte Neppert, Sabine Koster, Steffen Gais, Detlef Kompf, Christoph Helmchen, and Hubert Kimmig. Long-term eye movement recordings with a scleral search coil-eyelid protection device allows new applications. *Journal of Neuroscience Methods*, 170(2):305–309, 2008.

[26] Pavan Teja. Eye tracking devices in psychology of programming research. `http://www.psychtronics.com/2012/06/eye-tracking-devices-in-psychology-of.html`.

[27] Tobii. Tobii t/x series eye trackers, 2010.

[28] Tobii. Tobii technology the world leader in eye tracking and gaze interaction, 2012.

[29] Roel Vertegaal, Robert Slagter, Gerrit van der Veer, and Anton Nijholt. Eye gaze patterns in conversations: There is more to conversational agents than meets the eyes. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 301–308, 2001.

[30] zafersavas. Trackeye : Real-time tracking of human eyes using a webcam. `http://www.codeproject.com/Articles/26897/TrackEye-Real-Time-Tracking-Of-Human-Eyes-Using-a`.

Appendix: Source Code

The full source code for the project is included on the CD-ROM attached here.

Appendix: Plots of the processed data

The data obtained after processing have been plotted. For all four subjects, two sentences have been chosen: the 11^{th} and the 14^{th}. For each sentence, two plots have been realized: for the X coordinate and for the Y coordinate. Coordinates in pixels are plotted based on the time in seconds. The blue points represent the raw gaze data points obtained from the eye gaze tracker. The green line represents the reference change points and the red line represents the hypothesis change points. The details of all the plots realized are the following:

- **Figure 1** Subject 01, Sentence 11, X coordinate,
- **Figure 2** Subject 01, Sentence 11, Y coordinate,
- **Figure 3** Subject 01, Sentence 14, X coordinate,
- **Figure 4** Subject 01, Sentence 14, Y coordinate,
- **Figure 5** Subject 02, Sentence 11, X coordinate,
- **Figure 6** Subject 02, Sentence 11, Y coordinate,
- **Figure 7** Subject 02, Sentence 14, X coordinate,
- **Figure 8** Subject 02, Sentence 14, Y coordinate,
- **Figure 9** Subject 03, Sentence 11, X coordinate,
- **Figure 10** Subject 03, Sentence 11, Y coordinate,
- **Figure 11** Subject 03, Sentence 14, X coordinate,
- **Figure 12** Subject 03, Sentence 14, Y coordinate,
- **Figure 13** Subject 04, Sentence 11, X coordinate,
- **Figure 14** Subject 04, Sentence 11, Y coordinate,
- **Figure 15** Subject 04, Sentence 14, X coordinate,
- **Figure 16** Subject 04, Sentence 14, Y coordinate,

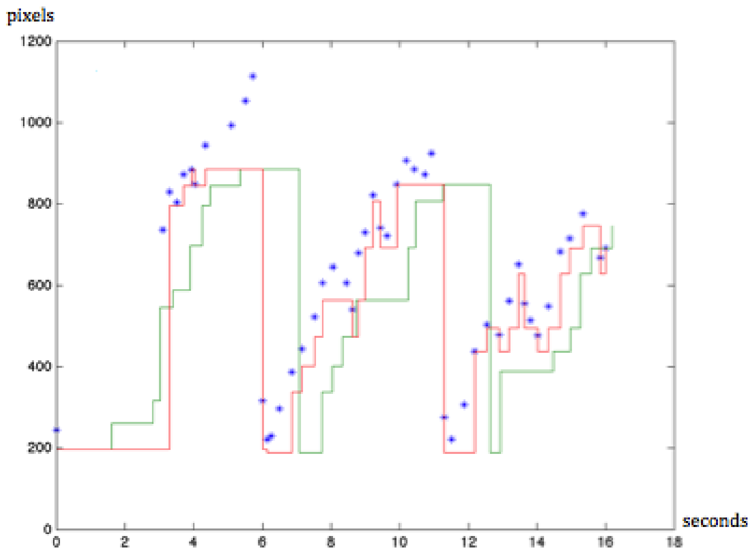

Figure 1: Subject 01 Sentence 11 for X coordinate

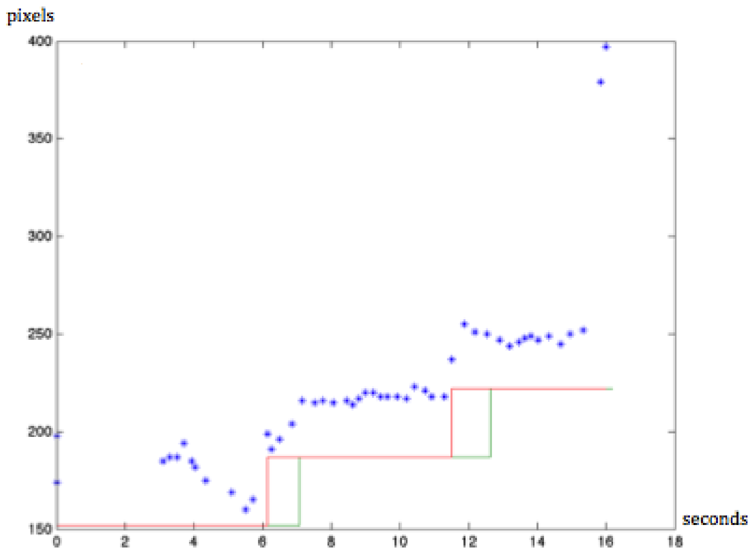

Figure 2: Subject 01 Sentence 11 for Y coordinate

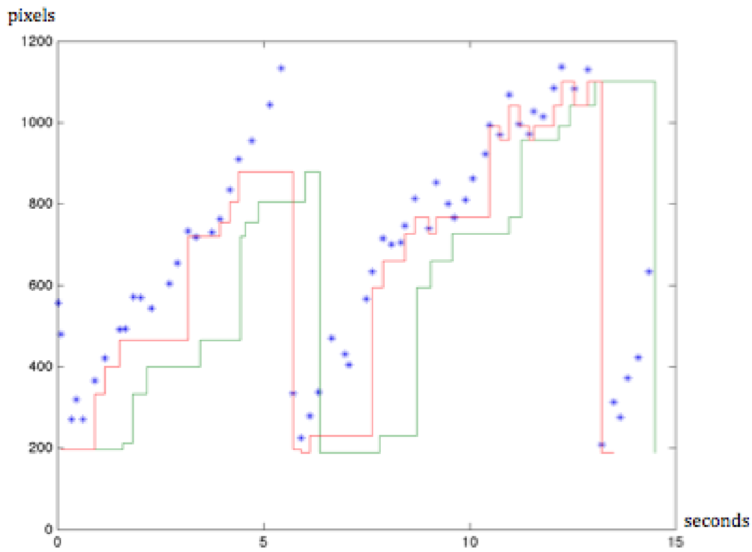

Figure 3: Subject 01 Sentence 14 for X coordinate

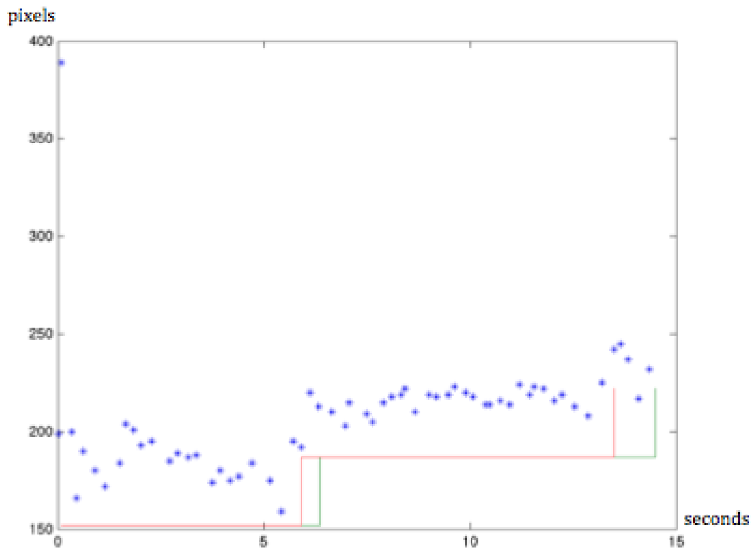

Figure 4: Subject 01 Sentence 14 for Y coordinate

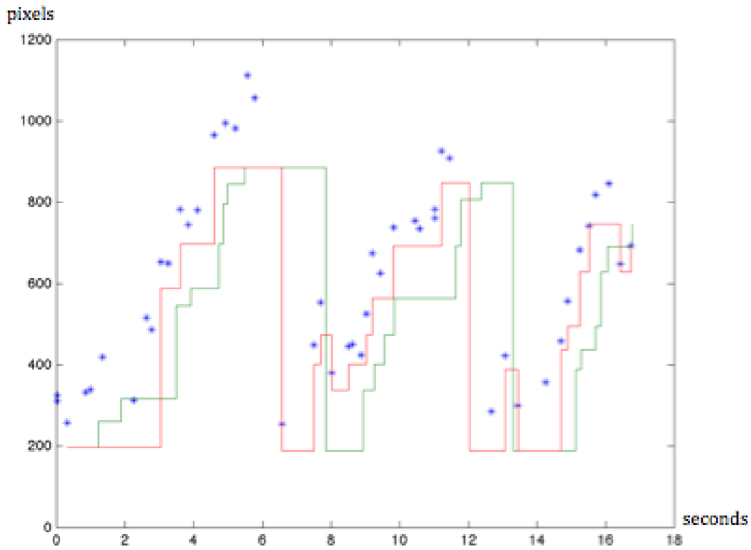

Figure 5: Subject 02 Sentence 11 for X coordinate

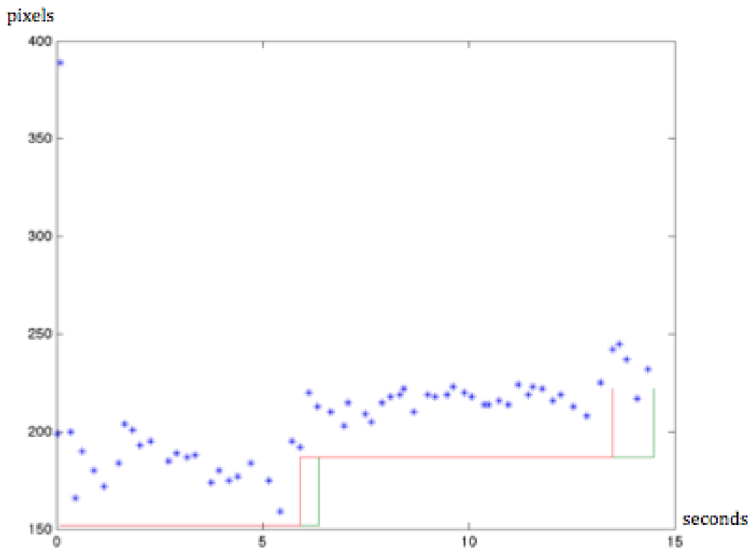

Figure 6: Subject 02 Sentence 11 for Y coordinate

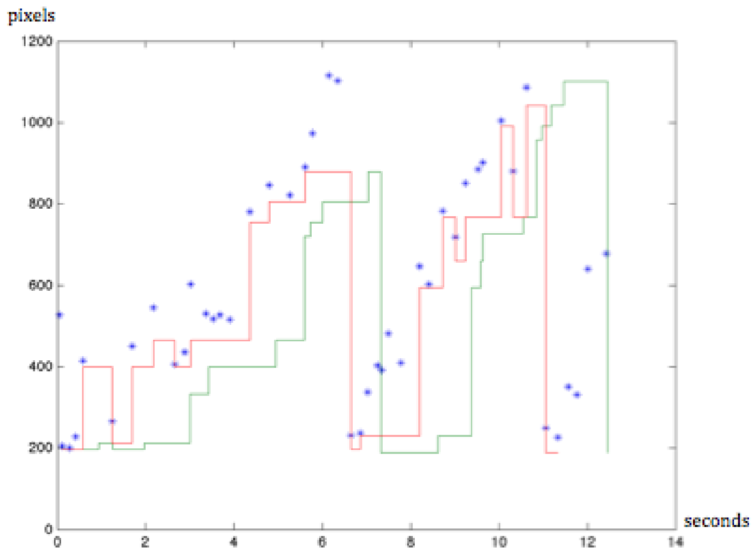

Figure 7: Subject 02 Sentence 14 for X coordinate

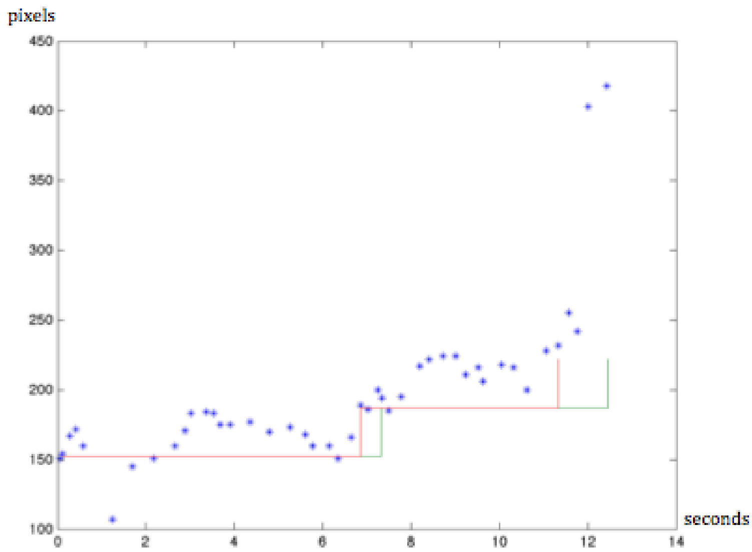

Figure 8: Subject 02 Sentence 14 for Y coordinate

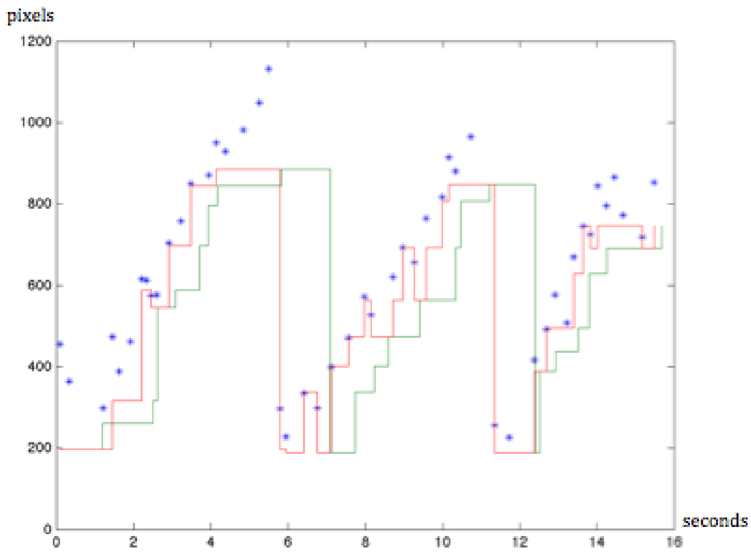

Figure 9: Subject 03 Sentence 11 for X coordinate

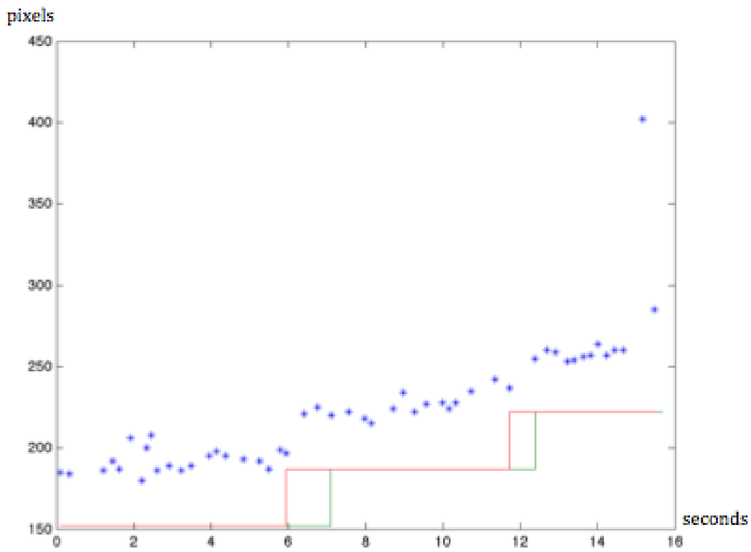

Figure 10: Subject 03 Sentence 11 for Y coordinate

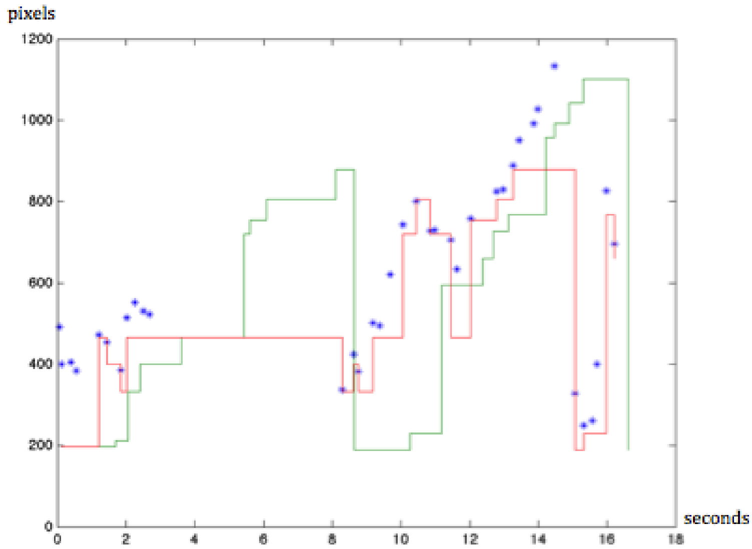

Figure 11: Subject 03 Sentence 14 for X coordinate

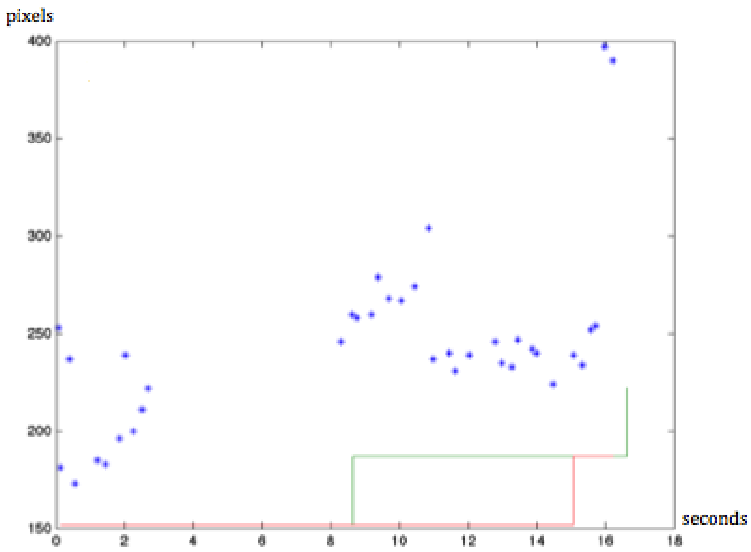

Figure 12: Subject 03 Sentence 14 for Y coordinate

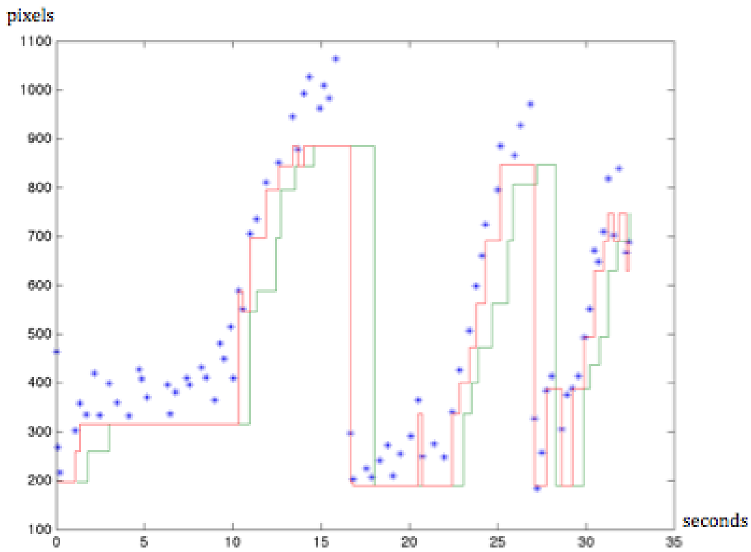

Figure 13: Subject 04 Sentence 11 for X coordinate

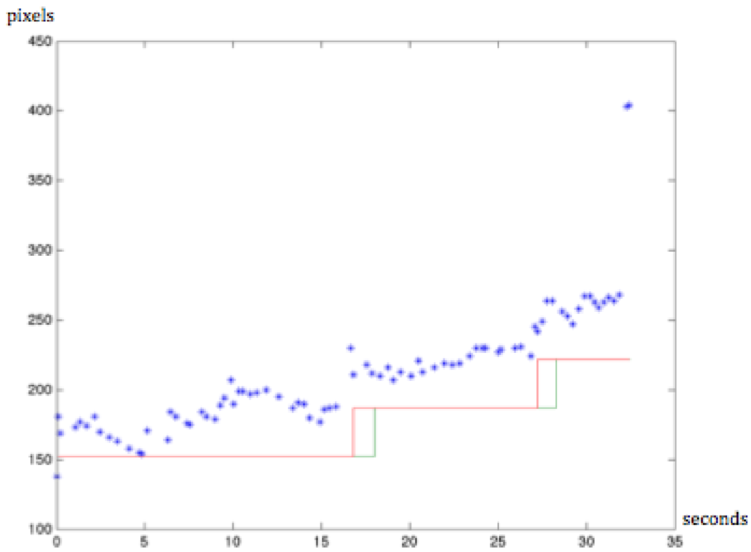

Figure 14: Subject 04 Sentence 11 for Y coordinate

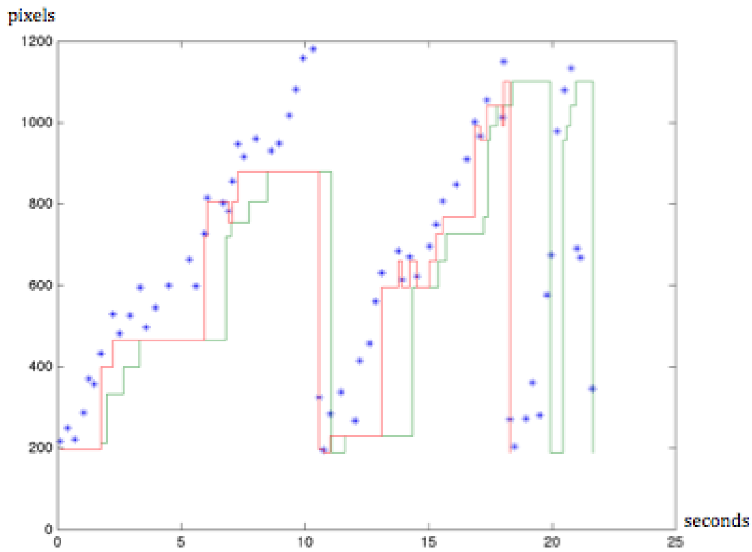

Figure 15: Subject 04 Sentence 14 for X coordinate

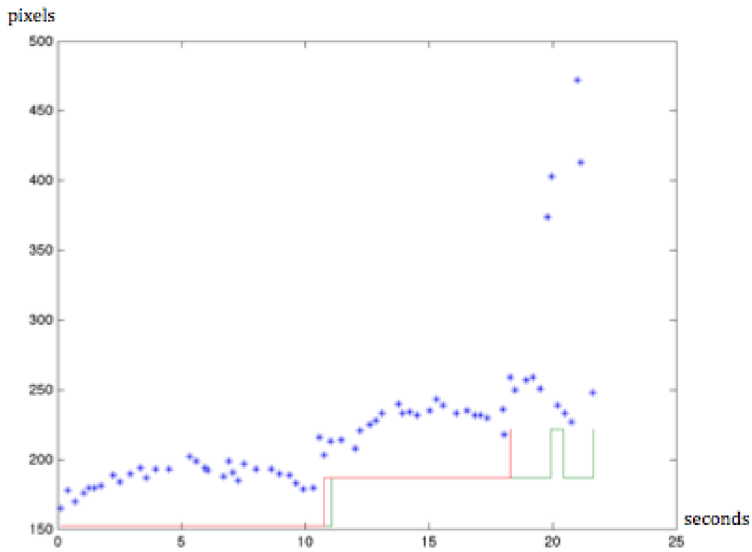

Figure 16: Subject 04 Sentence 14 for Y coordinate