NOVEL EYE FEATURE EXTRACTION AND TRACKING FOR NON-VISUAL EYE-MOVEMENT APPLICATIONS

BY
GEORGIOS DIAMANTOPOULOS

A THESIS SUBMITTED TO
THE UNIVERSITY OF BIRMINGHAM
FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

DEPARTMENT OF ELECTRONIC, ELECTRICAL
AND COMPUTER ENGINEERING
COLLEGE OF ENGINEERING AND PHYSICAL SCIENCES
THE UNIVERSITY OF BIRMINGHAM
FEBRUARY 2010

UNIVERSITY OF
BIRMINGHAM

University of Birmingham Research Archive

e-theses repository

This unpublished thesis/dissertation is copyright of the author and/or third parties. The intellectual property rights of the author or third parties in respect of this work are as defined by The Copyright Designs and Patents Act 1988 or as modified by any successor legislation.

Any use made of information contained in this thesis/dissertation must be in accordance with that legislation and must be properly acknowledged. Further distribution or reproduction in any format is prohibited without the permission of the copyright holder.

ABSTRACT

The Neuro-Linguistic Programming (NLP) Eye-Accessing Cues (EAC) model suggests that there is a correlation between eye-movements and the internal processing mode that people employ when accessing their subjective experience. Upon careful examination, the experimental methodologies of past research studies were based on assumptions informed by an incomplete or erroneous understanding of the EAC model that could have significantly influenced the experimental results. The reliability of the results can be further impacted by the absence of modern eye-tracking equipment to support the inherently complex task of reliably recording, selecting and rating eye-movements. While a plethora of eye-tracker designs is available to date, none of them has been designed to track non-visual eye-movements (eye-movements that are a result of neuro-physiological events and are not associated with vision), which tend to range outside the normal visual field and thus perform poorly in such cases. Therefore, this thesis introduces a set of novel algorithms for the extraction of relevant eye features (pupil position, iris radius and eye corners) that are combined to calculate the 2D gaze direction and to classify each eye-movement to one of eight classes from the EAC model. The applicability of the eye-tracker is demonstrated through a pilot study that serves as a real-world application case study. The performance of the eye-tracker is found to be practical for the intended purpose as it is lightweight, low-cost and can robustly perform the tasks of 2D gaze direction estimation and classification.

ACKNOWLEDGEMENTS

The contribution of certain people in our own lives and projects is often so subtle that we never acknowledge it to ourselves and them. I hope that, far in the future, when reading this short section of my thesis, I will have proved myself savvy of these subtleties. Here goes.

Thank you to my mother and my father who made "me" possible, to my supervisors Mike Spann and Sandra Woolley for their constant support, for sharing their wisdom and insights, for being friends as well colleagues and for being exceptionally patient with my changes of interest. Thank you to Paul Tosey, Jane Mathison and Richard Churches from the University of Surrey, who provided valuable guidance in the early stages of the PhD. Thank you to my friend Michael Antoniou for setting a good example for me and being the friend I can count on. Thank you to my friend Agnes Mariakaki for helping me break the laws of my personal physics and change the sinusoidal form of my motivation to peak higher and more often and dip less and less often. Thank to members of my family who always thought of highly of me and believed in me. Thank you to Prof. Martin Russell who as the head of postgraduate research at the time made my PhD scholarship possible and whoever else might have had something to do with it "behind the scenes". Thank you to Steve Quigley and Hooshang Ghafouri-Shiraz who were open to the shift in the goals of this PhD and allowed me to move forward. Thank you to James Gormley, David Checkley for their invaluable technical support and to Mary Winkles for her persistence in asking me whether I'm finishing up yet (amongst other things). Thank you to Mark Schulze for voluntarily offering stimulation and feedback in his own time.

Thank you to the naysayers that tried to influence me and my PhD directly or indirectly, to those who told me that I couldn't, wouldn't or shouldn't do it, to those who told me I could do it like this but not like that and of course, to those who directly opposed me – believe me when I say that, without you, this PhD would not have been possible. You were the extra fuel I needed to get "up" when I was "down". The universe works in mysterious ways.

If you are reading this and you feel you should have been mentioned but you were not it is merely because my mind has forgotten; my heart thanks you all the same.

TABLE OF CONTENTS

Chapter 1: Introduction 7

Chapter 2: An introduction to the NLP EAC model and critical review of past research 14

Neuro-Linguistic Programming and representational systems 15

The NLP Eye-Accessing Cues model 16

Past Eye-Accessing Cues model research 18

Brief overview of objectives and reported results of EAC studies 19

Eye-movement elicitation 20

Validation of the subject's cognition 23

Recording, rating and selecting eye-movements 24

Interpreting and analysing eye-movement data 28

Eye-movement research relevant to the EAC model 30

Eye-movements in dyadic interactions 31

Discussion 32

Summary and future directions 35

Chapter 3: Review of eye-tracking systems 37

Passive and active illumination 41

Gaze estimation 42

Calibration 42

Head-mounted eye-trackers 43

Remote eye-trackers 44

Eye-tracker invasiveness 47

Suitability of remote eye-trackers 48

Building a head-mounted eye-tracker 50

Chapter 4: Feature extraction 52

Acqusition and image properties 52
Eye-feature detection algorithm overview 64
Pupil contour estimation 65
Pupil contour refinement 69
Iris radius calculation 73
Eye corner detection 79
Clustering of eye-corner features 82
Calculation of the 2D gaze vector 89
Computational complexity 92
Chapter 5: Feature extraction evaluation 98
Testbed and sample video collection 99
Evaluation of the pupil detection algorithm 101
Evaluation of the iris boundary detection algorithm 111
Evaluation of the corner detection and clustering algorithms 119
Evaluation of the 2D gaze vector calculation algorithm 147
Comparison of the REACT eye-tracker with SR Research EYELINK-II 148
Evaluation of the REACT eye-tracker hardware usability 160
Chapter 6: Case study 164
Eye-tracking over time 164
Case study PILOT experiment 169
Discussion 170
Chapter 7: Conclusions and future work 184
Future work 188
Appendix A: Eye-tracking hardware 190
Appendix B: Case study full transcript 196
Appendix C: Published and submitted papers 230

List of references 249

CHAPTER 1: INTRODUCTION

This research work is concerned with the development of an eye-tracker that is able to track eye-movements over the maximum range of movement of the eyes and is targeted towards applications that are concerned with non-visual eye-movements. These eye-movements tend to extend beyond the normal field of view when people are looking at visual targets. In this way, it is probably the only eye-tracker that is able to track such extreme eye-movements and maintain similar levels of accuracy. It is also the only eye-tracker of its kind (head-mounted with close-up camera) to not use the glint, which is the reflection of the infrared light on the cornea, as a reference point but instead, detect and use the eye corners as reference points. Another advantage of the eye-tracker has been its ease of use; while a large proportion of other eye-trackers require that the camera is partially or fully calibrated and that each subject provides several calibration points, the REACT eye-tracker operates without camera calibration and required only one calibration point for each subject. Additionally, a low-cost, easy-to-assemble build has been maintained and the eye-tracker can be easily adapted to other applications, both in hardware and software. One of the adaptations that were considered to be important is the possible transition from 2D gaze to 3D gaze. For this particular application, 2D gaze has been sufficient and should it be necessary in the future, because the iris radius is calculated with great precision and if camera calibration data were available, 3D gaze could also be calculated. Last but not least, the REACT eye-tracker makes use of no models of great computational complexity and is thus able to perform fast. This research work was inspired by the Neuro-Linguistic Programming Eye-Accessing Cues model and beyond the major contribution of the REACT eye-tracker itself, it further contributes to the academic body of knowledge in relation to Neuro-Linguistic Programming with a critical review of past Eye-Accessing Cues model research which is presented in Chapter 2 and published elsewhere in a peer-reviewed publication (see Diamantopoulos *et al.*, 2008).

Movements of the eyes have fascinated academics for decades; both in sleep and waking, academic researchers have questioned their purpose, deconstructed their operation and analysed them both in terms of their physiological and psychological properties.

Perhaps this preoccupation with the eyes may be at first explained by considering that the eyes are an essential part of our experience and the visual channel helps us make a large part of our

decisions in everyday tasks. Secondly, even though vision appears smooth to us, it is only made possible by the immensely complex structure and behaviour of the eyes. Most commonly referred to as eye-movements, movements of the eyes have been classified into at least nine different classes (Carpenter, 1988; Wade and Tatler, 2005); with each class of eye-movement serving a different purpose in everyday visual tasks, it easily becomes evident just how complex eyes are and why there is such a wealth of research regarding eyes and their movements.

Of course, studying the eyes requires observation and the simplest of devices for this task is another pair of eyes. In his review of early "eye-movement detectors", Carpenter (1988), states that "with some practice one can probably detect movements of 1° or so without difficulty" (Yarbus, 1967 cited by Carpenter, 1988) which Carpenter judges to be adequate for preliminary clinical examination but unsuitable for anything but very crude quantitative measurements. Wade and Tatler (2005) explain that direct viewing is the oldest method but not a particularly fruitful one and this is for two reasons: a) the eyes can move very fast and thus only the initial and final locations are noted and b) the eyes have low temporal resolution and even with intense concentration, it is difficult to determine how the eyes have moved.

Thus, several man-made devices have been introduced from as early as 1901 when Dodge and Cline (1901 cited Carpenter, 1988) implemented a device that allowed permanent records of eye-movements to be made - a very primitive, yet functional, cinematic camera. It was not until the 1950s that video recording as we know it now was used for recording eye-movements as a slightly improved version of direct viewing; computer analysis of the video and the record eye-movements was introduced relatively recently, in the 1980s (Carpenter, 1988). Video recording was only one amongst many other devices and methodologies such as mirrors, photoelectric viewing, devices that use light reflected by the cornea or attachments to the eye, electro-oculography, electromagnetic recording, contact lenses and suction devices, etc. (Carpenter, 1988; Yarbus, 1967).

As happens with technology, eye-tracking has made several advancements and has matured over the years and high frequency eye-trackers that can track even the smallest of movements have been introduced. Video-based eye-trackers are now the mainstream choice and several different types of them are available, the two major ones being head-mounted and remote eye-trackers.

While eye trackers have thus changed dramatically over the last few decades, eyes are still mainly studied as a functional organism of vision. With the exception of studies in rapid eye-movements during sleep, it was only recently that eyes were studied as a part of the brain and its function (in the waking state). This paradigm shift may have been encouraged but the progressive price drop of eye-trackers which rendered them more affordable to research establishments. Having said that, while there is a reasonable amount of studies relating eye-movements to speech activities such as reading text or maps, very few studies have involved the recording and tracking of eye-movements during the performance of *non-visual* tasks.

To be clear, the term *non-visual tasks* refers to those tasks which do not explicitly rely on vision to be performed. One example of such an experiment is the "Hollywood Squares" paradigm, where the subject is shown a grid of four squares. Sequentially, an object is placed into one of those four grid squares and a fact is presented auditorily. The grid is then taken away and the subject is asked to answer a question relevant to one of the facts previously presented. Since the objects displayed in the grid can be identical, any visual significance is removed and thus this task does not rely on vision itself at any stage; as such, it can be termed as *non-visual*. In this task, it was found (Richardson and Spivey, 2000) that eye-movements played a key role in encoding the information presented in the auditory modality. Similarly, the term *non-visual eye-movements* refers to eye-movements concerned with non-visual tasks. Alternatively, visual eye-movements may be defined as eye-movements whose purpose is to change the visual stimulus falling on the fovea and non-visual as those eye-movements that are a result of neuro-physiological events and are not associated with vision.

Interestingly, models which make use of eye-movements but which are not related to visual tasks (at least not extrospective visual tasks) appeared in the last few decades. The Neuro-Linguistic Programming (NLP) Eye-Accessing Cues (EAC) model was introduced by Bandler and Grinder (1977) and suggests that the direction of non-visual eye-movements indicates the modality (visual, auditory, kinaesthetic) of the subjective experience a person is currently accessing.

Simply said, when a person[1] is looking down and to their right, they are accessing a feeling associated with the experience they are talking about or examining internally.

While it cannot be denied that eye-movements are hard-wired to brain function, the NLP EAC model was not scientifically validated by its authors. As it will be presented in detail in Chapter 2, several studies appeared at later dates that attempted to (dis-)prove the model but whose results suffered from severe methodological and experimental flaws. The use of direct viewing to record, select and rate the eye-movements has been the genesis of very significant limitations that will be discussed in full detail in Chapter 2.

It quickly becomes apparent that studies relevant to the NLP EAC model and other such models would have benefited by the use of eye-tracking systems. However, selecting a suitable eye-tracking system for this task does not prove to be as easy. This is for several reasons, the most important one being that eye-trackers to date were designed to track *visual* eye-movements. Whilst the classification of visual versus non-visual eye-movements is not significant in itself, visual eye-movements are normally bound by a much smaller field of view. By contrast, when a person is thinking his or her eyes will usually shift to one of the extremities of the eye socket (regardless of the direction). Thus, if the person was asked to consciously look in the same direction indicated by this shift, he or she would have turned his or her head and performed a much smaller eye-movement to reach the target; this behaviour is largely undocumented. What is, however, documented is the tendency of subjects to shift their eyes when asked to answer a question (not related to a visual task) and before they return to looking at the interviewer.

Therefore, it is no surprise that eye-trackers designed to track visual eye-movements fail to track non-visual eye-movements of such extremities and the need for development of an eye-tracker that is able to track such eye-movements is introduced. The technical detail of this inadequacy as well as a review of recent video-based eye-trackers will be presented in Chapter 3.

In designing and implementing this novel eye-tracker several non-functional requirements must be considered because of the nature of the particular application(s). For example, in the context of the EAC model and other models where the interviewer-subject relationship must be

[1] This is a generalization offered by Bandler and Grinder (1979) for a cerebrally normally-organized right-handed person. The EAC model is explained in more detail in Chapter 2.

characterized by harmony (often referred to as rapport), precision or accuracy may not be the key requirement for a particular research application. Instead, the invasiveness of the eye-tracker is a key requirement, as any discomfort experienced by the subject will "break" rapport.

Some of the aforementioned eye recording or tracking devices, especially the early ones, required that the subject's eyelids are pulled open with adhesive plaster or clamps and the device makes contact to the eyeball in order to perform the measurements (Carpenter, 1988; Yarbus, 1967). Other devices, even modern ones, require that the chin is fixated using a chinrest or bite bar. All of these requirements would most certainly increase the subject's discomfort during the experiment and thus render these eye-trackers unsuitable for such applications.

Several factors can influence the invasiveness of an eye-tracker such as: a) whether it requires contact to the eyeball or other parts of the body, b) whether it restricts any type of movement (e.g. movement of the head) and c) if it is mounted on the head or body, how much it weighs and how long it takes before wearing the eye-tracker becomes uncomfortable for the user.

The thesis presented here is concerned with the development of the Robust Eye-Accessing Cues Tracker (REACT) whose main requirements are to maintain a high level of accuracy while tracking non-visual eye-movements as well as minimize invasiveness and cost, which is always a concern. Finally, while initially targeted to non-visual eye-movement tracking for research applications such as the NLP EAC model, it is desirable that the REACT eye-tracker can be adapted to other eye-tracking applications and be easy to assemble.

The REACT eye-tracker is head-mounted but very lightweight (approx. 60 grams) and is thus minimally invasive. A head-mounted approach was chosen over a remote camera one as to minimize cost and increase the resolution of the captured images and consequently the accuracy viable by the eye-tracker. Moreover, a remote camera requires that the head is tracked and not only does that increase the computational complexity but it also decreases the ability of the eye-tracker to track the aforementioned extreme eye-movements because of limitations imposed by the camera viewing angle. Several of the design choices made will be contextualized in Chapter 3, where a comprehensive review of recent eye-trackers is given.

On the hardware-level, the eye-tracker works by illuminating the eye with near-infrared light while blocking most of the visible light spectrum with an infrared filter imposed over the camera

lens; the hardware design is discussed in detail in Appendix A. This produces the dark-pupil effect (discussed in more detail in Chapters 3 and 4) which allows for the easy detection of the pupil and, in a sense, offloads some of the processing to the hardware. The captured images are then processed in software and a set of three eye features are detected: the pupil centre and contour, the iris radius and the location of the eye corners. Finally, features are combined in order to calculate the 2D gaze angle.

In terms of the thesis organization, Chapter 2 is a brief introduction to Neuro-Linguistic Programming and the Eye-Accessing Cues model specifically and a complete and detailed critical review of past EAC model research. At least six out of ten studies that have investigated the EAC model since its first introduction in 1977 (Grinder, DeLozier and Bandler, 1977) have reported unsupportive results. In this review of past research, these studies and their respective experimental methodologies are examined and the reliability of their results is assessed. The review is extended by presenting findings from other relevant eye-movement research while discussing their relevancy to and implications for the EAC model. Thus, in this chapter, it is argued that there is substantial ground for further research into the EAC model and identify the requirements that should inform this work, a significant one of which is using an eye-tracker to perform any further research. In doing so, the motivation of this research work will be established in further detail.

After having established this motivation, a review of existing eye-tracking systems is presented in Chapter 3, with a view towards discussing their limitations in tracking extreme non-visual eye-movements, the requirements of a non-visual eye-tracker as well as fundamental decisions that have informed the design of the REACT eye-tracker. Overall, Chapter 3 aims to give a brief overview of other systems regardless of their modi operandi but go into detail where relevant to this research work.

Chapter 4 then explicates the algorithms involved in detecting the eye features and combining them to calculate the 2D gaze using a detailed description of the mathematical and computer vision concepts as well as several illustrations where appropriate. In Chapter 5, the performance of the eye-tracking hardware and the eye feature detection algorithms presented in Chapter 4 is evaluated and the results are discussed.

Chapter 6 presents a pilot study specifically designed to apply the eye-tracker in a real-world application which will serve as a case study for this work. The full transcript with the results visualised is included as Appendix B and selected parts are extracted to facilitate discussion in Chapter 6.

Finally, Chapter 7 offers some concluding remarks and discussion of future work. Appendix C includes a list of already published journal and conference papers and a paper for submission to the Special Issue of the Signal, Image and Video Processing Journal entitled “Unconstrained Biometrics: Advances and Trends”.

CHAPTER 2: AN INTRODUCTION TO THE NLP EAC MODEL AND CRITICAL REVIEW OF PAST RESEARCH

This chapter is a brief introduction to Neuro-Linguistic Programming (NLP) and the Eye-Accessing Cues (EAC) model specifically. Parts of this chapter have been published in the proceedings of the First International NLP Research Conference held at the University of Surrey, UK, on 5th July 2008 (Diamantopoulos *et al.*, 2008). For a more in-depth introduction to NLP and its models, the interested reader is referred to the relevant literature (Bandler and Grinder, 1975; Grinder and Bandler, 1976; Bandler and Grinder, 1979; Dilts and DeLozier, 2000).

Since its introduction in 1977, the EAC model has been investigated in ten studies. While six of these studies report unsupportive results, a clear conclusion as to the validity of the model has not been reached. Each one of these studies is considered in the first part of this chapter and it is shown that, upon careful examination, the respective experimental methodologies were based on assumptions informed by an incomplete or erroneous understanding of the EAC model that could have significantly influenced the experimental results. The reliability of the results can be further impacted by the absence of modern eye-tracking equipment to support the inherently complex task of reliably recording, selecting and rating eye-positions. Further doubt is raised as to the validity of the results as most studies reported statistically significant results (whether in favour of the model or not) and yet, the correlations reported are not in agreement across studies.

Review efforts have been made before (Sharpley, 1987; Heap, 1988; Richardson and Spivey 2004), where NLP is criticised as unsupported by research efforts. However, these reviews are drawn from the reported results of the referenced studies rather than a critical review of the literature with strong background knowledge of the models in question. Further, Heap (1988) bases his conclusions largely on results reported by masters' dissertational theses; of his large list of 66 references, 36 are dissertations. Thus, the present review is restricted to peer-reviewed publications that concern the EAC model only.

Further extending our survey in the second part of this chapter, recent eye-movement research from other fields is presented and its relevancy to and implications for the EAC model are discussed. Thus, it will become apparent that there is no published research that directly proves or disproves the EAC model and there is substantial ground for further research. Finally, in the

last part, drawing from the strengths and weaknesses of past research in the EAC model and the research findings from eye-movement and cognition research, this chapter attempts to identify the requirements that should inform future research.

NEURO-LINGUISTIC PROGRAMMING AND REPRESENTATIONAL SYSTEMS

The roots of Neuro-Linguistic Programming (NLP) descend from the work of Richard Bandler and John Grinder in the early 1970s. Their first seminal work (Bandler and Grinder, 1975) was based on their study of Virginia Satir and Fritz Perls, a family therapist and the father of Gestalt therapy respectively, and introduced the meta-model. The meta-model is a linguistic model "about the way language functions in modelling the world" (Tosey, 2006).

Bandler and Grinder continued their work by studying Milton Erickson, a successful hypnotherapist, and further publishing two books about his hypnotic techniques (Bandler and Grinder, 1975; Grinder, DeLozier and Bandler, 1977), as the Milton model.

Integral to NLP is the notion of *representational systems*; as defined by Bandler and Grinder (1979, p. 14) the representational system is the sensory system that a representation of a person's subjective experience is held or accessed in:

> *"What we noticed is that different people actually think differently, and that these differences correspond to the three principal senses: vision, hearing, and feeling – which we call kinaesthetics[2]. When you make initial contact with a person s/he will probably be thinking in one of these three main representational systems. Internally s/he will either be generating visual images, having feelings, or talking to themselves and hearing sounds."*

According to Dilts and DeLozier (2000, p. 1097), "the term *representational systems* refers to the neurological mechanisms behind the five senses" and thus, five representational systems may be defined, each one corresponding to one of our senses: visual, auditory, kinaesthetic, olfactory and gustatory (VAKOG). The representational system is different to the "lead system", which is the sensory system that the person uses to initiate the search for the representation of the

[2] According to Dilts and DeLozier (2000), "'kinaesthetic' is a term used in NLP to refer to feelings and body sensations […] it is used to encompass all types of feelings including tactile, proprioceptive and visceral".

experience; for example, a search for a visual representation may be initiated through an auditory or kinaesthetic representation. There is also the distinction of the *primary representational system* (PRS), which is introduced and loosely defined in the original NLP texts (Bandler and Grinder, 1975; Grinder and Bandler, 1976; Bandler and Grinder, 1979); an appropriate definition is found in Dilts and DeLozier (2000, p. 1102): "in NLP, a person is said to have a *primary representational system* when that person values or uses one of his or her senses over the others in order to process and organise his or her experience of the world."

Using this notion of representational systems, they suggested that observable body language cues such as eye-movements, voice tone and tempo, body posture, gestures and breathing patterns indicate which representational system the person is currently employing (Bandler and Grinder, 1979).

THE NLP EYE-ACCESSING CUES MODEL

The NLP Eye-Accessing Cues (EAC) model was first introduced by Grinder, DeLozier and Bandler (1977) and further refined by Bandler and Grinder (1979). The EAC model suggests that non-visual eye-movements (i.e. eye-movements that are not concerned with the visual pursuit of an object in the environment) indicate which representational system a person is currently using. According to the model, such eye-movement patterns are observed in all individuals regardless of handedness in an idiosyncratic fashion. In other words, each individual ought to display an eye-movement pattern correlated to the modality they are currently accessing that is consistent to them. An explicit generalisation is offered for normally-organised right-handed people (Bandler and Grinder, 1979, p. 25; Figure 1 below). It is a generalisation because the patterns are said to be idiosyncratic and in this sense, the pattern is generalised to normally-organised right-handed people.

In simple terms, assuming a normally-organised right-handed subject that conforms to the generalised pattern offered in Figure 1 below, the EAC predicts that when, for example, the subject is accessing a visual memory, s/he is going to look up and right (their left). Thus, if the subject was asked a question such as "do you know when tomatoes are ripe?" and an eye-movement up and to the right is observed, that s/he accessed a visual memory in answering the

question, perhaps that of a ripe tomato. In practice, responses are usually much more complex but this crude example serves to illustrate the concept of the EAC model to the unfamiliar reader.

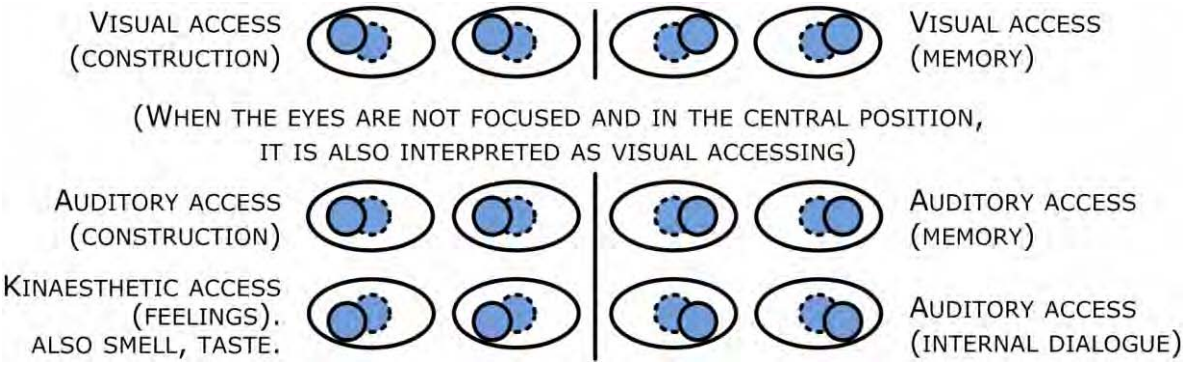


FIGURE 1: A GENERALIZATION OF THE EYE-ACCESSING CUES MODEL FOR A NORMALLY-ORGANISED RIGHT-HANDED PERSON (BANDLER AND GRINDER, 1979). IT IS A GENERALISATION BECAUSE THE PATTERNS ARE SAID TO BE IDIOSYNCRATIC AND IN THIS SENSE, THE PATTERN IS GENERALISED TO THIS PARTICULAR GROUP OF PEOPLE.

It can be said that the EAC model is a core component of NLP since it forms the basis of several advanced NLP techniques such as strategy elicitation and installation (Bandler and Grinder, 1979) which are said to have very practical uses. For example, Malloy (1987) found that pupils significantly improved their spelling ability if they looked up and to the right (from the observer's point of view) at the same time as visualising the word, versus visualising the word but looking in another direction and not visualising the word.

A very important part of the EAC model as presented by Bandler and Grinder (1979), is the process which elicits the eye-movements in question. This process, termed as *transderivational search* occurs when the subject recovers the *deep structure* from the *surface structure*, as shown in Figure 2.

The terms deep structure and surface structure were coined by Chomsky (1965) though a clear definition is elusive in both Chomsky's and the original NLP texts. A fairly comprehensive explanation can be found in Dilts and DeLozier (2000). Briefly explained, the deep structure consists of thoughts and ideas and their linguistic expression is the surface structure; derivation is a series of transformations which connects the deep structure with the surface structure (Bandler and Grinder 1975, p. 29) and hence the transderivational search. Chomsky (1965) originally used these terms to describe linguistic processes but Bandler and Grinder implicitly

extended these notions to neurological processes related to our sensory experience (Dilts and DeLozier 2000). Bandler and Grinder (1975) describe three transformative processes (deletion, distortion and generalisation) that are reflected both in the linguistic and the mental representation of the person's experience. Finally, "transderivational search is the process of accessing the meaning, which is equivalent to some set of images, feelings or sounds that are associated to that word" (Bandler and Grinder, 1979, p. 15).

Thus, transderivational search is the process that elicits the eye-movements that the EAC model focuses on. Thus, the first and foremost challenge in directly examining the EAC model as the following studies have done, is identifying an experimental methodology for eliciting eye-movements that follows a specific and precise definition of transderivational search. As useful as the above definitions may be, they require further refinement before they can be used experimentally. The methodology also needs to consistently achieve predictable responses in all instances and for all subjects. For the specific purpose of investigating the EAC model, it ought to recover visual, auditory, kinaesthetic and optionally olfactory and gustatory representations.

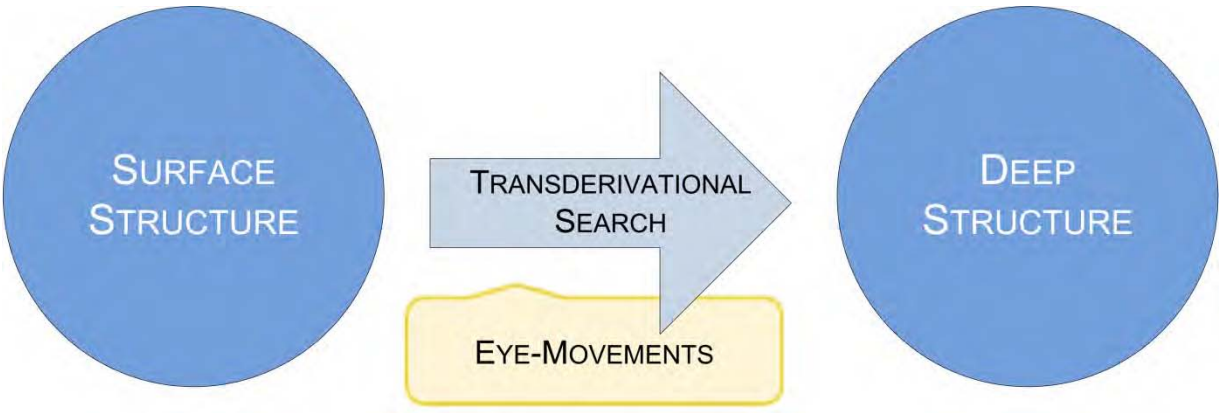


FIGURE 2: ELICITATION OF EYE-MOVEMENTS IN THE EAC MODEL AND TRANSDERIVATIONAL SEARCH.

PAST EYE-ACCESSING CUES MODEL RESEARCH

Of the relatively few academic studies of NLP, a large percentage of them have been concerned with the EAC model and the closely related notion of the primary representational system (PRS) introduced earlier. In this section, past research literature is reviewed and critiqued, with a sole focus on the EAC model. References to studies of the PRS will only be made where

methodological decisions are relevant to the discussion; otherwise, their results are regarded as irrelevant for this review.

Examination of the issues that revolve around handedness is beyond the scope of this review; the generic form of the model is assumed here, which, as stated earlier, suggests that patterns emerge in all individuals regardless of handedness.

BRIEF OVERVIEW OF OBJECTIVES AND REPORTED RESULTS OF EAC STUDIES

In the context of this review it will be said that a study has shown partial support for the EAC model if any of its results are statistically significant and consistent with the EAC model. Further, a study will be said to be unsupportive or to have shown no support for the model if none of its results are statistically significant or if none of its statistically significant results are consistent with the EAC model.

Thomason *et al*. (1980) attempted to test the EAC model hypothesis using questions to elicit visual, auditory and kinaesthetic (VAK) representations; their study was unsupportive of the model and was criticised by Beck and Beck (1984). Elich *et al*. (1985) used questions to elicit VAK representations and attempted to correlate eye movements and verbal predicates with question modality (interview-style); their conclusion was unsupportive of the model. Buckner and Reese (1987) asked subjects to report on VAK components of pleasant thoughts and found partial support for the model. Another test by Baddeley and Predebon (1991) correlated eye-movements with the corresponding verbal-report of the subjects' subjective experience and also found partial support. Burke *et al*. (2003) tested the relation between eye-movements and visual-kinaesthetic-gustatory (VKG) tasks both as hypothesised by the NLP model and idiosyncratically and found support for the idiosyncratic hypothesis.

Farmer *et al*. (1985) used recall of real stimuli and found no support for the model; a similar study was repeated by Wertheim *et al*. (1986) and found partial support. Dooley and Farmer (1988) repeated the experiment of Farmer *et al*. (1985) with aphasic subjects and found partial support.

Cheney *et al*. (1982) tested the relationship between eye-movements and reported imagery with the use of a questionnaire on vividness (Sheehan 1967); the results were unsupportive of the

model. A questionnaire was also used by Poffel and Cross (1985) with unsupportive results but it is unclear if the questionnaire was read out by an interviewer or completed by the subjects.

In an interview-style study (Ellickson 1983), the interviewers attempted to match the subjects' representational system, as determined in real-time by their eye-movements, through verbal predicates and tested the effect of predicate matching on perceived counsellor empathy.

Falzett (1981) used eye-movements to assess the PRS and determine the outcome of matching it through verbal predicates in counselling. Gumm *et al*. (1982) attempted to examine the agreement in the determination of the PRS using eye-movements, verbal predicates and self-report. Sandhu (1991) tested whether the PRS can be reliably determined from eye-movements by comparing its assessment from eye-movements, verbal predicates and self-report; he found no support for the EAC model.

Table 1 below is a summary of all the past research studies relevant to the EAC model.

EYE-MOVEMENT ELICITATION

Unknown questions were used by Thomason *et al*. (1980) and Poffel and Cross (1985); Falzett (1981) used questions from an unpublished doctoral thesis that are not reproduced in his publication. Gumm *et al*. (1982) used twenty questions to provide the subject with a variety of "cognitive tasks". It is unknown what exactly is meant by "cognitive tasks" in this case; it is likely that the tasks were unrelated to direct elicitation of sensory representations as other research of the time was concerned with generic mental tasks (e.g. Ehrlichman *et al*., 1974). Ellickson (1983) employed six "stimulus cues" during interviews that were apparently designed to elicit eye movements and neither the design criteria nor the stimulus cues are included.

Buckner and Reese (1987) asked their subjects to "think in silence of a single pleasant thought or memory" and after ten seconds, the interviewer asked the subject to report any VAK components; once again, the validity of this methodology in eliciting eye-movements as described above was not examined by Buckner and Reese (1987).

Eye-movement elicitation methodology was only sparsely informed by other research; the only instances are the papers by Cheney *et al*. (1982) and Elich *et al*. (1985) that utilised the questionnaire on mental imagery developed by Sheehan (1967). However, its relevance and

validity in the investigation of the EAC model was not discussed nor tested; even if the questionnaire is successful in eliciting eye-movements, it is not appropriate for this purpose as it explores the different properties of objects within a single representation (e.g. the colour of the dishes and the food on the breakfast table).

Sandhu (1991) was the first to note and give weight to the importance of "stressed recalls" and proposes that the subject's eyes will shift in potentially meaningful ways only when they "think hard" to answer the question. Despite this observation, the questions taken from the Sandhu PRS inventory are neither reproduced nor published in a journal publication elsewhere. Further, the inventory sample provided is arithmetic and not relevant to any one sensory modality.

Regardless of the particular method, the examples above highlight the ad-hoc selection of the eye-movement elicitation methodology and the implicit assumption that the respective methodology is equivalent of the transderivational search and consequently examines the desired eye-movements; no formal pilot studies were conducted and minimal emphasis was given on this pivotal aspect.

Baddeley and Predebon (1991) provided the full inventory of questions but there was another fundamental flaw. Duke (1968) found that a "complex" question will elicit a series of eye-movements. However, the more complex the question, the more difficult it is to isolate the cognitive process at work. Thus, questions that are too simple may not elicit any eye-movements and questions that are too complex may elicit too many eye-movements and/or cognitive processes for any useful distinctions to be made. More specifically, in the case of such complex questions, there is no guarantee that the representational system accessed by the subject is the same as intended by the author of the question. For example in a question that appears in Baddeley and Predebon (1991), "What colour are the walls in your bathroom?", which is reported as "visually remembered", the subject could retrieve a memory of their bathroom, kinaesthetically, e.g. by remembering the feeling of sinking into warm bath water (also reported by Beck and Beck, 1984). As mentioned earlier, in the EAC model, this is termed as the "lead system" (Bandler and Grinder, 1979, p. 28).

Table 1: Summary of past EAC research studies examined in this review. A study has shown partial support for the EAC model if any of its results are statistically significant and consistent with the EAC model. Further, a study is said to be unsupportive or to have shown no support for the model if none of its results are statistically significant or if none of its statistically significant results are consistent with the EAC model. Studies that related to the PRS are marked as "not relevant".

Publication	Purpose of study	Reported Results
Thomason *et al.* (1980)	Questions to elicit VAK representations; to correlate EMs with question type	Unsupportive
Falzett (1981)	Determine the effect of PRS as determined by EMs; questions to elicit EMs	Not relevant
Cheney *et al.* (1982)	Determine the correlation of EMs and reported imagery; Sheehan/Betts' questionnaire on imagery to elicit EMs	Unsupportive
Gumm *et al.* (1982)	Determine the effect of predicate of PRS based on EMs, predicates and self-report; questions used to elicit EMs	Not relevant
Dorn *et al.* (1983)	To assess the reliability of assessing the PRS through EMs	Not relevant
Ellickson (1983)	Determine the effect of real-time verbal predicate matching as determined by EMs	Not relevant
Elich *et al.* (1985)	Questions to elicit VAK representations; to correlate EMs and verbal predicates with question type	Unsupportive
Farmer *et al.* (1985)	Recall of real stimuli to elicit VAK representations; to correlate EMs with stimuli type	Unsupportive
Poffel and Cross (1985)	Questions to elicit VAK representations; to correlate EMs with question type	Unsupportive
Wertheim *et al.* (1986)	Recall of real stimuli to elicit VAK representations; to correlate EMs with stimuli type	Partial support
Buckner and Reese (1987)	To correlate the self-report of VAK components of subjective experience and recorded EMs	Partial support
Dooley and Farmer (1988)	Like Farmer *et al.* (1985) but with aphasic subjects	Partial support
Baddeley and Predebon (1991)	Questions to elicit VAK representations; to correlate EMs with question type and self-report	Unsupportive
Sandhu (1991)	Determine the agreement of PRS based on EMs, predicates and self-report	Not relevant
Burke *et al.* (2003)	Questions to elicit VKG representations; to correlate EMs to question type	Support for idiosyncratic case

VALIDATION OF THE SUBJECT'S COGNITION

This leads us to another inherent challenge in the direct examination of the EAC model: validation of the subject's cognition. While the representational system access required to answer a particular question can be linguistically pre-supposed (i.e. the access of a visual representation is required to recover purely visual information such as the colour of an object), how can one be truly certain that the subject has accessed the pre-supposed representational system to recover the information and nothing else without the use of neuroimaging technology such as functional magnetic resonance imaging (fMRI)? fMRI is a device which allows scientists to determine which part of the brain is active at any given point in time.

Farmer *et al*. (1985) provided subjects with real stimuli (pictures, tape-recorded sounds and textural objects) that they had to experience and later recall, presumably to guarantee that the representational system the subject used is the one intended. However, this is no different from pre-supposing that when the subject is asked to report on visual information, it is necessary for them to perform a visual access; the need for validation is still warranted. Take a textural object for example, such as a rock: even if it is supposed that the subject was blindfolded (no such mention by the authors), it is not necessary that the subject encoded only the kinaesthetic/textural aspect of the rock. Alternatively, the subject may form a mental visual image of what the rock may look like based on the kinaesthetic input – the "feel" of the rock. Thus, this methodology has failed to warrant the type of representational system accessed or to further involve the phenomenological, subjective experience of the subject. The same approach is taken by Wertheim *et al*. (1986) and is thus subject to the same criticism.

In an attempt to deal with this fundamental issue of validating the subject's cognition, some researchers collected accounts of the subject's subjective experience (Cheney *et al*., 1982; Elich *et al*., 1985; Baddeley and Predebon, 1991); however, the methodology was not informed by psycho-phenomenology literature (see Mathison and Tosey, 2008a; Mathison and Tosey, 2008b). For example, the method of introspective inquiry suggested by Beck and Beck (1984) in their critique of a related study (Thomason *et al*., 1980) is informed only by NLP literature. Imagery and introspection is an area of human psychology that has a long history of controversy (see e.g. Horowitz, 1983) simply because of its own very nature. The information is retrieved from a subjective source, the person, and the question of the reliability of any gained information is very

quickly raised (Mathison, 2006). NLP aims to study people's subjective experience and might thus be expected to have an interest in the methods of psycho-phenomenology; if NLP is to be explored academically and any potential links to be established, it is imperative that any enquiries into NLP are informed by established methodologies such as psycho-phenomenology.

RECORDING, RATING AND SELECTING EYE-MOVEMENTS

One cannot dismiss the inherent difficulty in recording and rating the eye-movements without an appropriate device such as an eye-tracker. To date, two different methodologies have been used: a) real-time scoring by human observers; or b) video recording the eye-movements and scoring them later. Especially in the first case, the question of who does the rating is especially relevant; the implicit assumption has been that eye-movements are easily discernable by (un)trained human observers in real-time or with the use of video-recording equipment.

The study of eye-movements has a long history and so does their measurement (Carpenter, 1988; Yarbus, 1967). The author hypothesises that direct-viewing was used in NLP studies because the EAC model is taught to be useful in real-time human interaction where the practitioner observes the eye-movements without technological aids. The use of inexperienced graduate student raters is advocated by Sharpley (1987) as traditional and a good measure of a procedure's readiness and robustness. This argument could only be valid if the model's suitability for adoption by untrained individuals was assessed and not its validity. The only available information regarding the accuracy and reliability of direct-viewing of eye-movements is that movements of less than 1° rotation (0.2mm movement of the retina) are not discernable by the naked eye (Yarbus, 1967; it is unclear whether this refers to a trained or untrained individual) and the question of sufficient reliability for the purposes of a scientific study quickly arises.

Indeed, several studies make no reference to the experience of the raters (Gumm *et al*., 1982; Poffel and Cross, 1985; Farmer *et al*., 1985; Wertheim *et al*., 1986; Burke *et al*., 2003), while others have regarded the use of naive (Thomason *et al*., 1980; Cheney *et al*., 1982; Baddeley and Predebon, 1991) or briefly trained raters (Falzett, 1981; Ellickson, 1983; Elich *et al*., 1985; Sandhu, 1991) as acceptable. NLP practitioners were used in one instance (Buckner and Reese 1987); again, this does not guarantee rating accuracy or reliability.

The case of some studies is strengthened because they used video-recording equipment that allows the rater to review eye-movements (Gumm *et al*., 1982; Cheney *et al*., 1982; Elich *et al*., 1985; Poffel and Cross, 1985; Wertheim *et al*., 1986; Sandhu, 1991; Baddeley and Predebon, 1991; Burke *et al*., 2003). In the methods used by most of these authors, the subject was forced to look at the camera thus restricting their head and body movement (Cheney *et al*., 1982). It is questionable whether all relevant eye-movements are discernable both because of relevant training and obscuring of the eye by blinks or head tilts and so on. The question of precision and reliability of rating has not been raised before other than inter-rater reliability tests which only certify a statistical agreement between raters and have no account for their individual abilities or other limitations imposed. In order to eliminate as many variables as possible, a recording and rating methodology whose error is known has to be used. In the methods described, no such precision/reliability tests have been performed. The question of what eye-movements occur during blinks and how they are relevant to the EAC model has not been considered in the literature other than by Buckner and Reese (1987) and Baddeley and Predebon (1991).

Even if an assumption that all relevant eye-movements can be precisely and reliably captured is made, another potent issue is which eye-movement to take into account. Firstly, the number of eye-movements in response to a stimulus cannot be predicted; to our knowledge there are no studies that show any statistically significant results in this respect. Therefore, by fixing the number of analysed eye-movements, bias is introduced in the selection process.

It is unknown how Thomason *et al*. (1980) selected the relevant eye-movement(s). In the study by Elich *et al*. (1985, p. 622), the authors specify that "eye-movements were recorded from the moment of asking the image-evoking question up through subject's description of the images experienced in response to the question"; however, the process of selecting the eye-movement judged as relevant to the question is also unknown. Similarly, Poffel and Cross (1985) provide no information on the matter.

In a study related to the PRS, Falzett (1981) selected the eye-movement prior to the acknowledgement of the subject that an internal response has been reached. This process was replicated by Farmer *et al*. (1985). Gumm *et al*. (1982) and Sandhu (1991) assessed the first eye-movement following the end of each question, while Wertheim *et al*. (1986) recorded the first eye-movement after the subject was asked to recall the stimuli as well as the last eye-movement

before the subject's acknowledgement of their internal response. Recording the first eye-movement after the end of the question was earlier done by Cheney *et al*. (1982); in their study, multiple eye-movements were regarded as a separate event and selection was not attempted. Ellickson (1983) made a distinction between occurrences of one and two eye-movements; in the latter case, the second eye-movement was selected. There are several flaws these studies have in common.

- There is an implicit assumption that eye-movements (or at least the relevant one) occur after the end of the question, which is not necessarily true and this is supported by Cheney *et al*. (1982) who pointed out that often the subject's eyes will shift before the end of the question.
- Bandler and Grinder (1979) also suggest that some eye-movements may reflect a speech preparation, rehearsal, or translation process or the first eye-movement may reflect the "lead" system, i.e. the representational system that the subject uses to bring the representation into consciousness; in the example offered earlier, the kinaesthetic system (the feeling of sinking into warm bath water) would be the lead system.
- In the case of asking the subject to acknowledge reaching an internal response before verbalising, it is possible that the last eye-movement corresponds to a process related to this acknowledgement.

Buckner and Reese (1987) recorded whether any eye-movement that matched the expected modality was present when asking their subjects if they were aware of VAK components in their thought. While the EAC model does not define a specific selection process, it is questionable if this methodology can yield objective results.

Baddeley and Predebon (1991) recorded a series of eye-movements in each part of their study; the two models they used are shown diagrammatically in Figure 3 together with the other approaches. This is an improvement over previous studies in that it attempts to record multiple eye-movements. However, there are three fundamentally problematic assumptions that cannot be predictably satisfied. Those are:

1. All subjects will always perform the same amount of eye-movements (no selection criteria are discussed).

2. The representational system targeted by the question will occur on the same eye-movement instance for all subjects. In reality, variations of cognitive and physiological responses can be expected in different people answering the same question.
3. Eye-movements have a one-to-one correspondence to internal representations or processes. In reality, one cannot be certain what these processes are; also pointed out by Cheney *et al*. (1982).

Recently, Burke *et al*. (2003) video recorded and scored all eye-movements and performed pattern analysis on sets of two, three, and more than three eye-movements. Cheney *et al*. (1982) reported that eye-movements will often transpire before the interviewer has reached the end of the question as early as 1982 and it is therefore surprising that no studies up to Burke *et al*. (2003) take this into account.

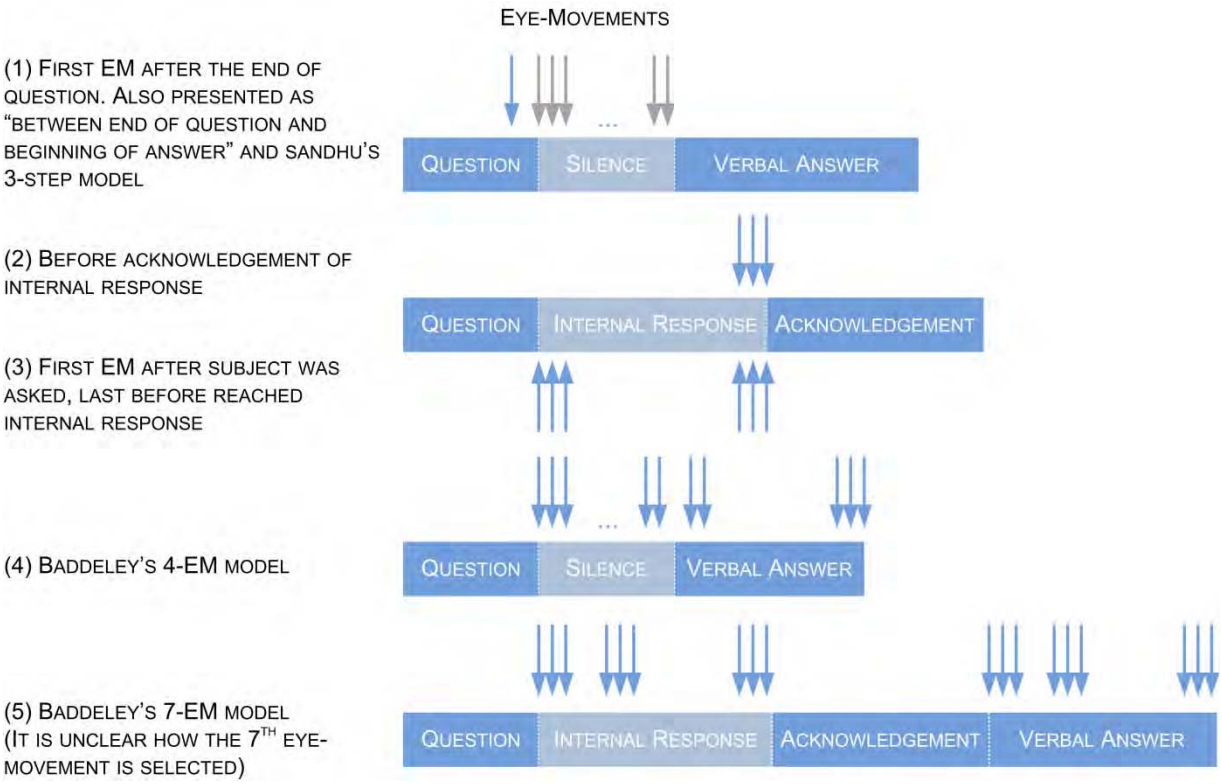


FIGURE 3: DIFFERENT MODELS OF EYE-MOVEMENT SELECTION FOUND IN LITERATURE. THE BLUE ARROWS INDICATE WHERE EYE-MOVEMENT MEASUREMENTS WERE MADE FOR EACH MODEL. THE GREY ARROWS DESIGNATE OTHER POSSIBLE INSTANCES OF THE CHOSEN EYE-MOVEMENT. DOTS ARE USED TO DESIGNATE AN INTERVAL WITHIN WHICH THE EYE-MOVEMENTS CAN OCCUR.

As mentioned earlier, the EAC model predicts that eye-movement patterns are observed in all individuals regardless of handedness and a generalisation is offered for normally-organised right-handed people (Bandler and Grinder 1979, p. 25; Figure 1). Given this is an explicit generalisation, it will not hold true for all right-handed people and it is thus not sufficient to screen for right-handed people. In order to investigate these claims it is necessary to test whether idiosyncratic patterns exist within any given individual. This idiosyncratic case was only tested by Burke *et al*. (2003) who had partially supportive results, while all the other studies used this generalised form in order to interpret the selected eye-movements.

INTERPRETING AND ANALYSING EYE-MOVEMENT DATA

Another interesting aspect of the studies that has never been commented on is the statistical variance of the results reported. Cheney *et al*. (1982) report no eye-movement 32% of the time while 18.9% of the responses were multiple eye-movements and not analysed. In the same study, eye-movements to the left and up and left were only 14.75% and 10.3% respectively. Poffel and Cross (1985) reported no eye-movement 50% of the time – vastly different results to those of Thomason *et al*. (1980) despite the similarity in methodologies.

Farmer *et al*. (1985) reported 49.6% baseline eye-movements with upwards movements coming second at 37% – a significant difference, especially in light of the roughly equal results reported on all VAK components by Thomason *et al*. (1980). Wertheim *et al*. (1986) did not include numerical data but report a majority of auditory responses regardless of question type. Dooley and Farmer (1988) and Farmer *et al*. (1985) report 44% stares for their aphasic subjects and 40% auditory for their control subjects. Finally, Baddeley and Predebon (1991) report 40% and 41.9% leftwards eye-movements in study one and two respectively.

The aforementioned incongruences observed in the results of the studies can lead to two possible logical conclusions: 1) eye-movements are random; or 2) there are variables that have not been considered (or perhaps discovered) and controlled for.

So, are eye-movements random? From the nine studies that performed statistical analysis on eye-movements and question/task modality, only three report no statistical significance (Cheney *et al*., 1982; Elich, 1984; Sandhu, 1991) whereas the remaining six reported some statistical significance even though they were unsupportive of the EAC model (Thomason *et al*., 1980;

Farmer *et al*., 1985; Wertheim *et al*., 1986; Dooley and Farmer, 1988; Baddeley and Predebon, 1991; Burke *et al*., 2003); this would suggest that eye-movements are indeed not random. Further, if the relationship was simple (e.g. baseline movements) the results would be more coherent. Later in this review, research from other fields that suggest that eye-movements are linked to internal processing will be considered.

From the aforementioned data, a trend is visible in some studies where a large sum of the elicited eye-movements has been stares or baseline eye-movements. It is surprising that this trend has not raised any suspicion about the validity of the questioning methodology in the past since the interpretation of both stares and baseline eye-movements is ambiguous. The term *stares* refers to the central position of the eye within the eye-socket when the eyes are not focused. According to the EAC model this position is associated with visual access. However, stares can also be regarded as failure to activate the transderivational search and it is unclear how to differentiate between that and visual access (also reported by Ehrlichman *et al*., 1974; Ehrlichman and Weinberger, 1978). This ambiguity is also true of baseline eye-movements; as per the EAC model, baseline eye-movements are associated with auditory eidetic access (auditory constructed sounds or words) but they can also be connected to an internal rehearsal or speech preparation process (Bandler and Grinder, 1979, p. 18). Once again, since there is no available distinction between the two cases, it is theoretically consistent with the EAC model that consistent elicitation of baseline eye-movements can also be regarded as a failure of the elicitation process to activate the "transderivational search".

Given the statistical variance of the recorded data of one to seven eye-movements, it is surprising that no past research has performed a comprehensive frequency analysis on the number of eye-movements that occur in response to questions or introspection and their temporal location. It is suggested here that if a predictable relationship exists between eye-movements and internal representations and if this relationship is to be discovered, it is necessary to record and analyse all eye-movements that subjects make (also pointed out by Ehrlichman and Weinberger 1978). Recent developments in eye-tracking technology may allow this to be done reliably and without the immense effort involved in manual rating; depending on the intrusiveness of particular system used, there can be minimal interference with subject-experimenter rapport.

EYE-MOVEMENT RESEARCH RELEVANT TO THE EAC MODEL

A very important finding for the EAC model by Christman *et al*. (2003) showed that the retrieval of episodic memories is selectively enhanced when it is preceded by saccadic eye-movements (fast movement of the eyes towards or away from an object, or without a visual stimulus) and not when preceded by pursuit eye-movements (eye-movements used to smoothly follow a moving object). Non-visual eye-movements such as those referred to by the EAC model fall under the category of saccadic eye-movements and thus an important link between eye-movements and memory retrieval is hereby established.

In a different area of eye-movement research and after a series of experiments (Brandt and Stark, 1997; Demarais and Cohen, 1998; Spivey and Geng, 2001), Richardson and Spivey (2000) adopted a "Hollywood Squares" paradigm where subjects were presented with a two-by-two grid of squares, each filled with an object and associated with an auditorily-presented semantic property. Consistent with earlier accounts, they found that when the objects were removed from the grid and the subjects were questioned about one of the properties, subjects tended to look at the blank region of space where the property had been previously presented. This spatial indexing effect is related to the eye fixation and not attentional focus, is independent to fixations on separate locations in absolute space and even though spatial location is irrelevant to the task, it is consistently and automatically encoded.

The spatial indexing phenomenon agrees with neurological research in imagery and perception where it has been shown that there is a large overlap of brain area (re-)activation between perception and imagery (Kosslyn, 2005; Buckner and Wheeler, 2001; Handy *et al*., 2004). Not only is there similar brain activation but the eye-movements are re-enacted and they play a functional role (Laeng and Teodorescu, 2002).

The NLP creators developed the EAC model based purely on their own observations of people's behaviour and it may be tempting to attribute these observations on spatial indexing but on closer inspection, such a conclusion would be erroneous. It would mean that every time a representation is accessed, our eyes move in the same direction as they did when this representation was encoded. In all probability, an illogical conclusion if we consider that most information in our lives is presented to us within a central attention window of a limited viewing range while the eye-movements associated with the retrieval of internal representations are

relatively spatially extreme and most probably outside the limits of this window. Also, the study conducted was concerned with very short-term recall – what about longer-term accesses? Of course, neither argument could make a strong stance without collecting further evidence.

EYE-MOVEMENTS IN DYADIC INTERACTIONS

The contexts where the EAC model is supposed to hold true are unclear; no explicit claims have been made by Bandler and Grinder (1979) and the original context is that of dyadic "natural" human interaction. It is possible that this is the only context where it holds true and only when a certain condition is met: rapport. The working assumption so far has been that the relationship between the researcher and the subject is not important and several studies have not reproduced this context (Thomason *et al*., 1980; Gumm *et al*., 1982; Burke *et al*., 2003). Further, some studies (Cheney *et al*., 1982; Elich *et al*., 1985; Baddeley and Predebon, 1991) required the subjects to interact with a light switch to enable blind rating of the eye-movements, which may have influenced the results by creating an artificial environment, thus jeopardising the rapport condition.

There are several important findings regarding eye-movements during dyadic interactions that are relevant to the EAC model:

- If a person is looking upwards and sideways and there is no apparent object to which their gaze is directed, 4-year-old children can infer the person is thinking (Baron-Cohen and Cross, 1992).
- There is some evidence that the relationship between the experimenter and subject affects the rate of the eye-movements (increased rate of eye-movements with high-anxiety questions, MacDonald and Hiscock, 1985). No change in direction was observed but the experiment controlled only for change in lateral direction.
- No conclusion may be made about whether the position of the experimenter (face-to-face versus behind subject) affects eye-movements (Kinsbourne, 1972; Ehrlichman and Weinberger, 1978) but it is certain that they occur even when no other person is present (Ehrlichman and Barrett, 1983; Kocel *et al*., 1972).
- According to McCarthy *et al*. (2006) eye-movements in dyadic interactions are also culturally-biased. In a complex-question task, all three groups of Trinidadians, Canadians

and Japanese subjects made less than 50% eye-contact and the direction of eye-movements was mainly (81%) up for the Trinidadian and Canadian subjects versus 75% down for the Japanese subjects. This was attributed to the different connotations of looking up or down during conversation for each group though there is no solid ground for this claim.

It was also recently explicitly shown that verbal questions will elicit eye-movements when there is nothing to look at (Ehrlichman *et al*., 2007); an interference theory that was originally put forward has been refuted. The remaining theory is that people naturally shift eyes rather than focus them and they suppress those eye-movements when there are useful visual cues in the environment such as the face of another person (Ehrlichman, 1981). In this view, eye-movements could be regarded as an integral part of thought and brain activity.

Another question that several studies have attempted to answer before is whether eye-movements are reliably consistent over time but the results are mixed and hence inconclusive. Templer *et al*. (1972) found them not to be reliable, in contrast to Bakan and Strayer (1973). Dorn *et al*. (1983) tested the same subjects with the same questions after a week and found that the eye-movements were different. However, during debriefing some subjects reported that they had recalled their previous response to each question instead of generating a new one.

Returning to the issue of question complexity examined earlier, several studies have made a distinction between questions that elicit eye-movements and questions that do not; reflective versus factual questions (Day, 1964; Duke, 1968), reflective versus over-learned (Ehrlichman *et al*., 1974) and complex processing versus over-learned, immediately available and syntactically simple (Ehrlichman and Weinberger, 1978).

DISCUSSION

In this review, the deficiencies of past EAC research have been identified and requirements that should inform future EAC research have been established.

The pattern that has emerged from reviewing past EAC model research is that working assumptions have not been identified fully. Granted, this is a difficult task for such a complex investigation and it is a privilege to be able to learn from past research. A fine example of this is

the use of the term *representational system*. In the original NLP texts (Grinder and Bandler, 1976) this term was defined very loosely and it was not until recently that definitions more suitable for academic research have appeared (Dilts and DeLozier, 2000). Linguistically speaking, people's ability to answer questions about the world such as "what colour is the sky?" presupposes that they are able to access those representations. Even though this presupposition may be sufficient for empirical and experimental research, it is customary to clearly state what definition of the term is assumed; this has been lacking and the definition has been taken for granted.

A new set of working assumptions is proposed (see Table 2) that with further research should lead to a more rigorous experimental methodology.

To begin, it is important to establish criteria for the eye-movement elicitation methodology. An attempt was made by Baddeley and Predebon (1991) to import the criteria from lateral eye-movement research but no discussion of its applicability or relevance was made. Certainly, a formalised classification of question complexity is required (e.g. reflective versus over-learned) or at least the responses have to be controlled; either by adopting a question model that continually refines the requested detail or with a consistency test of the eye-movement responses. Similarly, Ehrlichman and Weinberger (1978) suggested a test of which questions consistently elicit left or right eye-movements, and which do not. Alternatively, an established means of exploring the phenomenology of the subject's experience can be used (e.g. Varela and Shear, 1999).

It is also vital to record all eye-movements present because there is no way to predict the number of eye-movements without extensive question analysis. In order for the measurements to be reliable, human rating of the eye-movements should be avoided and a recording method whose accuracy and reliability is known ought to be used. Further, given the possibility that non-visual eye-movements are coupled during conversation (like visual eye-movements are, see Richardson and Dale, 2005), ideally both the eye-movements of the subject and the experimenter need to be recorded.

Last, but not least, subjects with a similar cultural background and same native tongue need to be selected to eliminate cultural bias and because internal translation processes would interfere with the results, respectively.

TABLE 2: OLD AND PROPOSED SET OF ASSUMPTIONS OF EAC MODEL RESEARCH.

Aspect	Old Assumption	Proposed Assumption
Questions	Response elicits transderivational search	Subject-specific
	Response elicits linguistically presupposed rep. system	Eliciting the presupposed rep. system requires refinement *
	One-step; isolation of cognitive processes assumed	Multi-step; isolation of cognitive processes by refinement *
Rating	Performed by raters of variable training	Performed with machine vision; automatic or semi-automatic
Selection	Specific eye-movements are relevant	All eye-movements are relevant
Pattern	Occurs across questions and/or subjects	Occurs idiosyncratically; secondary generic patterns may emerge
Eye-movements	Constant number	Variable number
	Stares = visual access Baseline = auditory access	Stares = visual access or no transderivational search * Baseline = auditory access or no transderivational search *
	All eye-movements in response to a question are non-visual	Eye-movements can be distinguished between visual and non-visual *
Relationship	Has no or minimal effect	Rapport is a necessary condition
Cultural background	Has no or minimal effect	Has potentially significant effect
Native language	Has no effect	Has potentially significant effect
Physiological feedback	Eye-movements only	Eye-movements and other physiological responses *

* requires further research before a methodology is developed

SUMMARY AND FUTURE DIRECTIONS

Thirty years after its introduction, NLP remains a largely unexplored field within academia and the EAC model was the main target of evaluation in the past. Even though the results of these evaluation efforts have not been consistent and thus conclusions can only be tentative, they have been used as evidence to discredit the model itself and NLP as a whole.

Perhaps the EAC model has been seen as a simplistic part of NLP but the inherent complexity of the EAC model and its study should be evident from the critique of past EAC model research that has taken place in this chapter. Past research has implicitly adopted incomplete and erroneous assumptions and therefore the EAC model requires further research attention.

More importantly, if the EAC model is considered a simplistic part of NLP and yet no definite conclusions can be drawn from this relatively large set of research, this is a clear indicator of how much more complex it would be to investigate larger and more complex NLP techniques or models. If academic value is to be extracted from NLP, more weight would need to be given by future research.

Though research in the EAC model has stopped for several years, it may now be a good time to continue these efforts, as the link between eye-movements and neurology is clearer and eye-tracking technology suitable for this purpose is going to presented in this thesis. As discussed, saccadic eye-movements were recently shown to aid the retrieval of episodic memories and even though (at least without further investigation) spatial indexing cannot account for the EAC model it is a positive indicator that eye-movements are related to the process of encoding information. Along with research in dyadic interactions that is now available, these advances justify the investigation of an ad-hoc model such as the EAC model which is still "current" in NLP circles.

What is especially interesting is that a theory similar to the EAC model has emerged fairly recently from an academic field unrelated to NLP (Ehrlichman *et al*., 2007); it is based on a buffer model which is used to hold representations. Eye-movements are associated with retrieval and fixations are associated with maintenance of the information in the buffer.

It is proposed that the EAC model is worthy of further research with an experimental methodology that incorporates the assumptions and requirements presented here. A significant

part of this future direction is the use of eye-tracking technology to detect, track and classify eye-movements. As current state-of-the-art eye-trackers are still limited in the range of eye-movements they can track because they are designed in with the working assumption of the subject looking at a screen or an object in the environment, a novel eye-tracker that is able to successfully track non-visual eye-movements is required. The design and implementation of such an eye-tracker has been the main objective of this thesis. Thus, Chapter 3 reviews current eye-tracking systems with respect to the requirements of such research and sets the background for the implementation of the Robust Eye-Accessing Cues Tracker (REACT), introduced in Chapter 4.

CHAPTER 3: REVIEW OF EYE-TRACKING SYSTEMS

In Chapter 2, the benefit of using an eye-tracking system for research relevant to the EAC model was established. In fact, the lack of a computerised eye-movement rating system has been a major flaw in previous studies. Direct viewing methods, even when performed by trained individuals, are error-prone and make objectivity seem impossible to achieve.

While there are several eye-tracking systems with good performance (both commercial and in academia), they are fundamentally designed for applications where the subject is looking at an object or a screen in the *external* world. Briefly mentioned in earlier chapters and reiterated here, such visual eye-movements are restricted by a relatively narrow field of view; Hansen and Ji (2010) report that fixations normally occur within two to five (2-5) degrees of central vision. Further, as it was found empirically, people prefer to turn their heads in the general direction of the object or screen and then shift their eyes a small amount to put the visual target in focus. A trivial example of this behaviour is television; if you were sitting in front of your computer reading this thesis and there was a television screen several degrees up and to one side, you would most likely turn your head towards the screen every time it gathered your interest and then shift your eyes to focus on any objects displayed on the television screen.

In contrast, non-visual eye-movements are usually characterized by a shift to an extreme location within the eye socket and the fundamental assumptions that current eye-tracking systems have been designed with render them incapable of tracking these eye-movements. For example, the vast majority of eye-trackers use infrared light and a glint-centred coordinate system (e.g. Ohno *et al.*, 2002; Li *et al.*, 2005); the glint is the reflection of the cornea and appears as a very bright spot. When the glint falls onto the sclera, which is also white, it can be very difficult to find its position. Thus, such eye-trackers operate on the implicit or explicit assumption that the glint will always fall within the iris or pupil, which would not hold true for extreme eye-movements.

Before going any further it is important to define what “extreme” means in the context of eye-movements and further explicitly list the requirements of a non-visual eye-tracker compared to a visual one.

Hansen and Ji (2010), cite the work of Tweed and Vilis (1990) when stating that “eye positions are restricted to a subset of anatomically possible positions described by Listing’s and Donder’s

laws". Indeed, in their study which is not specific to the oculomotor range but concerns the geometric relations of eye position and velocity vectors during saccades with respect to Listing's law, Tweed and Vilis (1990) briefly mention that the oculomotor range of their measurements is ±40° both horizontally and vertically. Other researchers (Guitton and Volle, 1987) however report that the human oculomotor range is ±55°, a considerably larger range. It is not uncommon to come across eye-movement studies which were performed over a range as large as ±70° (Collewijn *et al.*, 1988) and ±80° (Guitton and Volle, 1987) though such research also aims to test objectives that require eye-movements beyond the maximum human oculomotor range. In this research work, ±55° of angular range will be considered the maximum possible range. Further, the label of "extreme eye-movements" is given to those eye-movements that extend beyond the operational tracking range of existing eye-trackers, which is equal to or less than ±30° (e.g. SR Research, 2009).

Thus, the requirement that separates visual eye-movement trackers from non-visual eye-movement trackers is its ability to maintain similar accuracy across the complete range of eye-movements as defined by the aforementioned oculomotor range of ±55°.

This chapter will walk through the eye-tracking literature from the past decade or so in order to give an overview of currently available systems and their suitability for this application. In doing so, it will also form the basis for several fundamental decisions that have informed the design of the Robust Eye-Accessing Cues Tracker (REACT).

A recent review by Hansen and Ji (2010) is a comprehensive and fairly detailed source of technical information on video-based eye-tracking systems. It categorizes research work based on the particular area of focus:

a) Eye localization in the image that is concerned with:
    i. detecting the existence of eyes
    ii. interpreting eye positions in the image
    iii. tracking the detected eyes from frame to frame
b) Gaze estimation which is concerned with estimating where the person is looking in 2D or 3D or determining the 3D line of sight.

While this categorization is appropriate for the latter review, this review will categorise eye-trackers in a different fashion, one that facilitates the discussion of several design choices made with the applications requirements in mind:

- **Remote versus head-mounted.** Remote eye-trackers where one or more cameras are placed in a remote location and head-mounted eye-trackers which are directly mounted on the subject's head usually through a glasses-like frame or a helmet.
- **Light source(s).** The light source used in each eye-tracker dictates the image properties and to a large degree formulates the computer vision problem that is required to be solved. As such, the eye-trackers will be classed based on whether they use natural illumination (passive) or (near-) infrared illumination (active). In the case of active illumination several light sources may be used, each one of which will result in corresponding glint, which is the reflection of the light source on the cornea. The glint is a "nickname" for the first Purkinje image, as shown in Figure 4.
- **Number of cameras.** As this review focused on video-based eye-trackers only, it is important to include the number of cameras each system uses as more than one camera is often used.
- **Gaze estimation method.** There are two main methods of gaze estimation, the primary objective of eye-trackers, namely 2D and 3D gaze estimation. 2D gaze estimation is concerned with estimating where exactly the subject is looking on a surface such as a screen. On the other hand, 3D gaze estimation may estimate the gaze direction or point of regard in 3D space.
- **Calibration requirements.** As it will be briefly explained below, most eye-trackers require a calibration to be executed either once for each system, once for each subject, or both. This serves as another useful element for categorisation.

Further, eye localization schemes will not be discussed as they are only relevant to full-face images such as those taken by remote eye-trackers, unless relevant in terms of another technical aspect.

Before further discussing the requirements of an eye-tracker that is suitable for non-visual eye-movement application, it is useful to give a brief overview of existing designs based on the categories laid out above.

It seems that remote eye-trackers are by far the most common design, perhaps because it is considered to be less invasive than a head-mounted tracker, though this conception will be revisited later in this chapter. Thus, while there is a limited number of head-mounted designs (Ebisawa *et al*., 2002; Takegami *et al*., 2002; Li *et al*., 2005; Clarke *et al.*, 2002; Hansen and Pece, 2005) there are at seven to eight times more remote trackers found in the literature (Collet *et al*., 1997; Heinzmann and Zelinsky, 1998; Kim and Ramakrishna, 1999; Matsumoto and Zelinsky, 2000; Ohno *et al.*, 2002; Sirohey *et al*., 2002; Benoit *et al*., 2005; Sun *et al*., 2006; Wallhoff *et al*., 2006; Liang and Houi, 2007; Chen and Ji, 2008; Valenti *et al*., 2008; Yamazoe *et al*., 2008; White *et al*., 1993; Morimoto *et al.*, 2000; Morimoto *et al.*, 2002; Coutinho and Morimoto, 2006; Hennessey *et al.*, 2006; Meyer *et al.*, 2006; Li *et al.*, 2007; Ramdane-Cherif and Nait-ali, 2008; Newman *et al.*, 2000; Shih *et al.*, 2000; Andiel *et al.*, 2002; Ji and Yang, 2002; Beymer and Flickner, 2003; Ishima and Ebisawa, 2003; Noureddin *et al.*, 2004; Ohno and Mukawa, 2004; Shih and Liu, 2004; Park and Kim, 2005; Yoo and Chung, 2005; Merad *et al.*, 2006; Tsuji and Aoyagi, 2006; Park, 2007; Chen *et al.*, 2008; Guestrin and Eizenman, 2008; Kohlbecher and Poitschke, 2008; Hennessey and Lawrence, 2009; Nagamatsu, 2009; Wang *et al*., 2005).

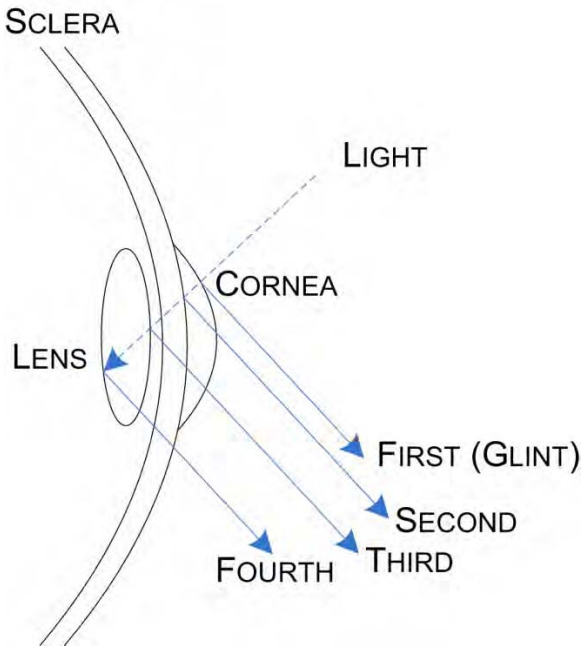


FIGURE 4: ILLUSTRATION OF PURKINJE IMAGES. ADAPTED FROM HANSEN AND JI (2010).

PASSIVE AND ACTIVE ILLUMINATION

Equally limited is the number of eye-trackers (regardless of whether they are head-mounted or remote) that use passive illumination (Hansen and Pece, 2005; Li *et al.*, 2005; Colombo *et al.*, 2007; Newman *et al.*, 2000; Wang *et al.*, 2005; Heinzmann and Zelinsky, 1998; Yamazoe *et al.*, 2008; Matsumoto *et al.*, 2000). Eye-trackers that use infrared light illumination offer several advantages over passive illumination:

- Depending on the exact configuration, the pupil appears as very dark or very bright (called dark- and bright-pupil effect respectively), which allows the pupil to be detected easily and with accuracy. Because of the different reflection properties of the iris and the pupil, this is possible even if the iris appears as dark as the pupil in natural light. The bright-pupil effect is produced when the light source is coaxial to the camera and the dark-pupil effect otherwise (Ebisawa, 1998).
- If a filter is put on top of the camera lens to block non-infrared light, brightness and contrast are kept constant and barely, if at all, affected by other light sources from the external environment. Similarly, shadows may only be formed by the diffusion properties of the infrared source which is within the control of the eye-tracker designer. Last but not least, reflections from objects in the environment do not appear in images captured with active illumination.
- Infrared light is invisible to the human eye and therefore does not distract the subject or cause the pupil to contract.

As can be seen from the list above, active illumination provides several important advantages over passive illumination. In addition, it costs very little to add infrared illumination to any hardware design and there has to be a very compelling reason to use passive illumination. There only two disadvantages to active illumination:

- Active illumination does not perform quite as well in the outdoors and eye-trackers designed to work outdoors (Hansen and Pece, 2005) avoid the use of infrared illumination.
- There is additional light emitted to the eye which will soon be regulated by international safety standards (Hansen and Ji, 2010). However, in all probability, this will only be a

concern for eye-trackers that use multiple infrared sources, especially extreme configurations such as the 3x3 grid of infrared light sources used by Li *et al.* (2007).

GAZE ESTIMATION

In order to determine the gaze, which is primary objective of eye-trackers, a feature-based approach is by far the most commonly used (e.g. Ebisawa *et al.*, 2002; Li *et al.*, 2005; Ohno *et al.*, 2002; Benoit *et al.*, 2005). A feature-based approach means that the eye-tracker identifies a set of feature points in the image such as the pupil and the glint (for active illumination eye-trackers) in order to determine the gaze. In this category, two schemes are possible: (a) 2D regression-based gaze estimation and (b) 3D model-based gaze estimation (Hansen and Ji, 2010). One of the main challenges faced by eye-trackers which use the glint as their reference point is that with movement of the eyes and depending on the where the light source is placed, the glint may fall onto the sclera which makes it very hard to detect as they are both very bright regions. The eye-corners are also used as reference points in some natural-light trackers where the glint is not available (Zhu and Yang, 2002; Valenti *et al.*, 2008).

2D regression-based gaze estimation is performed simply by asking the subject to look at several different calibration points $C_{i...N}$ whose geometry on a surface (usually the screen) is known. At the same time, the pupil or iris locations $F_{i...N}$ are recorded and the calibration points and feature point locations are used to calculate a function f such that $C_i = f(F_i)$. Using this function, consequent pupil positions in the image can be mapped to screen coordinates. In schemes that use the corneal reflection, the vector between the glint and the pupil or iris location is used.

CALIBRATION

On the other hand, 3D model-based gaze estimation, as the name suggests, uses the set of detected feature points in combination with a model of the eye and in some cases the scene to estimate the gaze direction or point of regard in 3D space.

All gaze estimation methods require that a set of parameters are determined through a calibration process; such processes can be categorised into the following categories (Hansen and Ji, 2010):

- Camera calibration, which refers to determining the intrinsic camera parameters (focal length, image sensor size and principal point). As long as the parameters do not change value (e.g. by changing the camera focus setting), they only need to be calculated once.
- Geometric calibration, which refers to determining the relative positions and orientations of the eye-tracker components (camera and light sources) and target surface (screen). As long as the geometry does not change, this needs to be calculated only once.
- Personal or subject calibration, which refers to determining parameters specific to the individual, such as the cornea curvature and the angular offset between visual and optical axes. Such parameters need to be calculated once for each subject.
- Gazing mapping calibration, which refers to determining the eye-to-surface mapping functions. As mentioned earlier, this is usually done by having the subject look at points on the target surface whose geometry is known.

A *fully calibrated system* is a system whose camera intrinsic parameters *and* geometry are known. A *partially calibrated system* is a system whose camera intrinsic parameters *or* geometry is known.

HEAD-MOUNTED EYE-TRACKERS

Most of the head-mounted eye-trackers found in the literature have followed very similar designs and methodologies.

Ebisawa *et al.* (2002; Ebisawa, 1998) present a head-mounted tracker that uses active illumination and two light sources to alternatively produce the dark- and bright-pupil effect and using simple image processing algorithms, the positions of the pupil and glint are detected, the two feature points used to determine the gaze. A good example of the common misconception that the glint does not move when the eyeball moves, when in fact it does; in such cases it is erroneously implicitly assumed that the corneal surface is a perfect mirror and thus if the head is kept fixed, the glint will remain stationary even when the cornea rotates. In some cases, a simplifying assumption that the glint is stationary may however give satisfactory results.

Takegami *et al.* (2002) use a rigid setup where the subject has to rest his chin on a metal frame while holding onto it with their hands, similar to Ramdane-Cherif and Nait-ali (2008). The camera is calibrated and active illumination is used; the algorithm extracts the pupil contour,

from which the pupil flatness (ratio of major and minor axis of an ellipse) is calculated. In this paradigm, the eye is modelled as a sphere and the pupil as a circle. However, the pupil will only appear as a circle in the image if it is exactly coplanar with the camera lens. Otherwise, it appears as an ellipse and by determining the flatness of this ellipse, the subject's gaze can be estimated. It is reported that in this setup no subject calibration is necessary.

Li *et al.* (2005) present an active illumination head-mounted eye-tracker and a novel method to estimate the glint and pupil feature points. Once the glint is located, radial lines are extended to locate candidate edge points for the pupil contour, which are then optimised using the Random Sample Consensus (RANSAC) algorithm. The distance between the glint and the pupil centre is then used to calculate 2D gaze on a screen.

Clarke *et al.* (2002) present a high-frequency (400Hz) head-mounted system that makes use of binocular expensive CMOS cameras that are placed on the side of the headset (the eye images are captured through mirrors). The system presented by Hansen and Pece (2005) is quite unique in that it is reported to be able to track in any lighting conditions and can switch between infrared and non-infrared configurations without changing its parameters. However, it uses particle filtering to do the tracking, which is generally computationally complex and is not easy to implement in real-time (Kwok *et al.*, 2004).

REMOTE EYE-TRACKERS

Remote eye-trackers can be easily categorised between:

- Trackers which use a single camera and none or a single infrared source (Collet *et al.*, 1997; Heinzmann and Zelinsky, 1998; Kim and Ramakrishna, 1999; Matsumooto, 2000; Ohno *et al.*, 2002; Sirohey *et al.*, 2002; Benoit *et al.*, 2005; Sun *et al.*, 2006; Wallhoff *et al.*, 2006; Liang and Houi, 2007; Chen and Ji, 2008; Valenti *et al.*, 2008 and Yamazoe *et al.*, 2008).
- Trackers which use a single camera but multiple infrared sources (White *et al.*, 1993; Morimoto *et al.*, 2000; Morimoto *et al.*, 2002; Park *et al.*, 2005; Coutinho and Morimoto, 2006; Hennessey *et al.*, 2006; Meyer *et al.*, 2006; Li *et al.*, 2007 and Ramdane-Cherif and Nait-ali, 2008).

- Trackers which use multiple cameras and one or more infrared sources (Newman *et al.*, 2000; Shih *et al.*, 2000; Andiel *et al.*, 2002; Clarke *et al.*, 2002; Ji and Yang, 2002; Beymer and Flickner, 2003; Ishima and Ebisawa, 2003; Noureddin *et al.*, 2004; Ohno and Mukawa, 2004; Shih and Liu, 2004; Yu and Eizenmann, 2004; Park and Kim, 2005; Yoo and Chung, 2005; Merad *et al.*, 2006; Tsuji and Aoyagi, 2006; Park, 2007; Zhu and Ji, 2007; Chen *et al.*, 2008; Guestrin and Eizenman, 2008; Kohlbecher and Poitschke, 2008; Hennessey and Lawrence, 2009 and Nagamatsu, 2009).

One of the major problems with remote eye-trackers is movement of the subject's head. Not only is the resolution of the eyes reduced because of the distance between the subject and the camera, but the subject's head is able to move unrestrictedly and the eye-tracker must be able to cope with that if it is going to track the subject's gaze successfully.

Single camera remote systems use a variety of different methods to compensate for head-movement and calculate the gaze.

Collet *et al.* (1997) detect the location of the eyes and nose and use these feature points to calculate face orientation and gaze. Several similar schemes appear in the literature; Heinzmann and Zelinsky (1998) use the mouth and eye corners, Wallhoff *et al.* (2006) use the eyes and mouth, Chen and Ji (2008) use the nose and eye corners, Valenti *et al.* (2008) use the eye corners only and Yamazoe *et al.* (2008) use the mouth, nose and eye corners. Head-pose estimation is done similarly with stereo camera systems; for example, Newman *et al.* (2000) use the eye corners and mouth corners.

In a screen setup where the distance of the subject from the screen is known, Kim and Ramakrishna (1999) use the point between the eyes to compensate for small head-movement and the iris length to calculate the distance between the camera and the eyeball. Finally, 3D gaze is calculated using the iris centre. Two similar setups are employed by Matsumoto (2000) who uses the eye corners, anthropological data and the iris radius to initialise a 3D eye model that is used then to estimate 3D gaze; the mouth and eye corners are used to estimate face orientation. Wang *et al.* (2005) also calculate the iris radius from the image to facilitate a 3D model of the eyeball and use the eye-corners to disambiguate between two possible solutions for the gaze vector. The eye corners and iris centre are also used by a few other systems (Tian *et al.*, 2000;

Benoit *et al*., 2005). Sirohey *et al.* (2002) present a system where the iris and eyelids are detected and tracked.

The system described by Ohno *et al.* (2002) is "traditional" in that is uses the glint and pupil feature points but also includes an eyeball model to calculate 3D gaze. The system by Sun *et al.* (2006) uses a similar model which also includes the eye-corners. Neural networks have been used to determine gaze in some systems (e.g. Stiefelhagen *et al.*, 1997).

A single-camera remote system that can classify eye-movements in the classes defined by the NLP EAC model (up left, up, up right, left, centre, right, bottom left, bottom and bottom right) is proposed by Liang and Houi (2007), which classifies gaze into eight (8) different classes by calculating the difference between the pupil looking forward and the current pupil location.

With increased cameras and/or light sources, it is possible to use 3D models that result in greater theoretical accuracy. The detailed description of these models and how they operate in a multi-glint or multi-camera setup is beyond the scope of this review and the interested reader is referred to the excellent reviews already available (Guestrin and Eizenman, 2006; Villanueva *et al.*, 2007; Villanueva and Cabeza, 2007). In single-camera, multiple-glint cases, calibration to the subject is still required; in their work, Villanueva and Cabeza (2008) mathematically prove that a system with one camera and two glints requires a minimum of one calibration point to give geometrically correct results. Two systems that do not abide to this rule are the systems by Morimoto and Flickner, (2002), which is reported to have lower accuracy, and Kohlbecher and Poitschke (2008), which use the pupil ellipse to extract the 3D orientation. Systems with more than one camera and sources are able to operate calibration-free (Shih *et al.*, 2000; Nagamatsu, 2009).

Arrays of infrared sources larger than two have been used in limited occasions; for example, Coutinho and Morimoto (2006) use five sources (four on the screen and one on the camera), Meyer *et al.* (2006) use four infrared sources, Li *et al.* (2007) use an array of 3x3 infrared sources, Guestrin and Eizenman (2008) use four infrared sources. These arrays are used either for the ability to calculate 3D parameters or to overcome the problem mentioned earlier when the glint is positioned in the sclera.

Appearance-based methods (Stiefelhagen *et al.*, 1997; Tan *et al.*, 2002; Xu *et al.*, 1998) are an alternative approach to feature-based tracking reviewed so far. These methods attempt to detect and track the eyes by directly using their photometric appearance (either through image intensity or through its response to a filter), instead of extracting features from it. From the large list of appearance-based eye-trackers in the comprehensive review by Hansen and Ji (2010), only one was designed to work with a head-mounted eye-tracker (Hansen and Pece, 2005). The latter eye-tracker uses particle filtering to track gaze and while it is very robust, it is also very complex. There are several reasons why appearance-based methods are less favoured for this application:

a) They usually require a large amount of training data.
b) They are used in remote eye-tracking systems which means that they would most likely require significant modification to be used with the close-up pictures of a head-mounted tracker. As will be discussed in more detail in Chapter 4, the task of modelling or detecting eye features becomes harder as the camera gets closer to the eye. First, the appearance of the eye-corners is significantly different when viewed close up than when viewed from a remotely placed camera and second, the change in the camera view angle may significantly change the appearance of the eye. Both changes would probably decrease the accuracy of an appearance-based approach or require ever larger training data sets.
c) They are much more difficult to evaluate as exact landmarks are not easily defined because they are based on contours.

The review of remote eye-trackers and the methods involved has been intentionally brief for two reasons. First, as it will be argued below, remote eye-trackers are unsuitable for this application and thus, delving into the complexities of such systems would only serve to deviate from the scope of this thesis. Second, there are already detailed reviews of such systems (Guestrin and Eizenman, 2006; Hansen and Ji, 2010).

EYE-TRACKER INVASIVENESS

At the top of the requirement list is the minimisation of invasiveness, the ability to track even the most extreme eye-movements and ease of use. While a formal definition of invasiveness has not

been found in the eye-tracking literature, it is normally regarded as a combination of the following factors:

a) whether it requires contact to the eyeball or other parts of the body
b) whether it restricts any type of movement (e.g. head) and
c) if it is mounted on the head or body, how much it weighs and how long it takes before this becomes uncomfortable for the user

With invasiveness defined by the aforementioned factors, a remote eye-tracker is the least invasive type of eye-tracker that can be developed as it is not mounted on the subject and thus does not impose any further weight. Also, as mentioned earlier, because most remote eye-trackers encompass some form of head pose estimation, some head-movement is acceptable. Of course, how much movement is acceptable is solely defined by the performance of the head pose estimation.

Another important factor that determines the invasiveness of an eye-tracker and is rarely, if ever, explicitly mentioned in the literature is how much the subject is aware of his or her eyes being tracked. The feeling of being "watched" often makes people self-conscious and aware of every movement they make. Depending on what the task of the experiment is, it may also trigger performance anxiety. In any case, in experiments where rapport between the subject and the experimenter is important, it surely does not help if the subject is aware of being the subject of not only the experiment itself but the eye-tracker too. Similarly, if an elaborate subject calibration procedure is required, it can remind the subject of the eye-tracker's presence and thus contribute towards reducing their comfort during the experiment.

Other than an out-dated comparison of five commercial eye-trackers on comfort (Williams and Hoekstra, 1994), there is no formal study of the invasiveness of different eye-trackers and the subjective experience of subjects during an experiment.

SUITABILITY OF REMOTE EYE-TRACKERS

Whether remote eye-trackers are any less invasive or not, they are definitely much more expensive to build as they usually require more than one camera and because of the distance

from the camera to the subject's eyes, higher image resolution and fidelity is required, which makes them more expensive than cameras that may be used on a head-mounted solution.

For this particular application, remote eye-trackers may prove impractical for several additional reasons:

- In applications where the subject is required to look at a screen (such as tracking how people browse a website), the camera can be hidden in the screen and thus minimize invasiveness in this way. However, in an interview between subject and experimenter, this is significantly harder to achieve.
- Future developments in this kind of research may require that both the subject and experimenter's eye movements are tracked simultaneously. If two or more cameras are required for each pair of eyes tracked, the setup quickly becomes more expensive and even harder to conceal. Furthermore, the more the cameras, the harder the system becomes to setup and simplicity is considered to be a key requirement in this application.
- Head-pose estimation is required and it comes at the cost of additional computational complexity and where more than one camera is involved, additional hardware costs. In fact, as referenced earlier, some remote eye-trackers that use infrared illumination and a glint-centred reference system go to great lengths to ensure head-pose invariance by using multiple infrared light sources.
- Depending on the accuracy and limits of head-pose estimation, head-movement may be restricted. Also, regardless of how good the head-pose estimation is, there may be cases when the pupil is not captured sufficiently well or at all by the remote camera. This may happen if for example the camera is placed below head level, the subject's head is tilted upwards and the subject performs an extreme upwards eye-movement.
- A fully calibrated system may be required (e.g. Meyer *et al.*, 2006). This means that not only the camera's intrinsic parameters need to be known but also the geometric topology of the camera(s) and the subject. While the calibration process is plausible for a rigid setup in front of a computer monitor, doing so for an interview-type experiment may be impossible as it requires much more flexibility.
- While processing power is much cheaper than it used to be, the enhanced processing of a remote system significantly demotes its attractiveness.

BUILDING A HEAD-MOUNTED EYE-TRACKER

A lightweight head-mounted eye-tracker can be minimally invasive (no contact to the eye, small weight, user experience similar to wearing vision glasses) and it does not require integration of different and isolated systems (remote systems require head pose estimation which is usually done with a second camera). In terms of accuracy, both types of systems can perform well depending on the hardware and setup used (e.g. Meyer *et al.*, 2006; Villanueva *et al.*, 2007). Further, given that there exists a publicly-available low-cost design of a light-weight head-mounted eye-tracker that is also easy to assemble (Babcock *et al*., 2003; Babcock and Pelz, 2004), a head-mounted design is a good choice for this project.

Amongst head-mounted eye-trackers, there are more design and implementation choices to be made, those of light source (natural light or passive versus infrared light or active) and gaze estimation method.

Adding infrared light source(s) is a relative easy procedure and the choice between active and passive illumination significantly affects the problem formation. When passive illumination is used, the input image is subject to severe brightness changes depending on the light source of the environment and artefacts caused by shadows, reflections from objects in the environment and pupil tracking becomes very hard to impossible depending on the subject's eye colour; in fact, as seen earlier, it is common for passive eye-trackers to track the iris as it is indistinguishable from the pupil.

Of course, using infrared light does not solve every problem. One very important challenge of designing an eye-tracker (whether active or passive) is that of finding a reliable reference feature point. In most cases, a glint-centred reference system is used (e.g. Ebisawa *et al.*, 2002; Li *et al.*, 2005; Ohno *et al.*, 2002; Benoit *et al.*, 2005) where the distance between the pupil and the glint is what determines the output of the eye-tracker (usually a mapping to the screen or 3D world). Specifically for remote systems, while eye corners are often used for head pose estimation (Lam and Yan, 1996; Zhang, 1996; Feng and Yuen, 1998; Tian *et al.*, 2000; Sirohey *et al.*, 2002; Wang *et al.*, 2005; Xu *et al.*, 2008) they are a lot less preferred as reference points (Zhu and Yang, 2002; Valenti *et al.*, 2008).

In terms of gaze estimation, 2D regression-based estimation requires a surface to map eye-movements onto; in the case of an interview where the subject is not looking at a screen, this method cannot be applied and thus rules out the design for this application. In the case of 3D model-based gaze estimation the camera's intrinsic parameters and scene geometry (relative position of screen, subject, camera and light source(s)) need to be known. This is a great disadvantage, especially for head-mounted systems where the scene geometry is significantly affected by the subject's face morphology (i.e. even if the camera is always placed at a fixed location, the variance of eye cavities in each subject will determine the exact distance between the camera and the eye). Furthermore, this information can be difficult to obtain and it restricts the eye-tracker to a rigid setup. Thus, for this application gaze mapping techniques are unnecessary.

From the above survey, it is appropriate to conclude with a remark made by Hansen and Ji (2010), who state that "each technique has its advantages and limitations, but the optimal performance of any technique also implies that its particular optimal conditions with regard to image quality are met". In other words, each technique will work well for the image quality that it has been designed for. In the case of the REACT eye-tracker, the application requirements have driven the selection of the design that offers the best compromise and in Chapter 4 the feature detection algorithms that make tracking with the REACT eye-tracker possible are outlined.

CHAPTER 4: FEATURE EXTRACTION

After the review of eye-tracking systems in Chapter 3 and the relevant discussion in terms of their requirements, a novel eye-tracker is introduced in this chapter which fulfils them. The hardware design of the eye-tracker is deliberately omitted from this chapter as it is largely based on a previous design by Babcock and Pelz (2004) and is described in Appendix A. Instead, this chapter focuses on the algorithms involved in extracting the features from the input images and calculating the 2D gaze angle.

Thus, what follows is an extensive discussion of the image acquisition and image properties that will help gain insight into the computer vision problem of extracting the eye features and its complexity. Then, the algorithms that are responsible for extracting the pupil, iris radius and eye corners are described in-depth. Finally, the calculation of the 2D gaze vector is also described in-depth before moving onto the evaluation of these respective components in Chapter 5. In each section, the intermediate steps are visually illustrated; sample visualisations of the complete set of features for each subject is shown in Table 8 at the end of this chapter while randomly selected frames throughout each test sequence with the pupil marked are shown in Table 9, which is also located at the end of this chapter. The last section of this chapter discusses the computational complexity of the algorithms involved in extracting the features.

ACQUSITION AND IMAGE PROPERTIES

This section intends to examine in detail the properties of the source images taken with the REACT eye-tracker in order to better understand the problem at hand and how it may be best solved.

Camera and light source

The camera used is Supercircuits PC206XP which is a grey-level pinhole camera and it captures images 640 pixels wide by 480 pixels tall at 29.97 frames per second (NTSC). A standard infrared (IR) LED that transmits light at a wavelength of 940nm is used as the light source and a Kodak Wratten 87c IR filter is placed on top of the camera to filter out non-infrared light. For further details on the hardware of the REACT eye-tracker, please refer to Appendix A.

Interlacing

First and foremost, the input images are interlaced and thus introduce a severe artefact to be dealt with. Interlacing is a technique that was first developed with the introduction of the cathode ray tube (CRT) televisions and was a means to improve the quality of the picture without increasing the bandwidth requirements (Luther and Inglis, 1999).

Interlacing works by labelling each frame into odd and even fields (fields are synonymous to scan lines or just lines) and refreshing the odd fields at different intervals to the even fields (Figure 5).

For vision processing, non-progressive (interlaced) images create several problems as if there is any motion in the image, no entity is continuous (this is often described as motion blur). An example of this can be seen in Figure 6 (a); as the eye is moving quickly upwards and the two fields are updated at different points in time, three different pupil outlines can be seen in the image: one from the pupil position at time T (odd field), another from the pupil position at time T+1 (even field) and their overlap. Thus, computer vision algorithms would face significant difficulties in detecting the pupil from the original image (or other features for the matter).

De-interlacing methods are largely undocumented in computer vision literature and are mostly discussed informally on the World Wide Web, e.g. Wikipedia, 2009. Of the reported methods of de-interlacing a video, some are more complex (and slower) than others. For the REACT eye-tracker, the most simplistic[3] of methods was used to de-interlace the video: splitting each field to a frame of its own (referred to as half-sizing, Wikipedia, 2009). Thus, the odd fields in frame number N are collated to compose frame number $2 \times N$ and the even fields in frame N are collated to compose frame number $2 \times N + 1$. The result video has twice the frame-rate of the original; an example output of the de-interlacing may be seen in Figure 6 (b). When the pupil or any other object is moving really fast some smearing will still be visible; however this smearing cannot be removed completely using de-interlacing as it is a limitation imposed by the speed that the camera captures frames at.

[3] More complex schemes usually involve using interpolation to recover missing samples and/or motion compensation (Wikipedia, 2009).

It is also worth noting here that because of this de-interlacing, the image is vertically cut in half i.e. vertical lengths are half their original length.

From here on, all displayed images will be de-interlaced or the products of the de-interlaced source.

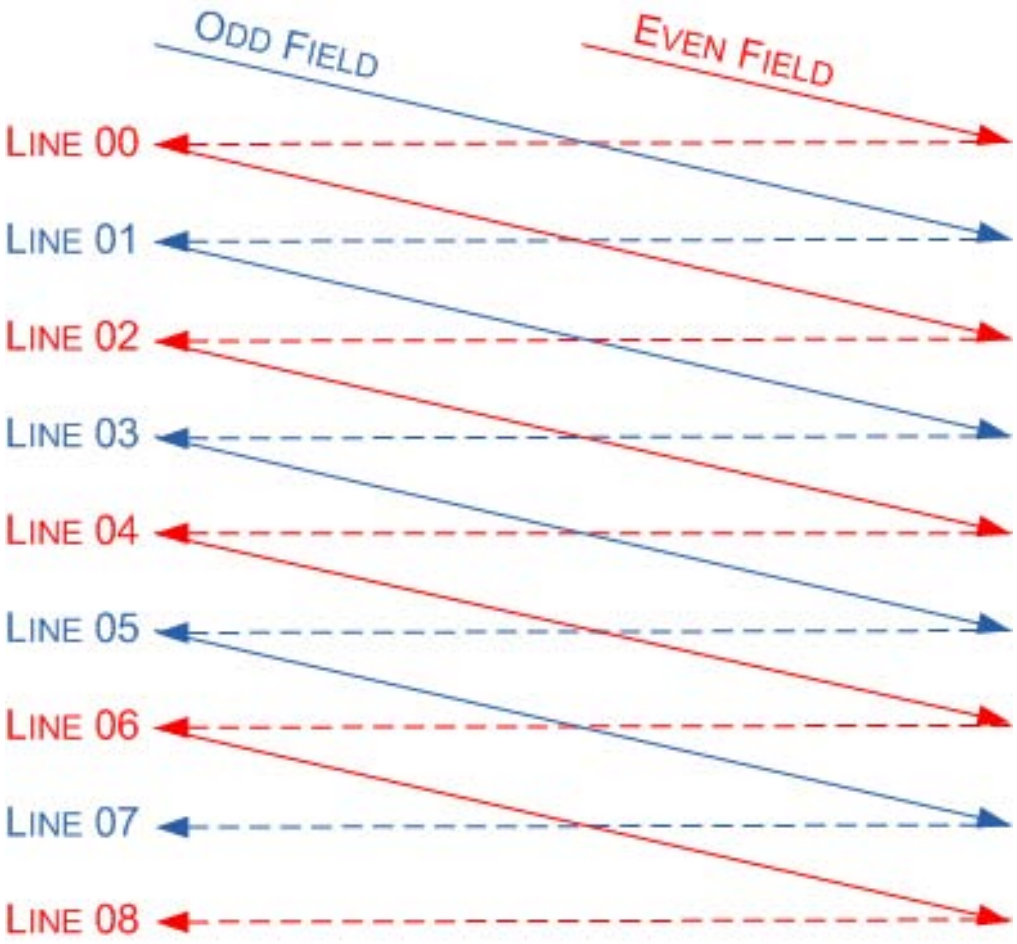
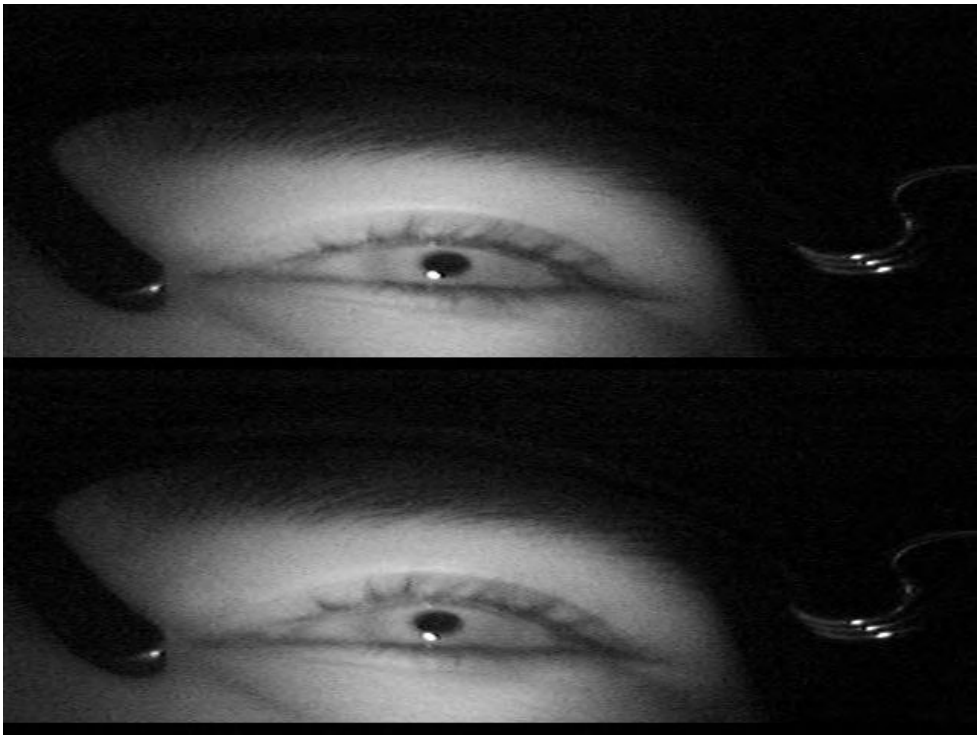


FIGURE 5: ILLUSTRATION OF THE INTERLACING PROCESS.

(A) EXAMPLE OF INTERLACED FRAME WHEN THE EYE IS MOVING.



(B) EXAMPLE FRAME AFTER IT HAS BEEN SPLIT INTO TWO FIELDS.

FIGURE 6: ILLUSTRATION OF DEINTERLACING

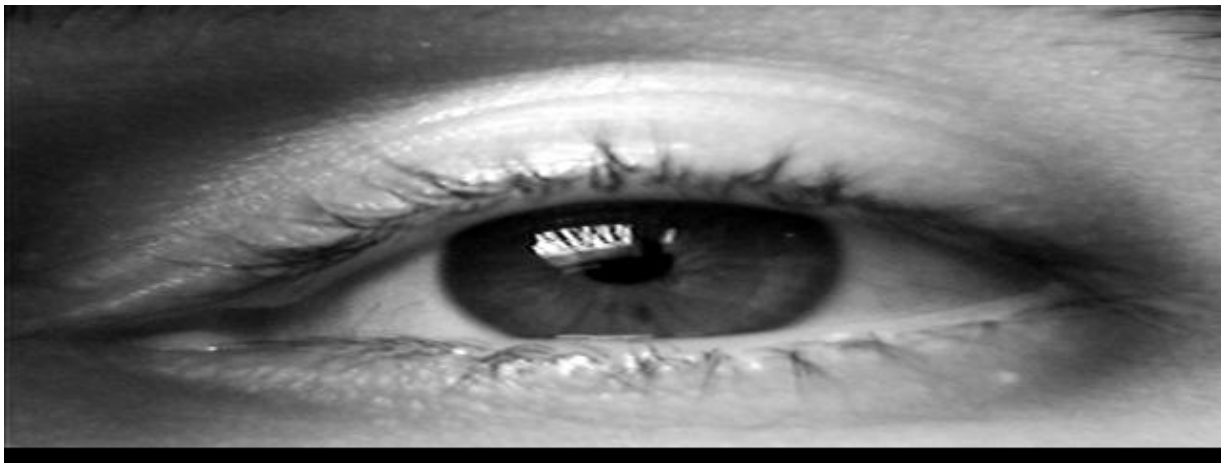
Infrared lighting

As discussed in Chapter 2, the REACT eye-tracker was purposefully built to use infrared lighting due to the immediate advantages it offers over normal light. Re-iterated here for completeness, recording the eye-tracker images without the infrared light source and non-infrared filter can potentially introduce severe changes in the output's histogram, contrast and brightness. Natural light eye-trackers (e.g. Hansen and Pece, 2005) face a great challenge in dealing with such variations; these variations can be a result of two potential reasons:

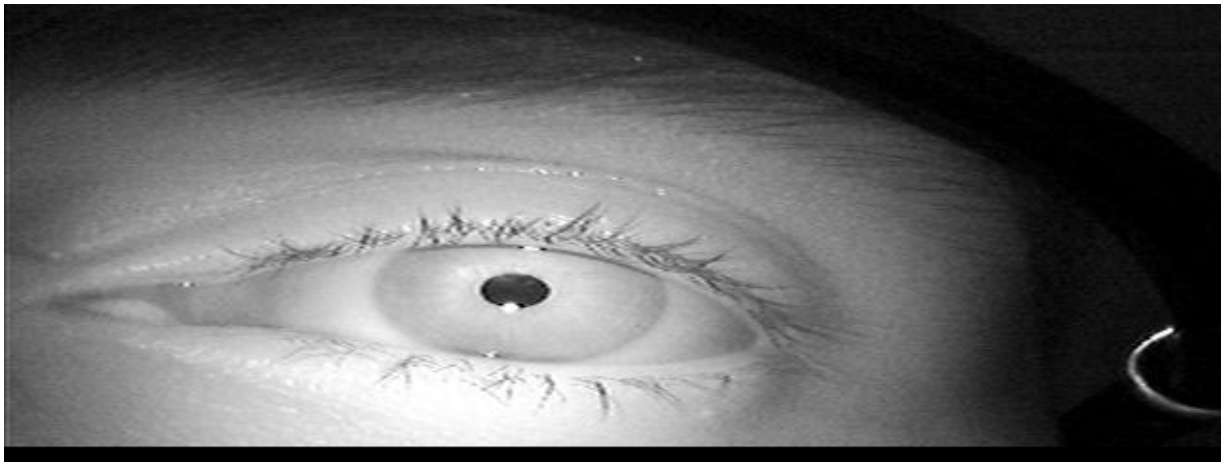
- First of all, natural environment light sources during the day (e.g. sun) can vary depending on the current point in time.
- Secondly, because we use the normal light spectrum in our everyday lives, it is open to interference from several sources whether working inside or outside. These interferences would be tough and potentially expensive to control. In a usual home or office environment, there are several types of light sources that can cause such interference; this may happen either when the light source varies the intensity of its output over time as noted earlier or, when the subject changes the orientation of his or her head with respect to the light source (for example, when turning the head towards or away a lamp).

Thus, the infrared setup of the REACT eye-tracker is ideal with respect to the above-mentioned problems as infrared lighting provides a consistent light source that does not fluctuate over time and it is not affected by the subject's position in space as the light source is fixed in a position relative to the subject's head. This effect can be clearly seen in Figure 7; in (a) where the eye is illuminated by visible light, the reflection of a window can be seen on top of the pupil, as well as the non-uniform lighting distribution across the image. In contrast, in (b) where the eye is illuminated by infrared light, the image brightness is solely dependent on the diffusion properties of the infrared source (infrared LED).

Furthermore, the infrared setup offers the significant advantage of the dark pupil effect where the pupil appears very dark and the iris appears very light independent of the subject's eye colour, due to the different infrared light reflection properties of the pupil and the iris. In comparison, the visible light image makes the separation of iris and pupil very hard, especially considering the reflections of other objects on the eye.



(A) IMAGE TAKEN IN THE VISIBLE LIGHT SPECTRUM (I.E. WITHOUT INFRARED ILLUMINATION).



(B) IMAGE TAKEN IN THE INFRARED LIGHT SPECTRUM

FIGURE 7: ILLUSTRATION OF IMAGES TAKEN IN THE VISIBLE VERSUS INFRARED LIGHT SPECTRUM.

Figure 8 illustrates a set of example source images and their corresponding heat map images. Heat maps are generated by mapping the [0, 255] grey scale range to a special ordering of the RGB (Red-Green-Blue) colour space where low intensity values are coloured blue (cold) and high intensity values are colour red (hot). Heat map images were found to be useful in visually inspecting grey scale images as they expand the 8-bit [0, 255] range to $2^{24} = 16{,}777{,}216$ possible values and can provide insights to the structure of the image by increasing the visibility of intricacies in the image that would otherwise be hardly distinguishable.

As can be seen in Figure 8, the infrared light source is directly pointed to the eye centre from below and thus the majority of "heat" can be seen along the lower eyelid and on the eyeball. The dark pupil effect can also be easily noticed both in the original and heat map images. The darkest/coldest area is usually the pupil whereas the brightest/hottest area is the *glint*[4]. Once again, it can be seen how the pupil is always dark whereas the iris appears as a shade of grey, regardless of the subject's eye colour. As already mentioned, this is because of the different reflection properties of the pupil and iris.

To further establish the difficulty in detecting the eye features, especially the iris boundaries and the eye corners, the performance of several standard edge and corner detectors and other similar algorithms is presented below. The aforementioned advantage of infrared illumination of the separation between the pupil and iris is also a disadvantage in detecting the iris boundaries. This is because in some subjects, the iris appears as a very light shade of grey and it can hardly be distinguished from the sclera (see Figure 8 for such examples). This is because the iris usually darkens on the border to the sclera but this dark border is less evident (thinner) in some people.

Further, because the camera is placed close to the eye, the inner and outer corners appear to have a different shape. This appears to be due to the appearance of the tear gland which is part of the inner eye corner. Thus, the inner eye corner extends further than the outer eye corner, as measured from the eyeball centre.

[4] The small white circle that is the direct reflection of the infrared LED on the cornea and appears as red in the heat-mapped images.

Figure 9 visually illustrates the performance of standard edge and corner detectors as well as some custom ones[5]:

- Sobel edge operator (Gonzalez and Woods, 2001), with a 3x3 kernel.
- Canny edge detector (Canny, 1986) with two different set of parameters.
- Laplace operator (Gonzalez and Woods, 2001), with filter aperture of 3.
- Laplacian of Gaussian (LoG) filter (Gonzalez and Woods, 2001) – 5x5 Gaussian kernel, sigma set to 1.4 and Laplace filter aperture of 3. Both the absolute value of the filter output and the zero-crossings are illustrated.
- Minimum and maximum Eigen-values of the 2x2 gradient (or covariance) matrix extracted on a 3x3 window for each pixel. These Eigen values can be a reliable source of corners or other features and are used in other popular vision algorithms such as the Harris operator (Harris and Stephens, 1988) and the KLT feature tracker (Shi and Tomasi, 1994).
- Harris edge and corner detector (Harris and Stephens, 1988).
- Variance map – computed as the variance of a 5x5 window for each pixel.
- Partial x- and y-derivatives, with and without local maxima suppression.

As can be seen in Figure 9, most complex corner detectors perform poorly both for the detection of the iris boundaries as well as the detection of the eye corners. In fact, in most cases the iris boundaries are not preserved as strong edges in the output whereas the filtered output near the eye corners is heavily obstructed by edges from shadows on the sclera and eyelashes. In the next section, where the feature detection is explained in depth, the partial x- and y- derivatives were chosen over the more complex detectors because they were found to give similar output at a fraction of the time.

[5] The original image has been smoothed using a Gaussian 5x5 kernel before being processed with each algorithm.

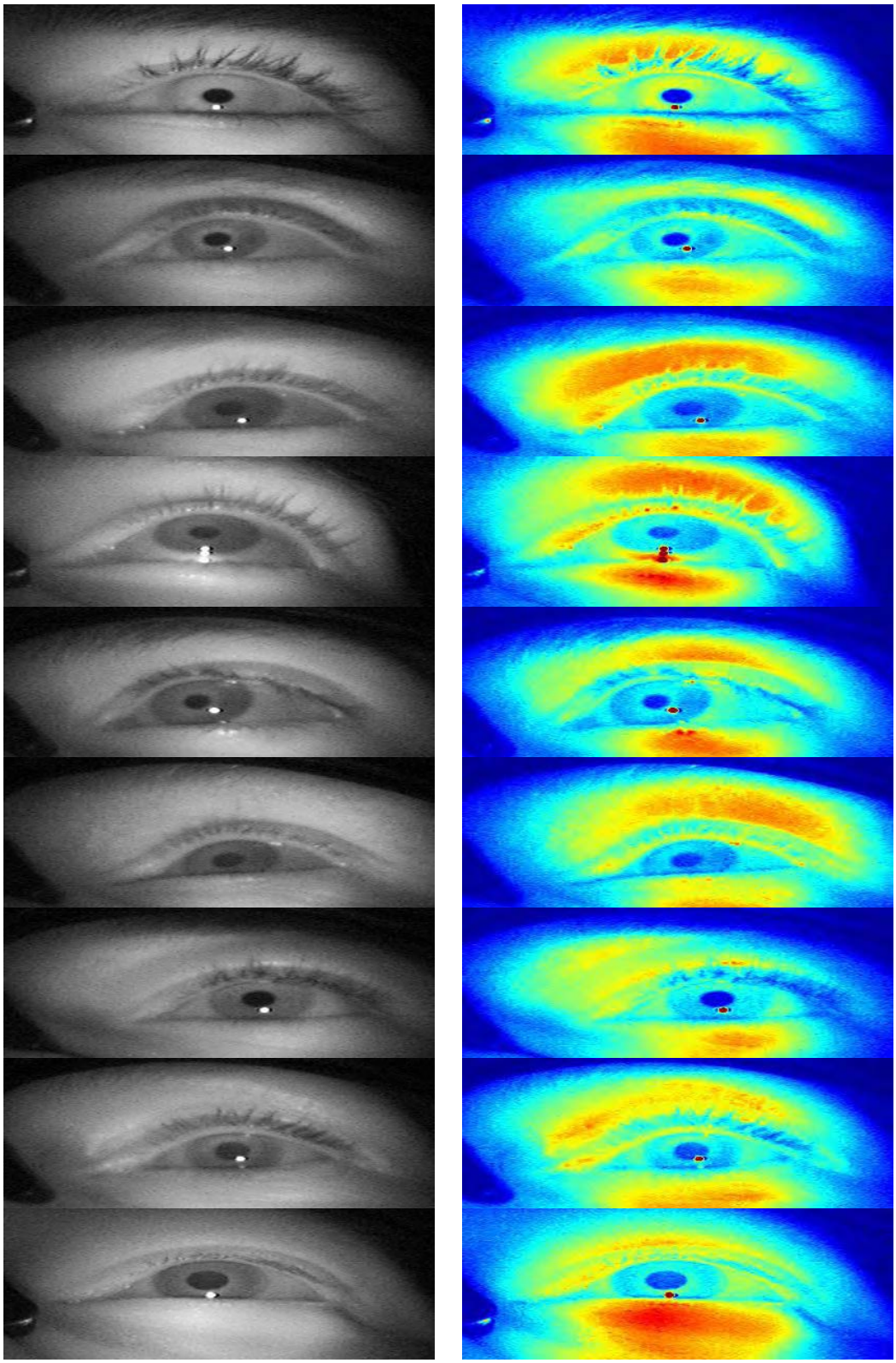
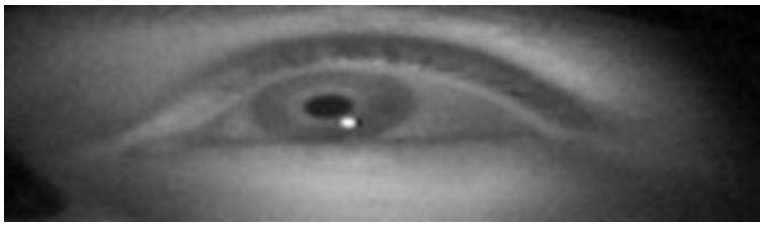
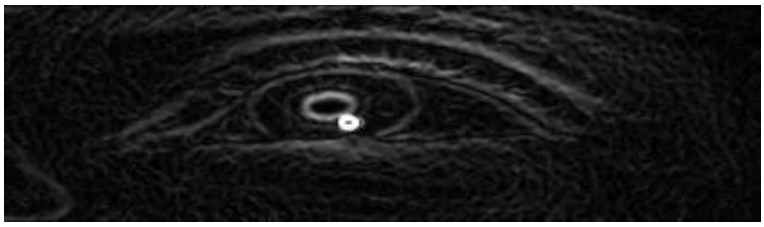


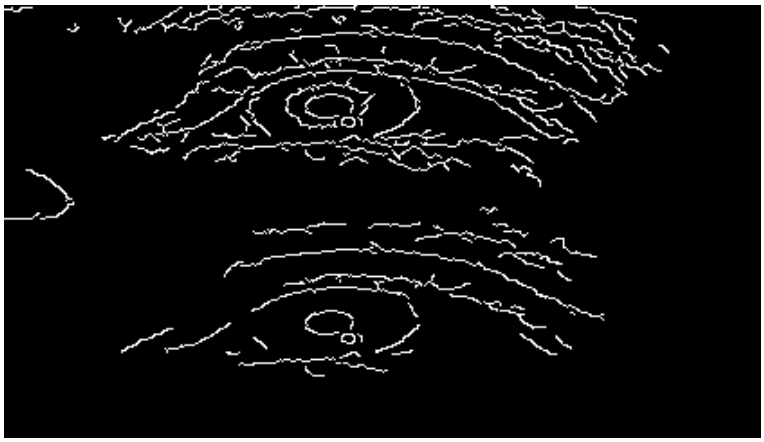
FIGURE 8: EXAMPLES OF INPUT IMAGES (LEFT) AND CORRESPONDING HEAT MAPS (RIGHT) FOR SUBJECTS 1 THROUGH 9. IMAGES WERE CROPPED TO INCREASE VISIBILITY.



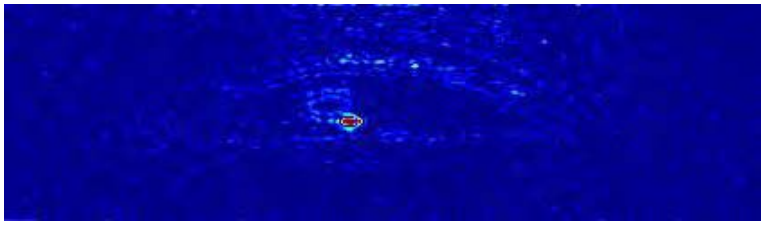
ORIGINAL (GAUSSIAN-SMOOTHED WITH A 5X5 KERNEL)



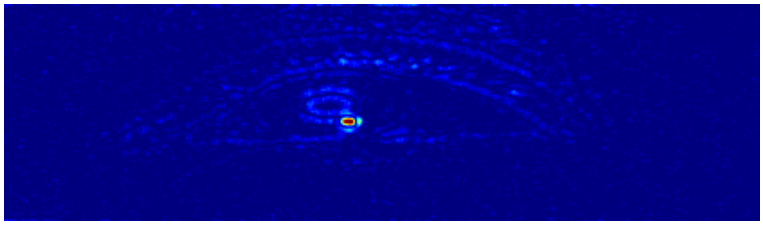
SOBEL 3X3



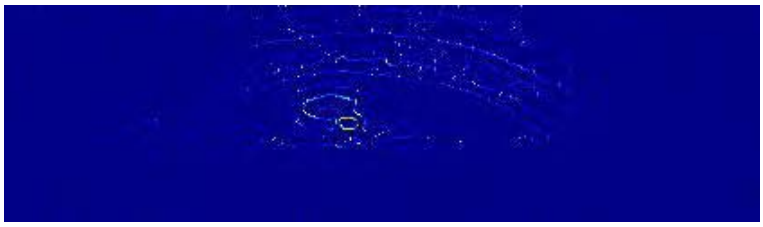
CANNY (TWO DIFFERENT SETS OF PARAMETERS)



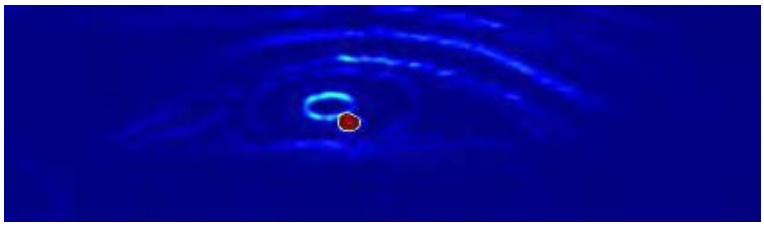
LAPLACE: ABSOLUTE VALUE (APERTURE = 3)



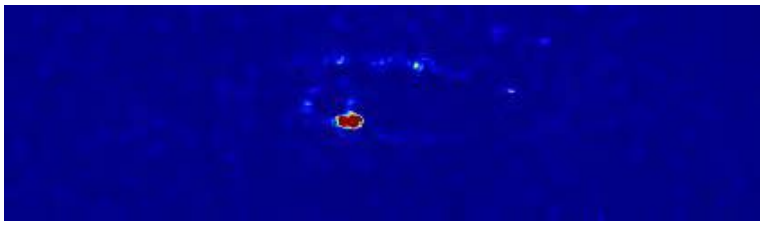
LAPLACIAN OF GUASSIAN: ABSOLUTE VALUE (GAUSSIAN 5X5 KERNEL, SIGMA = 1.4, LAPLACE APERTURE = 3)



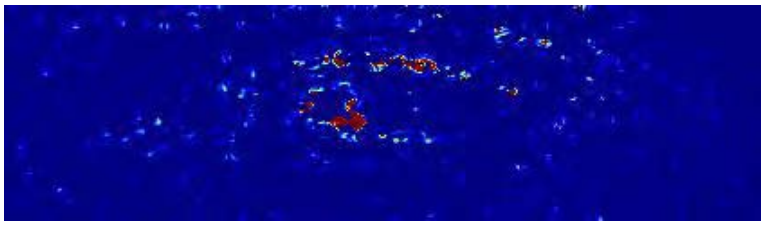
ZERO-CROSSINGS IN LAPLACIAN OF GUASSIAN



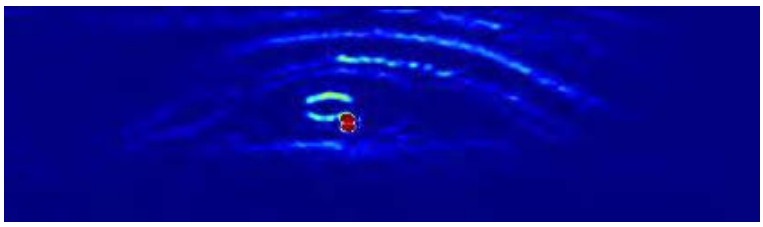
MAP OF THE MAXIMUM EIGEN-VALUES (BLOCK SIZE = 3, APERTURE = 3)



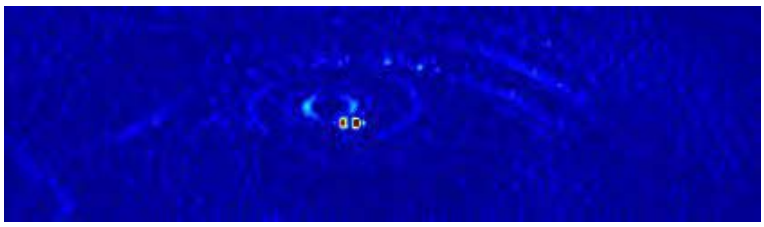
MAP OF THE MINIMUM EIGEN-VALUES (BLOCK SIZE = 3, APERTUERE = 3)



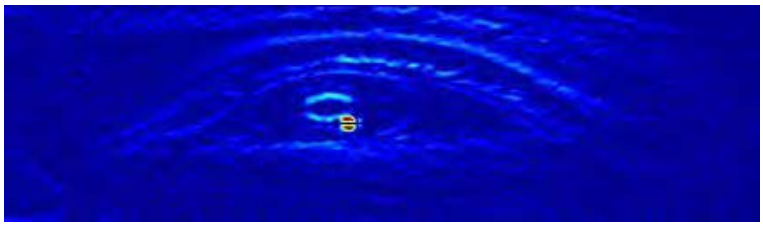
HARRIS CORNER DETECTOR OUTPUT



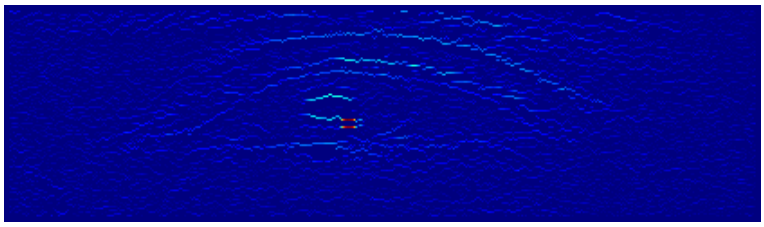
VARIANCE MAP (5X5 KERNEL)



PARTIAL X-DERIVATIVE



PARTIAL Y-DERIVATIVE



PARTIAL Y-DERIVATIVE, LOCAL MAXIMA SUPPRESSED (1X3 WINDOW)

FIGURE 9: EXAMPLES THE PERFORMANCE OF POPULAR CORNER DETECTORS. IMAGES WERE CROPPED AND SCALED WHERE APPROPRIATE TO INCREASE VISIBILITY.

EYE-FEATURE DETECTION ALGORITHM OVERVIEW

The REACT eye-tracker has been designed to detect a set of eye features (pupil location, iris radius and eye corners location) that can potentially be used to set up various different eye-tracking projects. While other configurations can be easily implemented, the default configuration of the REACT eye-tracker uses these features to determine the two-dimensional direction of gaze, in degrees. A high-level block diagram of the feature detection and 2D gaze calculation is shown in Figure 10. A classification scheme (e.g. up and to the left) is proposed and implemented for the case study in Chapter 6. The eye-tracker processes images off-line although this is an implementation detail and not a requirement.

Taking advantage of the dark pupil effect discussed earlier, estimating the location of the pupil can be achieved using a simple thresholding technique and ellipse fitting. Global thresholding (versus adaptive) is used and therefore further refinement of the pupil contour is performed using an active contour (also known as a "snake"; Blake and Isard, 1998).

Then, using the concept of grey level edge strength used in cell segmentation tasks (Zhou and Pycock, 1997) the iris radius is detected and quantified. The eye feature detection is completed by initiating a search for the eye corners at the iris boundaries; edge information is obtained by performing local-maxima suppression of the y-derivative of the input image.

Both the iris and eye corner detection require that the subject is looking *approximately* straight ahead. This requirement is because of previously-mentioned reasons: a) as the eye moves, the iris is easily obscured by the eyelids/eyelashes and will also change shape b) as the eye moves, the eyelids near the eye corners will change shape and reveal or obscure more of the eyeball, thus severely altering the appearance of the corners.

Detecting the iris serves both to localize the eye corners position and to calculate the 3D gaze when a calibrated camera is available, like in the system presented by Wang *et al*. (2005). Further, the eye corners serve as reference points and are used to calculate the reference-axes for the calculation of the gaze angle as well as re-calibration of the eye-tracker over long sequences when the glasses may have changed position on the subject's head.

Thus, even though it does not require an explicit, interactive subject calibration, this default configuration of the REACT eye-tracker requires only one frame to calibrate for each individual subject.

In the following sub-sections, the eye feature detection and direction of gaze calculation algorithms are extensively discussed.

PUPIL CONTOUR ESTIMATION

The first step in detecting the pupil is thresholding the input image I_{input} to a binary image I_t with a threshold value T such that:

$$I_t(x,y) = \begin{cases} 0, when\ I_{input(x,y)} > T \\ 255, otherwise \end{cases}$$

Example output of thresholding is demonstrated in Figure 11.

Using connected components labelling (Gonzalez and Woods, 2002), the binary image can be converted to a higher-level description of the image content. In brief, the labelling algorithm works by scanning the image top-to-bottom, left-to-right and labelling each pixel and those neighbour pixels that are connected to it with the same label, either with a 4- or 8-way connectivity criterion (see Figure 12). A second image scan is done in order to adjust equivalent labels. The resulting image contains the same intensity value (label) for each set of connected pixels (blob).

As an extension to the standard connected components labelling algorithm, during the labelling process, the following information is collected about each blob: a) the bounding rectangle, defined by the top-, bottom-, left- and right-most pixels, b) the area, defined by the number of pixels that have this blob's label and c) a list of the locations of all the pixels that compose this blob. This information is used to assist consequent connected component processing; for example, a minimum and maximum blob area is defined (A_{min} and A_{max} respectively) and used to filter blobs resulting from random noise (too small) and blobs that are formed in the darker areas of the image (where the infrared illumination fades rapidly and the resulting blobs occupy a very large area).

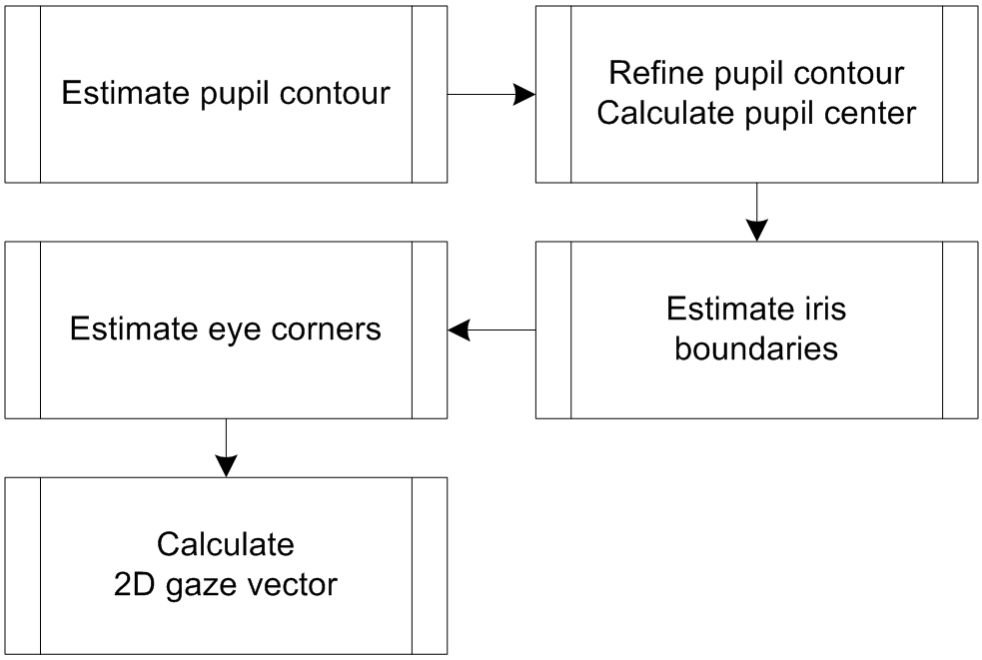
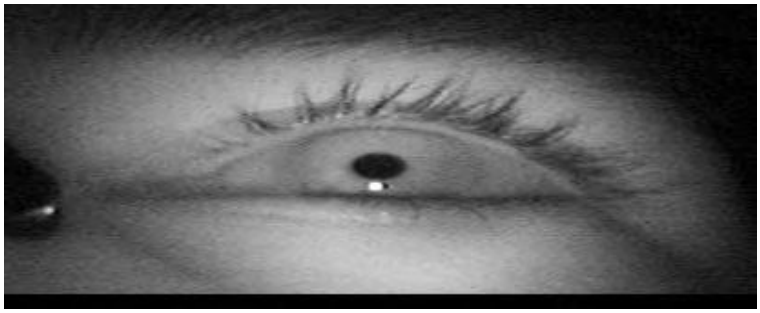


FIGURE 10: HIGH-LEVEL BLOCK DIAGRAM FOR THE REACT EYE-TRACKER IMAGE PROCESSING ALGORITHM.



ORIGINAL

THRESHOLDED IMAGE

FIGURE 11: EXAMPLE OUTPUT OF THRESHOLDING.

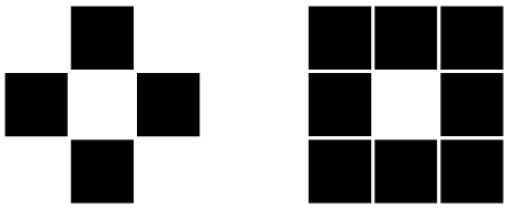


FIGURE 12: 4-CONNECTIVITY (LEFT) AND 8-CONNECTIVITY (RIGHT).

Mathematically put, from the original set of blobs S_b, a new set S_{bf} is formed by discarding any elements B where

$$area(B) < A_{min} \; or \; area(B) > A_{max}$$

S_{bf} is now the set of pupil candidates. If the set is empty, the default threshold value T is adjusted to $T - 10, T - 20, T + 10$ and $T + 20$ and the thresholding and labelling processes are applied again. This adjustment is necessary because the threshold is not chosen adaptively and in some cases may include too much or too little of the pupil.

At this stage, it is possible to take advantage of the pupil morphology. If we were to point a camera lens at a subject who is looking straight ahead, the pupil would appear approximately circular. The exact eyeball morphology is fairly complex and such approximations are necessary to design reduced complexity systems (Villanueva and Cabeza, 2007). In the case of the REACT eye-tracker, the pupil always takes an approximately elliptical shape for two reasons:

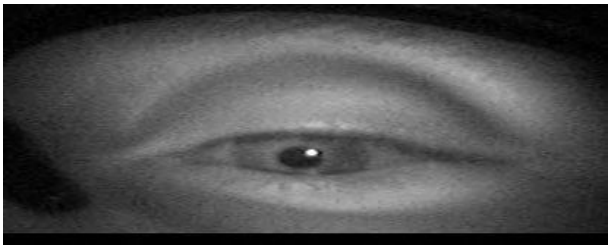
- the camera is pointed to the eyeball from below and is off-centre
- the video is de-interlaced by splitting the odd and even field into separate frames, thus vertically "cutting" the image in half (see Figure 6)

Therefore, by taking into account the elliptical shape of the pupil, the algorithm to select the pupil blob from the set of candidate blobs $S_{bf} = \{B_0, B_1, \dots, B_n\}$ is as follows.

For each blob B_x in S_{bf}, the outmost pixels $outmost(B_x)$ are selected by scanning each line within the blob's bounding rectangle and selecting the first and last pixel that matches the blob's label. The set of pixels $outmost(B_x)_{0..m}$ is then used to fit an ellipse $E(B_x)$ using the algorithm described by Halir and Flusser (1998). For each ellipse $E(B_x)$, a measure of the fit error $e_{E(B_x)}$ is

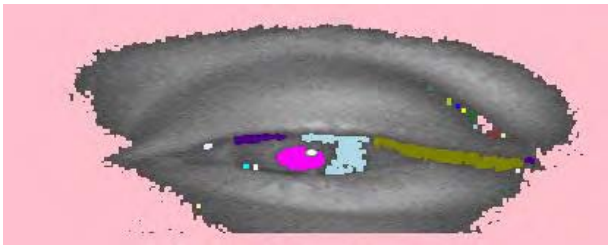
calculated as the average of the Euclidean distance between the detected points to the ellipse contour (see Figure 14). Finally, the pupil contour estimate blob B_{pupil} is found by selecting the blob in set S_{bf} which has the minimum error.

Each step of the above process is visually illustrated in Figure 13.

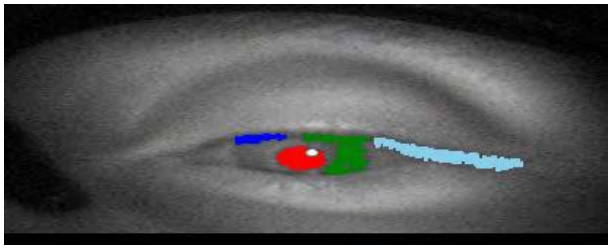


ORIGINAL

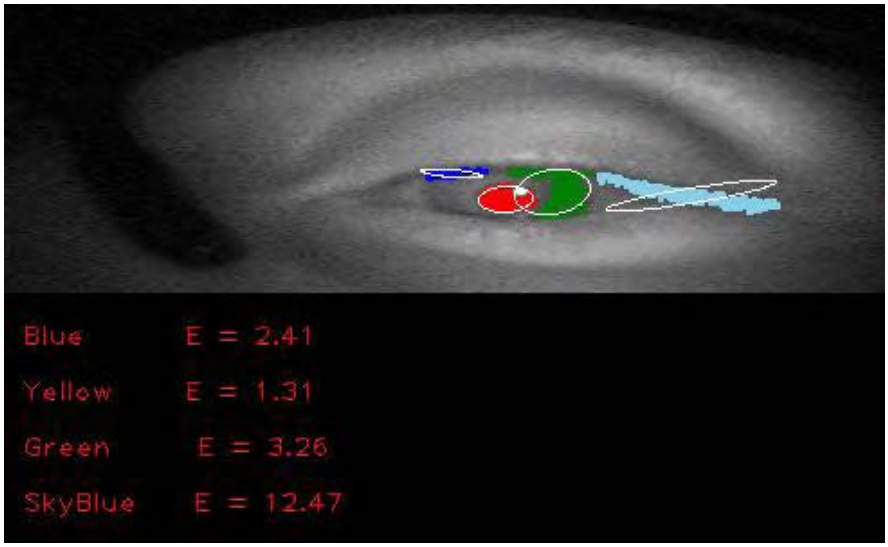
TRESHOLDED IMAGE



LABELLED IMAGE



LABELLED IMAGE AFTER THE CONNECTED COMPONENT SIZE CONSTRAINT HAS BEEN APPLIED



LABELLED IMAGE AFTER AN ELLIPSE HAS BEEN FITTED TO EACH COMPONENT, SHOWING THE ERROR VALUE ASSIGNED (THE YELLOW BLOB BEING THE LOWEST IN ERROR WAS SELECTED AND RE-PAINTED RED TO ILLUSTRATE THE SELECTION).

FIGURE 13: ILLUSTRATION OF THE INTERMEDIATE STEPS OF THE PUPIL CONTOUR ESTIMATION ALGORITHM.

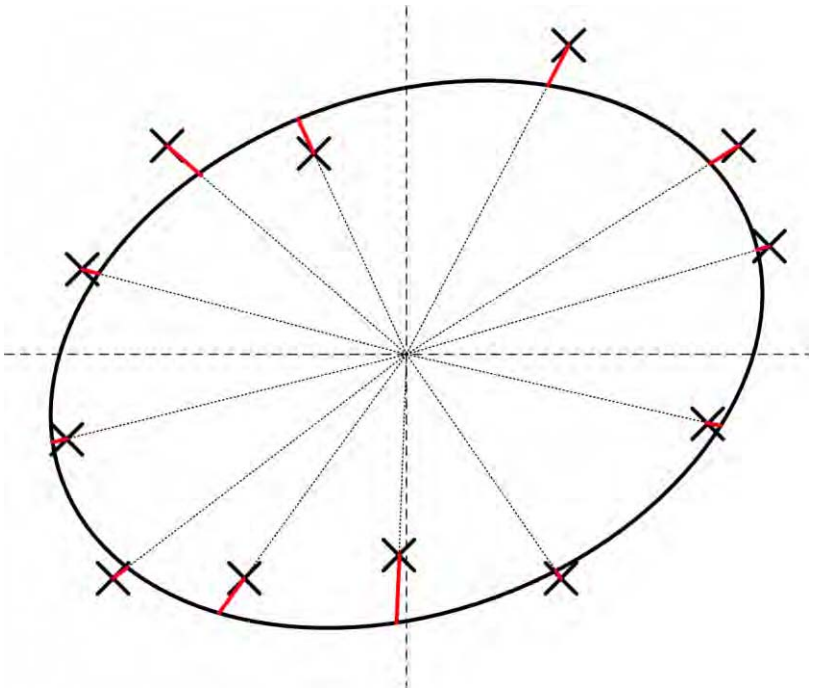
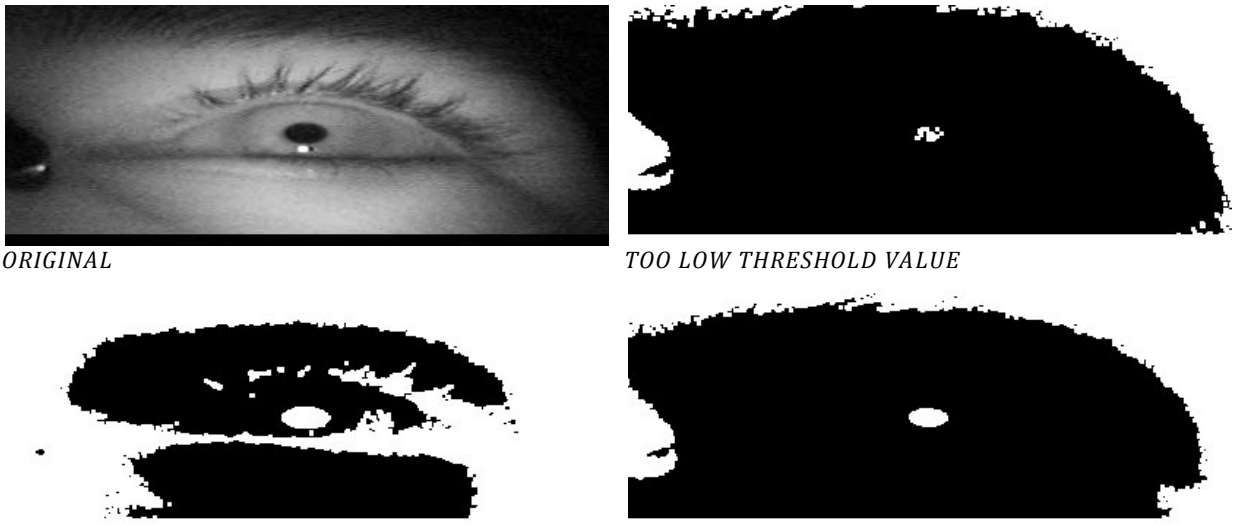


FIGURE 14: ILLUSTRATION OF THE DISTANCES USED TO CALCULATE THE ELLIPSE FIT ERROR ON A HYPOTHETICAL PUPIL ELLIPSE (RED LINES DESIGNATE THE DISTANCES WHOSE AVERAGE IS THE ERROR ESTIMATE).

PUPIL CONTOUR REFINEMENT

After performing an initial estimation of the pupil contour from the thresholded image, it is necessary to further refine it; since global thresholding is used, the results are very much dependent on the threshold value used. Whilst the pupil is *always* dark in the images, just how dark it is will depend on the camera topology (distance from the eyeball and angle of positioning i.e. exactly how the infrared light falls on the pupil), its pupil reflection properties and how they vary between subjects.

Explained simply, a strict threshold value $T = 30$, that is too low, may mark less or even no pixels from the pupil region as foreground. On the other hand, a permissive hypothetical threshold $T = 110$, that is too high, may mark more pixels than those inside the pupil region as foreground. On the other hand, a more balanced threshold $T = 50$ performs much better as shown in the comparison of Figure 15. As mentioned in the above section, different values of T are tried if a match for the pupil is not found; only in this sense the thresholding algorithm is adaptive.



ORIGINAL *TOO LOW THRESHOLD VALUE*

TOO HIGH THRESHOLD VALUE *BALANCED THRESHOLD VALUE*

FIGURE 15: ILLUSTRATION OF THE DELICATE BALANCE REQUIRED BY GLOBAL THRESHOLDING.

An alternative route to global thresholding would have been to use local or adaptive thresholding (Gonzalez and Woods, 2002). Local thresholding is where the thresholding function applied depends on some local property of each pixel. Adaptive thresholding is where the thresholding function applied also depends on the spatial coordinates of each pixel. For example, one possible scheme of adaptive thresholding would be to examine small or medium-size neighbourhoods around each pixel and set the threshold to the mean value, median value or the average of the minimum and maximum value. While these operations are simple compared to other, more complex, schemes (e.g. Chow and Kaneko, 1972 cited by Gonzalez and Woods, 2002), they still increase the computational complexity[6] thus making them less suitable for real-time applications. At 59.97 frames a second, even small additions can make a significant difference in processing speed. The results from the next chapter (evaluation) support that this global thresholding scheme performs sufficiently well for this application despite its simplicity.

Now that the need for contour refinement has been established, more detail can be added as to how this is done in the REACT eye-tracker through the use of active contours.

[6] For example, a scheme that examines the mean value of a $m \times n$ neighborhood, would require $m \times n$ more additions and one division *per pixel*.

An active contour or "snake" is an energy minimizing spline that deforms to fit local minima (Blake and Isard, 1998); since active contours match local minima, their initial position must be explicitly defined. Snakes are versatile and can be adapted to wrap around various types of objects; thus, they are widely used in computer vision problems. Ramadan *et al.* (2002) used a snake to perform pupil tracking; that is, the task of tracking the pupil was performed solely with a modified snake (new pressure model and curvature formulation). In our case, a standard snake is used only to refine the contour obtained from thresholding.

The energy function that the snakes minimize is:

$$E_{snake} = \int E_{internal}\big(v(s)\big) + E_{image}\big(v(s)\big) + E_{constraint}(v(s)) \cdot ds$$

$v(s) = (x(s), y(s))$ is a parametric representation of the snake's position, $E_{internal}$ is the internal energy of the spline due to stretching and bending, E_{image} is a measure of the attraction of image features such as contours and $E_{constraint}$ is a measure of the external constraints imposed either from higher-level shape information or user applied energy. For further details on the theoretical grounding of snakes and how these energies may be generically derived, the interested reader is referred to Blake and Isard (1998).

An adjustable, generic form of the above snake can be modelled as (Williams and Shah, 1992):

$$E_{snake} = \int \big(\alpha(s)\, E_{continuity} + \beta(s)\, E_{curvature} + \gamma(s)\, E_{image}\big)\, ds$$

In this form of the snake function, the first and second terms are first- and second-order continuity constraints and correspond to $E_{internal}$ in the original snake function. The third and last term of the snake function is equivalent to E_{image} and can measure some image quantity such as edge strength or image intensity. Finally, the weights α, β, γ are used to control the relative influence of each term to the snake energy – thus they usually default to $\alpha = \beta = \gamma = 1.0$ and are adjusted depending on the particular application.

The greedy snake algorithm implemented in the REACT eye-tracker is derived from the work by Williams and Shah (1992) and is as follows.

The snake is initialized with the set of outmost points of the selected blob in the previous step, $outmost(B_x)$. When considering new locations for a point of the snake, a 7x7 neighbourhood is considered. Since a very good initial approximation of the spline is given from the thresholding, $E_{continuity}$ is computed such that the snake will not shrink but rather will favour points which have distance near the average distance between consecutive points:

$$E_{continuity} = \bar{d} - \left|p_{candidate} - p_{previous}\right|$$

$\bar{d}$ is the average distance between consecutive points of the snake, $p_{candidate}$ is the candidate point for which $E_{continuity}$ is being calculated, $p_{previous}$ is the previous point in the spline formation and $\left|p_{candidate} - p_{previous}\right|$ is the Euclidean distance between the two.

Curvature is computed at each point using a formula that has been shown (Williams and Shah, 1992) to be computationally efficient as well as favour evenly spaced points:

$$E_{curvature} = |\vec{u}_i - \vec{u}_{i+1}|^2 = \left|p_{previous} - 2 \times p_{current} + p_{next}\right|^2$$

E_{image} is simply calculated from the image gradient (Williams and Shah, 1992).

Before each term is substituted for the final snake energy calculation, they are normalized to [0, 1.0]. Specifically, $E_{continuity}$ and $E_{curvature}$ are both divided by $\max(E_{continuity})$ and $\max(E_{curvature})$ respectively whilst a different normalization function is used for E_{image} that accentuates the differences in gradient magnitude (Williams and Shah, 1992):

$$E_{image}{}' = \frac{\min(E_{image}) - grad(p_{candidate})}{\max(E_{image}) - \min(E_{image})}$$

Note that all *min* and *max* functions mentioned above are calculated within the candidate neighbourhood.

The snake is iterated a predefined number of times and the new spline points define the refined pupil contour $C_{pupil} = \{p_1, \dots, p_n\}$. The final output of this component comprises C_{pupil} and the pupil centre:

$$\bar{C}_{pupil} = average(C_{pupil})$$

IRIS RADIUS CALCULATION

Locating the iris is a problem that has been tackled several times before (e.g. Wang *et al*., 2000; Sirohey *et al*, 2002; Chapter 3). However, a novel solution was required for the REACT eye-tracker because the image formation is different to that of other setups and:

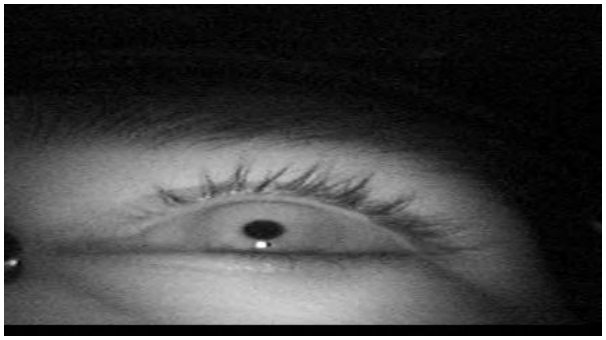
- In many grey scale images that are illuminated using infrared light, the iris-sclera edge prominent in full-colour images is diminished (Figure 16). This can be easily explained if it is considered that the iris is coloured and its colour is reduced to grey-level.
- While the iris-sclera edge is not preserved, infrared illumination will often accentuate the texture of the iris which can create strong edges that complicate the use of edge detectors.

Even though it may have been possible to design the eye-tracker without locating the iris boundaries, doing so provides two significant advantages:

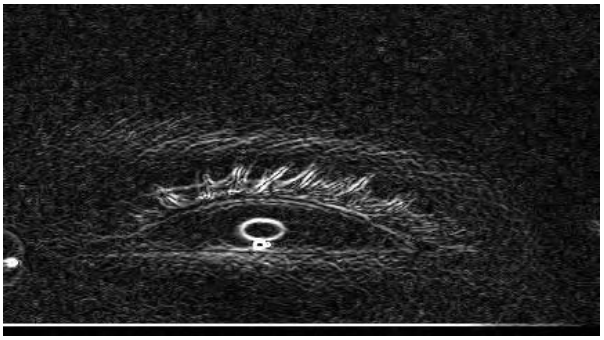
(a) It provides a robust starting point for the challenging task of locating the eye corners.
(b) Most importantly, it allows the eye-tracker to be extended to calculate the 3D gaze provided the camera is fully calibrated. One such approach is found in the remote eye-tracker system developed by Wang *et al*. (2005):
    i. The iris radius and averages taken from anthropological data are used to estimate the radius of the eyeball.
    ii. Assuming a simple eye model where the eyeball is a sphere, an ellipse is fitted to the iris contour (the iris is more suitable for a remote system because it is much bigger than the pupil and because the pupil can hardly be distinguished from the iris without infrared illumination) from which two solutions of the corresponding 3D circle are estimated using techniques outlined by Safaee-Rad, *et al.* (1992 cited by Wang *et al.*, 2005). Additionally, an ellipse is fitted because the pupil and iris appear as a circle only if the person is looking straight ahead and the camera lens is parallel to the eye lens.
    iii. The correct solution is chosen by using a distance constraint based on the position of the eye-corners.

CAMERA-SHOT IMAGE (FULL-COLOUR CONVERTED TO GREYSCALE)

SOBEL OUTPUT OF THE CAMERA-SHOT IMAGE



EXAMPLE IMAGE EXTRACTED FROM THE EYE-TRACKER SAMPLES



SOBEL OUTPUT OF THE EYE-TRACKER IMAGE

FIGURE 16: ILLUSTRATION OF THE EDGE-LOSS IN THE INFRARED IMAGES VERSUS A FULL-COLOUR IMAGE SHOT WITH A CAMERA FROM A DISTANCE. THE SCLERA IS THE WHITE PART OF THE EYEBALL.

A similar task to the detection of the iris boundaries is cell segmentation (e.g. Zhou and Pycock, 1997).

Cells, much like the iris, are fairly uniform in terms of the pixels' intensity levels on the inside. In a similar vein, cell image background is also uniform, like the sclera. Thus, the edge is not necessarily defined by the change in grey-level intensity, which is the basis of most edge-detectors, but rather by a change in the uniformity over a range of pixels.

In the original paper, in images of cells, dark regions are identified to locate the cell interiors. From the centre of each cell, several candidate boundary points are generated in regular intervals through 2π. For each radial, a set of feature measures are calculated, one of which is the edge strength; these features then combined and used to select the final boundary points from the candidates. The calculation of the edge strength is done in the same way here but its

application in identifying the iris boundaries is novel. Further, the modification to the algorithm presented below, specialises the edge strength algorithm for the REACT eye-tracker.

As defined by Zhou and Pycock (1997), for a set of pixels M that are divided into two subsets m and M-m, the edge strength or maximum likelihood ratio mlr is

$$mlr = \frac{\hat{\sigma}_M}{\hat{\sigma}_m \, \hat{\sigma}_{M-m}}$$

where $\hat{\sigma}_M$, $\hat{\sigma}_m$ and $\hat{\sigma}_{M-m}$ is the standard deviation of the grey-level pixel sets M, m and M-m respectively. The edge strength is calculated for several different divisions of M and peaks are observed where an edge is prominent.

In the original edge strength calculation algorithm by Zhou and Pycock (1997) shown above, the whole population is considered; however, it was empirically found that with eye image data, edges attributed to eyelashes and eyelids can severely alter the standard deviation of each population and thus make the algorithm fail or return erroneous (in this context) results. Thus, the original formula was modified to work within a constrained window such that for a line of pixels L, given a constant window size W, the edge strength at an index i within L is equal to:

$$mlr_i = \frac{\hat{\sigma}_{[i-W,i+W]}}{\hat{\sigma}_{[i-W,i]} \, \hat{\sigma}_{[i,i+W]}}$$

W can be chosen from averages taken from training data or adaptively to ½ the pupil radius. Thus, in this case $M = 2 \times W$ and $m = W$. The edge strength is calculated for a number of lines to the left and right of the pupil centre. The complete iris boundary detection algorithm is illustrated in Table 3 using pseudo-code. The selection of populations in the equations above is visually illustrated in Figure 17.

The iris boundary detection algorithm is based on two fundamental assumptions:

a) The subject is looking approximately straight ahead and therefore the vertical position of the pupil approximately coincides with the semi-major axis of the iris ellipse (the iris is closer to a circle but appears as an ellipse because of the camera angle and the de-interlacing, like the pupil).

b) The length between the two iris boundaries is approximately constant for lines $[PupilCenter.Y - Iterations, PupilCenter.Y + Iterations]$. $Iterations$ is the predefined number of consecutive lines examined by the algorithm.

The first assumption simplifies the detection of the iris boundary while the second assumption increases the algorithm's robustness by generating several matches and discarding outliers.

As illustrated in Table 3, the algorithm is fundamentally simple and computationally efficient. A search window of size $\left[SearchLength - SearchOffset - \frac{PupilRectangle.Width}{2}\right], [2 \times Iterations]$ is defined and the edge strength is calculated for each line using the equation given above for a window of size N. The local maxima are extracted as candidates for the iris boundary.

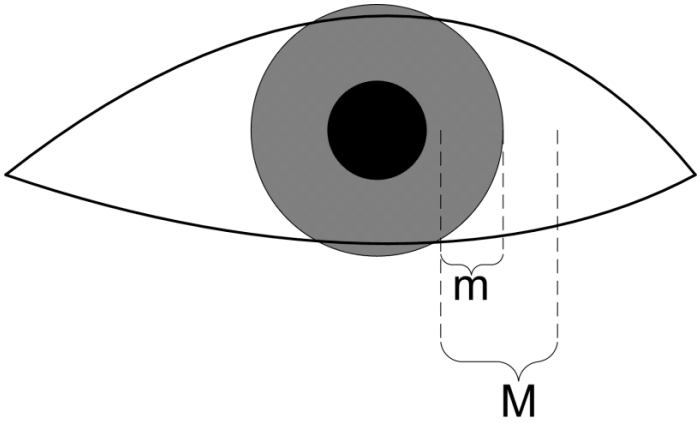


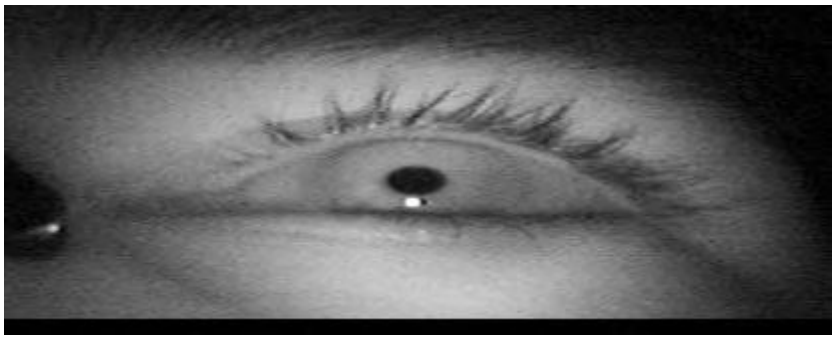
FIGURE 17: EDGE STRENGTH POPULATIONS USED TO DETECT IRIS BOUNDARIES. THE EDGE STRENGTH IS CALCULATED ALONG SEVERAL HORIZONTAL LINES OF FIXED WIDTH ON SEVERAL POINTS WITHIN EACH LINE. AT EACH POINT, FIXED SIZE POPULATIONS ($M = 2 \times W$ AND $m = W$) ARE USED TO CALCULATE THE EDGE STRENGTH.

Outliers, or maxima that lie an abnormal distance from other candidates, are discarded using the algorithm illustrated in Table 4. In essence, the filtering algorithm discards any candidate points whose x-coordinate falls outside a specified confidence interval σ. If the resulting set is empty after the filtering, the interval is progressively reduced using values $\sigma = 2.576$ (99%, initial interval), $\sigma = 2$ (95.5% interval), $\sigma = 1.645$ (90% interval) and $\sigma = 1$ (68.27% interval). Most often, the matches are congregated around the mean value; however, even in cases when the matches found are more spread out, reducing the confidence interval allows the algorithm to complete successfully, at the cost of a slightly less accurate result.

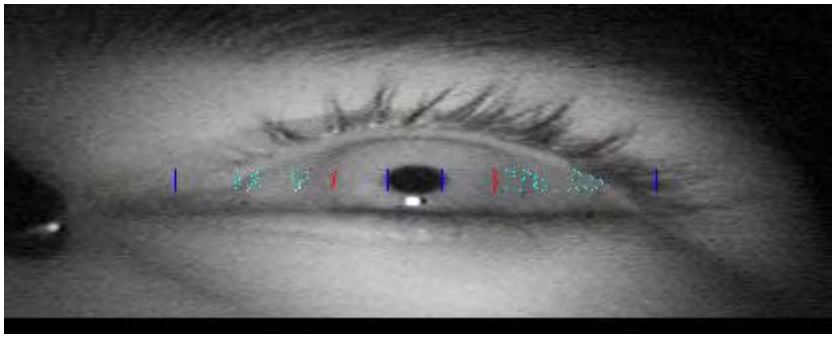
Each step of the iris detection algorithm is visually illustrated in Figure 18.

TABLE 3: PSEUDO-CODE ILLUSTRATION OF THE IRIS BOUNDARY DETECTION ALGORITHM (LEFT BOUNDARY ONLY – THE ALGORITHM IS SIMPLY REVERSED FOR THE RIGHT BOUNDARY).

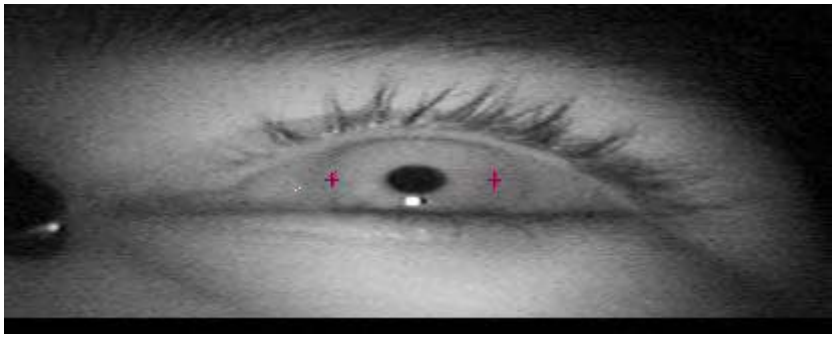
```
Function FindLeftIrisBoundary
const N
const SearchLength
const Iterations

searchOffset = pupilContour.Width

leftSearchBoundary = PupilCenter.X - searchLength
rightSearchBoundary = PupilRectangle.Left - searchOffset

for(y = [PupilCenter.Y - Iterations, PupilCenter.Y + Iterations])
 line = image[leftSearchBoundary ... rightSearchBoundary, y]
 mlr = EdgeStrength(line, mlrWindowSize = N)
 maxima = LocalMaxima(mlr)

 matches.Add(LeftMost(maxima))
end

filteredMatches = FilterOutliers(matches, 2.576 sigma)
if( filteredMatches.Count == 0 )
 filteredMatches = FilterOutliers(matches, 2 sigma)
if( filteredMatches.Count == 0 )
 filteredMatches = FilterOutliers(matches, 1.645 sigma)
if( filteredMatches.Count == 0 )
 filteredMatches = FilterOutliers(matches, 1 sigma)
if (filteredMatches.Count == 0 )
 filteredMatches = matches

leftBoundary = Point(average(X in filteredMatches), PupilCenter.Y)
```



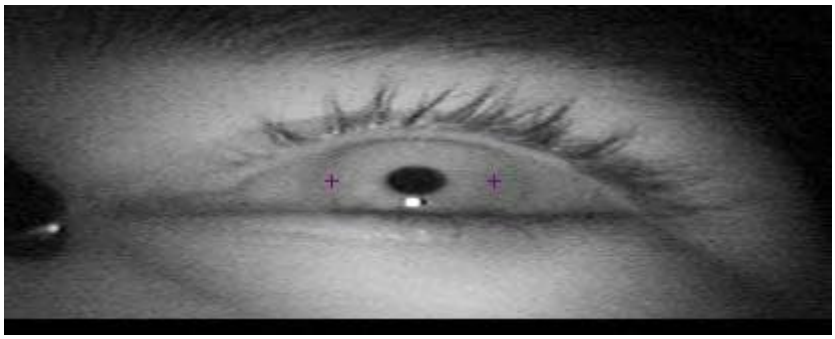
ORIGINAL



THE INITIAL CANDIDATE POINTS GENERATED BY FINDING PEAKS IN EDGE STRENGTH



THE FILTERED CANDIDATE POINTS



FINAL OUTPUT

FIGURE 18: ILLUSTRATION OF THE INTERMEDIATE STEPS OF THE IRIS DETECTION ALGORITHM.

TABLE 4: PSEUDO-CODE ILLUSTRATION FOR THE OUTLIER FILTERING FUNCTION.

```
Function FilterOutliers(matches, Z)

meanX = mean(X in matches)
errorX = Abs(meanX - X in matches)
sigma = stdev(errorX)

if(Z * sigma < 1.0) return matches
filteredMatches = matches where errorX < Z * sigma
```

EYE CORNER DETECTION

Locating the eye corners is probably the most significant challenge for the set of input images taken with the REACT eye-tracker. The problem of locating the eye-corners has been tackled before (Lam and Yan, 1996; Zhang, 1996; Feng and Yuen, 1998; Tian *et al.*, 2000; Sirohey and Rosenfeld, 2001; Sirohey *et al.*, 2002; Wang *et al.*, 2005; Xu *et al.*, 2008) but the systems in question operated, without exception, on a full-face, sometimes colour, image.

In a close-up image, surprising as it may be, the additional level of detail creates several problems making it more difficult to locate the eye corners. With the higher-resolution of an otherwise low-cost camera, more noise is preserved[7] and thus, corner detectors output many false positives. This includes random salt-and-pepper noise as well as structured noise such as shadows caused by the diffusion pattern of the illuminator and eyelashes.

Furthermore, at this level of detail, the inner eye corner does not appear as a corner; as illustrated in Figure 19, the inner corner morphology can greatly vary between people. On the upper left-hand image of Figure 19, the inner corner morphology resembles that of a corner as defined in computer vision, it is approximately symmetrical to the outer corner and the tear gland is hidden. On the contrary, on the upper right-hand image of Figure 19, the inner corner is asymmetrical to the outer corner and the upper eyelid continues to extend all the way to the tear gland. In this case, because of the size of the eyelids, a corner detector would fail to detect the point as a corner and would generate several false positives, as shown in the lower row images of Figure 19.

[7] For the difference of two consecutive frames (before de-interlacing) mean squared error values of ≅ 30 are typical.

Detecting the location of the corners is also a significant task as it provides two static points of reference. In the current configuration, these two reference points are useful in two ways:

(a) To calculate the principal axis by which to calculate the 2D gaze angle (next section). Especially in cases when the camera is rotated around the Y-axis (with Z- pointing upwards), this offers a correction of several degrees which significantly increases the accuracy of the eye-tracker. The aforementioned rotation of the camera can be a result of camera misplacement by the experimenter or slippage of the frame due to the weight of the cables or otherwise.
(b) In long sequences where the absolute position of the eyeball centre is bound to change over time (e.g. frame slippage etc.), the eye corners can be used to detect whether a re-initialization of the eye tracker is required.

Additionally, if the eye-tracker was to be configured to detect the 3D gaze (using a fully calibrated camera), the eye corners are essential to disambiguating the 3D vector solution. For more details, the interested reader is referred to Wang *et al*. (2000).

In order to ease the task of finding the eye corners, it is therefore necessary to remove some detail as well as noise before tackling the problem. A computationally efficient way that reduces image resolution as well as removes noise is Gaussian Pyramid Decomposition (Gonzalez and Woods, 2002). After a Gaussian filter with a 5x5 kernel is convolved over the original image, even-numbered rows and columns are discarded and the output image is a quarter of the size of the original.

Because of the aforementioned differences in morphological structure between the two corners, the corner detection algorithm is specialized for the inner and outer corner separately. Table 5 illustrates the pseudo-code for both versions of the algorithm.

First and foremost, the input image is scaled down to ¼ of its original size using the Gaussian Pyramid Decomposition mentioned above. Then, the partial x- and y-derivative of the scaled image f are calculated:

$$dx(x, y) = |f(x, y) - f(x - 1, y)|$$

$$dy(x, y) = |f(x, y) - f(x, y - 1)|$$

For the y-derivative, non-maxima are suppressed locally using a 1x3 window. Whilst this is an irregular window (usually square windows are used for computer vision operations), it has been empirically found that it preserves the vertical edges better than a 3x3 window. This is most likely because the derivative is a one-column operation too.

As mentioned earlier, slightly different algorithms are used to detect the inner and outer corner due to the different eye morphological structure evident at this image resolution. Both algorithms are however based on the same principle:

1. It is assumed that the edges formed between the eyelids and the sclera are within the top N local maxima for a restricted window ($N = 2$). This assumption was empirically tested.
2. A grouping process begins near the iris boundary previously found and continues outwards, grouping all local maxima that are connected, using an 8-connectivity criterion.
3. The groups are searched for a set of two predefined patterns (shown in Figure 21) and if found, the grouping is terminated at that point. These patterns have been empirically found to occur when the lower eyelid edge is joined with another face line edge and thus the purpose of this step is to separate the two edges.
4. The final corner is selected from the group (outer corner) or pair of groups (inner corner) that demonstrate the maximum derivative energy. The energy of a group of points g is calculated as:

$$e(g) = sum(dy(g))$$

For a pair of groups $g1, g2$:

$$e(g1, g2) = e(g1) * e(g2)$$

The added steps and differences between the two algorithms are summarized here and the complete algorithms are summarised using pseudo-code in Table 5. In Step 1 above, the search window includes both the upper and lower eyelid edges for the inner corner but only the lower eyelid edges for the outer corner. This is done for several reasons:

a) For the inner corner: typically, the lower eyelid edge does not meet the upper eyelid edge. Thus, a distance criterion between the two edges has to be applied to find the corner. Further, often, the lower eyelid edge will be joined to a face line. The pattern

detection offered in the main algorithm will in some cases alleviate this problem but only in combination with the distance constraint is the algorithm robust.

b) For the outer corner: typically, the camera is rotated around the vertical axis towards the outer corner thus making the eyelid-half on the inner corner side appear longer and the eyelid-half on the outer corner side appear shorter. Thus, the upper eyelid on the outer corner side is sloped several degrees more than the inner corner side. For this reason, grouping local maxima points on the top eyelid results in several disjointed groups. In order to robustly find the outer corner, pattern matching is combined with a refinement based on the partial x- derivative of the image. Since the upper eyelid edge and the lower eyelid edge always meet on the outer corner side, a strong maxima is created in the partial x-derivative image. This maxima is used to refine the corner in the last step and is found by searching a 10x5 window.

Figure 22 illustrates the intermediate steps of the algorithm visually.

CLUSTERING OF EYE-CORNER FEATURES

To increase the robustness of the corner detection results which can be sensitive to noise, corner detection is applied over a group of *N* consecutive frames centered around the target frame and the resulting eye corners are calculated by the algorithm shown in Table 6. In essence, this clustering algorithm is an adaptation of the outlier filtering algorithm presented in Table 4 (used to filter outliers of the iris boundary candidates) but modified to take into account the y-coordinate of each point. It was found that compared to no clustering or a weighted-sum clustering scheme, this algorithm performs the best (see evaluation in Chapter 5 for comparison chart). This strategy is only employed for the corner detection for two reasons.

First, while false positives are output by the pupil detection algorithm also, the pupil often changes position even in the time-space of one field (1/2 frame or 1/59.94 sec). Thus, the only suitable strategy to filter out false positives is to employ a distance constraint, optionally with a motion predictor, as proposed in the pupil detection section. Also, the iris radius detection algorithm is accurate enough to not require this. In other words, only the corner detection algorithm is suitable for this kind of filtering.

Second, the most important reason however is how the pupil feature-point is used versus the corners. A false positive of the pupil position at a random point in time would only affect one sample; on the other hand, a false positive of the eye corners during the calibration of the eye-tracker would affect several samples, until the eye-tracker re-initializes itself. With pupil detection, false positives occur rarely and only in extreme positions of the pupil. In contrast, false positive in eye corner detection can occur at any point in time, including the calibration frames thus making it of paramount importance to filter out false positives of the eye corners.

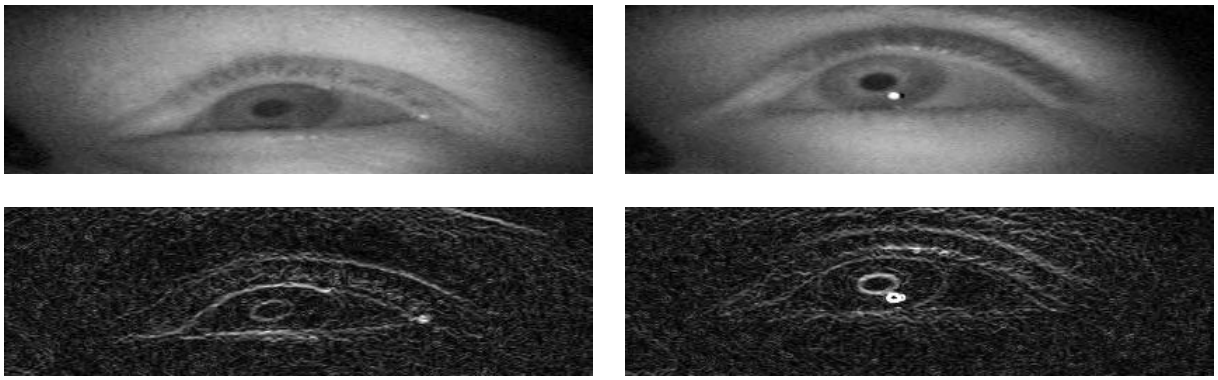


FIGURE 19: INNER EYE CORNER MAY OR MAY NOT APPEAR AS A CORNER. EXAMPLE IMAGES FROM TWO DIFFERENT SUBJECTS (UPPER ROW) AND THE RESPECTIVE OUTPUT FROM A CONVENTIONAL CORNER DETECTOR (LOWER ROW).

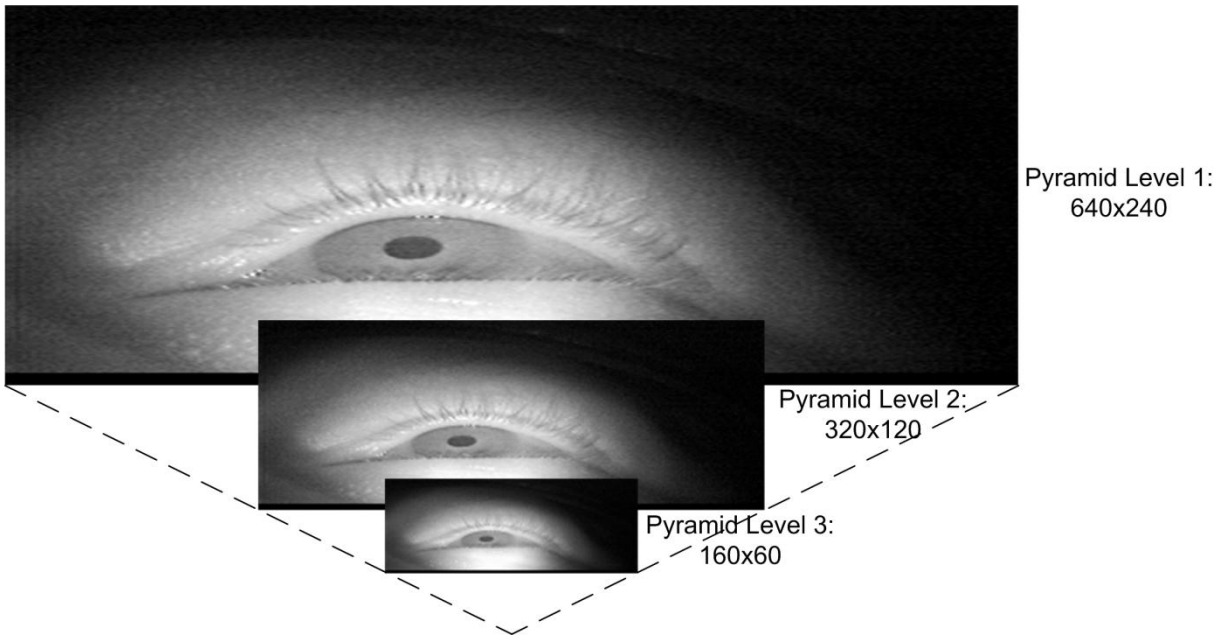
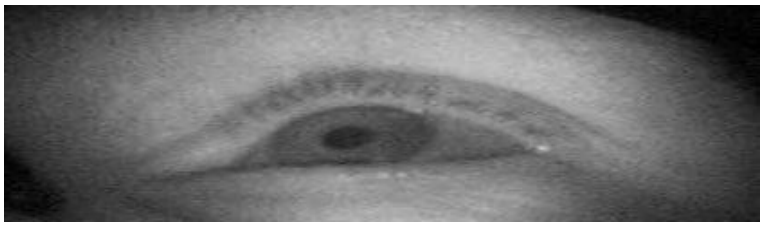
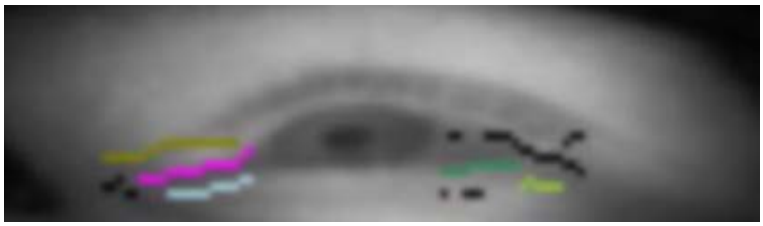


FIGURE 20: EXAMPLE OF GAUSSIAN PYRAMID DECOMPOSITION.

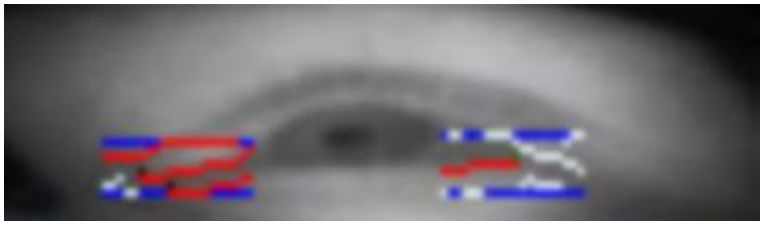
FIGURE 21: PREDEFINED SEARCH PATTERNS FOR THE OUTER CORNER SEARCH. THE PATTERNS ARE REVERSED FOR THE INNER CORNER.



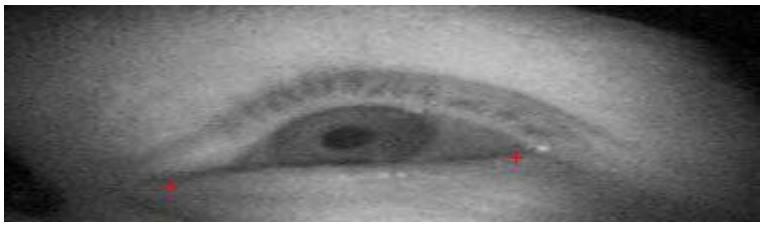
ORIGINAL



GROUPINGS GENERATED BY APPLYING A DISTANCE CRITERION TO LOCAL MAXIMA OF THE Y-DERIVATIVE IMAGE AFTER NON-MAXIMA HAVE BEEN SUPPRESSED (SCALED UP X4)



SELECTED GROUPINGS (SCALED UP X4)



FINAL OUTPUT

FIGURE 22: ILLUSTRATION OF THE INTERMEDIATE-STEPS OF THE CORNER-DETECTION ALGORITHM

TABLE 5: PSEUDO-CODE ILLUSTRATION FOR THE CORNER DETECTION ALGORITHM.

```
Function PreProcess

pyrDownImg = PyrDown(PyrDown(source image))

derivX = partial x- derivative of pyrDownImg
derivY = suppressNonMaxima(partial y- derivative of pyrDownImg)

/////

Function DetectInnerCorner

const SearchOffsetX
const SearchWindowHeight
const PairingDistance
const MinimumGroupSize

searchInitX = InnerIrisBoundary.X - SearchOffsetX
searchTopY = leftIrisBoundary.Y - 0.25 * SearchWindowHeight
searchBottomY = leftIrisBoundary.Y + 0.75 * SearchWindowHeight

allMaxima = list()
while(searchInitX >= 0)
 sample = derivY [x, searchTopY : searchBottomY]
 maxima = FindLargestValues(sample, N = 4, distance > 1)
 allMaxima.Add(maxima)
 x--
end

initialGroups = MaximaToGroups(allMaxima)
groups = initialGroups where n(group) >= averageSize(initialGroups)

for each(group in groups)
 patternIndex = find(pattern1 or pattern2 in group)
 if found
     remove points in group with index <= patternIndex
 end
end

pairs = select pairs of groups
    where (group1 contains point p1 and group2 contains point p2
                such that dist(p1, p2) <= PairingDistance) and
          group1.Size > MinimumGroupSize and
          group2.Size > MinimumGroupSize

candidates = select from pairs where selectFunction:
 pairLeftMostPoint = LeftMostPoint(group1).X,
                  average(LeftMostPoint(group1), LeftMostPoint(group2))

 if pairLeftMostPoint.Y < leftIrisBoundary.Y
     discard
```

```
 end

 if pairLeftMostPoint.X >= leftIrisBoundary - 0.75 * SearchWindowHeight
     discard
 end
end

groupMatch = select from candidates where energy = max:
 energy = sum(derivY along group1) * sum(derivY along group2)
end

corner = LeftMostPoint(groupMatch[group1]).X,
                      average(LeftMostPoint(groupMatch[group1]),
                              LeftMostPoint(groupMatch[group2]))

/////

Function DetectOuterCorner

const SearchOffsetX
const SearchWindowHeight
const searchWindowLength
const minimumGroupSize

searchInitX = OuterIrisBoundary.X + SearchOffsetX
searchTopY = rightIrisBoundary.Y
searchBottomY = rightIrisBoundary.Y + SearchWindowHeight

allMaxima = list()
searchX = searchInitX
while(searchX <= searchInitX + searchWindowLength)
 sample = derivY [x, searchTopY:searchBottomY]
 maxima = FindLargestValues(sample, N = 2, distance > 3)
 allMaxima.Add(maxima)
 x++
end

initialGroups = MaximaToGroups(allMaxima)
groups = initialGroups where n(group) > minimumGroupSize and
                          group.points.Y > searchTopY

groupMatch = select from groups where energy = max:
 energy = sum(derivY along group)
end

patternIndex = find(pattern1 or pattern2, groupMatch)
if found
 cornerCandidate = group[patternIndex]
else
 cornerCandidate = group.last()
end
```

```
derivXBlock = GetBlock(derivX, size = 10x5, center = cornerCandidate)
corner = where derivXBlock = max

/////

Function MaximaToGroups

groups = list()

while(maxima has elements)
 select currentPoint

 search groups where (p in groups) is connected to (currentPoint)
 if found
     add currentPoint to group match
 else
     create new group with currentPoint
 end
end
```

TABLE 6: PSEUDO-CODE ILLUSTRATION OF THE CALCULATION OF THE EYE CORNERS USING A CLUSTER OF RESULTS.

```
Function FilterOutliers2(matches, Z)

mean = mean(matches)
error = dist(mean - matches)
sigma = stdev(error)

if(Z * sigma < 1.0) return matches
filteredMatches = matches where error < Z * sigma
```

CALCULATION OF THE 2D GAZE VECTOR

In this research work, the 2D gaze vector is defined as the vector between the pupil position in the source image when the subject is looking straight ahead and the current pupil position. This is not to be confused with the 3D gaze vector defined as the vector between the subject's eye and the point the subject is looking at in 3D space or a surface such as a screen (e.g. Morimoto and Mimica, 2005).

Calculating a 3D gaze vector would have required a fully calibrated camera (Wang *et al*., 2005). Even though there are modern means of camera calibration that greatly simplify the process (see Bouguet, 2008), it is still too involved to be performed by the user of a system like the REACT eye-tracker. For this reason and given that when investigating non-visual eye-movements 3D gaze offers little or none additional information over 2D gaze, it was decided that 2D gaze calculation would suffice.

As mentioned in an earlier chapter, simplicity was a key requirement that influenced the design and development of the REACT eye-tracker. Thus, a complex calibration procedure was intentionally avoided and the eye-tracker requires only one calibration point – that of the subject looking approximately straight ahead.

This calibration point can be provided on-line (in real-time while recording and tracking at the same time) or off-line (after the video has been recorded) and initializes the tracker by calculating the initial pupil position P_0 and contour C, the iris radius R and the eye corners locations P_{Co} and P_{C1}.

At each point in time, the 2D gaze vector is calculated as follows:

- If the current pupil position P_t falls within the initial pupil contour C, it is assumed that the subject is looking straight ahead.
- Otherwise:
    - A reference x-axis and corresponding y-axis are established using the line that connects the two corners P_{Co} and P_{C1}.
    - The centre of the reference axis's is translated to coincide with the initial pupil position P_0.

- The 2D gaze vector between $P_0(x_0, y_0)$ and $P_t(x_t, y_t)$ is calculated as well as the gaze angle θ:

$$\theta = \tan^{-1} \frac{y_t - y_0}{x_t - x_0}$$

The above process is visually illustrated in Figure 23 and the pseudo-code is presented in Table 7.

When the tracker is set to process long video sequences, it re-calculates the position of the corners when the subject is looking straight, i.e. when the current pupil position is within the initial/calibrated pupil contour. This allows the tracker to adjust to changes of the frame position on the subject's head and semi-permanent changes in the appearance of the eye (such as squinting). If a significant change in the location of the eye corners is detected, the initial pupil position and reference axis centre P_0 is updated to the new value P_n such that $|P_0 - P_{C0}| = |P_n - P_{C0}|$ and $|P_0 - P_{C1}| = |P_n - P_{C1}|$.

The single subject calibration point required by the eye-tracker offers the significant advantage that it can even be acquired without the user's explicit knowledge. For example, the interviewer may incite the subject to look forward and provide an audio signal that will allow the frames in question to be marked for calibration, off-line. On the other hand, it can be a source of error but if necessary, more complex calibration schemes can be incorporated easily, at the cost of additional complexity and perhaps the increase of invasiveness (by making the subject more self-conscious of his/her eyes being tracked). For example, two calibration points C_1 and C_2 may be acquired by asking the subject to look left and right (on the baseline) and classifying each eye-movement as follows:

- On the x-axis (left/middle/right) by the horizontal distance between the pupil position and C_1, C_2.
- On the y-axis (down/middle/up) by the vertical distances between a) the pupil position and C_1 or C_2 and b) the pupil position and the corresponding eye corner.

It is important to mention here that the eye corners have been used as reference points to calculate gaze only in remote systems before (Zhu and Yang, 2002; Valenti *et al.*, 2008). The usual

approach is that of using the glint as the point of reference which introduces the problem of the glint falling into the sclera during extreme eye-movements thus making it very hard to detect.

As explained earlier, the problem of locating the corners is significantly different for the case of a head-mounted setup as the image is taken close-up and is much more detailed which is in fact a disadvantage. For calculating gaze, a model-based approach such as the system by Matsumoto and Zelinsky (2000), which calculates the 3D centre of the eyeball as the middle point between the two corners and then calculates the 3D vector between the eyeball centre and the pupil centre, would not be valid for the system presented in this thesis. As explained in the corner detection section, the appearance of the eye corners significantly is changed versus a remote system and the eye corners no longer appear as symmetrically placed. In other words, the distance between the inner corner and the eyeball centre would not be equal to the distance of the outer corner and the eyeball centre. Thus, a significantly more elaborate 3D eye model would need to be used, in coordination with a more elaborate subject calibration that determines these distances.

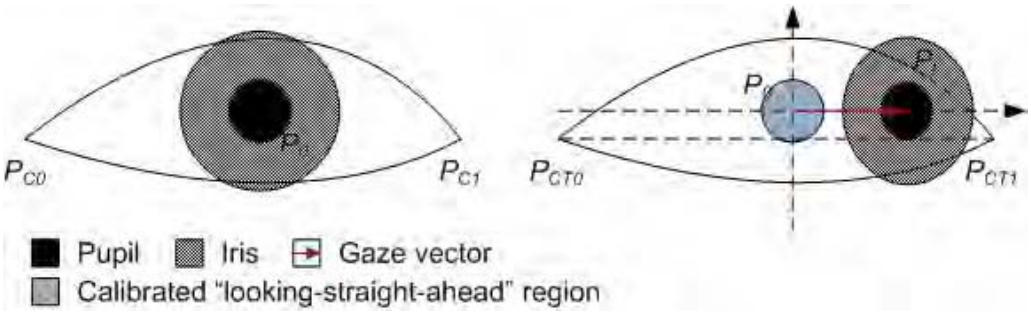


FIGURE 23: CALCULATION OF THE 2D GAZE VECTOR.

TABLE 7: PSEUDO-CODE ILLUSTRATION OF THE 2D GAZE VECTOR CALCULATION.

```
Function CalcGaze(InitPupilPos, InitContour, Corners, PupilPos)

if(InitContour.Contains(PupilPos))
 return SubjectIsLookingStraightAhead

cornerLine = FitLine(Corners.Inner, Corners.Outer)
axisAngle = cornerLine.Theta
gazeVector = PupilPos - InitPupilPos

return theta(gazeVector) + axisAngle
```

COMPUTATIONAL COMPLEXITY

The computational complexity of the algorithms presented in this thesis is often mentioned and this section aims to specifically discuss the basic time complexity of each component and their sum.

The pupil detection is involves estimating the pupil contour (thresholding, connected components labelling and blob ellipse fitting) and refining it through the use of a snake. For N number of pixels, thresholding requires one comparison and one assignment per pixel, thus being of $O(N)$complexity. During connected components labelling, the image is scanned twice and it thus requires approximately $2 \times N$ operations, being also of $O(N)$ complexity. The time taken by the ellipse fitting that takes place after labelling does not depend on the number of pixels but on the number of labels or blobs which is small compared to the number of pixels and thus its complexity is approximately $O(1)$. Refining the contour using a snake also does not depend on the number of pixels but on the number of the points in the contour; since this number is always very small compared to the number of pixels and the snake is iterated a constant amount of times, it is also of $O(1)$ complexity. Overall, detecting the pupil is of $O(N)$ complexity.

When calibrating the eye-tracker, other than the pupil, both the iris and eye corners are detected.

Calculating the iris radius also requires a very small amount of operations. The edge strength is calculated over a fixed number of lines L and across a fixed width W for each line; its calculation is equivalent to two multiplications and three calculations of standard deviation (approximately $3 \times m$ number of additions and m number of multiplications for a sample of size m) of a small number of pixels compared to the total number of pixels N and therefore the total complexity can be approximated by $O(1)$. Filtering the outliers requires an approximately constant time and therefore is of complexity $O(1)$.

When detecting the eye corners, the most significant operation is the Gaussian Pyramid Decomposition (convolution of 5x5 Gaussian kernel and subsampling) whose complexity is $O(25 \times N) = O(N)$. Relatively to this, the maxima selection and grouping that follows is of constant complexity and thus the total complexity of the corner detection is $O(N)$.

Thus, for calibration frames where both iris detection and corner detection takes place, the complexity is $O(N) + O(1) + O(N) = O(N)$.

TABLE 8: THE CALIBRATION FRAMES FOR SUBJECTS 1-9 WITH THE FEATURE POINTS ILLUSTRATED.

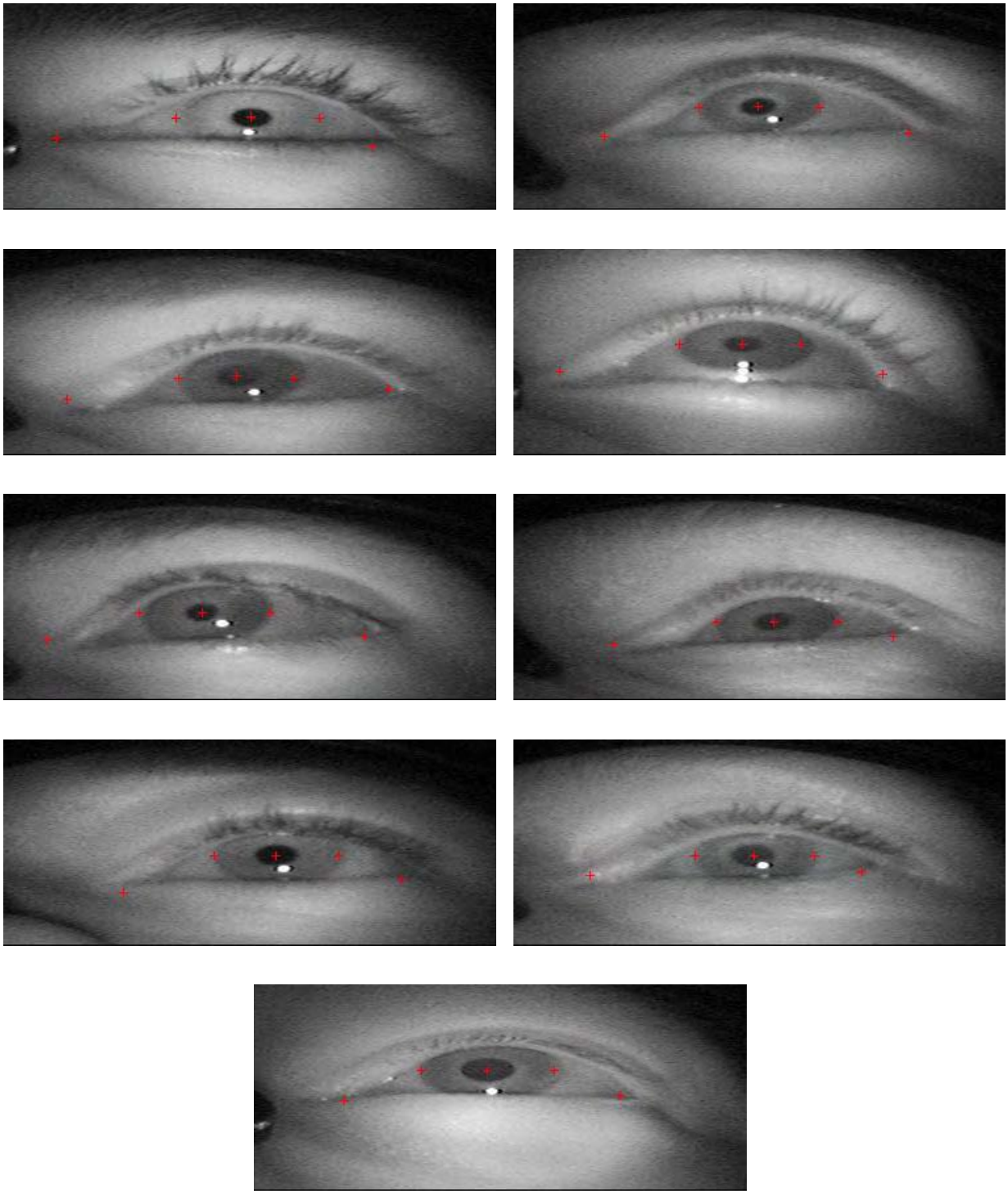
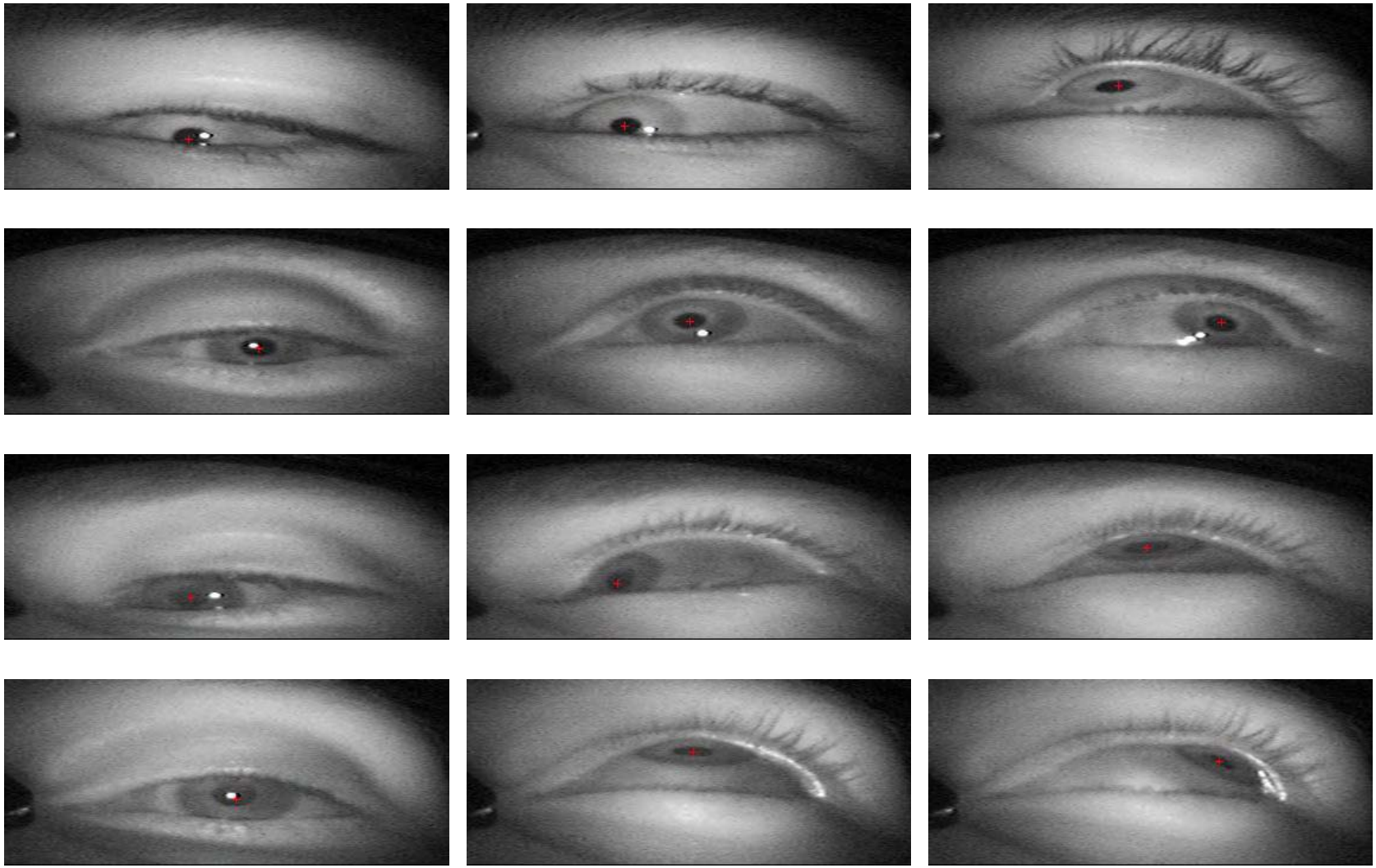
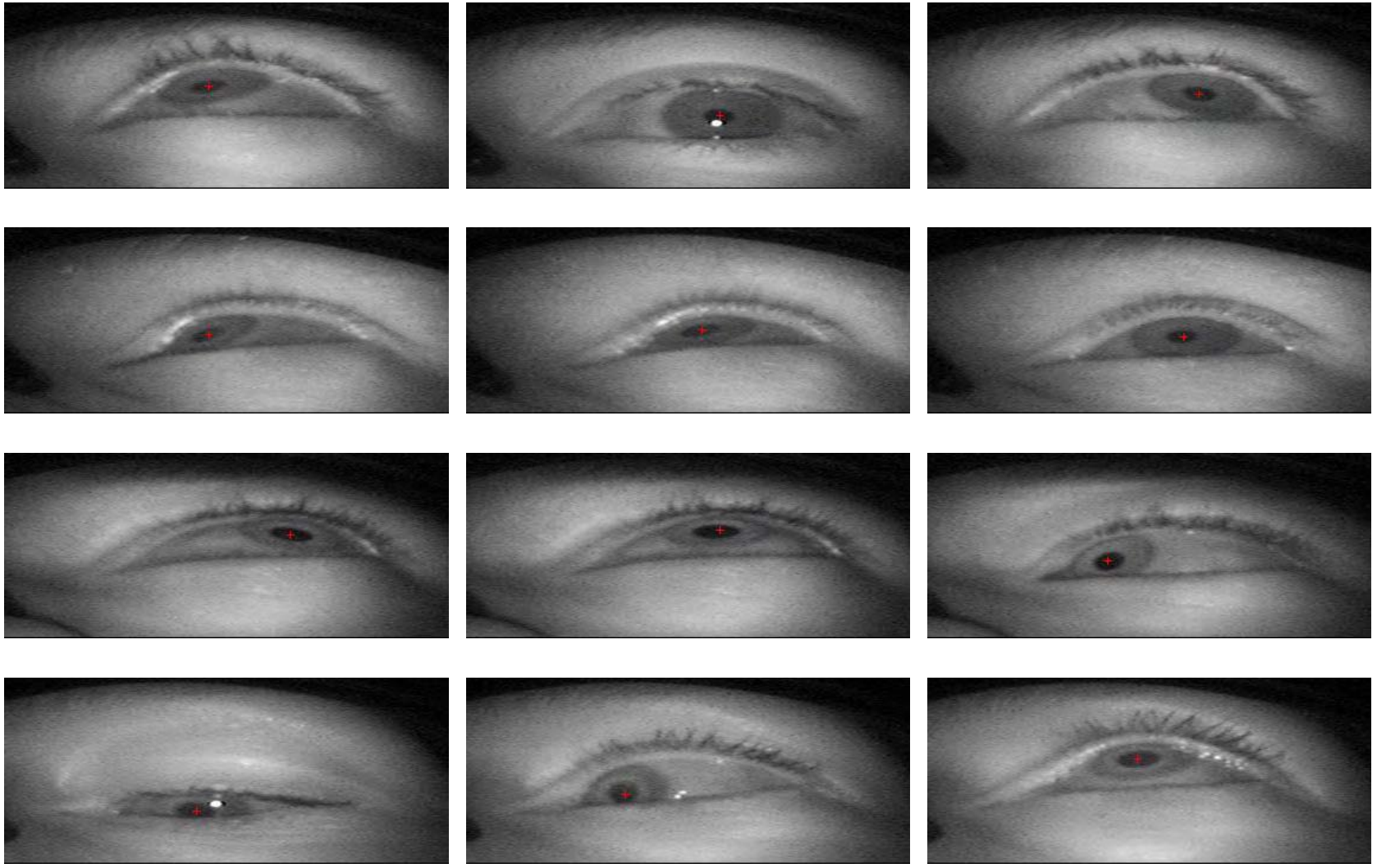
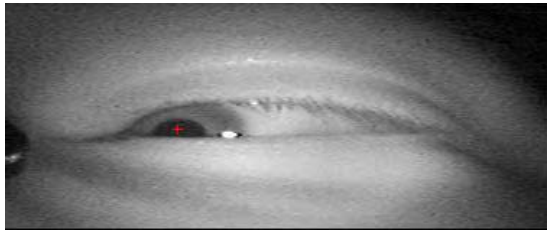
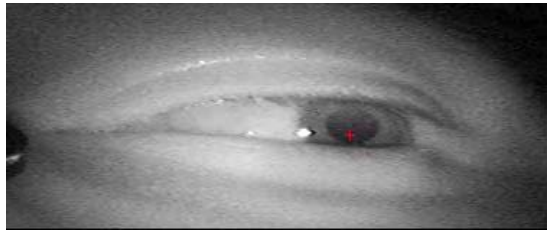
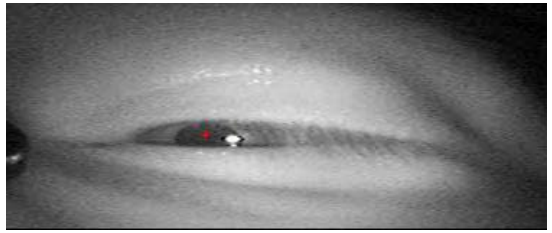


TABLE 9: RANDOMLY SELECTED FRAMES FOR SUBJECTS 1-9 WITH THE PUPIL MARKED BY THE EYE-TRACKER (ONE SUBJECT PER ROW).





CHAPTER 5: FEATURE EXTRACTION EVALUATION

Given that the REACT eye-tracker is feature-based, it makes sense to evaluate its performance in extracting these features by calculating the Euclidean distance between each feature point as extracted by the eye-tracker and the feature point as manually marked by the author.

Thus, on each intermediate step of the 2D gaze calculation (detecting the pupil, calculating the iris radius and locating the corners), the appropriate set of frames was selected from the test video database and the errors were measured. To render that possible, a software application that allows for relatively easy manual marking of feature points on frames was written and used. The set of manually marked frames will also be referred to as the *validation data set*.

It is desirable to assess the performance of each component separately and thus, for the iris radius and corners extraction algorithms that depend on previous outputs (pupil location and pupil location, iris radius respectively), they were taken from the validation data set such that there is no interference from errors from other components. The validation test data set was composed of nine (9) different subjects, of which two subjects were female and seven male, one subject was Black, one Asian and the rest seven Caucasian. Finally, two subjects were in the 35-25 group and the other seven subjects were in the 25-35 age group.

Beyond assessing each individual component separately, it is also desirable to assess the performance of the eye-tracker as a whole. Thus, the 2D gaze angle calculation algorithm was evaluated by comparing the 2D gaze angle calculated using inputs (pupil position, iris radius, corners locations) as calculated by the eye-tracker versus using inputs from the validated data set.

Finally, in order to assess the usability performance of the eye-tracker as hardware, a simple questionnaire was designed and distributed to the subjects that have used the eye-tracker.

The evaluation of the REACT eye-tracker can thus be split into the following parts:

1. Evaluation of the pupil detection algorithm
2. Evaluation of the iris boundary detection algorithm
3. Evaluation of the corner detection algorithm

4. Evaluation of the 2D gaze vector calculation algorithm
5. Evaluation of the eye-tracker hardware usability

TESTBED AND SAMPLE VIDEO COLLECTION

Collecting sample videos was an important process in order to develop, test and evaluate the REACT eye-tracker. In order to do this, a software application was designed that projects points with known locations on a screen. By placing the subject at a fixed distance from the screen and aligning the centre of the projected points and the subject's direct point of gaze, it is possible to consistently generate extreme eye-movements that cover regular intervals of the complete 360° view.

The aforementioned test bed was implemented with a NEC MT1065 projector which is configured with a special mirror to project images from the back of an 84" screen.

The latter configuration was essential to collecting video samples successfully from this test bed:

- The subject needs to be located fairly close to the screen such that it is possible to generate "extreme" eye-movements outside of the normal field of view, as a simulation of eye-movements that occur during thinking. For the same reason, running the test on a regular 17" desktop screen would not suffice as the generated eye-movements would be restricted in a narrow field of view and the screen is required to be large.
- Using a regular ceiling-mounted projector configuration, it is impossible to display the screen correctly at the same time as recording the subject's eye-movements as part projector's beam would be occluded by the subject's head and shoulder. Thus, projecting to the screen from the back was absolutely vital.

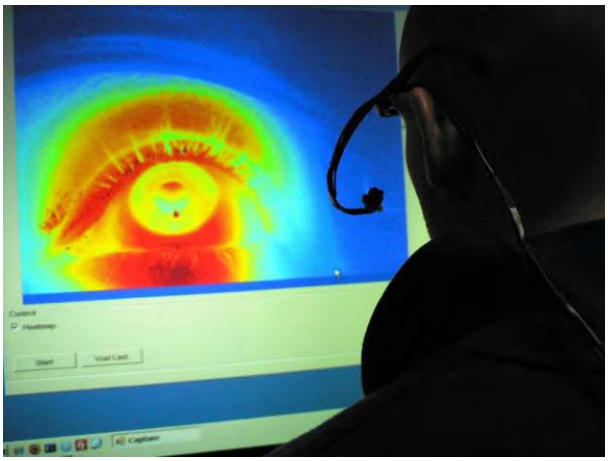


FIGURE 24: SUBJECT STANDING IN FRONT OF THE PROJECTOR BEFORE RUNNING A RECORDING SESSION.

The test bed software was designed to display a set of twelve points from an imaginary circle that fills the screen as shown in Figure 24. If the centre circles coincides with the centre of a Cartesian plane, then the points start at 0° and continue at 30° increments to complete a full circle. The display order of the points is randomized to simulate a more realistic testing environment for the eye-tracker and prevent the subjects moving their eyes in anticipation of the next eye-movement. At the same time as displaying the points on screen, the video input from the eye-tracker is being recorded to disk. The synchronization is as follows:

- *Subject calibration phase*. All twelve points and the centre of the circle are displayed all at once. The researcher asks subjects to indicate where the centre of the circle ought to be such that it is the focus of their direct gaze. Then, subjects are asked to make sure they can look at each and every point without moving their head or straining their eyes. If necessary, the circle radius is adjusted to allow subjects to keep their head still and comfortably view all the points throughout the recording. Video recording is off during this phase. When subject are ready, they are asked to press a key to move on to the next phase.
- *Tracker calibration phase*. Once subject calibration is completed, the centre of the circle is displayed once again and subjects are asked to press a key when their eyes are focused on the centre point. This is done such that the first recorded eye-movement is that of the person looking straight ahead and the tracker may be calibrated. At the beginning of this phase, video recording is initiated. On key press, the next phase is initiated.

- *Circle phase*. The twelve points are displayed one by one; subjects are asked to focus on each point and press a key when ready. When the key press occurs, the next point is displayed. When all points have been displayed, the key press initiates the last phase.
- *Final phase*. This phase is identical to the *Tracker calibration phase* and was added in case consistency checks between the two pupil positions (original and repeated) were later required. At the end of this phase, video recording is turned off and the software exits.

At the same time as recording the video from the eye-tracker, the coordinates of the point currently displayed on screen are also recorded. Keystrokes may be detected by the change of displayed point. Figure 25 illustrates the recording software in operation.

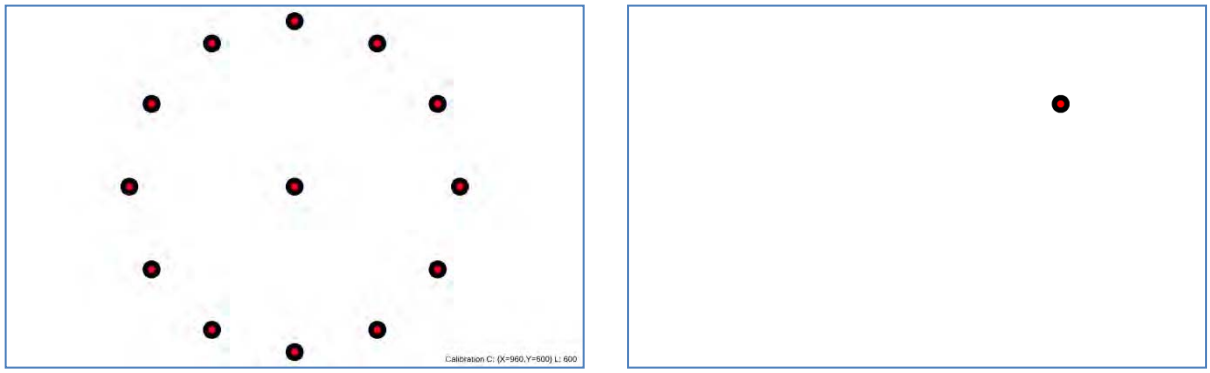


FIGURE 25: THE RECORDING SOFTWARE IN OPERATION; CALIBRATION (LEFT) AND RECORDING (RIGHT).

EVALUATION OF THE PUPIL DETECTION ALGORITHM

In order to evaluate the robustness and the accuracy of the pupil detection algorithm, the pupil was manually marked in a set of nine (9) sample videos from nine (9) different subjects, giving a total of 12,334 frames. The pupil was marked on each frame by manually drawing an adjustable transparent ellipse over the pupil and then automatically calculating the centre of the drawn ellipse as the pupil centre, as illustrated in Figure 26.

The tracker was set to process all the frames in the above data set and the error was quantified as the Euclidean distance between the calculated pupil position and the manually marked position. The errors were then classified to *negligible*, *acceptable* and *unacceptable* to aid the interpretation of the results. As negligible were considered any errors less than or equal to 2

pixels because it is estimated that errors up to 2 pixels may be a result of the manual marking; despite the use of manual software that eases the marking task, it is impossible to mark the data set 100% accurately as a) in some cases the exact pupil boundaries cannot be distinguished and b) the boundaries are often not bound to one single pixel. As acceptable were considered any errors between 2 and 8 pixels and as unacceptable any errors over 8 pixels.

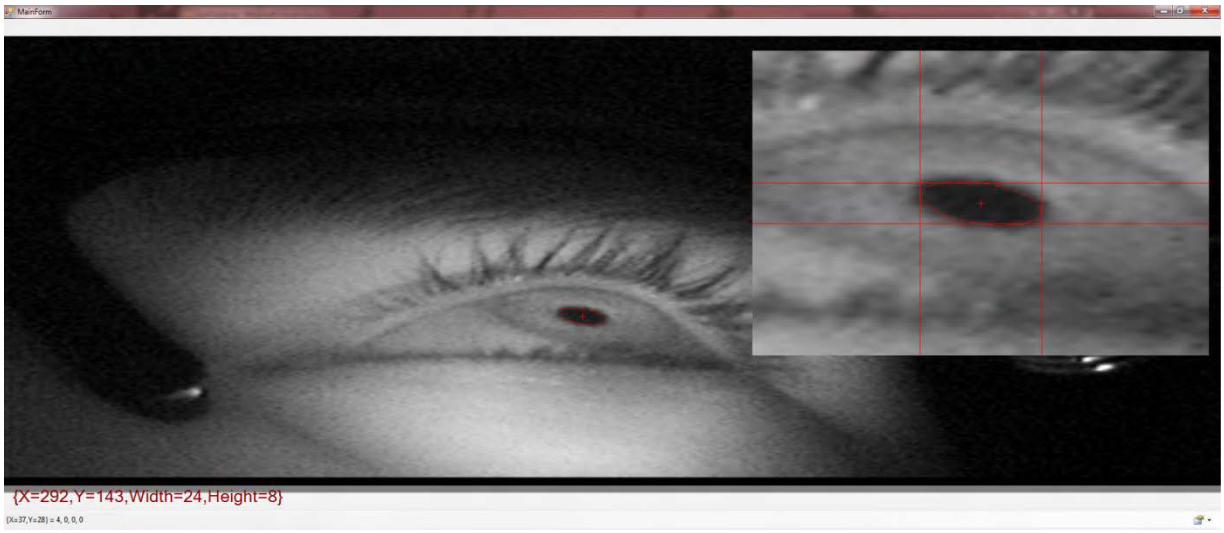


FIGURE 26: MANUAL MARKING OF THE PUPIL CONTOUR AND LOCATION IN THE VALIDATION VIDEOS.

Table 10 and Table 11 below display the error and error classification statistics. Specifically, in Table 10, the first column designates the test sample, the second column displays the processed number of frames for the particular sample, the third and fourth column the average error and standard deviation for the sample respectively and the fifth column displays the maximum error across the frames of the particular sample. Finally, the last row of Table 10 shows the average error, standard deviation and maximum error for the whole data set taken together. In Table 11, a similar format is used; the sample is shown on the first column, the total number of frames on the second column, followed by the number of frames of each class and the number of frames that failed to process. The final row of Table 11 shows the total number of frames for each class, both as a cardinal number and a percentage.

As can be seen from the tables above, the pupil detection algorithm performs quite well despite using a global threshold; on average, the error is 2.04 ± 3.32 pixels. The good performance of the algorithm can be attributed to the use of infrared lighting and the accompanied dark-pupil effect.

Poor performance may only be observed in sample Subj09 where the error for 326 of the 1361 total frames has been classified as unacceptable. This was due to poor placement of the camera; specifically, the camera was erroneously placed almost directly in front of the eye. Not only does that obscure the subject's vision more than placing it pointing upwards but it creates a reflection similar to that of the bright-pupil effect introduced in Chapter 3 (and similar to the red-eye effect seen in photographs) which causes the thresholding and snake to perform poorly, as shown in Figure 27.

The maximum error is quite large (over 20 pixels) for test sequences except Subj02 and Subj08; this is simply due to a failure of the pupil detection algorithm. In rare cases, the fit error calculated for each ellipse favours an image blob that is not the pupil, as shown in Figure 28. However, as shown in Table 12 which illustrates the errors for each test sequence as bar graphs, this is extremely rare (in the order of 2-3 frames per sequence, for most sequences). Additionally, such high errors rarely occur for two consecutive frames. Thus, such errors could easily be filtered when tracking over time by defining a maximum pupil movement between frames and discarding frames that exceed this value or alternatively by using a polynomial model to predict the current pupil position based on the movement from the last few frames and discarding frames for which the tracked position does not fit the model. As explain in Chapter 4 however, this is not as important for the pupil as it is for the eye corners.

Finally, where the eye-tracker has failed to detect the pupil (11 of 12,334 frames or 0.09%), it means that after filtering by size, no connected components were found in the image. If this effect were of any significant occurrence, it could be remedied by lowering the minimum blob size threshold if no components were found with the default setting.

TABLE 10: PUPIL EVALUATION - ERROR STATISTICS, IN PIXELS.

Sample	Frames	Average	σ	Maximum
Subj01	2088	1.06	1.76	71.22
Subj02	1238	1.19	0.88	5.80
Subj03	1372	2.42	3.24	87.91
Subj04	927	2.32	1.97	23.53
Subj05	1259	1.83	3.28	75.70
Subj06	1308	1.94	3.67	73.09
Subj07	1383	1.38	4.23	109.68
Subj08	1398	1.66	1.21	7.94
Subj09	1361	5.15	5.09	38.34
Overall	**12334**	**2.04**	**3.32**	**109.68**

TABLE 11: PUPIL EVALUATION - ERROR CLASSIFICATION STATISTICS, NUMBER OF FRAMES.

Sample	Frames	Negligible	Acceptable	Unacceptable	Failed
Subj01	2088	1861	225	2	0
Subj02	1238	1039	199	0	0
Subj03	1372	771	557	44	0
Subj04	927	493	425	9	1
Subj05	1259	927	312	20	10
Subj06	1308	910	376	22	0
Subj07	1383	1162	217	4	0
Subj08	1398	993	405	0	0
Subj09	1361	509	526	326	0
Total	**12334 (100%)**	**8665 (70.25%)**	**3242 (26.28%)**	**427 (3.46%)**	**11 (0.09%)**

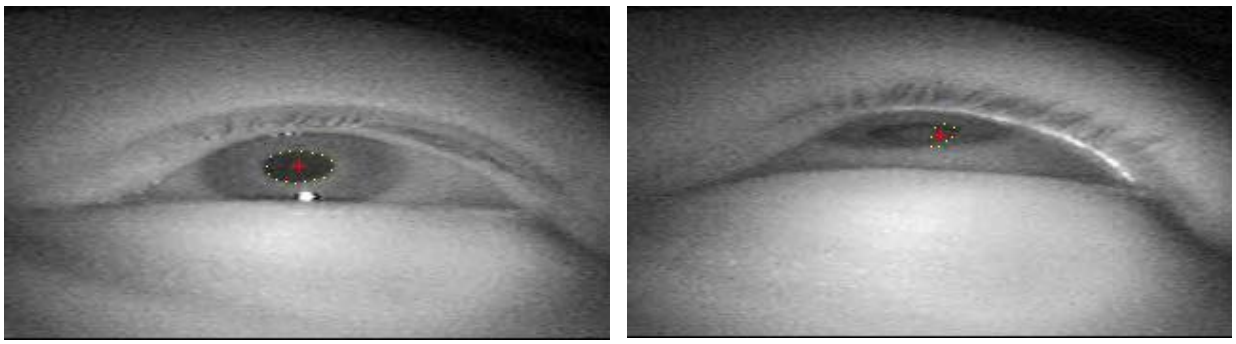


FIGURE 27: ILLUSTRATION OF POOR CAMERA PLACEMENT CAUSING PUPIL DETECTION TO GIVE HIGH ERRORS. SUCCESSFUL INITIALISATION IS SHOWN ON THE LEFT WHILE PARTIAL BRIGHT PUPIL EFFECT IS SHOWN ON THE RIGHT (SAMPLE: SUBJ09).

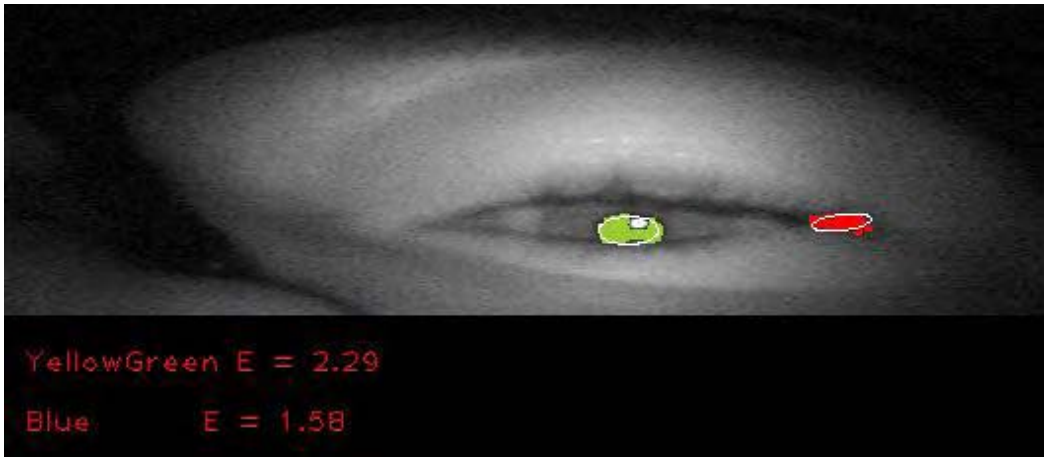
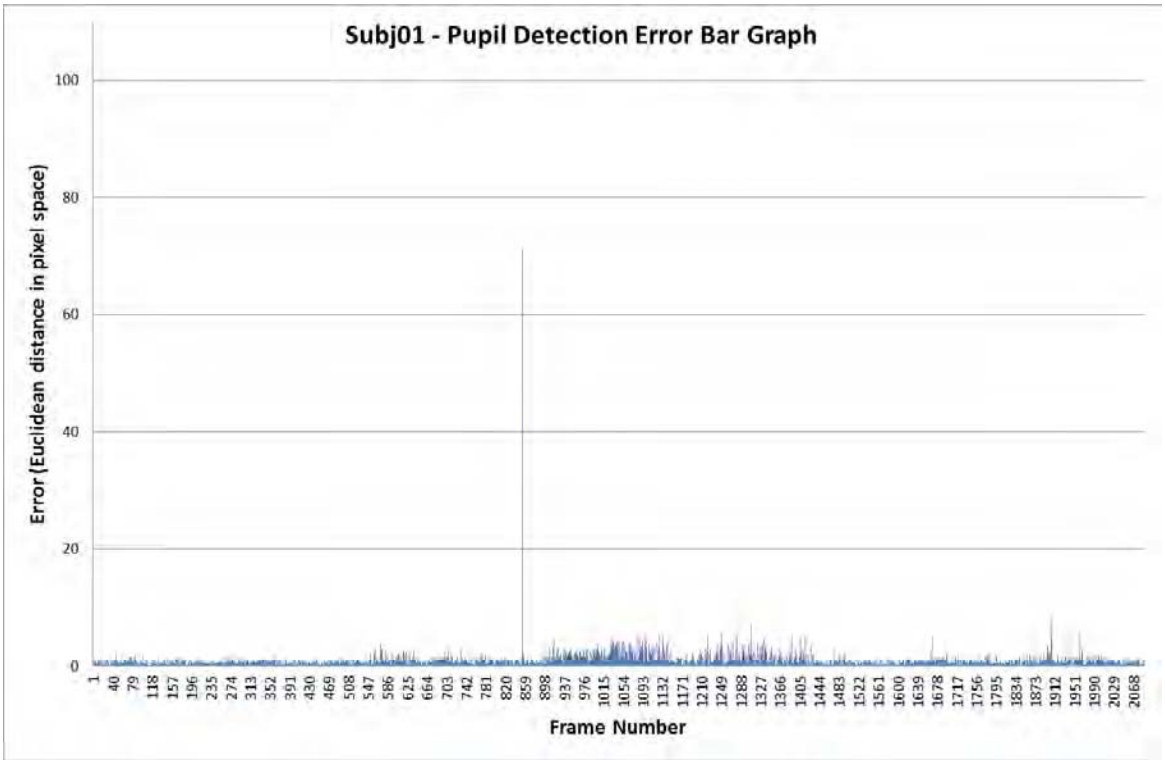
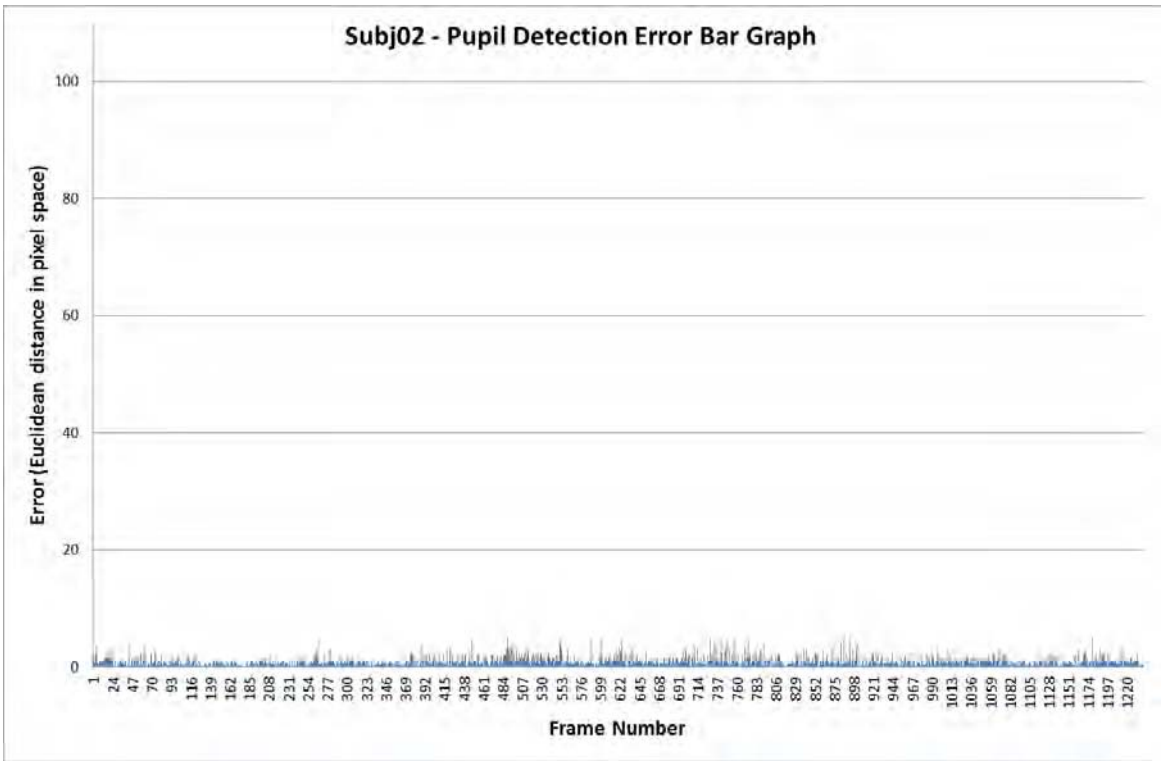
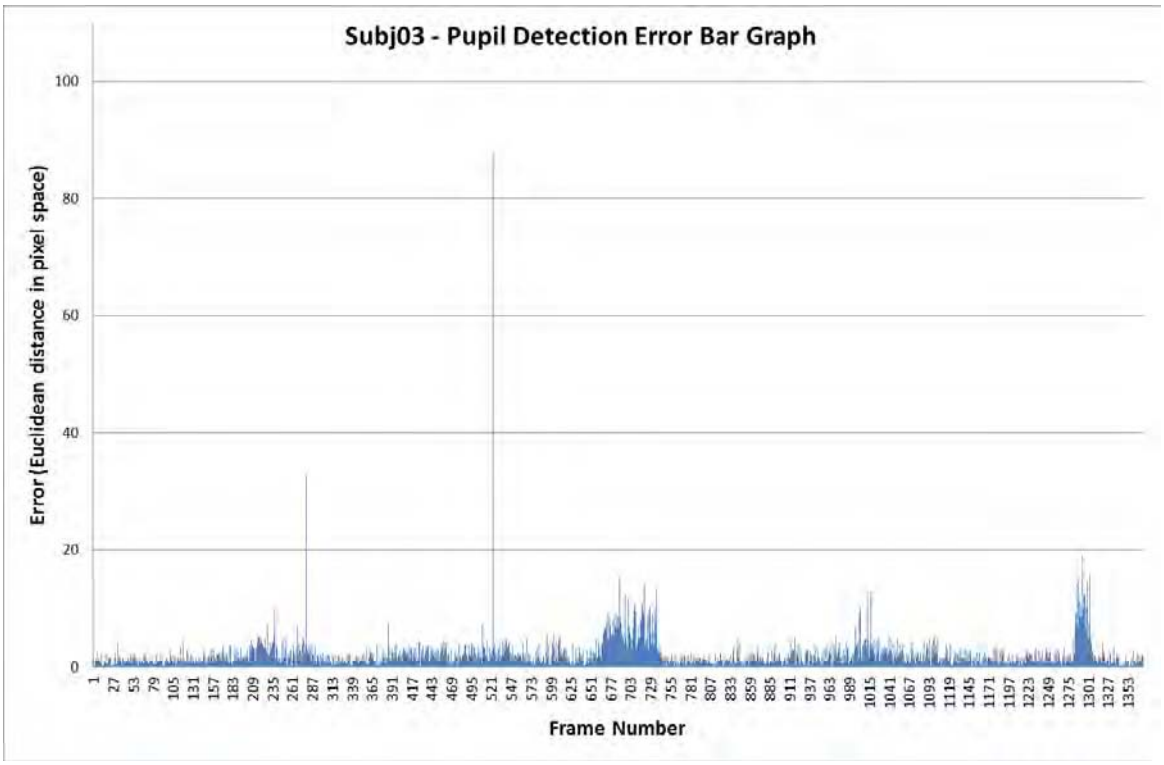


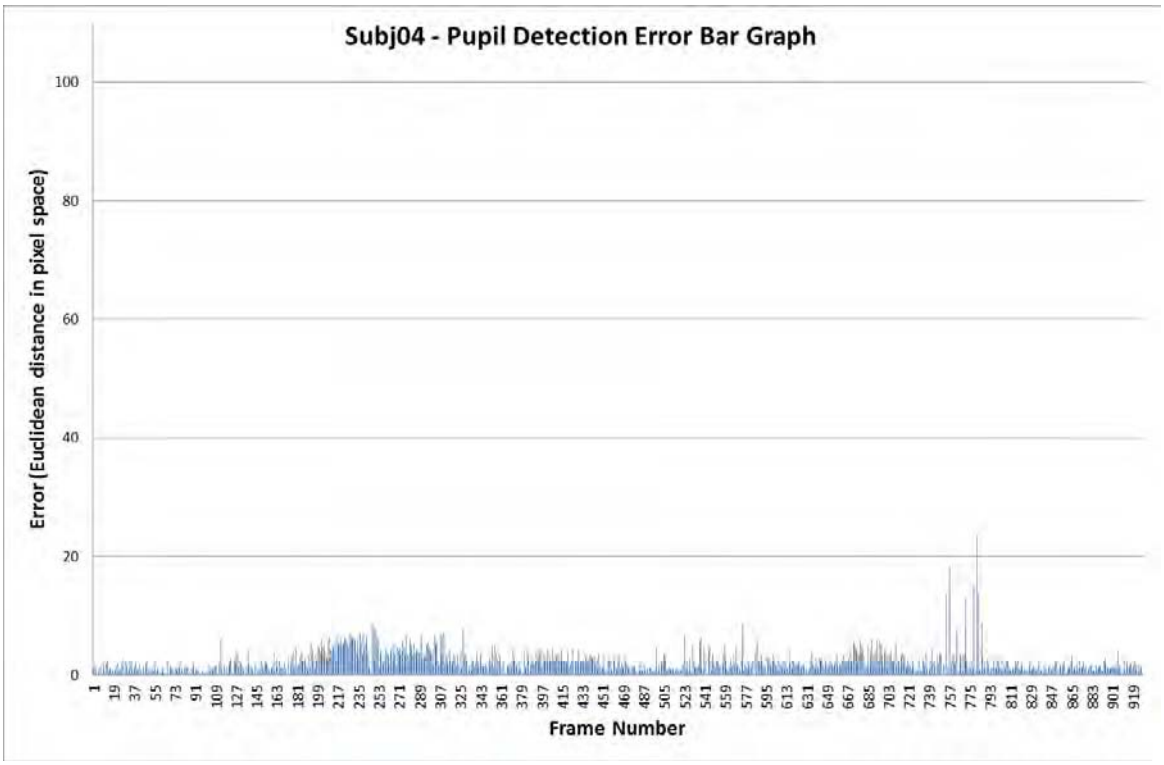
FIGURE 28: EXAMPLE OF A FAILED SELECTION OF THE PUPIL DETECTION ALGORITHM.

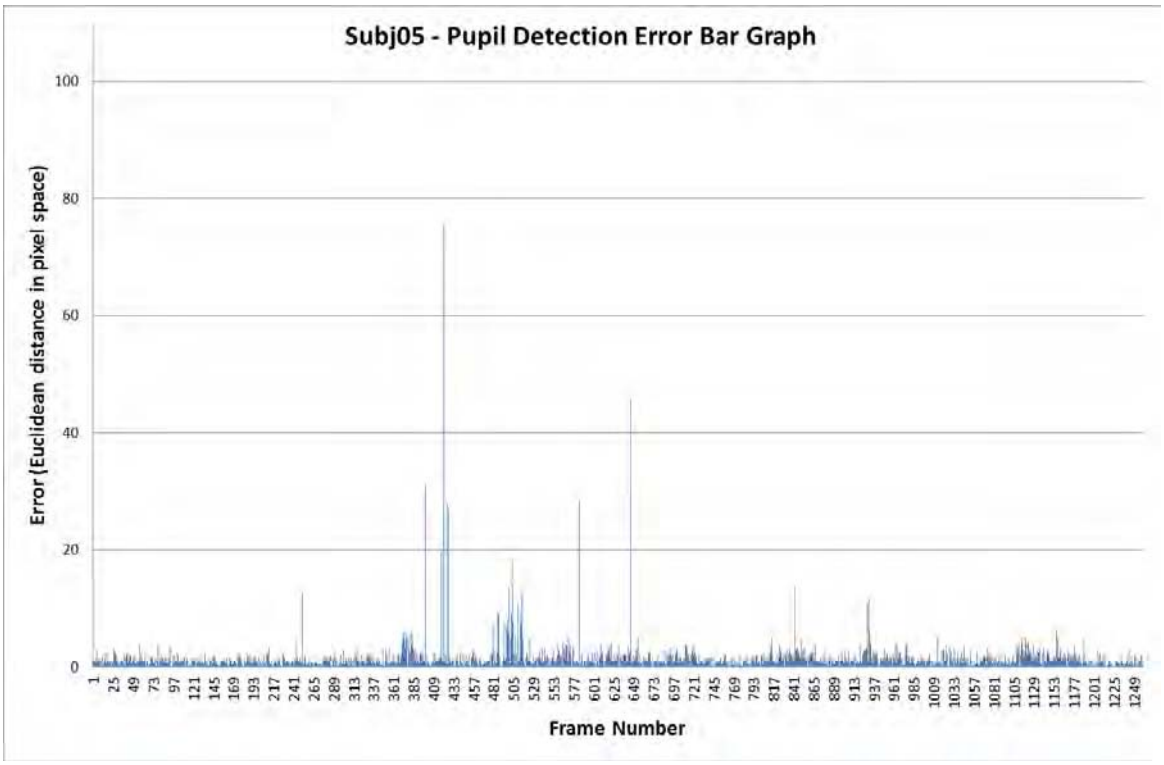
TABLE 12: BAR GRAPHS OF THE PUPIL DETECTION ALGORITHM ERROR FOR EACH TEST SEQUENCE.

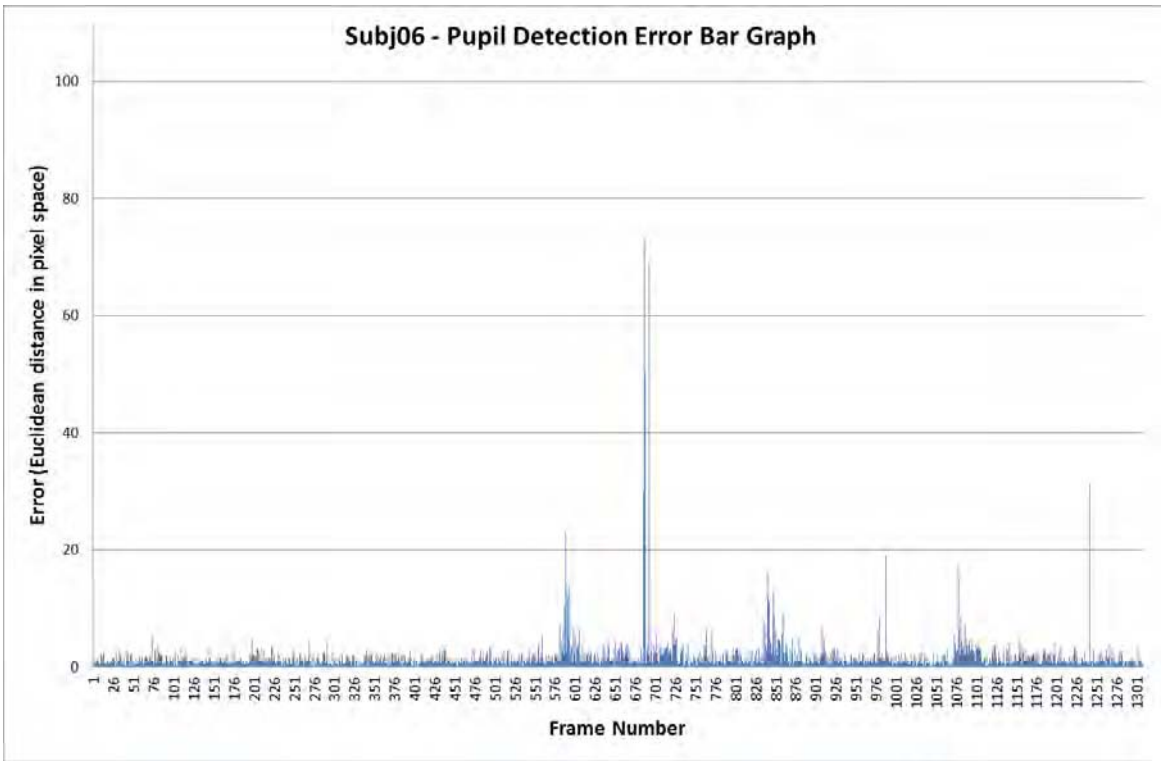


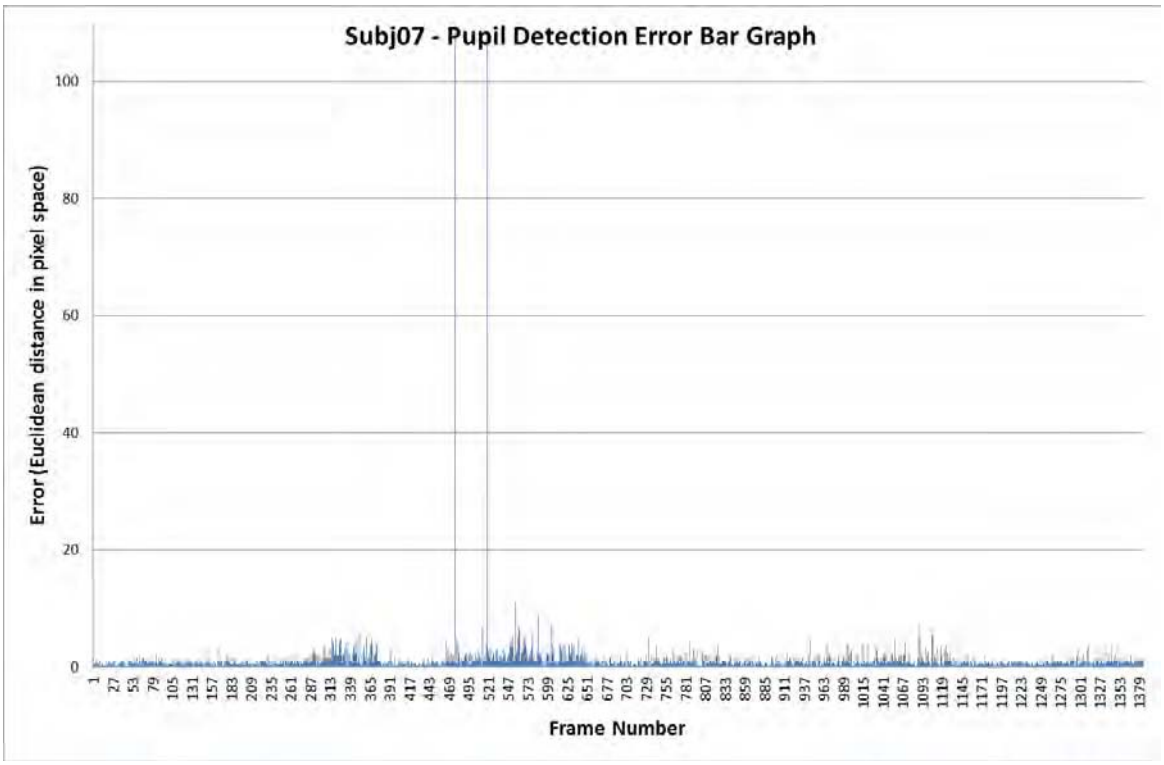


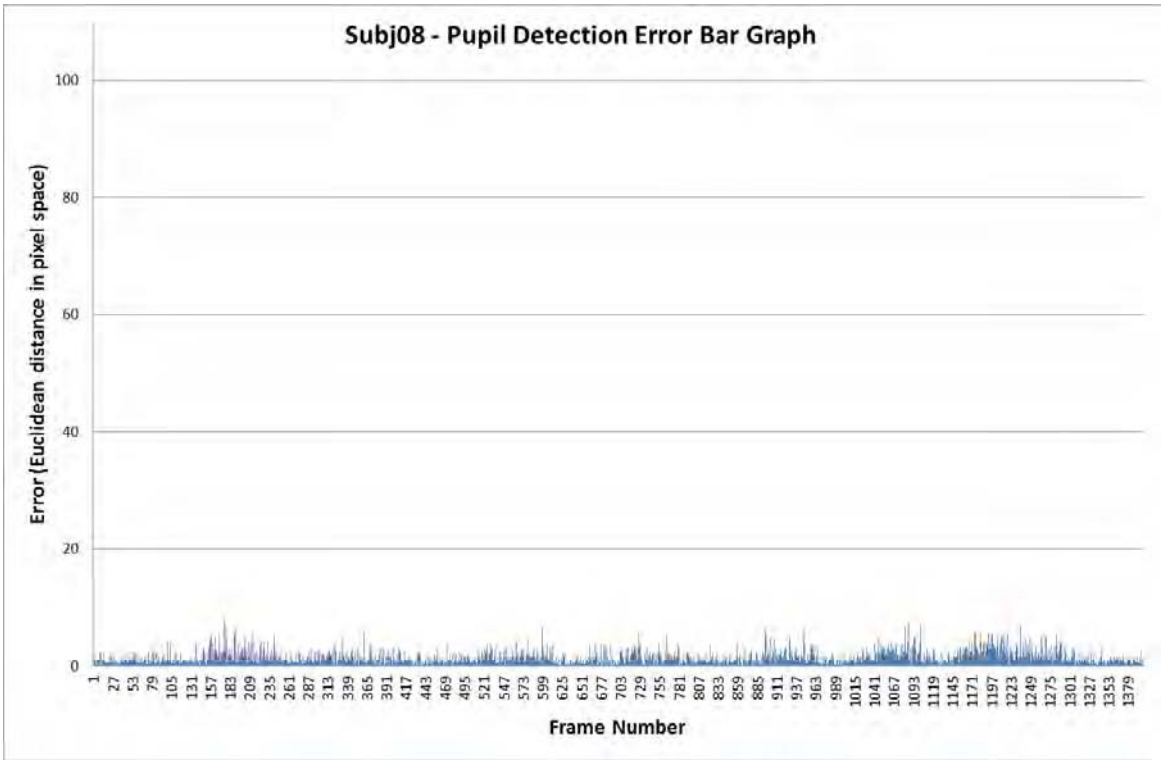

Subj03 - Pupil Detection Error Bar Graph
Error (Euclidean distance in pixel space)
Frame Number

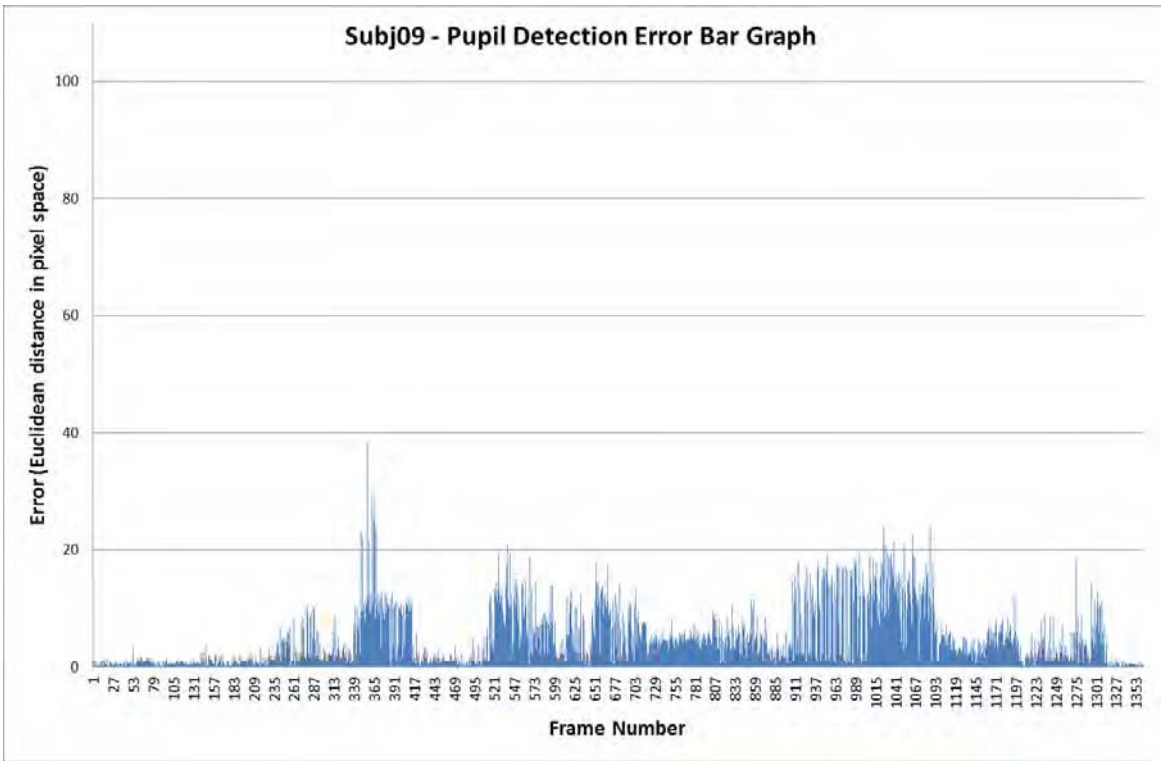

Subj04 - Pupil Detection Error Bar Graph
Error (Euclidean distance in pixel space)
Frame Number


Subj05 - Pupil Detection Error Bar Graph
Error (Euclidean distance in pixel space)
Frame Number


Subj06 - Pupil Detection Error Bar Graph
Error (Euclidean distance in pixel space)
Frame Number


Subj07 - Pupil Detection Error Bar Graph
Error (Euclidean distance in pixel space)
Frame Number


Subj08 - Pupil Detection Error Bar Graph
Error (Euclidean distance in pixel space)
Frame Number


Subj09 - Pupil Detection Error Bar Graph
Error (Euclidean distance in pixel space)
100
80
60
40
20
0
1
27
53
79
105
131
157
183
209
235
261
287
313
339
365
391
417
443
469
495
521
547
573
599
625
651
677
703
729
755
781
807
833
859
885
911
937
963
989
1015
1041
1067
1093
1119
1145
1171
1197
1223
1249
1275
1301
1327
1353
Frame Number

EVALUATION OF THE IRIS BOUNDARY DETECTION ALGORITHM

The iris boundary detection algorithm was evaluated in a similar way to the pupil detection algorithm. The iris boundaries were manually marked in the same sample set as used for the pupil detection algorithm (*only frames where the subject is looking straight ahead were used; as already mentioned in Chapter 4, this is a requirement for the iris boundary detection algorithm* – a total of 1,856 frames) and the iris radius was automatically extracted from the marked boundaries as the ½ of the difference between the two x-coordinates. Figure 29 illustrates the software used to mark the iris test data set.

The tracker was set to process all the frames in the above data set and the error was quantified as the difference between the calculated iris radius and the iris radius calculated from the manually marked boundaries. The errors were then classified to *negligible*, *acceptable* and *unacceptable* to aid the interpretation of the results. As negligible were considered any errors less than or equal to 2 pixels because it is estimated that errors up to 2 pixels may be a result of the manual marking; despite the use of manual software that eases the marking task, it is impossible to mark the data set 100% accurately as a) in some cases the exact iris boundaries cannot be distinguished and b) the boundaries are often not bound to one single pixel. As acceptable were considered any errors between 2 and 8 pixels and as unacceptable any errors over 8 pixels.

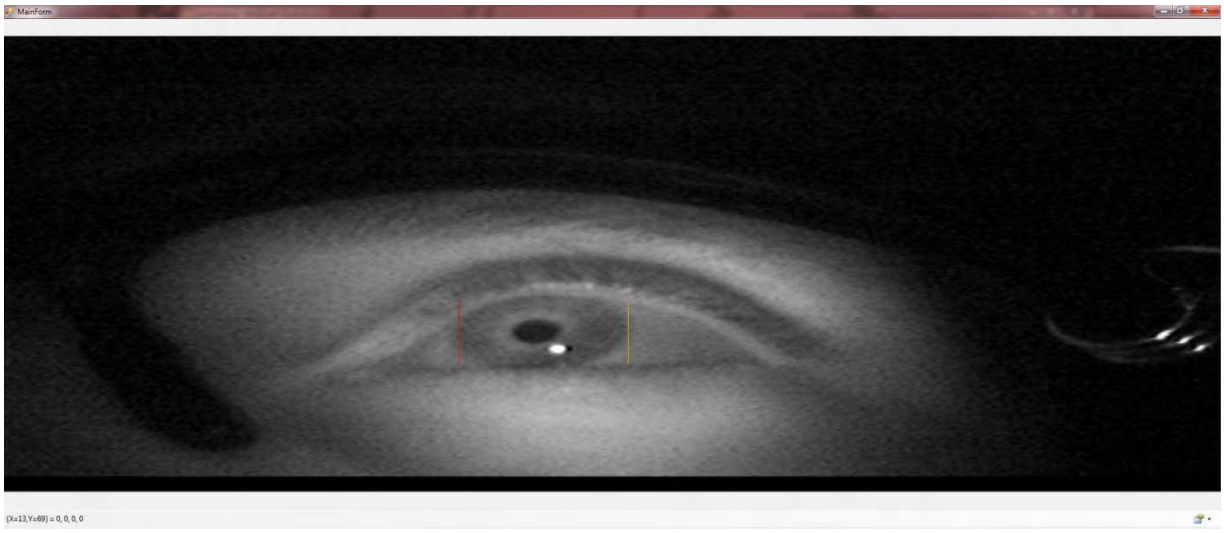


FIGURE 29: MANUAL MARKING OF THE IRIS BOUNDARIES.

Table 13 and Table 14 below display the error and error classification statistics. Specifically, in Table 13, the first column designates the test sample, the second column displays the processed number of frames for the particular sample, the third and fourth column the average error and standard deviation for the sample respectively and the fifth column displays the maximum error across the frames of the particular sample. Finally, the last row of Table 13 shows the average error, standard deviation and maximum error for the whole data set taken together. In Table 14, a similar format is used; the sample is shown on the first column, the total number of frames on the second column, followed by the number of frames of each class and the number of frames that failed to process. The final row of Table 14 shows the total number of frames for each class, both as a cardinal number and a percentage.

As can be seen from the tables above, the iris radius detection algorithm performs very well and consistently; in all of the test sequences, the error is never high enough to be classified unacceptable. On average, the iris radius error is 2.11 ± 1.42 pixels and the maximum error across all sequences is less than 8 pixels, 7.92 pixels to be exact.

Table 15 shows the error distribution for each test sequence using bar graphs.

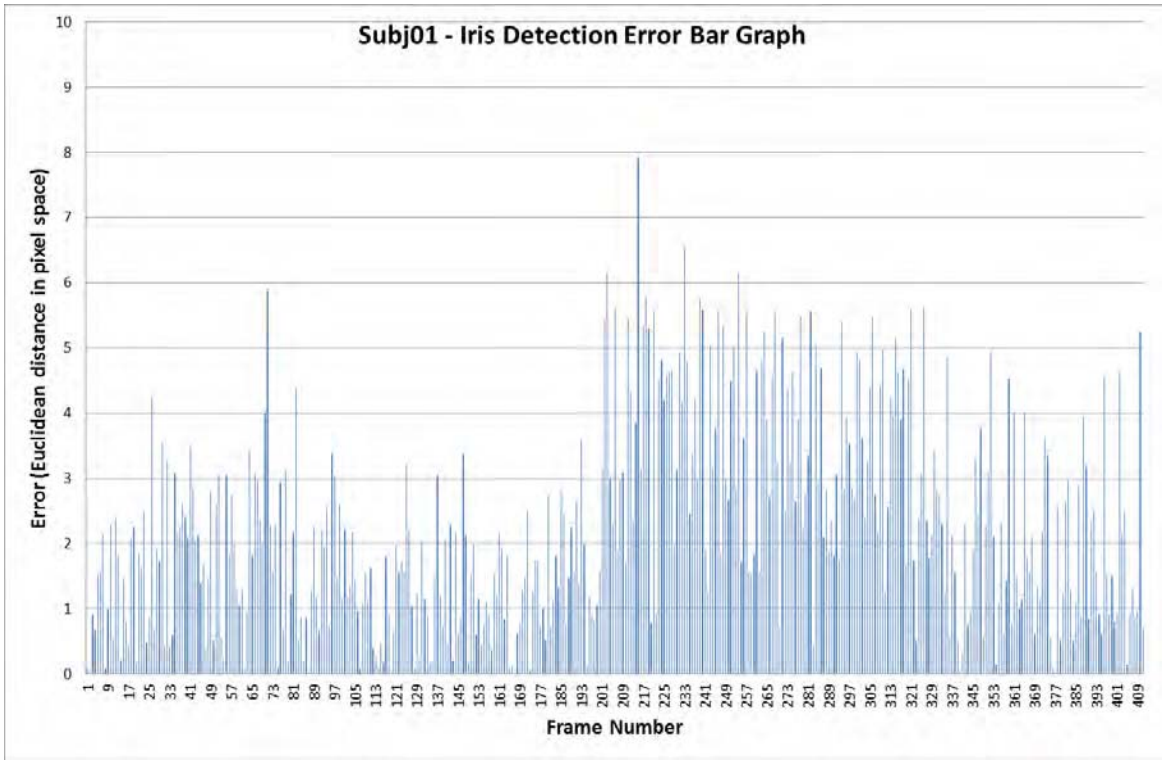
TABLE 13: IRIS EVALUATION - ERROR STATISTICS, IN PIXELS.

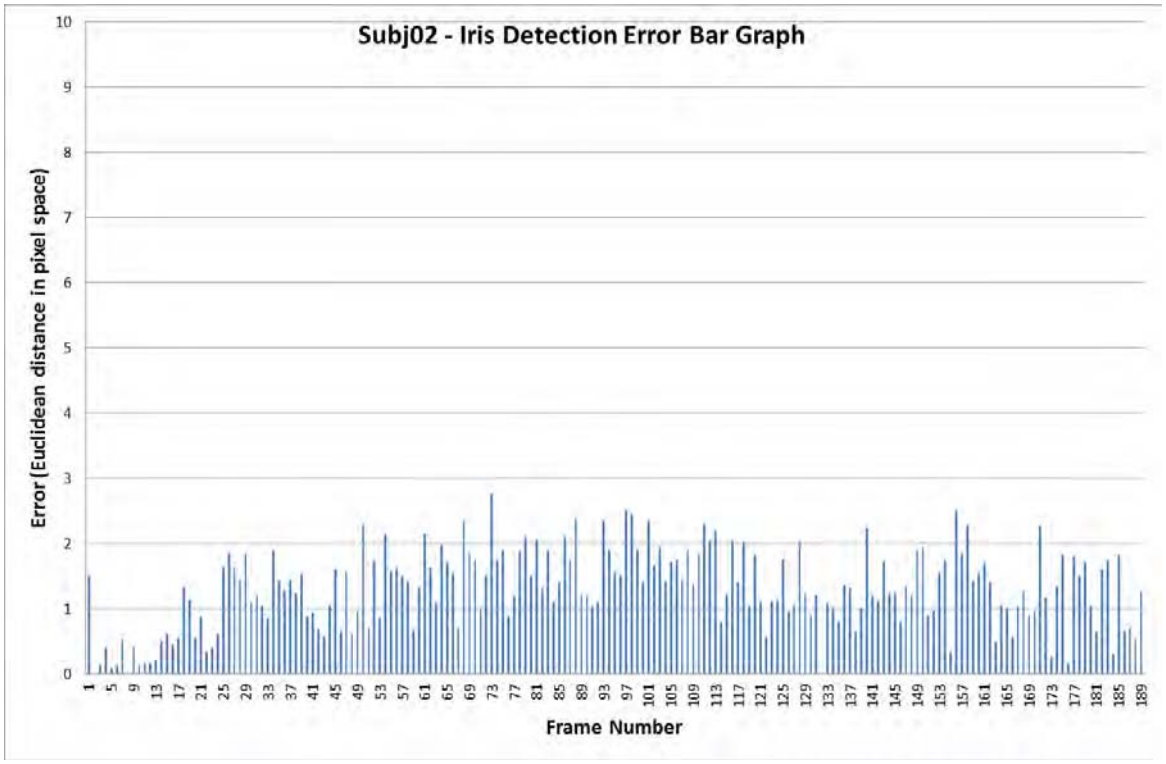
Sample	Frames	Average	σ	Maximum
Subj01	411	2.24	1.57	7.91
Subj02	189	1.29	0.60	2.76
Subj03	282	3.45	1.39	7.92
Subj04	44	0.54	0.48	2.07
Subj05	169	0.79	0.47	1.95
Subj06	115	0.89	0.72	2.92
Subj07	210	2.43	1.29	6.43
Subj08	235	1.62	0.55	3.44
Subj09	201	3.17	0.72	4.88
Overall	**1856**	**2.11**	**1.42**	**7.92**

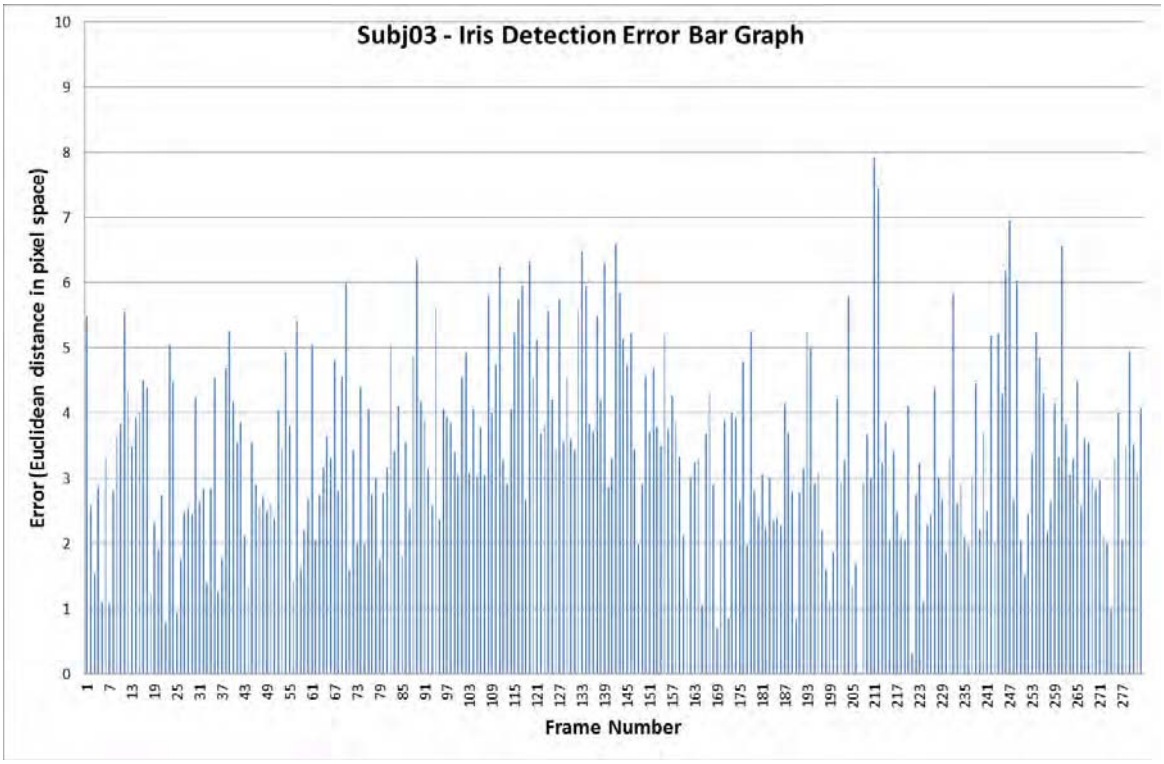
TABLE 14: IRIS EVALUATION - ERROR CLASSIFICATION STATISTICS, NUMBER OF FRAMES.

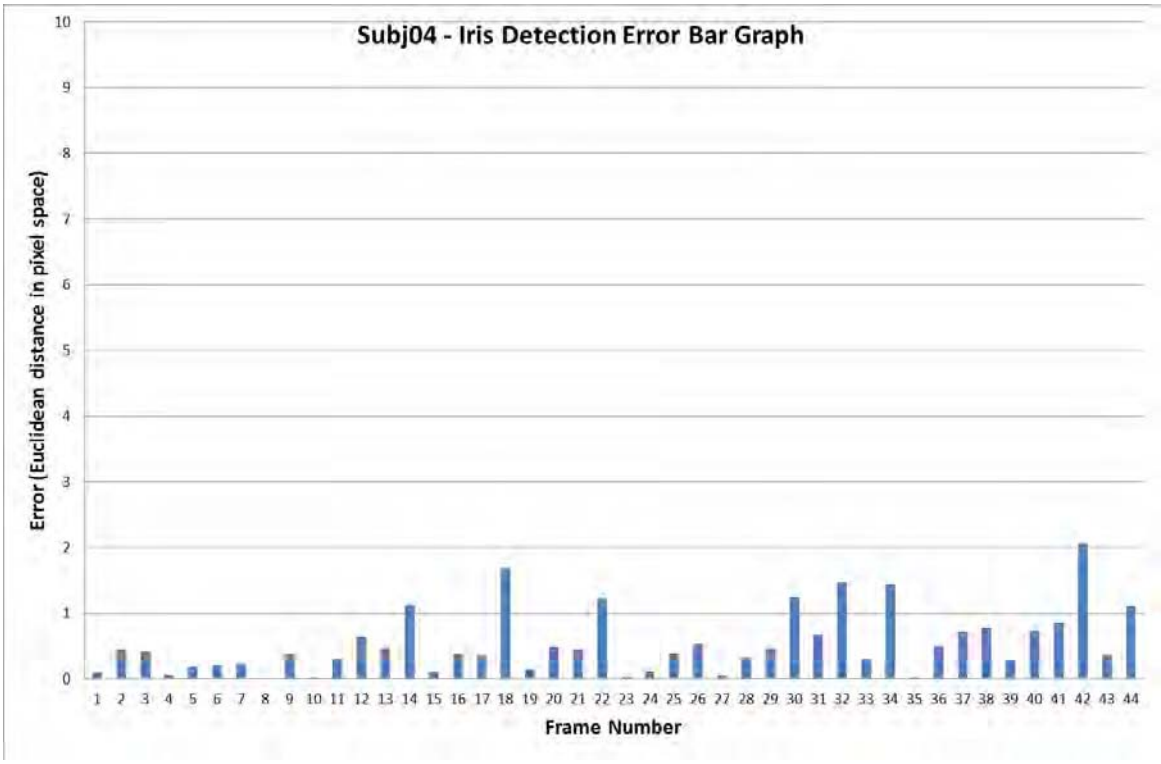
Sample	Frames	Negligible	Acceptable	Unacceptable	Failed
Subj01	411	208	203	0	0
Subj02	189	166	23	0	0
Subj03	282	38	244	0	0
Subj04	44	43	1	0	0
Subj05	169	169	0	0	0
Subj06	115	102	13	0	0
Subj07	210	81	129	0	0
Subj08	235	181	54	0	0
Subj09	201	8	193	0	0
Total	**1856 (100%)**	**996 (53.66%)**	**860 (46.33%)**	**0 (0%)**	**0 (0%)**

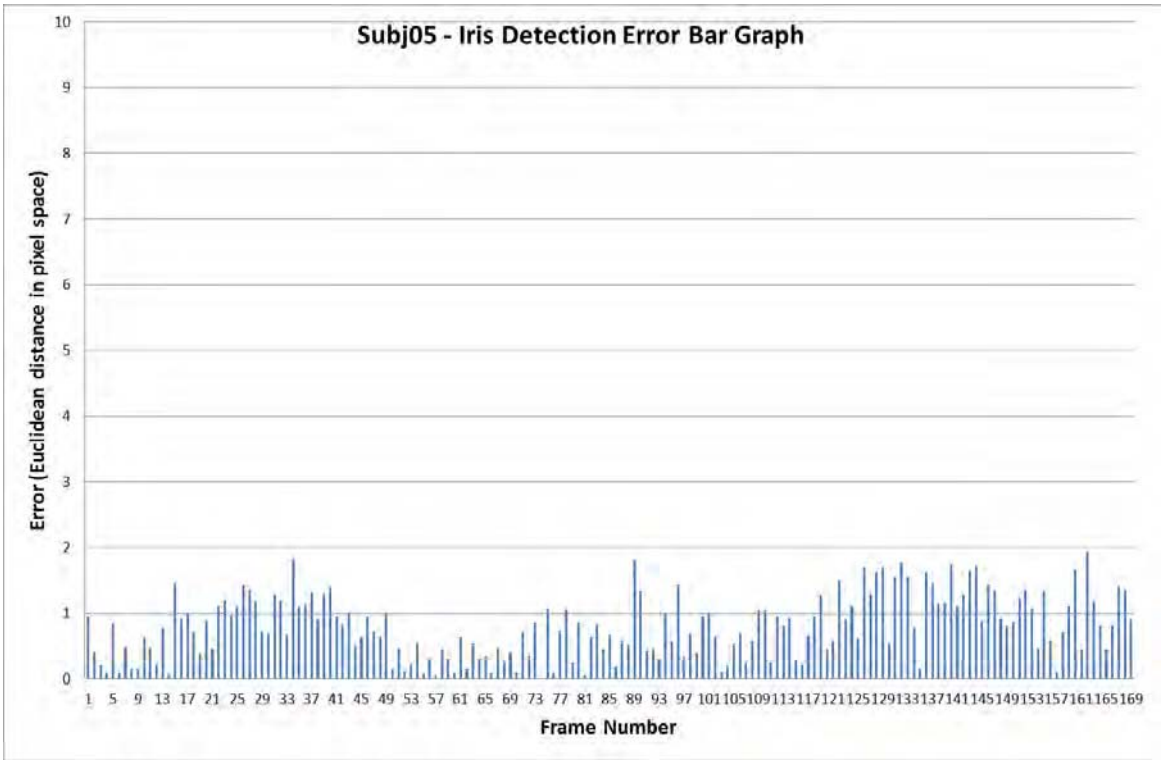
TABLE 15: BAR GRAPHS OF THE IRIS DETECTION ALGORITHM ERROR FOR EACH TEST SEQUENCE.

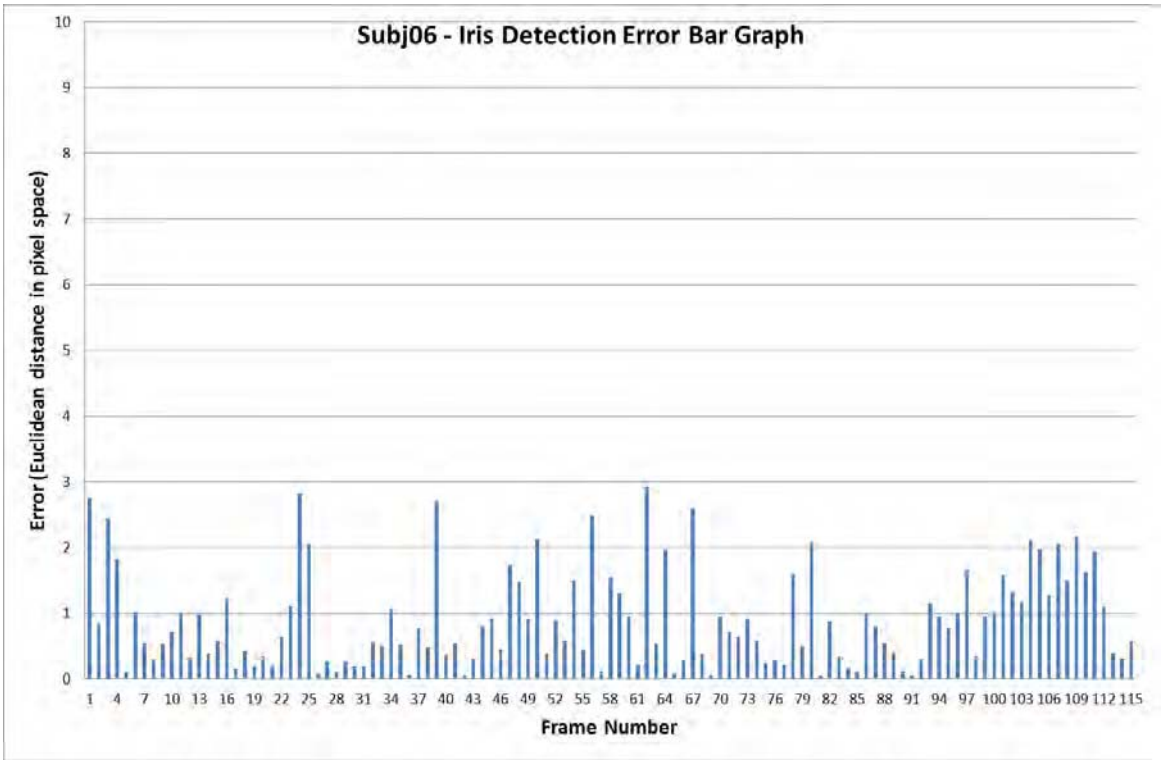


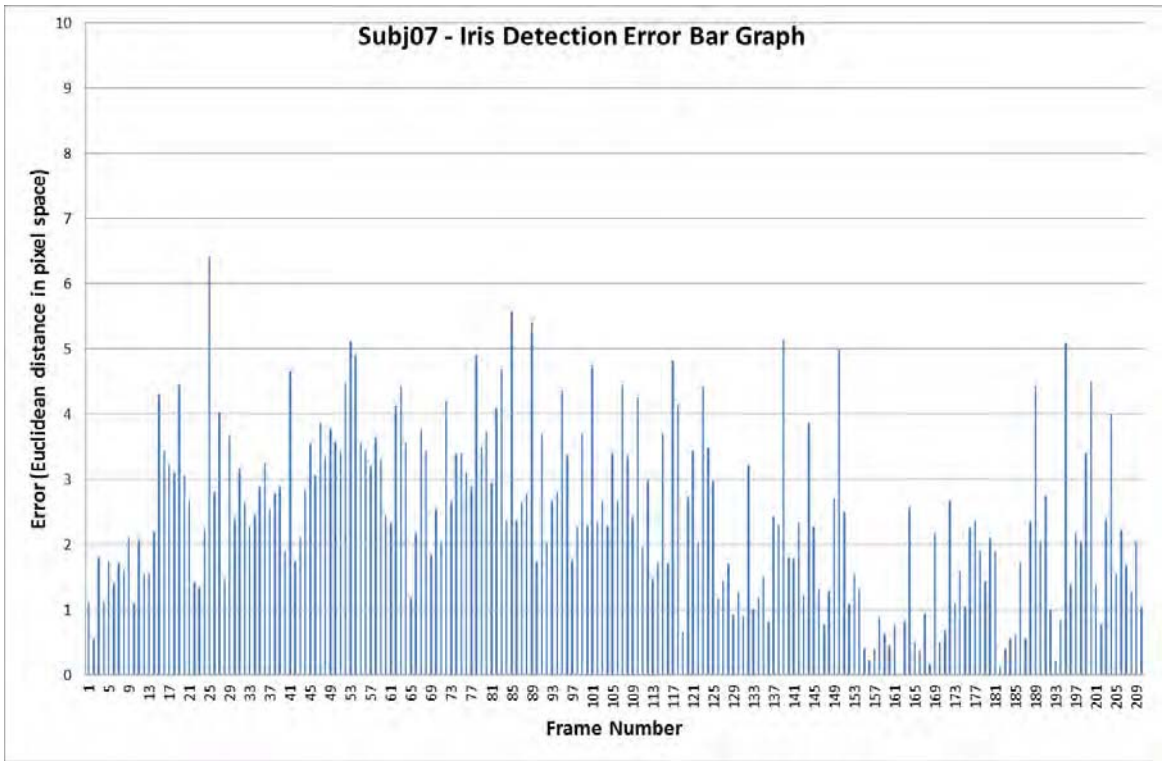


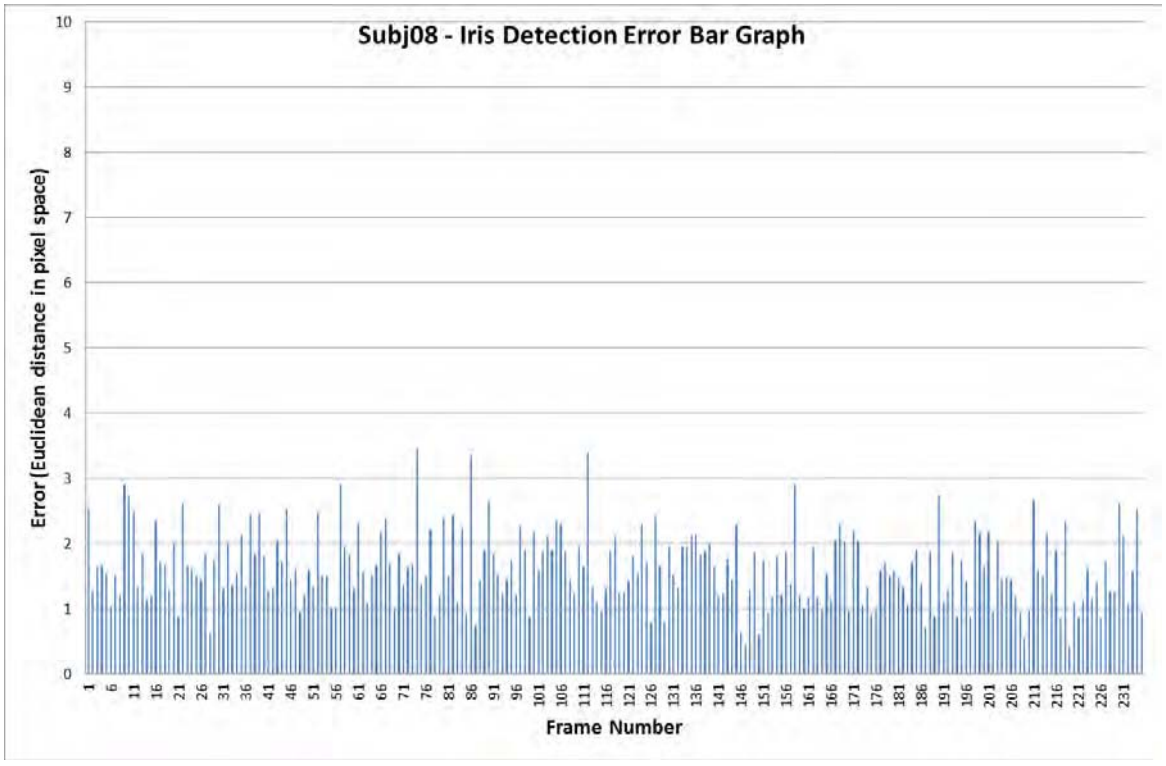

Subj03 - Iris Detection Error Bar Graph
Error (Euclidean distance in pixel space)
Frame Number

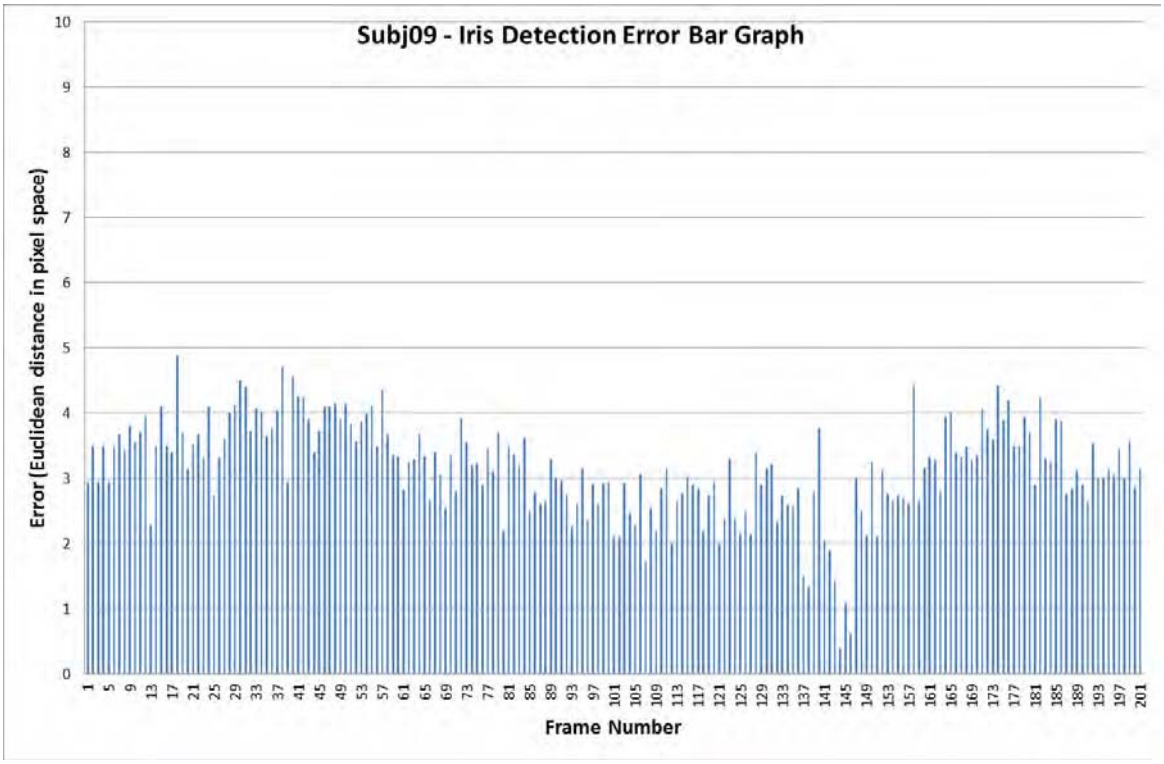

Subj04 - Iris Detection Error Bar Graph
Error (Euclidean distance in pixel space)
Frame Number


Subj05 - Iris Detection Error Bar Graph
Error (Euclidean distance in pixel space)
Frame Number


Subj06 - Iris Detection Error Bar Graph
Error (Euclidean distance in pixel space)
Frame Number


Subj07 - Iris Detection Error Bar Graph
Error (Euclidean distance in pixel space)
Frame Number


Subj08 - Iris Detection Error Bar Graph
Error (Euclidean distance in pixel space)
Frame Number


Subj09 - Iris Detection Error Bar Graph
Error (Euclidean distance in pixel space)
Frame Number
0
1
2
3
4
5
6
7
8
9
10
1 5 9 13 17 21 25 29 33 37 41 45 49 53 57 61 65 69 73 77 81 85 89 93 97 101 105 109 113 117 121 125 129 133 137 141 145 149 153 157 161 165 169 173 177 181 185 189 193 197 201

EVALUATION OF THE CORNER DETECTION AND CLUSTERING ALGORITHMS

The evaluation of the corner detection algorithm has been a significant challenge for the simple reason that the corners will change appearance in coordination to the subject's eye morphology over time (both momentary e.g. blinks or eyelid movement because of eye movement as well as more lasting changes e.g. squinting). In other words, the corners may appear to have moved if the subjects squints or simply looks in a different direction and their eyelids move significantly. For the evaluation data set, frames of the subject is looking straight ahead (as already mentioned in Chapter 4, this is a requirement for the iris boundary and corner detection algorithms) were included and frames where the corners could not be clearly marked were excluded, giving a total of 1,856 frames.

In some cases, because of the particular eye morphology of the individual subjects, there were two candidates for the inner eye corner, as shown in Figure 30. As the right-most candidate changes appearance more often than the left-most candidate and in order to be consistent across-subjects, the left-most candidate was always marked thus marking the point nearest to the tear gland as the inner eye corner. Figure 31 illustrates the marking software in operation.

The tracker was set to process all the frames in the above data set and the error was quantified as the Euclidean distance between the calculated corners and the manually marked corners. The errors were then classified to *negligible*, *acceptable* and *unacceptable* to aid the interpretation of the results. As negligible were considered any errors less than or equal to 4 pixels. As acceptable were considered any errors between 4 and 10 pixels and as unacceptable any errors over 10 pixels.

Slightly higher thresholds (4 pixels versus 2 pixels for *negligible* and 10 pixels versus 8 pixels for *acceptable/unacceptable*) were used for the evaluation of the eye corners detection algorithm for several reasons:

a) Just like the pupil and iris evaluation, manually marking these positions over several hundreds of frames is a somewhat error-prone process; there is no objective means to mark the exact location of the points in question and as careful as one may be, errors will occur in such a repetitive and tedious task. This problem is even more evident with the

marking of the eye corners as their appearance is much less distinct than the pupil and iris.

b) The eye corners detection algorithm subsamples the input images to ¼ of its original size before eye corner detection takes place and later up samples the result by multiplying the eye corner locations by 4. Thus, an error of N pixels in the scaled-down image will be equivalent to $4 \times N$ pixels movement in the original image. As such, it is reasonable to consider deviation of the eye corner detection of 4-10 pixels as acceptable.

In practice, it is hypothesised that the manual marking errors are higher than assumed but the above low-threshold values were used for the classification because this cannot be proved objectively and to produce conservative error measurements.

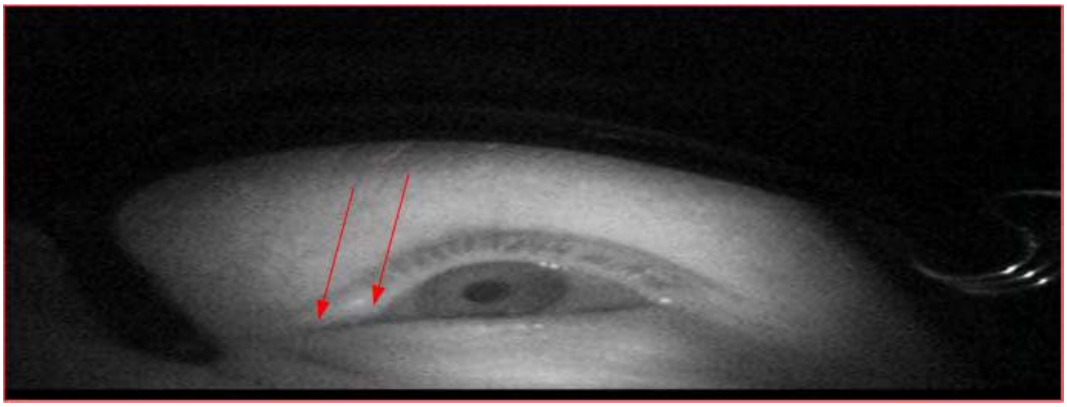


FIGURE 30: "TRUE" INNER EYE CORNER MAY HAVE TWO CANDIDATES, SHOWN IN RED.

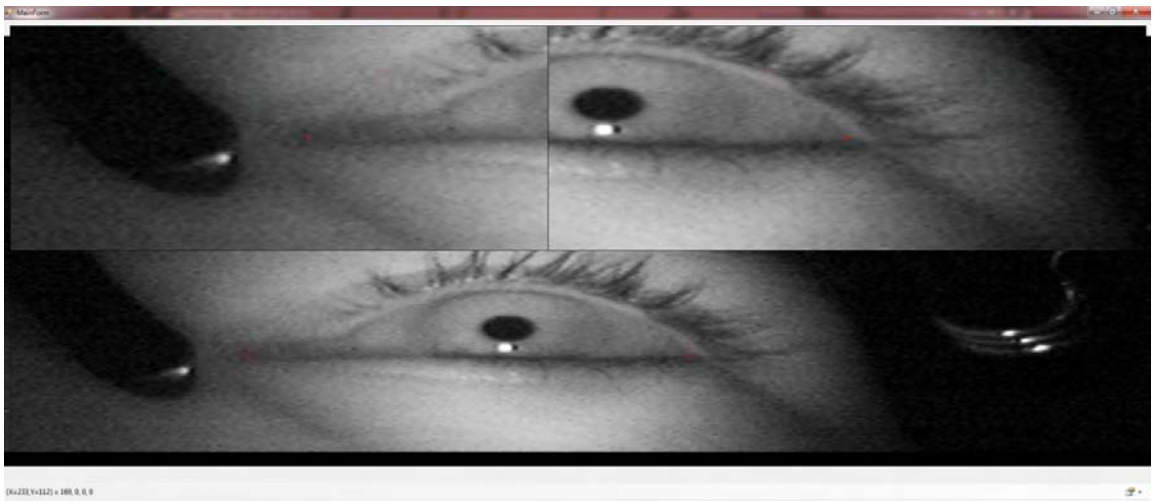


FIGURE 31: MANUAL MARKING OF THE EYE CORNERS.

Table 16, Table 18 and Table 17, Table 19 below display the error and error classification statistics for the inner and outer corner respectively. As before, in the error statics table, the first column designates the test sample, the second column displays the processed number of frames for the particular sample, the third and fourth column the average error and standard deviation for the sample respectively and the fifth column displays the maximum error across the frames of the particular sample. Finally, the last row shows the average error, standard deviation and maximum error for the whole data set taken together. In the error classification statistics table, the sample is shown on the first column, the total number of frames on the second column, followed by the number of frames of each class and the number of frames that failed to process. The final row shows the total number of frames for each class, both as a cardinal number and a percentage.

Detecting the location of the eye corners is the hardest problem to tackle and this is indeed reflected in the error measurements from the corner detection, as shown in the tables above. Both the inner and outer corner detection algorithm show very similar error measurements; 8.32 ± 5.78 pixels and 8.41 ± 5.40 pixels respectively. If it is considered that the algorithm subsamples the original image to a quarter of its size, this corresponds to 2-3 pixels of error in the down-sampled image, which is an acceptable result. The error is amplified when the result is up-sampled.

The maximum error across the whole data set is 50.24 pixels and 44.38 pixels for the inner and outer corner detection respectively. This shows that the algorithm will in some cases output false positives and this is the reason that the clustering algorithm was integrated into the corner detection as explained in Chapter 4.

The evaluation of the corner detection after clustering has been applied was performed in the same manner as the corner detection algorithm and the results are presented in tables 18 through 21. As can be seen, there is a significant reduction in error with the average error being 7.41 ± 3.78 pixels for the inner corner and 6.49 ± 3.21 pixels for the outer corner. Additionally, by filtering outliers (false positives), the maximum error was reduced to 16.76 pixels and 25.83 pixels respectively.

Figure 32 and Figure 33 graphically illustrate how different clustering methods and settings affect the average error measurements for the inner and outer corner detection respectively. The

clustering methods tested are as outlined in Chapter 4 (no clustering, weighted sum clustering, outliers filtering).

As can be seen in this comparison:

- Both clustering algorithms offer a significant improvement versus no clustering, especially when the error is high.
- In cases of small error, the two clustering methods perform similarly.
- In cases of larger error, the clustering method used in the eye-tracker performs better than the weighted sum clustering.
- In most cases, there are no significant improvements offered for $N > 15$.

TABLE 16: CORNER DETECTION EVALUATION (INNER CORNER) - ERROR STATISTICS, IN PIXELS.

Sample	Frames	Average	σ	Maximum
Subj01	411	5.18	5.55	44.18
Subj02	189	9.27	3.44	23.60
Subj03	263	13.99	4.33	35.00
Subj04	44	11.99	6.17	23.43
Subj05	162	8.01	9.16	50.24
Subj06	109	9.17	4.63	30.08
Subj07	210	8.01	4.42	22.02
Subj08	231	6.59	3.35	21.09
Subj09	201	7.75	3.47	21.02
Overall	**1820**	**8.32**	**5.78**	**50.24**

TABLE 17: CORNER DETECTION EVALUATION (INNER CORNER) - ERROR CLASSIFICATION STATISTICS, NUMBER OF FRAMES.

Sample	Frames	Negligible	Acceptable	Unacceptable	Failed
Subj01	411	221	161	29	0
Subj02	189	9	128	52	0
Subj03	263	0	100	163	19
Subj04	44	0	24	20	0
Subj05	162	34	114	14	7
Subj06	109	0	85	24	1
Subj07	210	31	138	41	0
Subj08	231	0	180	51	4
Subj09	201	25	104	72	0
Total	**1820 (98.33%)**	**320 (17.29%)**	**1034 (55.86%)**	**466 (25.18%)**	**31 (1.67%)**

TABLE 18: CORNER DETECTION EVALUATION (OUTER CORNER) - ERROR STATISTICS, IN PIXELS.

Sample	Frames	Average	σ	Maximum
Subj01	411	9.53	3.24	26.07
Subj02	189	8.82	3.64	21.37
Subj03	282	6.57	3.58	22.20
Subj04	44	11.27	4.56	25.61
Subj05	169	13.28	9.34	33.61
Subj06	110	7.47	4.62	38.63
Subj07	210	9.60	5.24	26.92
Subj08	235	6.54	6.53	44.38
Subj09	201	5.02	1.50	8.94
Overall	**1851**	**8.41**	**5.40**	**44.38**

TABLE 19: CORNER DETECTION EVALUATION (OUTER CORNER) - ERROR CLASSIFICATION STATISTICS, NUMBER OF FRAMES.

Sample	Frames	Negligible	Acceptable	Unacceptable	Failed
Subj01	411	11	263	137	0
Subj02	189	5	146	38	0
Subj03	282	63	203	16	0
Subj04	44	0	30	14	0
Subj05	169	0	98	71	0
Subj06	110	2	95	13	0
Subj07	210	7	132	71	0
Subj08	235	72	150	13	0
Subj09	201	102	99	0	0
Total	**1851 (100%)**	**262 (14.15%)**	**1216 (65.69%)**	**373 (20.15%)**	**0 (0%)**

TABLE 20: CORNER DETECTION EVALUATION AFTER CLUSTERING (INNER CORNER) - ERROR STATISTICS, IN PIXELS.

Sample	Frames	Average	σ	Maximum
Subj01	279	3.99	2.41	15.09
Subj02	161	6.51	2.17	12.11
Subj03	254	13.36	1.74	16.76
Subj04	30	9.72	1.14	12.16
Subj05	99	5.95	1.46	11.21
Subj06	62	7.49	1.18	16.32
Subj07	168	7.23	2.95	13.24
Subj08	207	5.97	2.29	11.63
Subj09	145	7.30	2.38	11.78
Overall	**1405**	**7.41**	**3.78**	**16.76**

TABLE 21: CORNER DETECTION EVALUATION AFTER CLUSTERING (INNER CORNER) - ERROR CLASSIFICATION STATISTICS, NUMBER OF FRAMES.

Sample	Frames	Negligible	Acceptable	Unacceptable	Failed
Subj01	279	108	162	9	0
Subj02	161	15	127	19	0
Subj03	254	0	12	242	0
Subj04	30	0	19	11	0
Subj05	99	3	95	1	0
Subj06	62	0	61	1	0
Subj07	168	20	113	35	0
Subj08	207	0	184	23	0
Subj09	145	24	105	16	0
Total	**1405 (100%)**	**170 (12.10%)**	**878 (62.49%)**	**357 (25.41%)**	**0 (0%)**

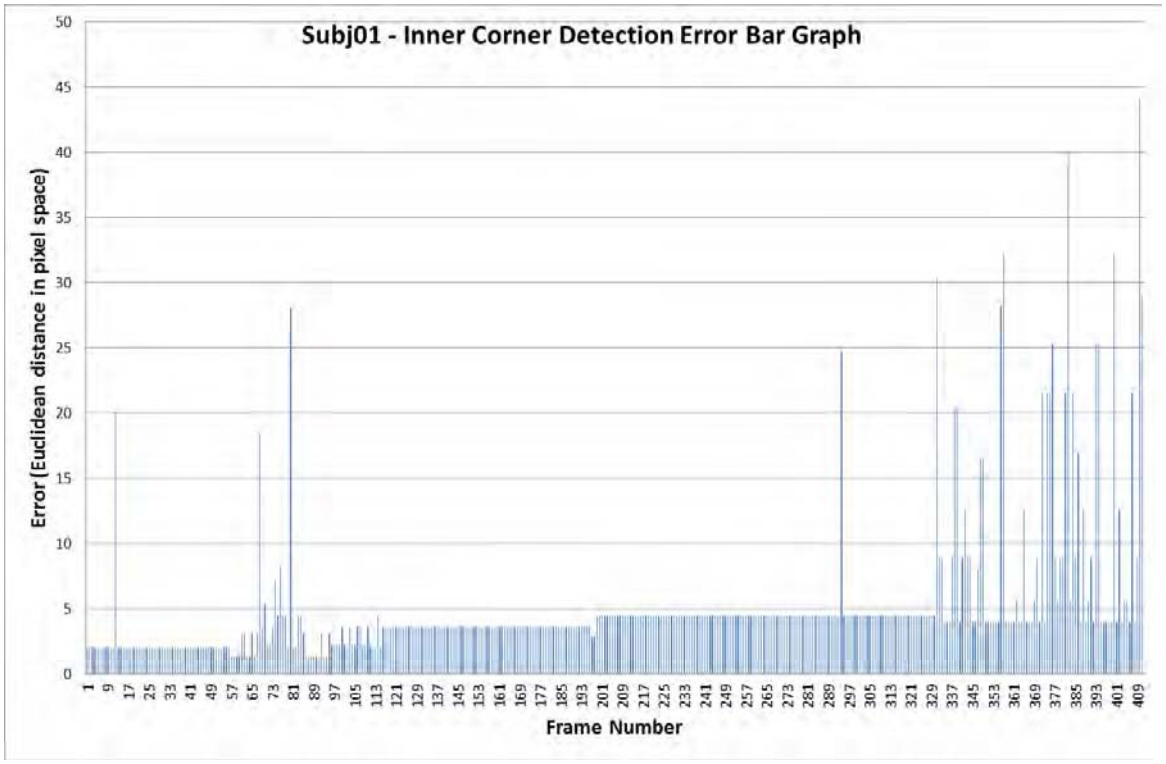
TABLE 22: CORNER DETECTION EVALUATION AFTER CLUSTERING (OUTER CORNER) - ERROR STATISTICS, IN PIXELS.

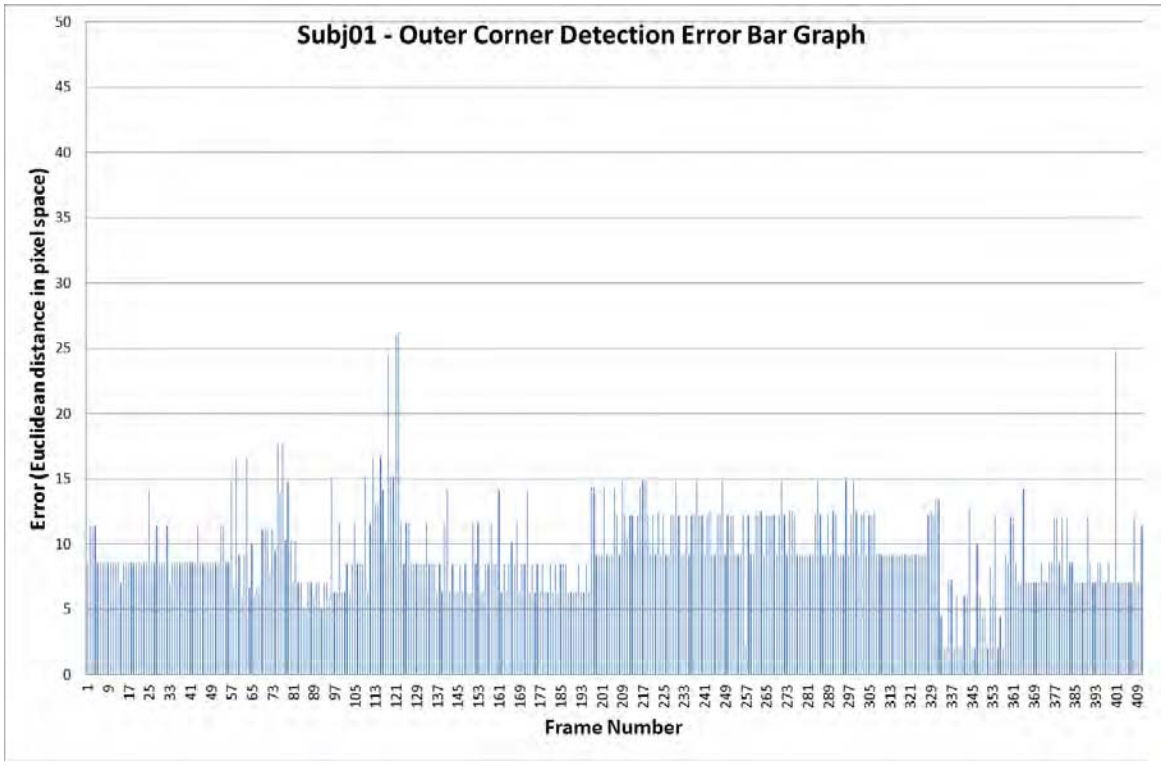
Sample	Frames	Average	σ	Maximum
Subj01	279	8.97	1.96	12.20
Subj02	161	5.59	1.37	9.67
Subj03	254	5.24	2.01	10.00
Subj04	30	8.97	0.29	9.73
Subj05	99	11.10	6.68	25.83
Subj06	62	5.33	0.53	6.40
Subj07	168	6.32	1.66	14.57
Subj08	207	4.25	2.20	7.28
Subj09	145	5.10	1.34	7.28
Overall	**1405**	**6.49**	**3.21**	**25.83**

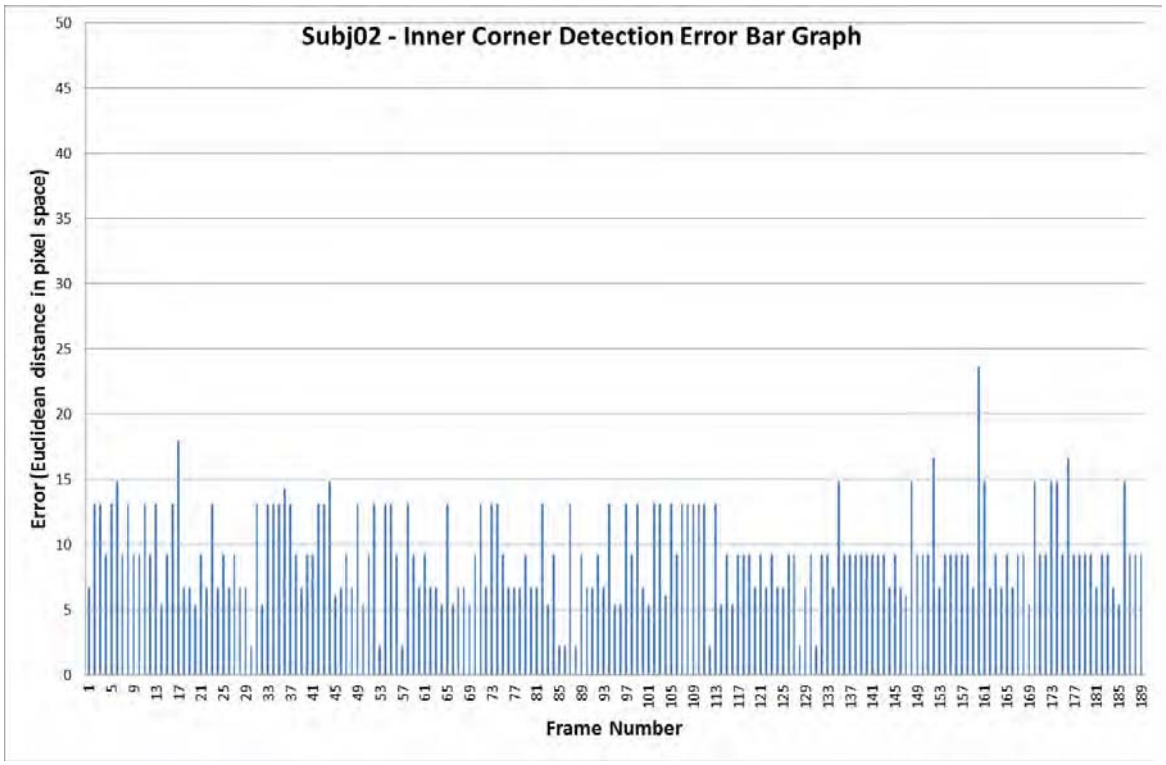
TABLE 23: CORNER DETECTION EVALUATION AFTER CLUSTERING (OUTER CORNER) - ERROR CLASSIFICATION STATISTICS, NUMBER OF FRAMES.

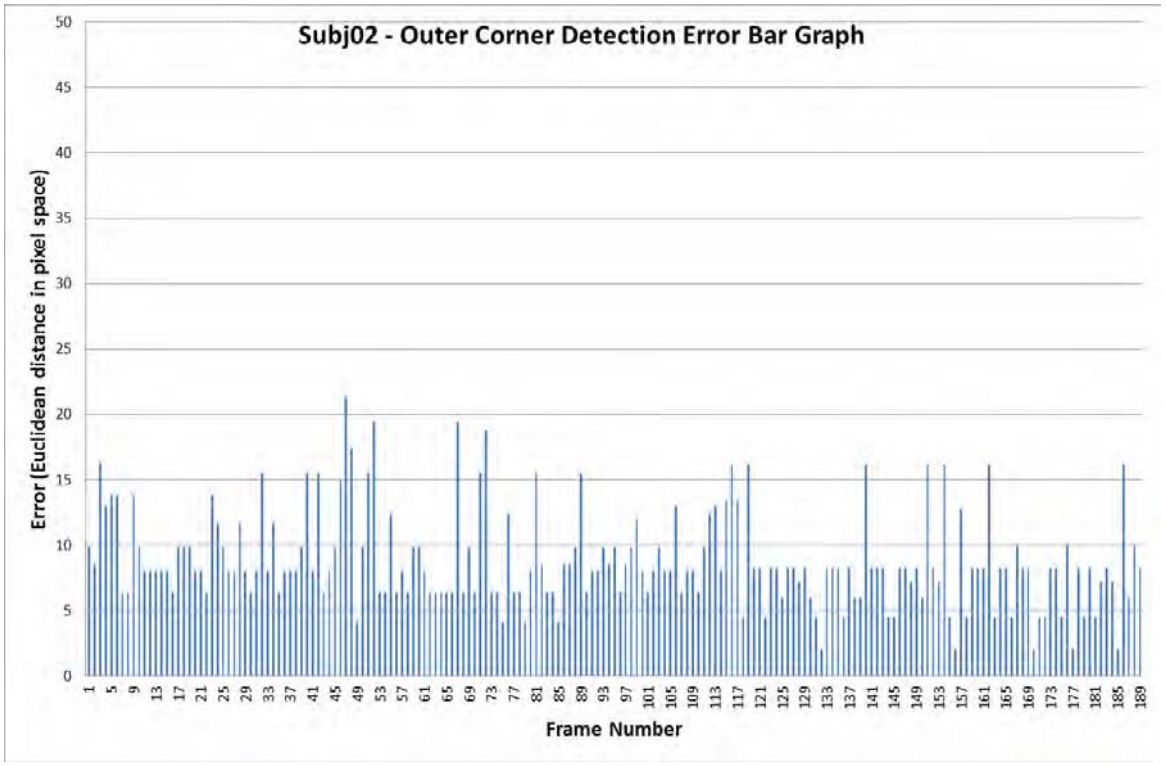
Sample	Frames	Negligible	Acceptable	Unacceptable	Failed
Subj01	279	13	186	80	0
Subj02	161	17	144	0	0
Subj03	254	60	194	0	0
Subj04	30	0	30	0	0
Subj05	99	0	60	39	0
Subj06	62	0	62	0	0
Subj07	168	0	162	6	0
Subj08	207	82	125	0	0
Subj09	145	59	86	0	0
Total	**1405 (100%)**	**231 (16.44%)**	**1049 (74.66%)**	**125 (8.90%)**	**0 (0%)**

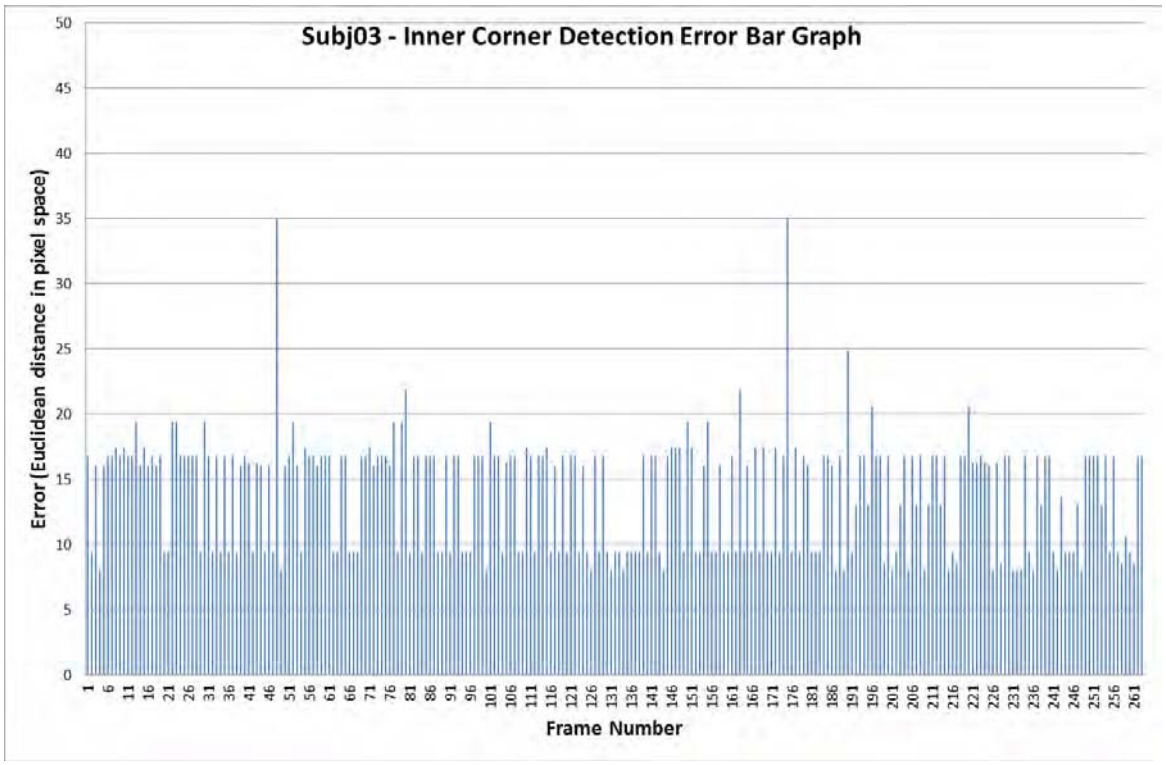
TABLE 24: BAR GRAPHS OF THE CORNER DETECTION ALGORITHM ERROR (INNER AND OUTER CORNER) FOR EACH TEST SEQUENCE.

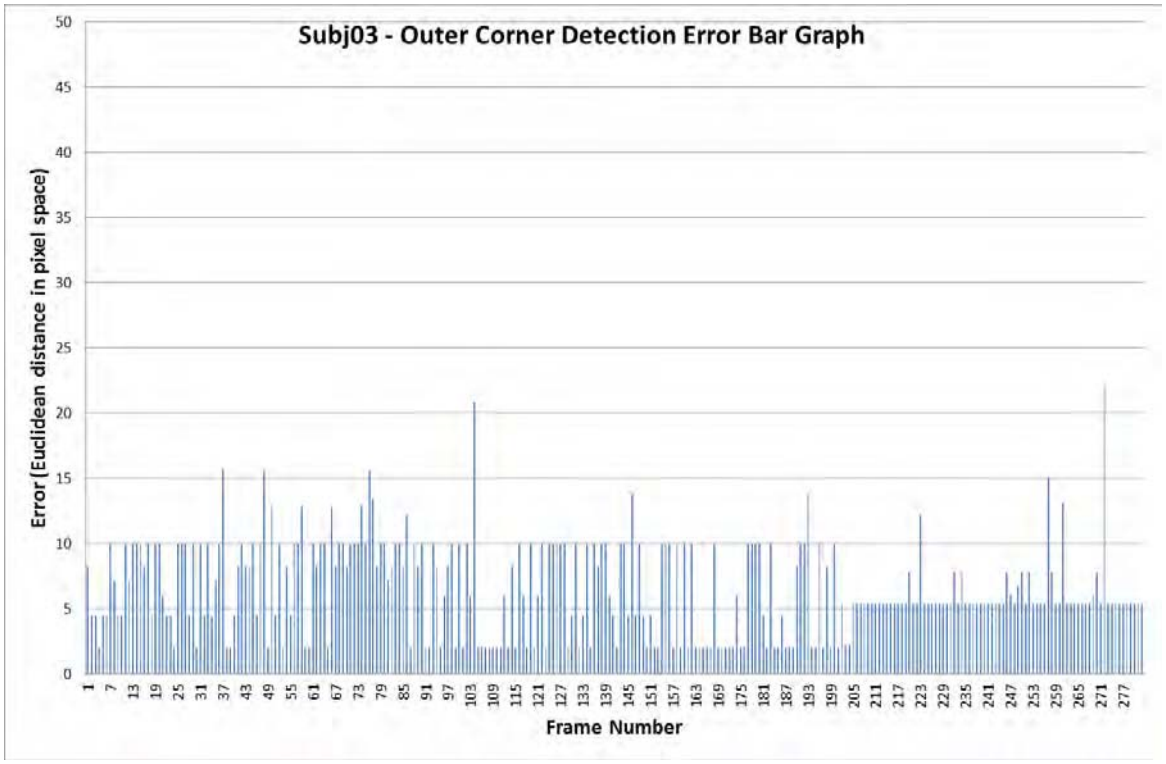


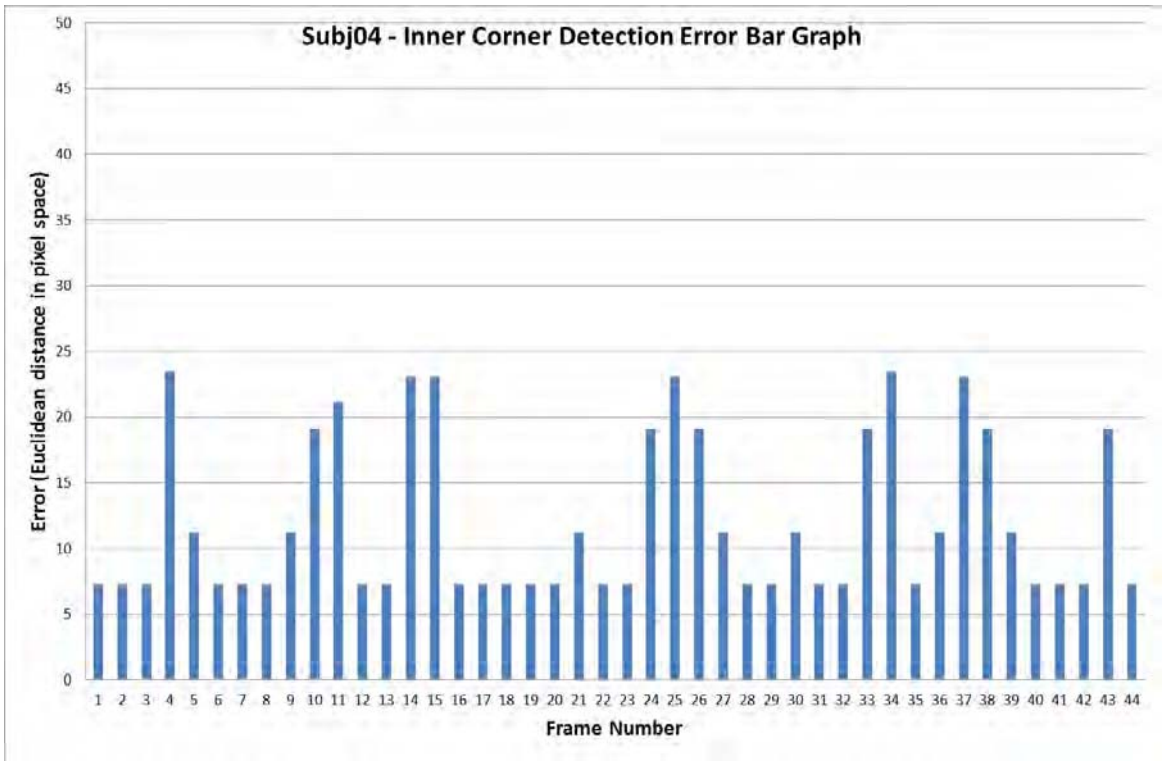


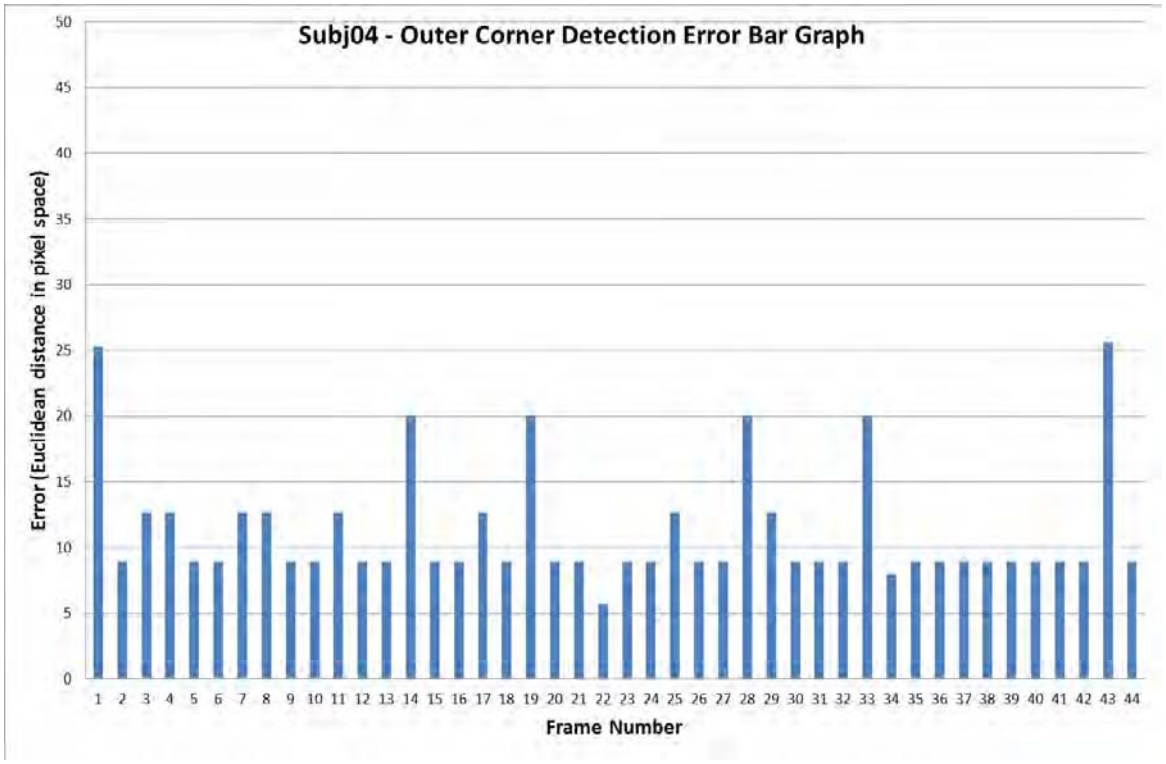

Subj02 - Inner Corner Detection Error Bar Graph
Error (Euclidean distance in pixel space)
Frame Number

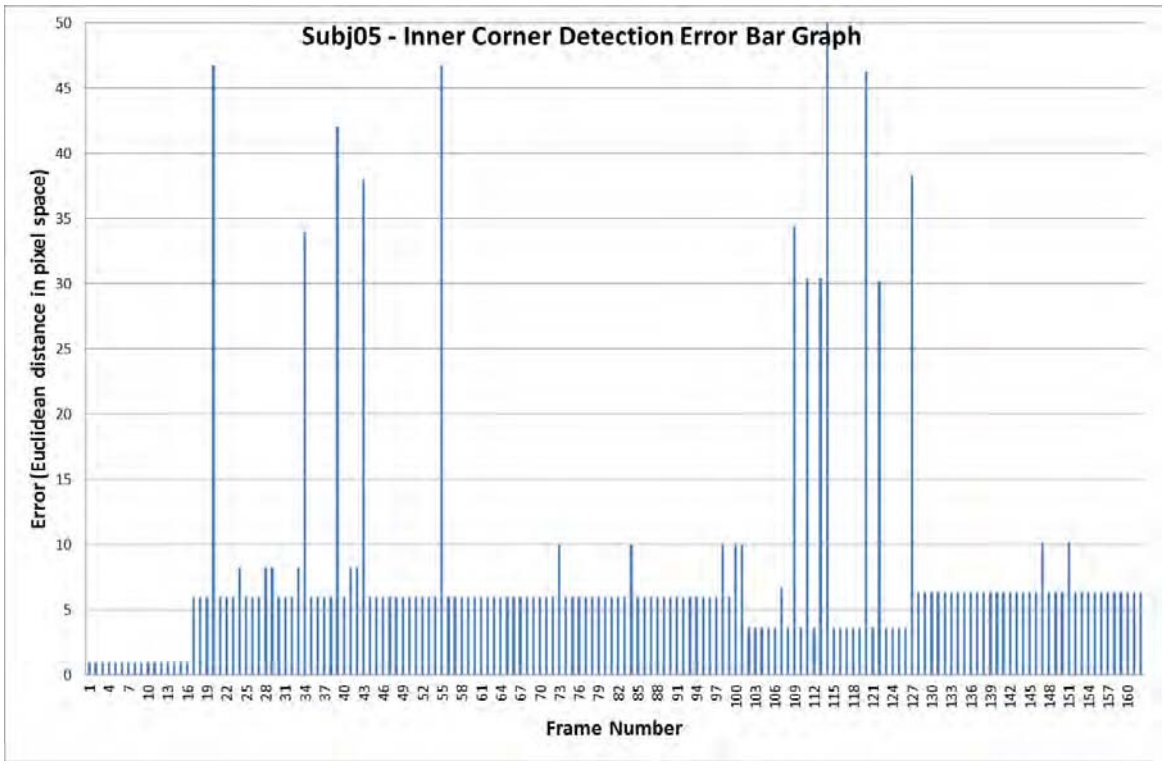

Subj02 - Outer Corner Detection Error Bar Graph
Error (Euclidean distance in pixel space)
Frame Number

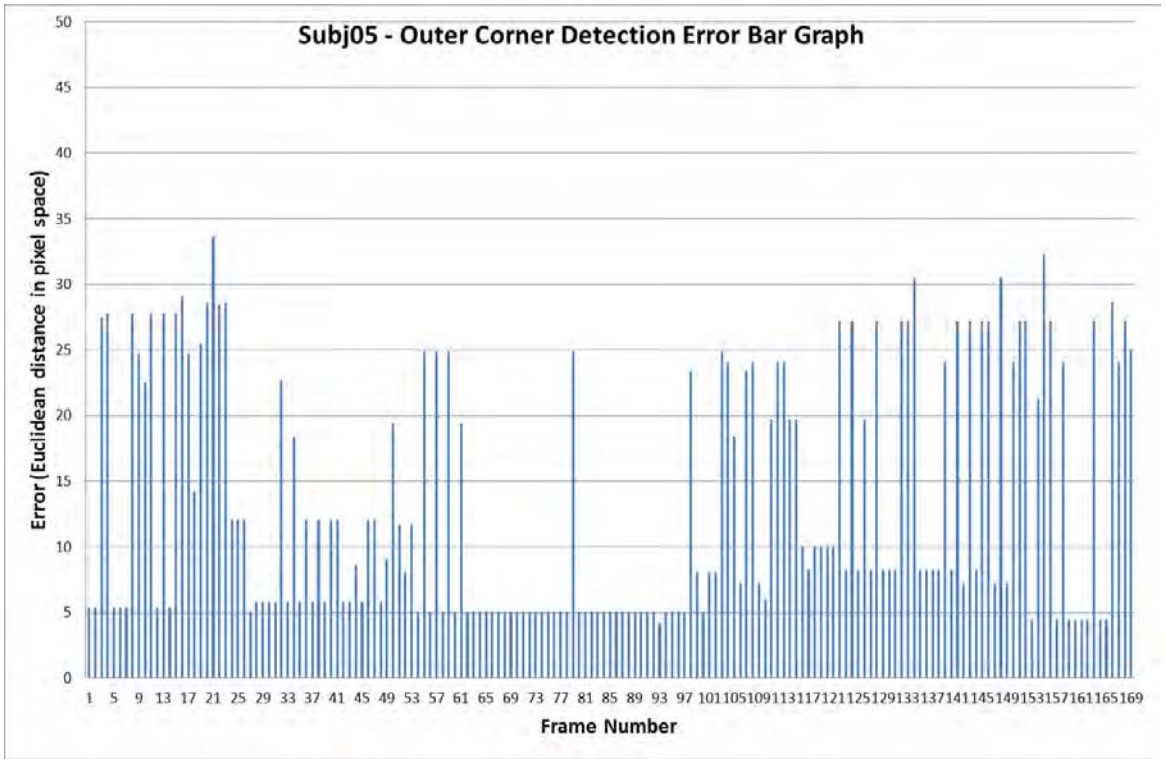

Subj03 - Inner Corner Detection Error Bar Graph
Error (Euclidean distance in pixel space)
Frame Number

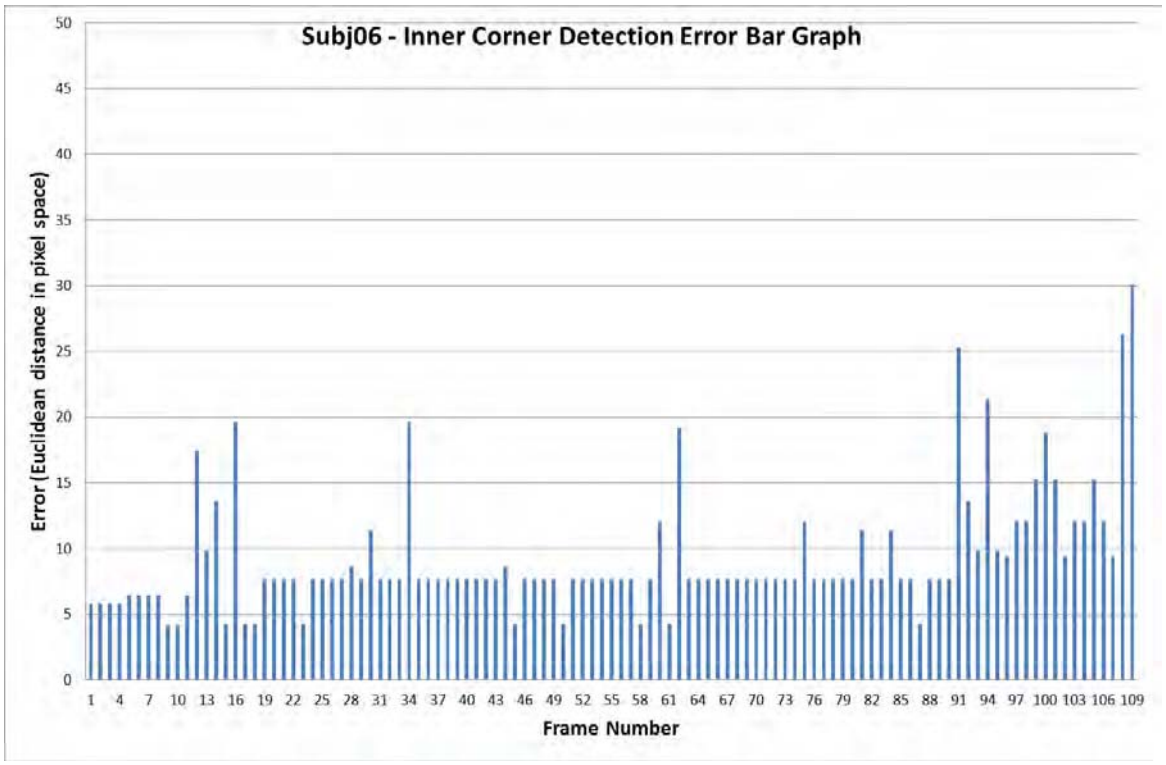

Subj03 - Outer Corner Detection Error Bar Graph
Error (Euclidean distance in pixel space)
Frame Number

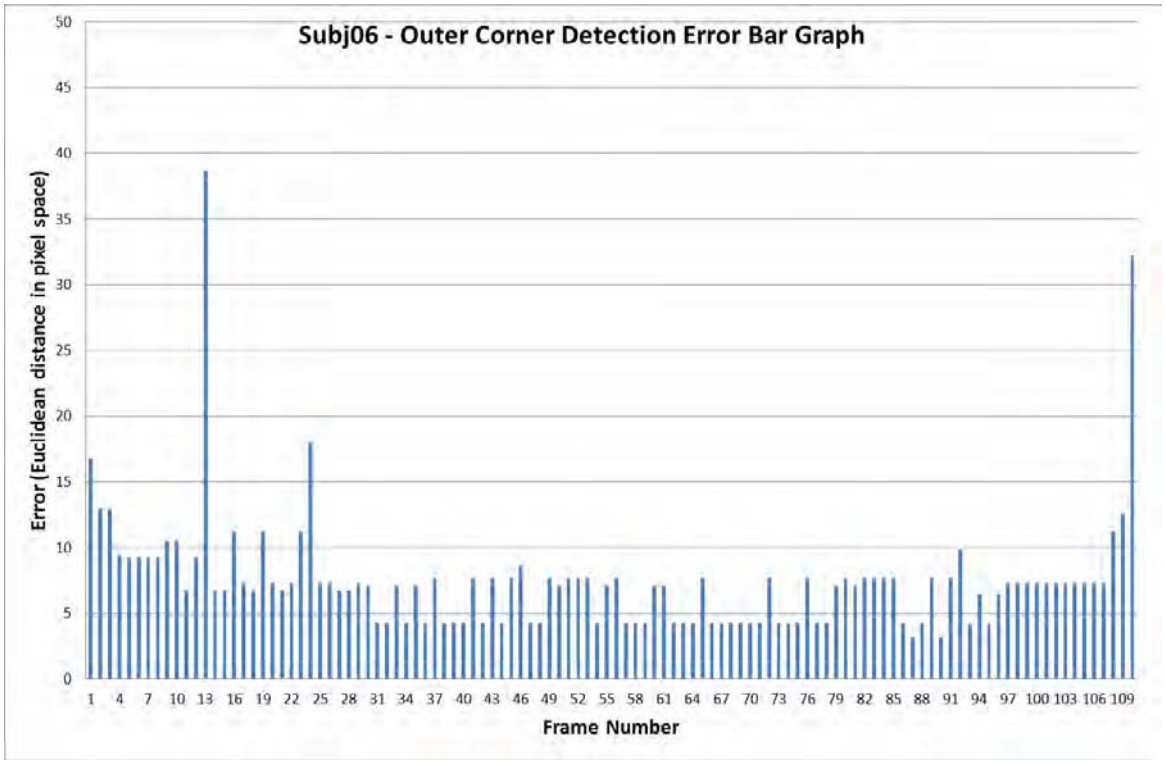

Subj04 - Inner Corner Detection Error Bar Graph
Error (Euclidean distance in pixel space)
Frame Number

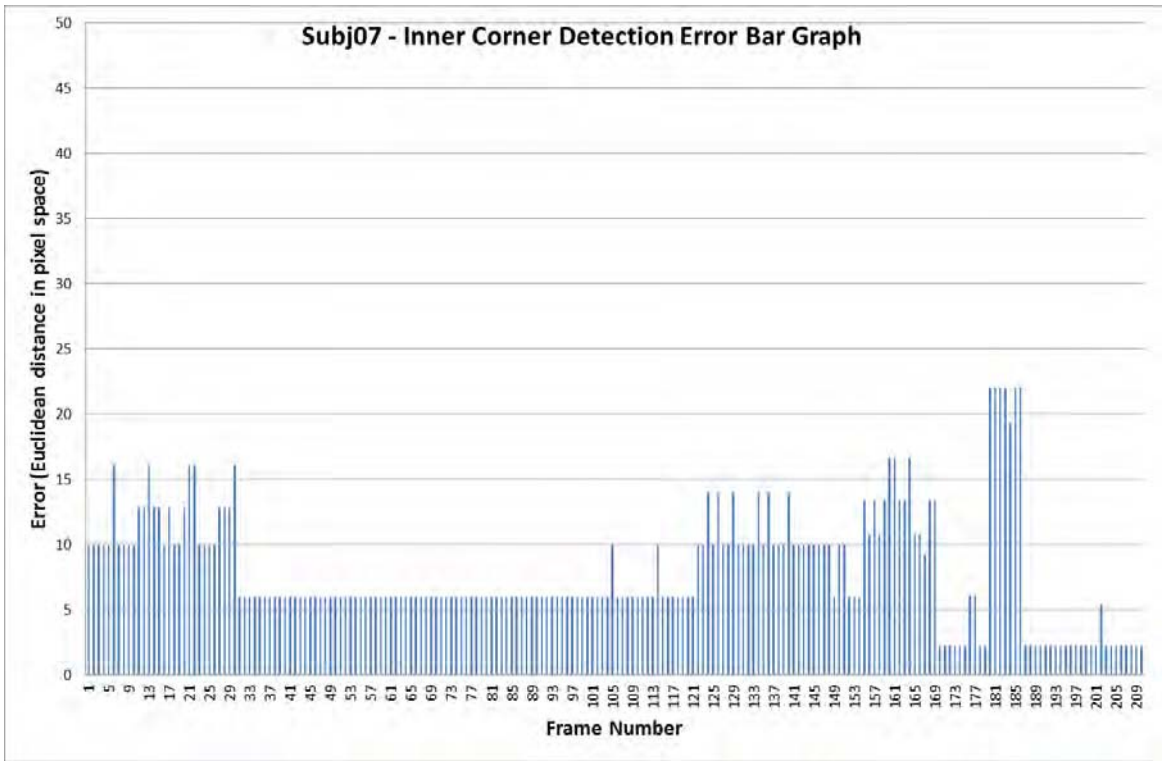

Subj04 - Outer Corner Detection Error Bar Graph
Error (Euclidean distance in pixel space)
Frame Number

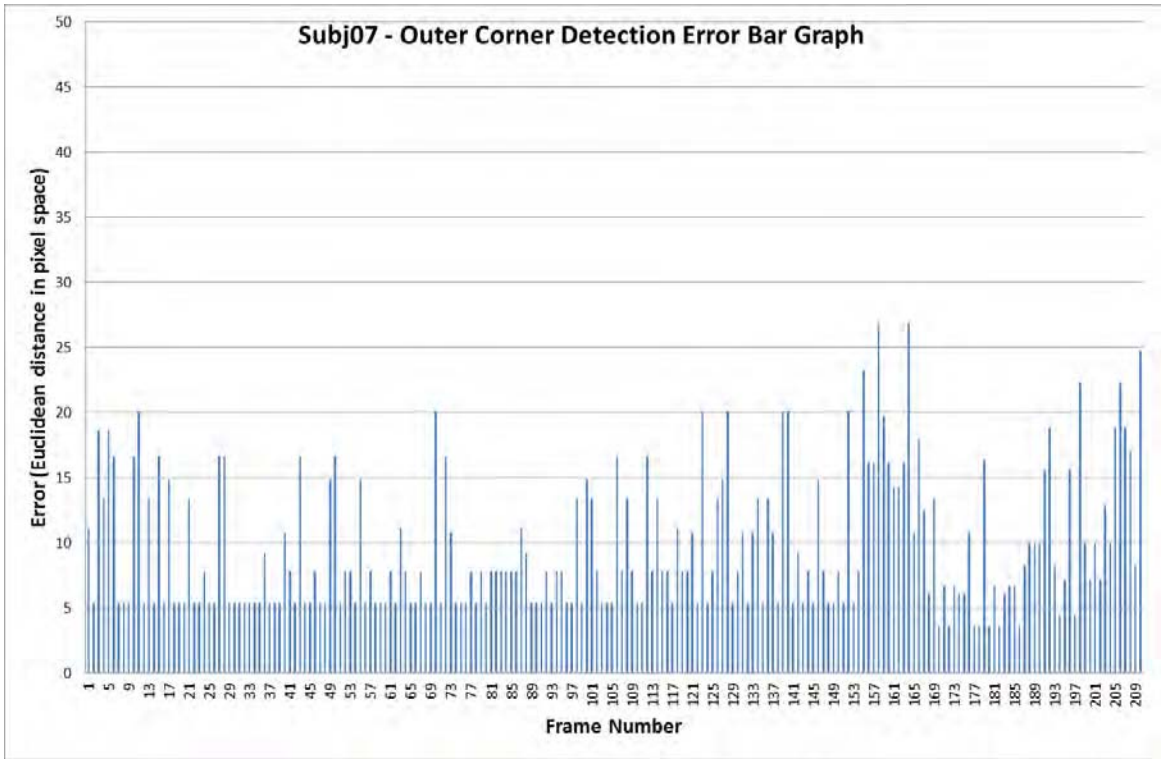

Subj05 - Inner Corner Detection Error Bar Graph
Error (Euclidean distance in pixel space)
Frame Number

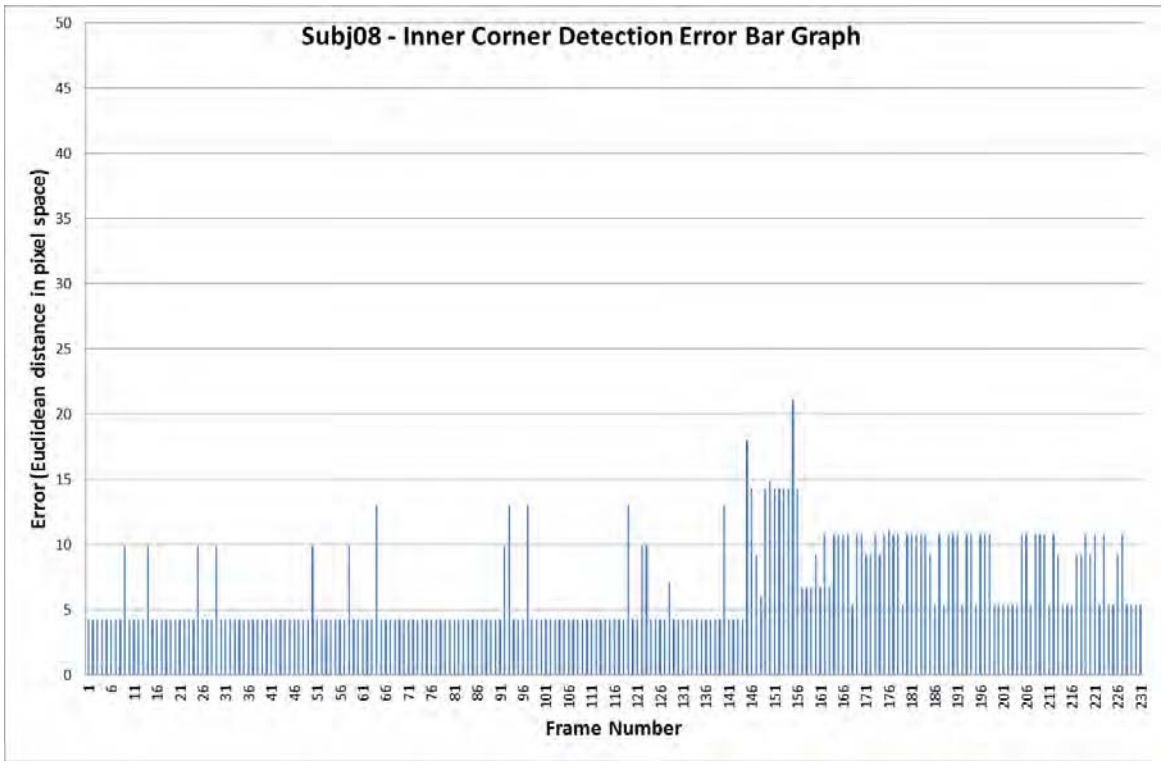

Subj05 - Outer Corner Detection Error Bar Graph
Error (Euclidean distance in pixel space)
Frame Number

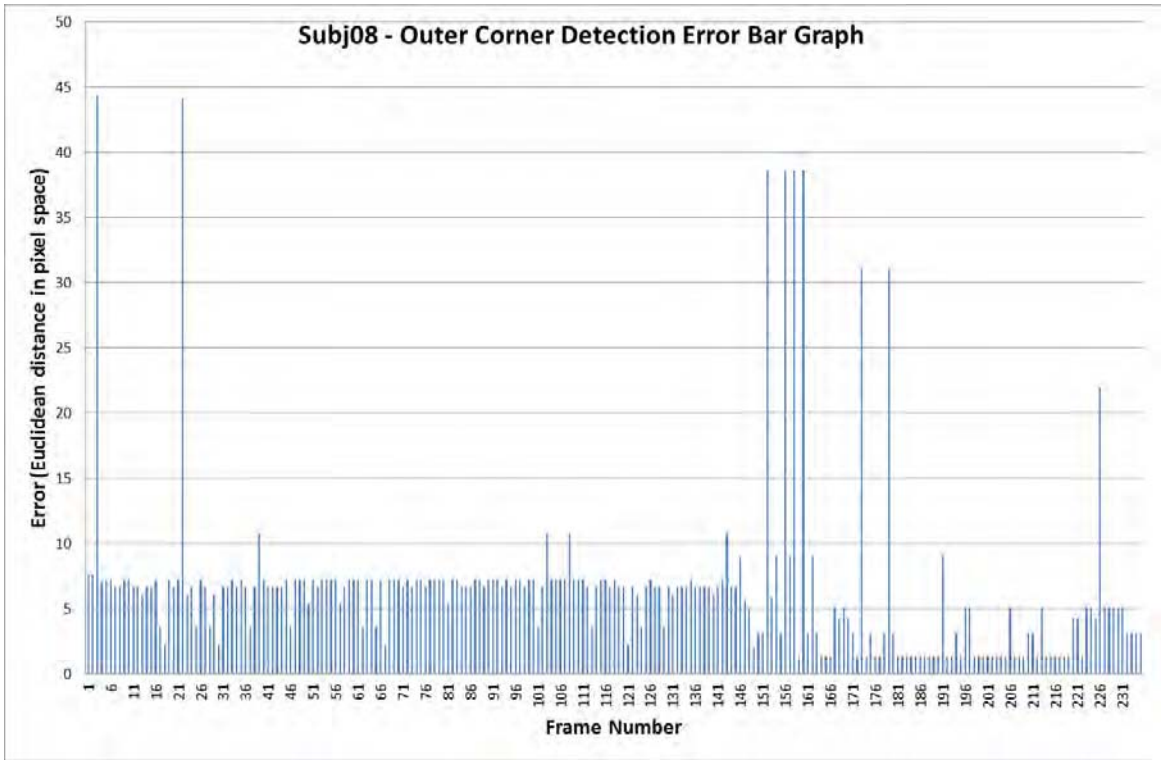

Subj06 - Inner Corner Detection Error Bar Graph
Error (Euclidean distance in pixel space)
Frame Number
0 5 10 15 20 25 30 35 40 45 50
1 4 7 10 13 16 19 22 25 28 31 34 37 40 43 46 49 52 55 58 61 64 67 70 73 76 79 82 85 88 91 94 97 100 103 106 109

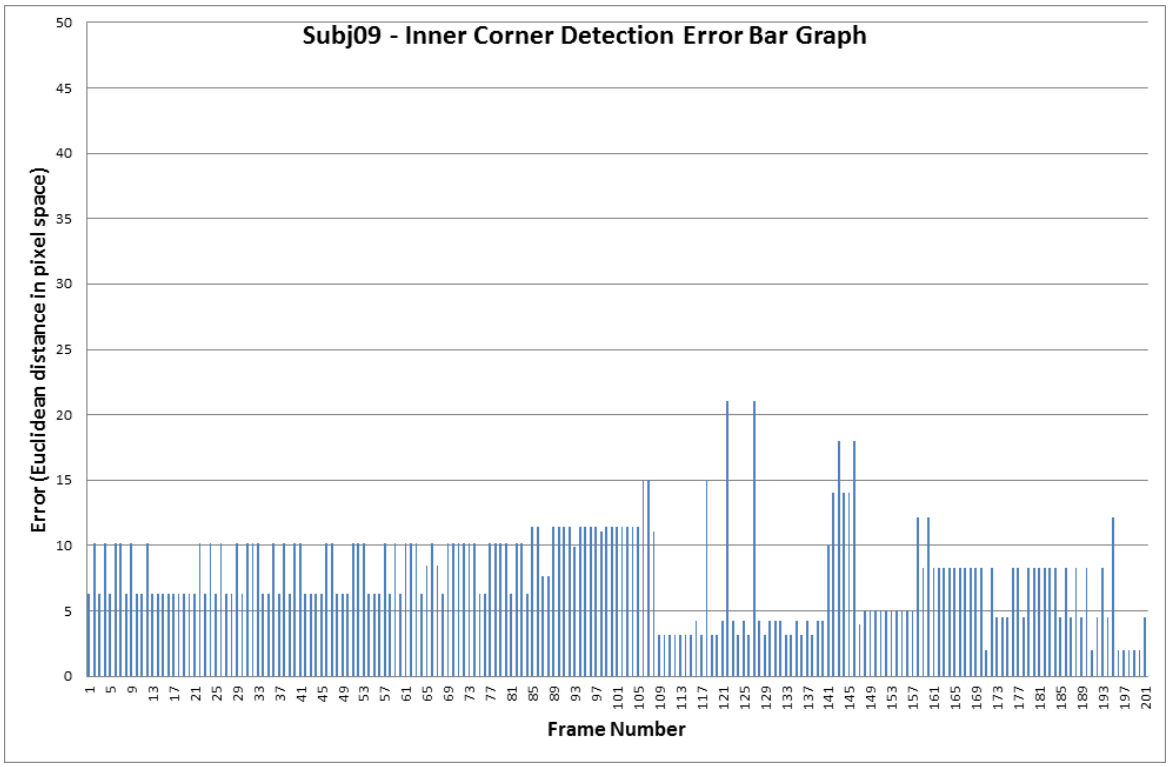

Subj06 - Outer Corner Detection Error Bar Graph
Error (Euclidean distance in pixel space)
Frame Number
0 5 10 15 20 25 30 35 40 45 50
1 4 7 10 13 16 19 22 25 28 31 34 37 40 43 46 49 52 55 58 61 64 67 70 73 76 79 82 85 88 91 94 97 100 103 106 109


Subj07 - Inner Corner Detection Error Bar Graph
Error (Euclidean distance in pixel space)
Frame Number


Subj07 - Outer Corner Detection Error Bar Graph
Error (Euclidean distance in pixel space)
Frame Number


Subj08 - Inner Corner Detection Error Bar Graph
Error (Euclidean distance in pixel space)
Frame Number


Subj08 - Outer Corner Detection Error Bar Graph
Error (Euclidean distance in pixel space)
Frame Number


Subj09 - Inner Corner Detection Error Bar Graph
Error (Euclidean distance in pixel space)
Frame Number

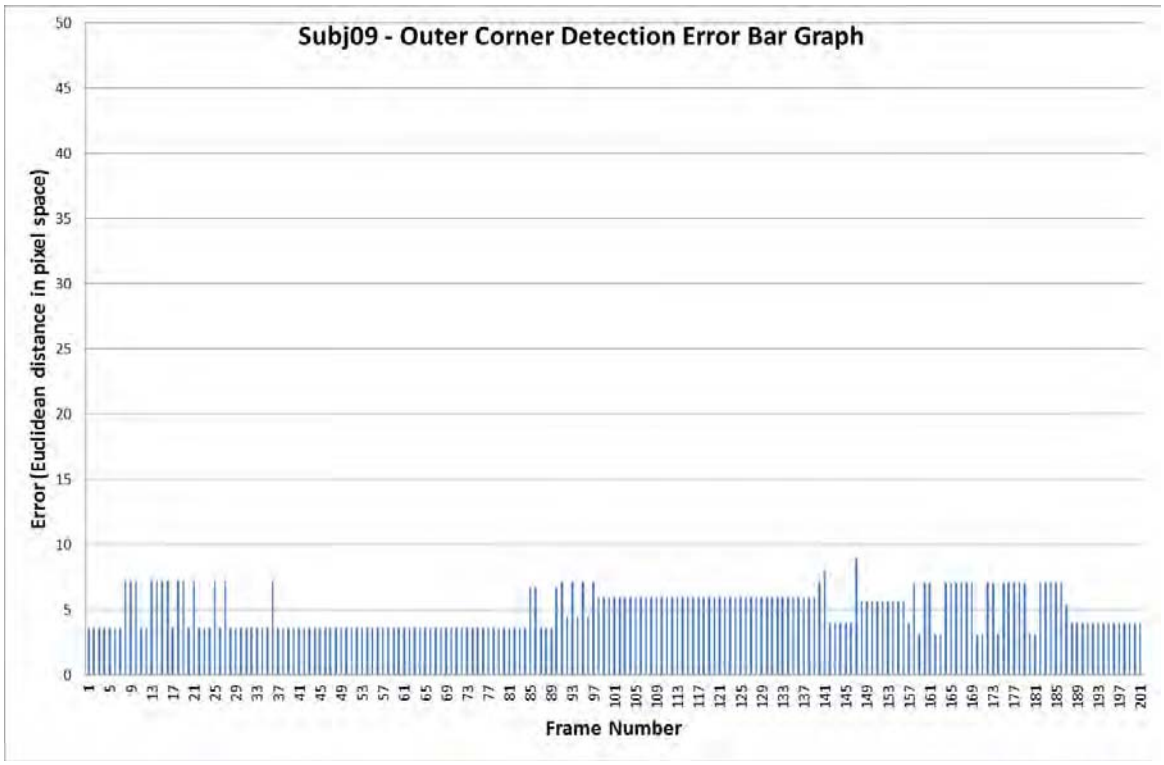
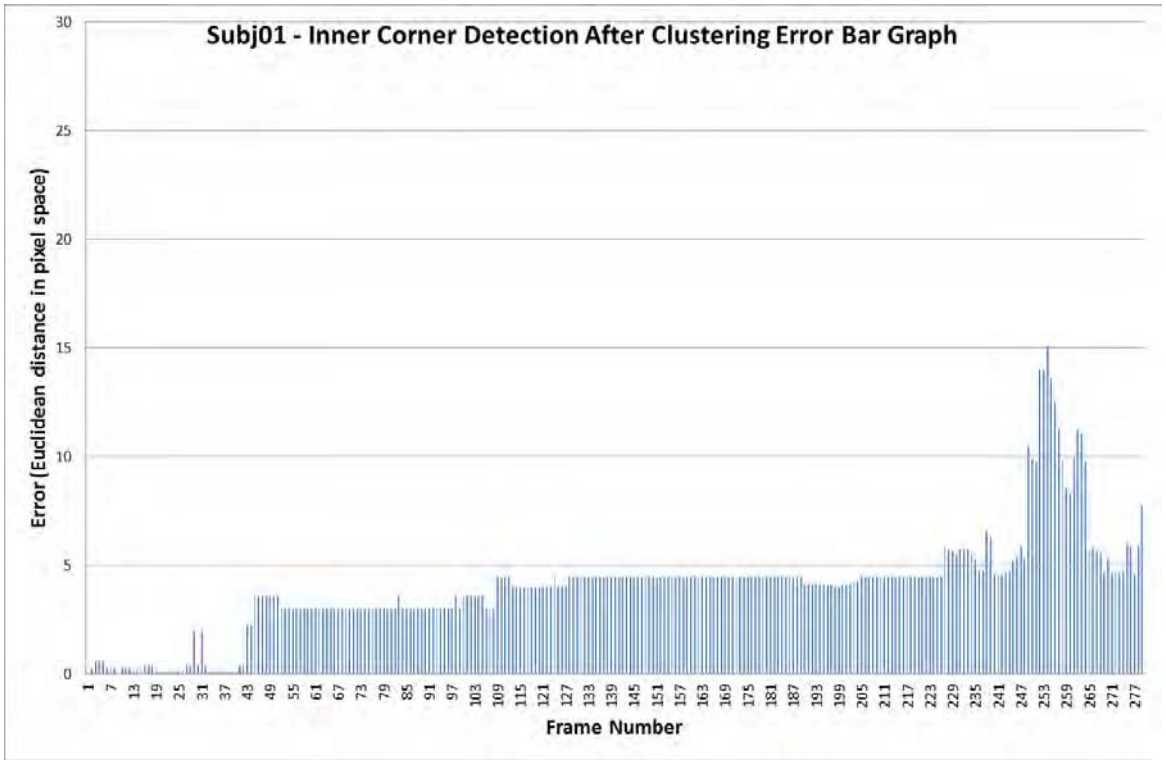
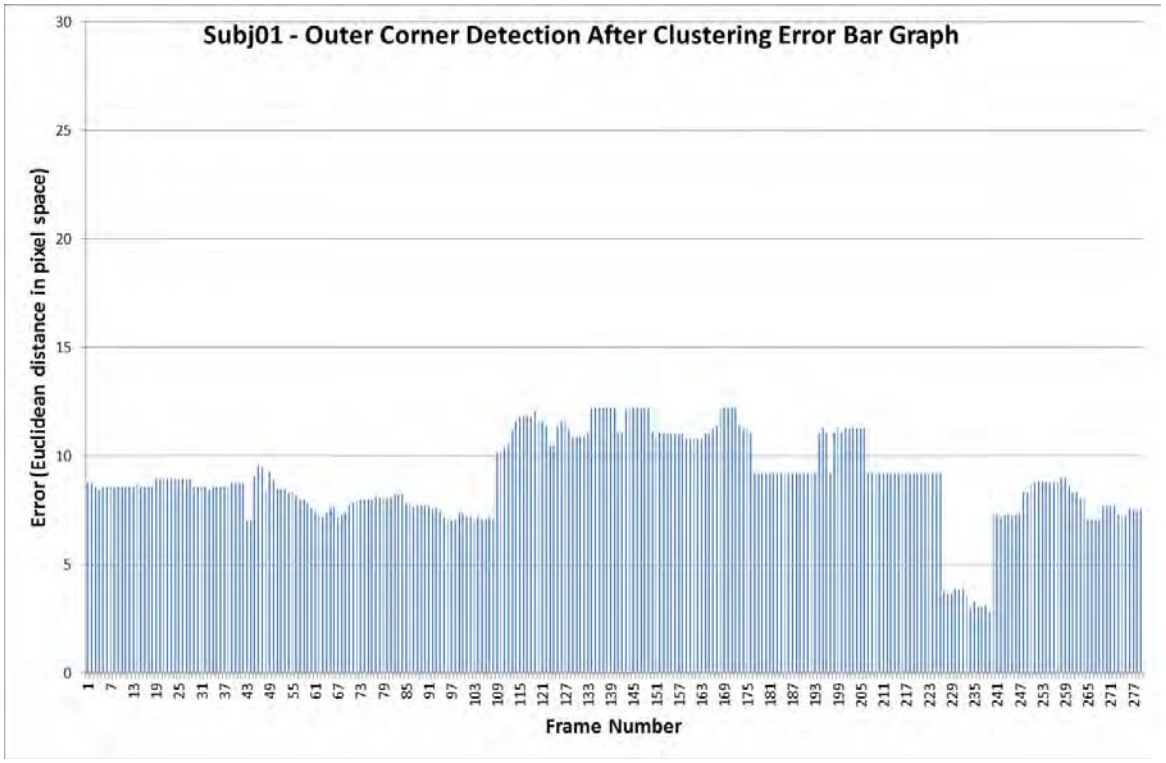
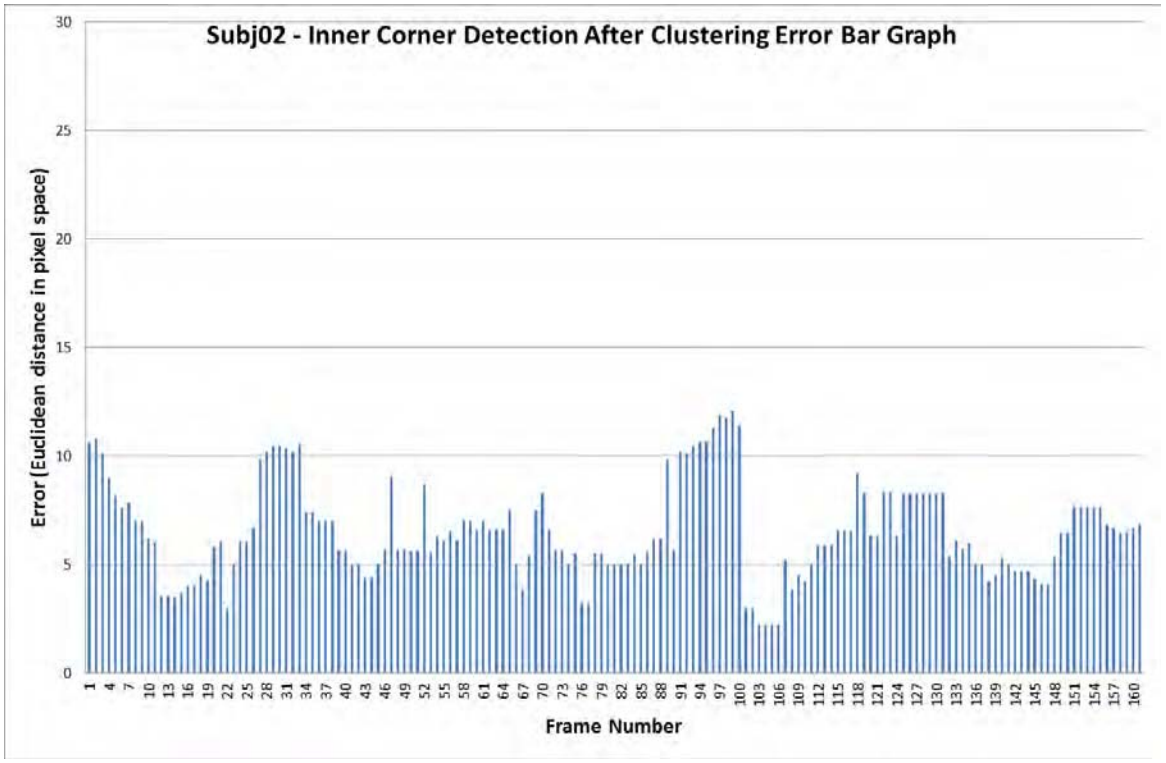
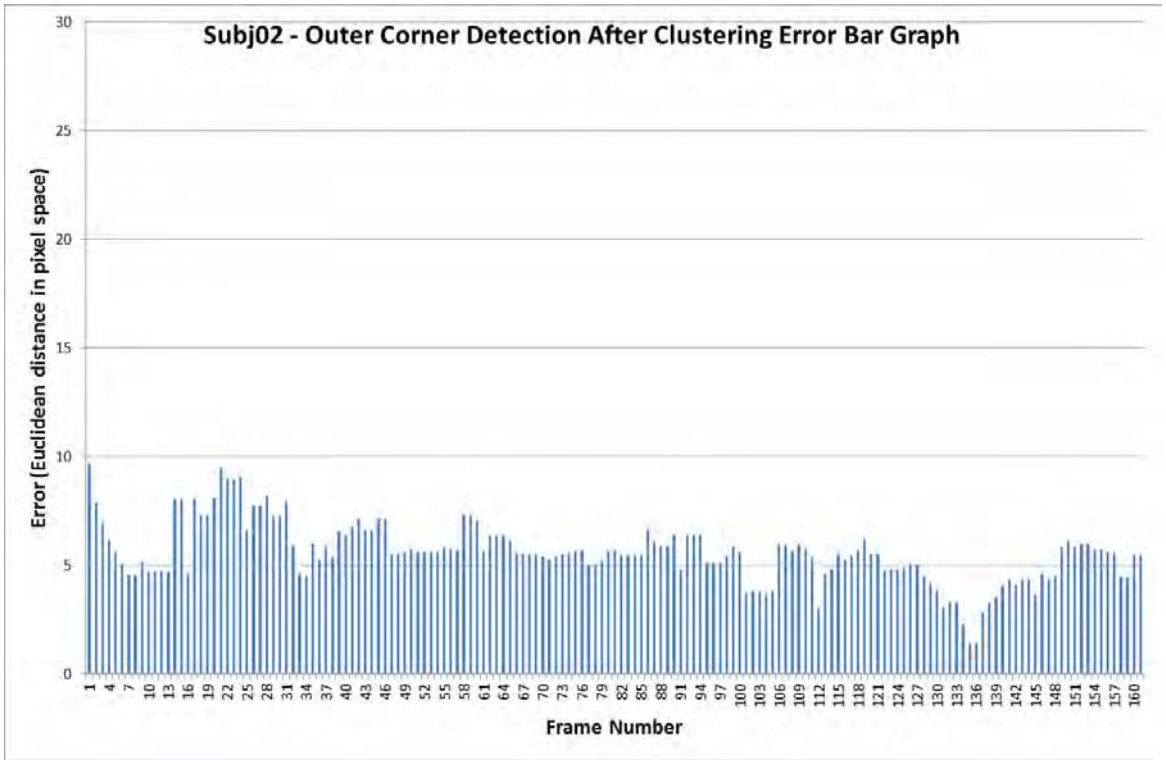

Subj09 - Outer Corner Detection Error Bar Graph
Error (Euclidean distance in pixel space)
Frame Number

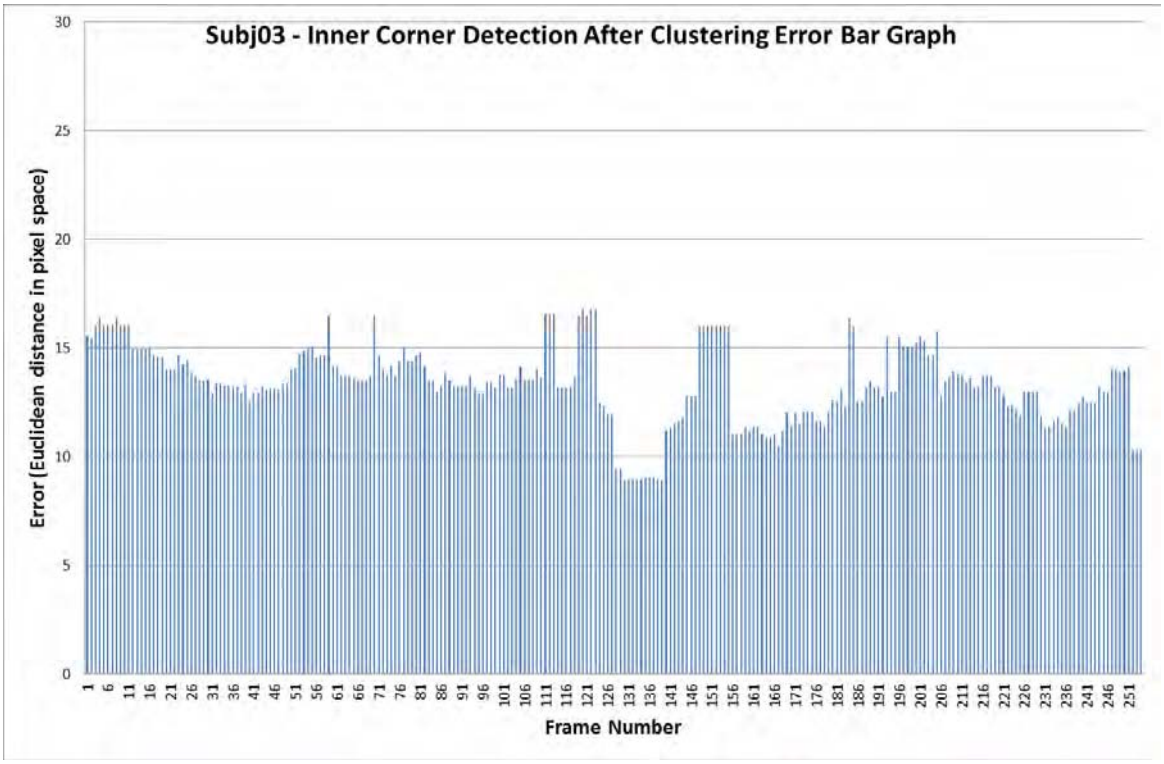
TABLE 25: BAR GRAPHS OF THE CORNER DETECTION ALGORITHM ERROR AFTER CLUSTERING (INNER AND OUTER CORNER) FOR EACH TEST SEQUENCE.

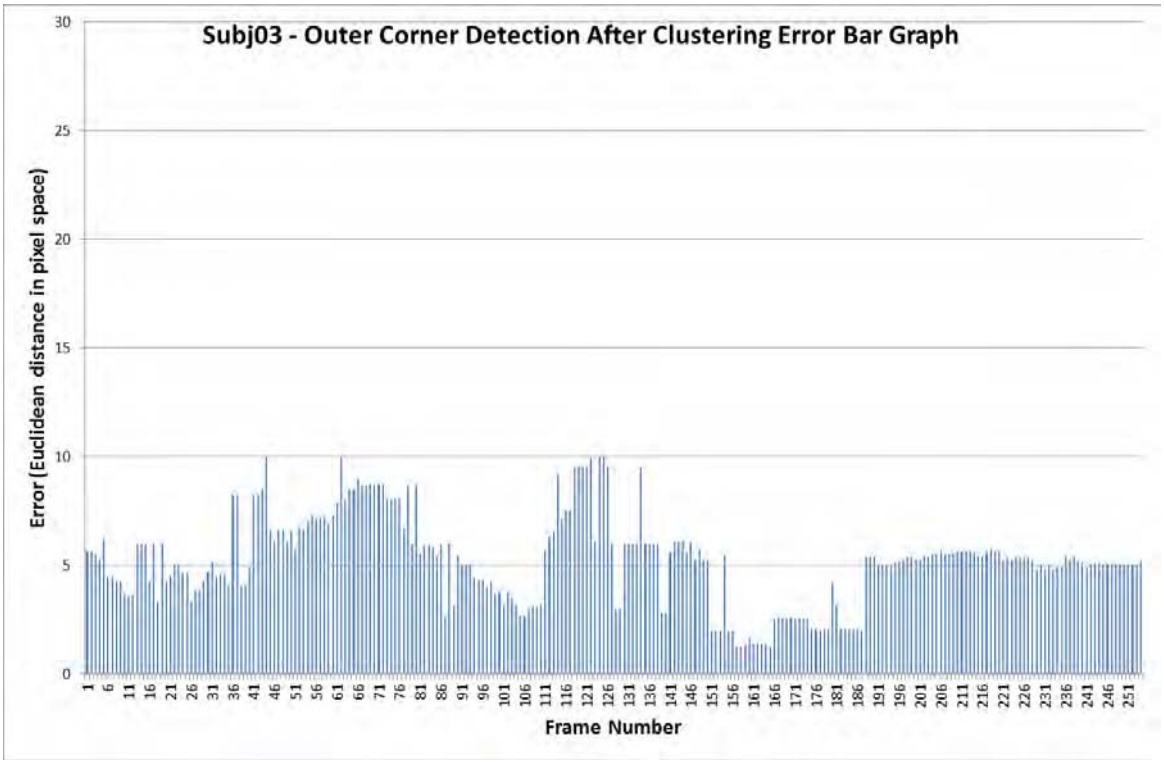


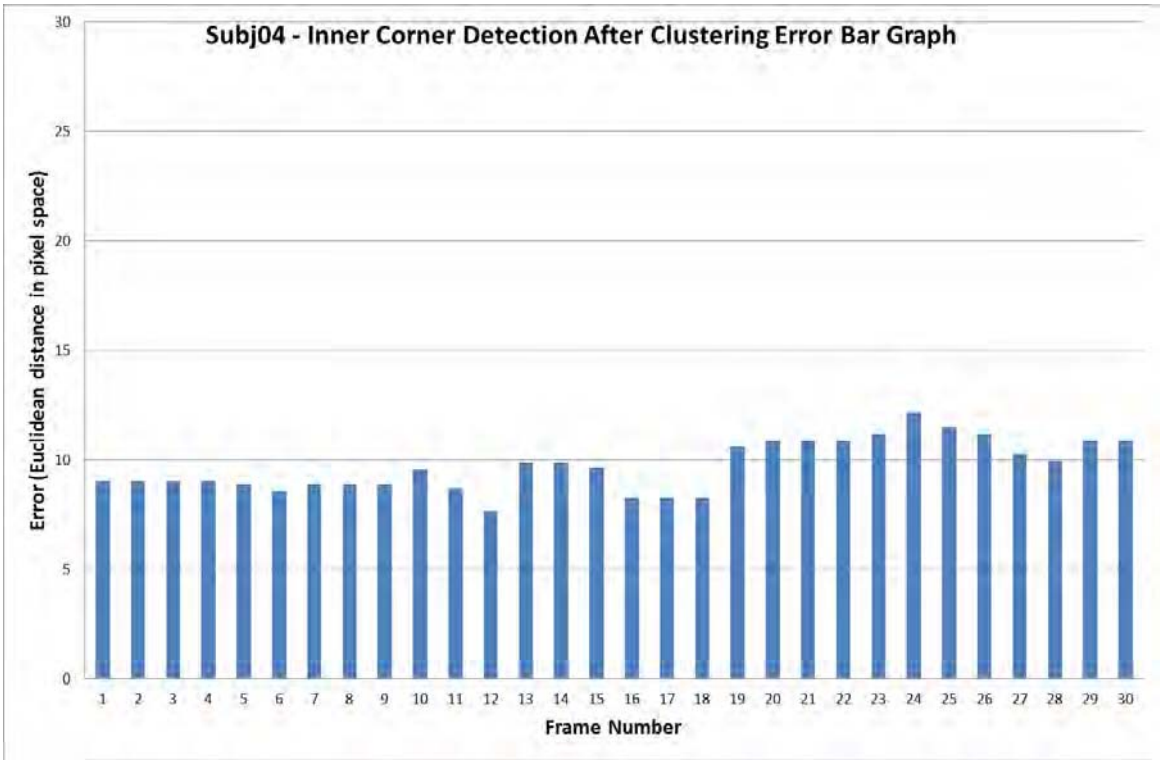


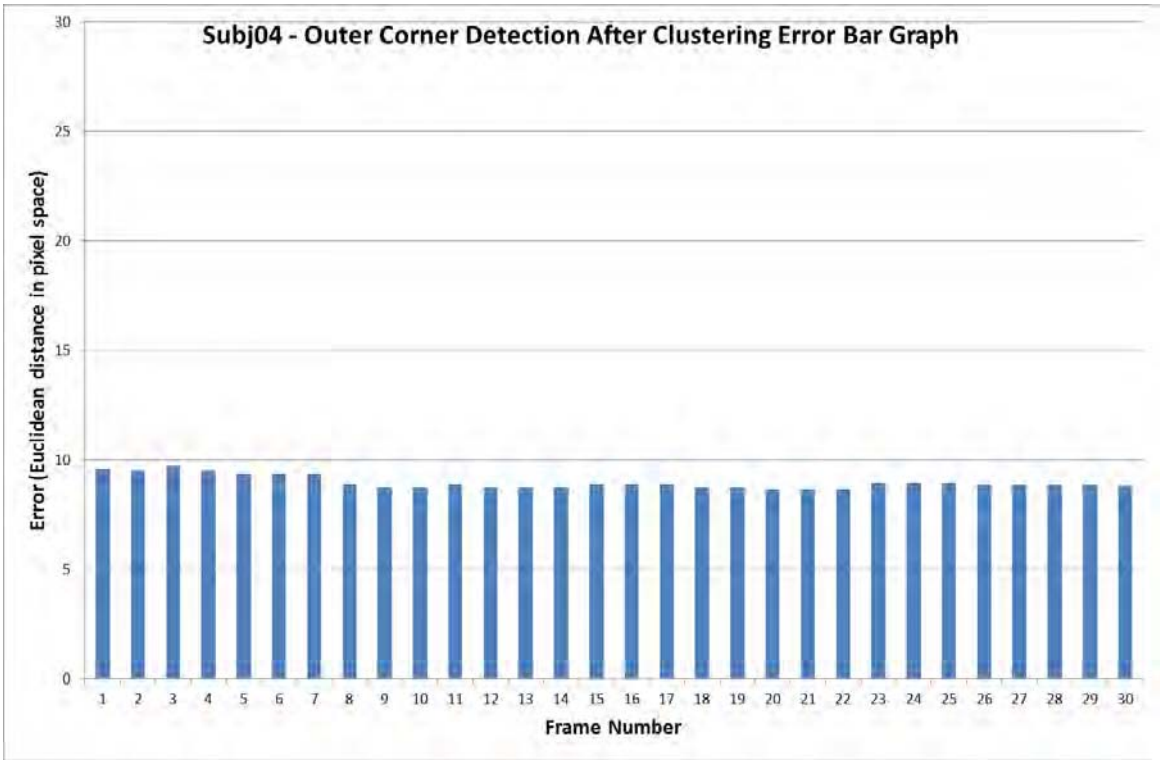

Subj02 - Inner Corner Detection After Clustering Error Bar Graph
Error (Euclidean distance in pixel space)
Frame Number

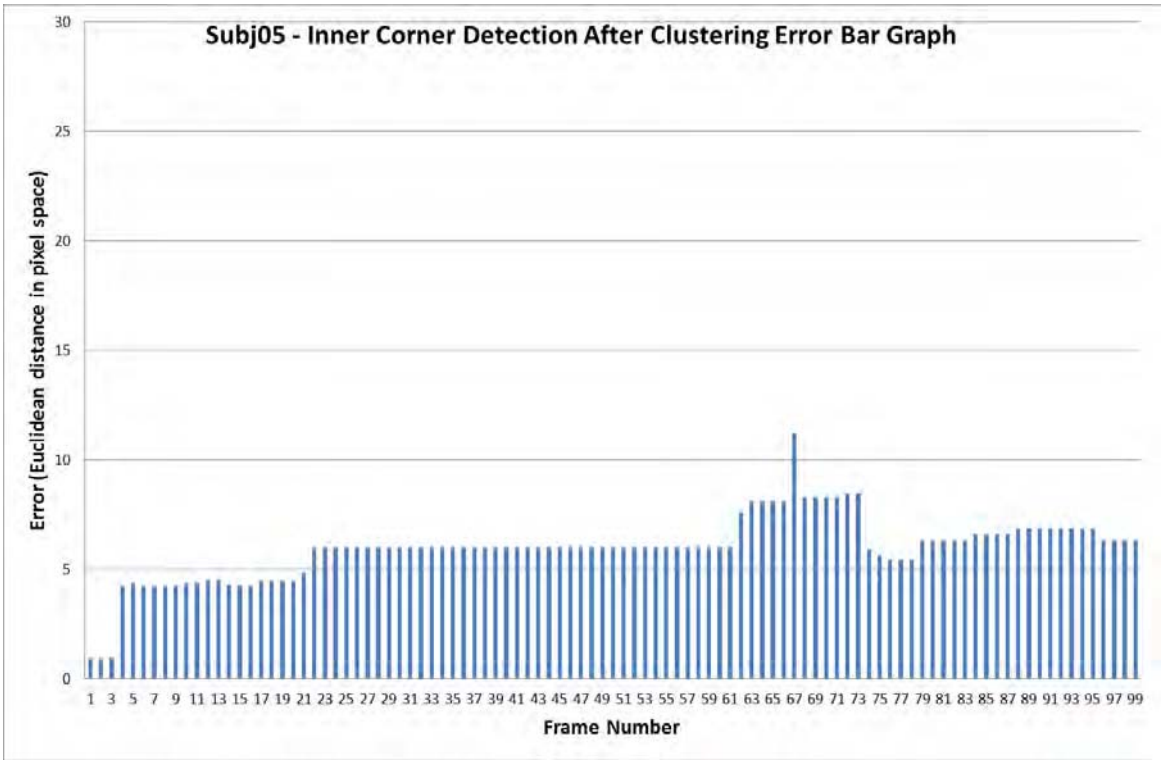

Subj02 - Outer Corner Detection After Clustering Error Bar Graph
Error (Euclidean distance in pixel space)
Frame Number

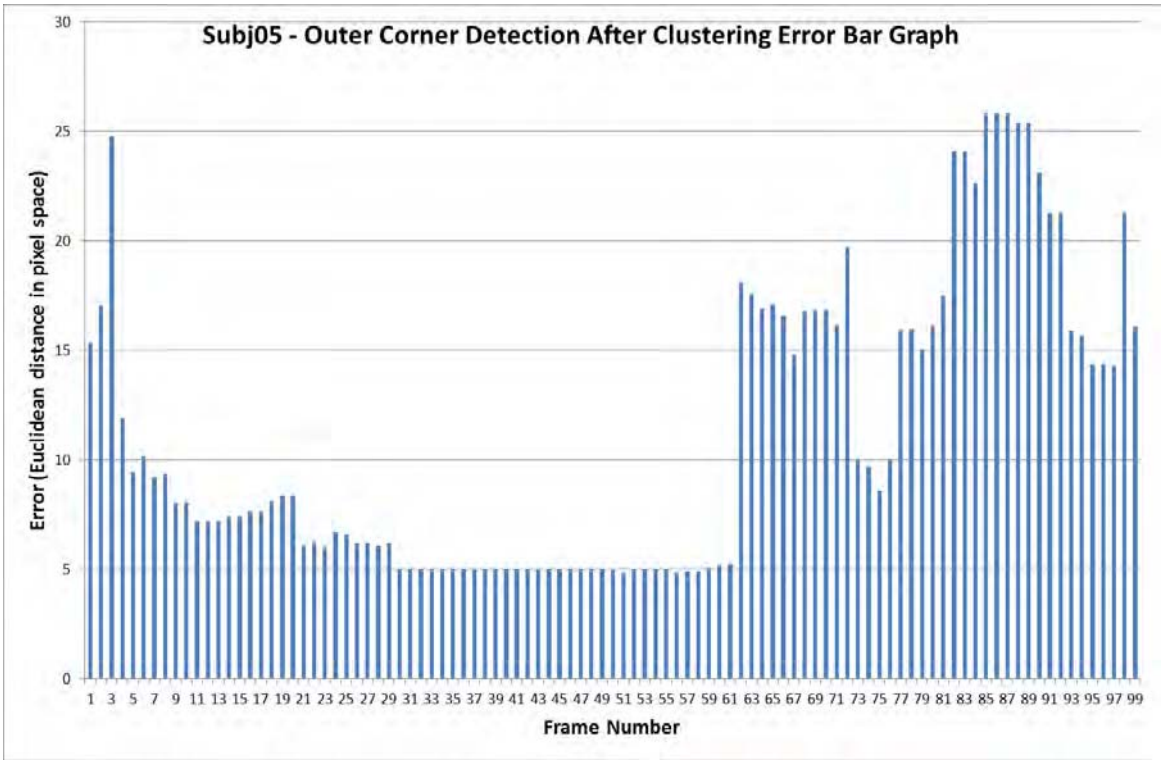

Subj03 - Inner Corner Detection After Clustering Error Bar Graph
Error (Euclidean distance in pixel space)
Frame Number

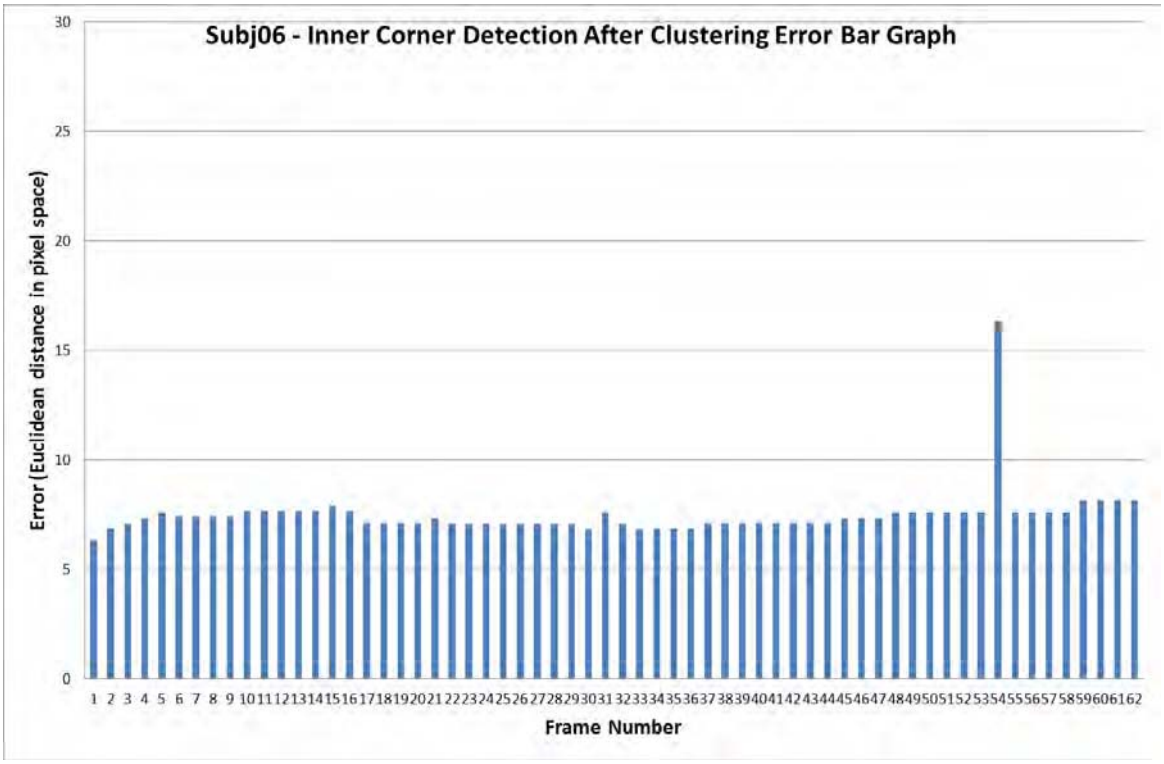

Subj03 - Outer Corner Detection After Clustering Error Bar Graph
Error (Euclidean distance in pixel space)
Frame Number

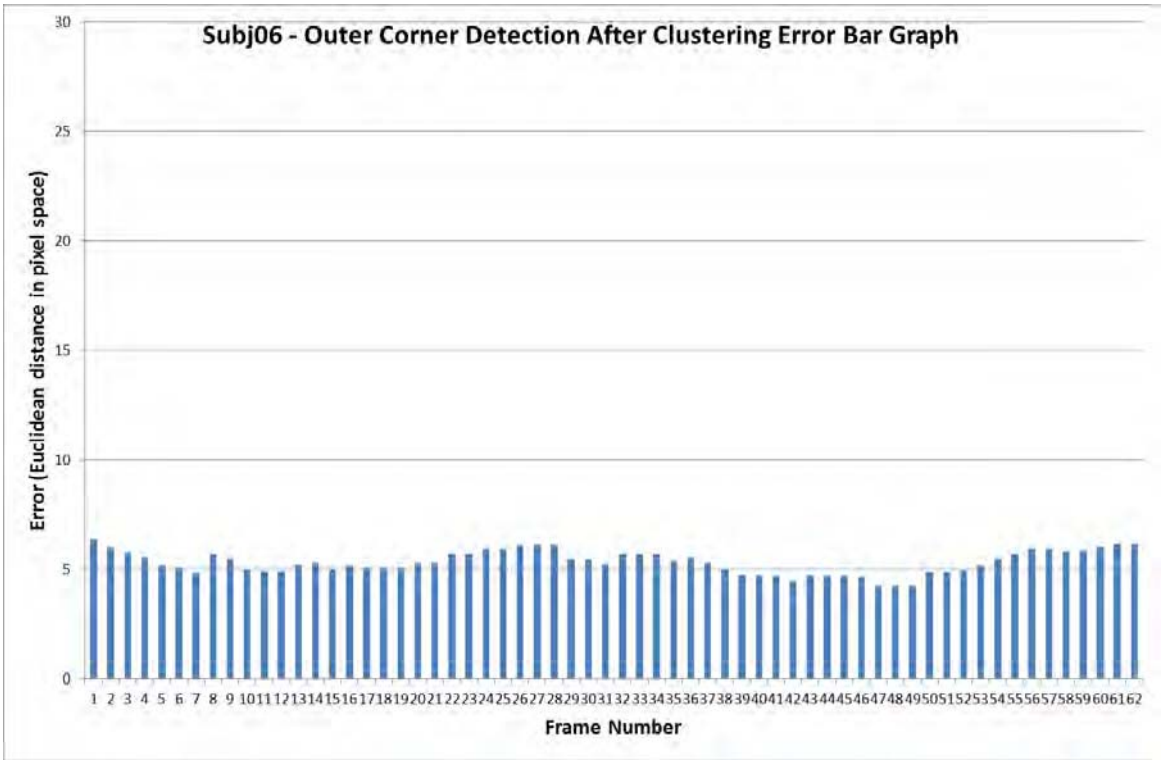

Subj04 - Inner Corner Detection After Clustering Error Bar Graph
Error (Euclidean distance in pixel space)
0
5
10
15
20
25
30
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30
Frame Number

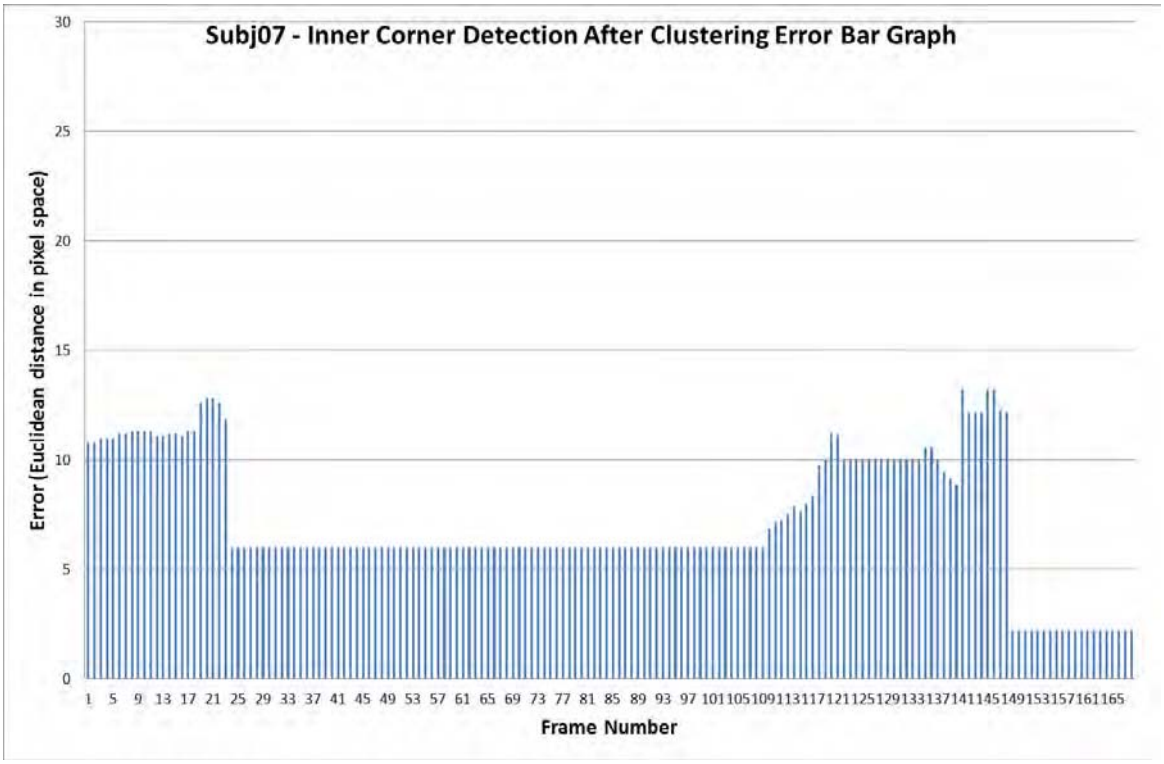

Subj04 - Outer Corner Detection After Clustering Error Bar Graph
Error (Euclidean distance in pixel space)
0
5
10
15
20
25
30
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30
Frame Number

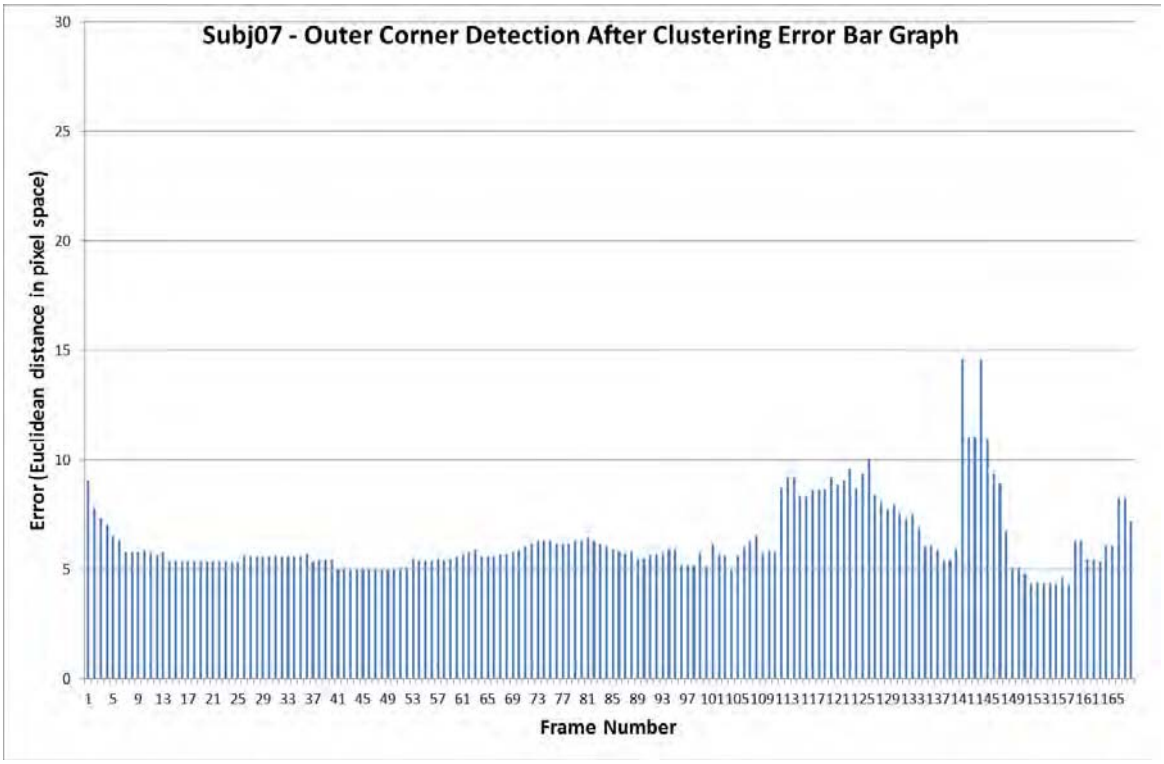

Subj05 - Inner Corner Detection After Clustering Error Bar Graph
Error (Euclidean distance in pixel space)
Frame Number

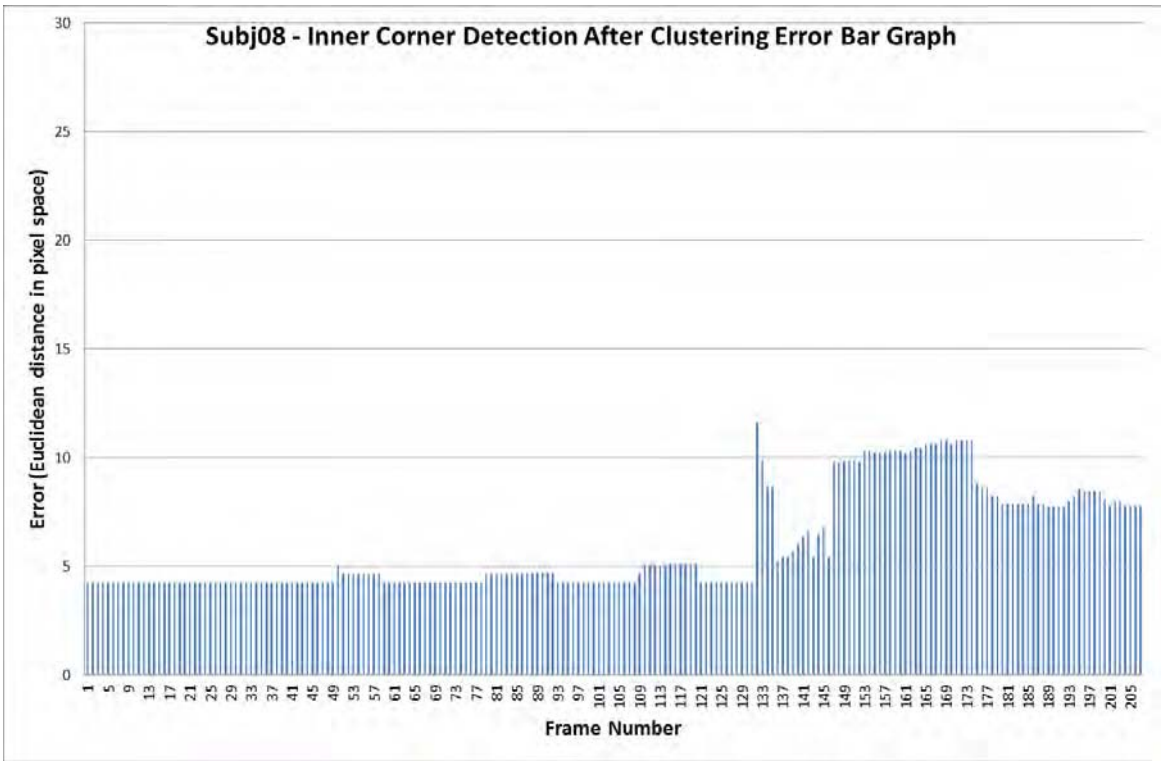

Subj05 - Outer Corner Detection After Clustering Error Bar Graph
Error (Euclidean distance in pixel space)
Frame Number

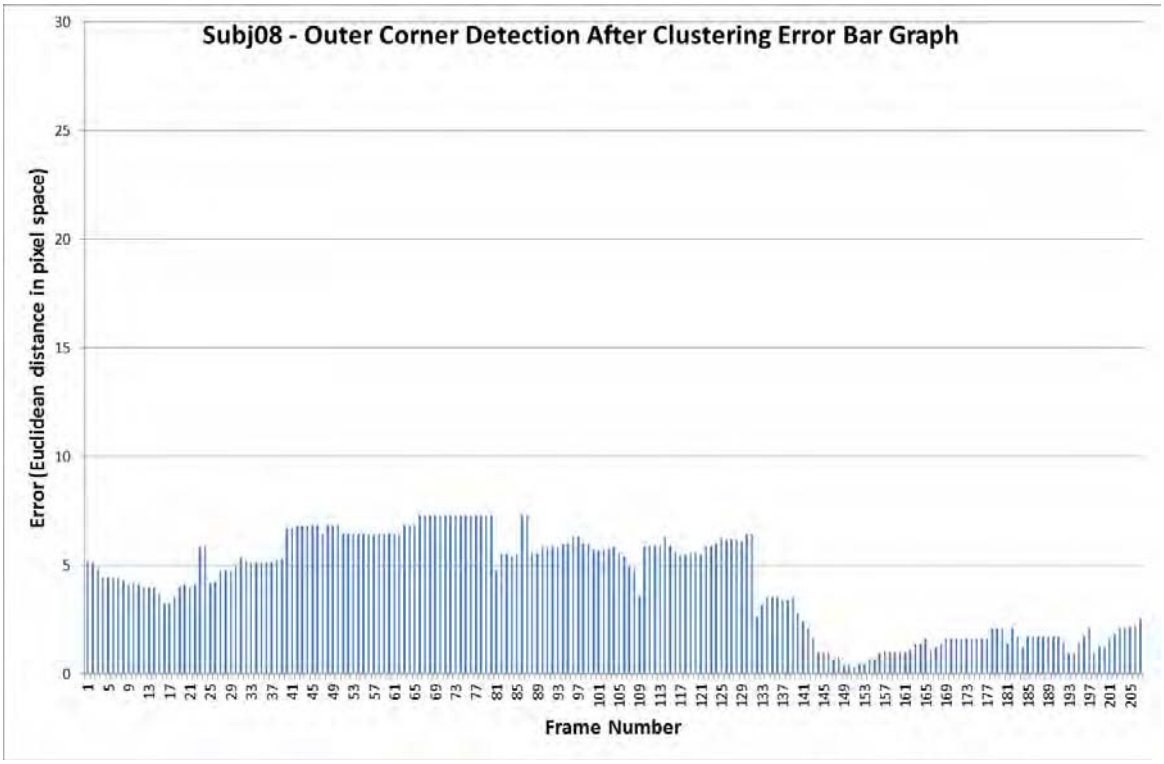

Subj06 - Inner Corner Detection After Clustering Error Bar Graph
Error (Euclidean distance in pixel space)
Frame Number

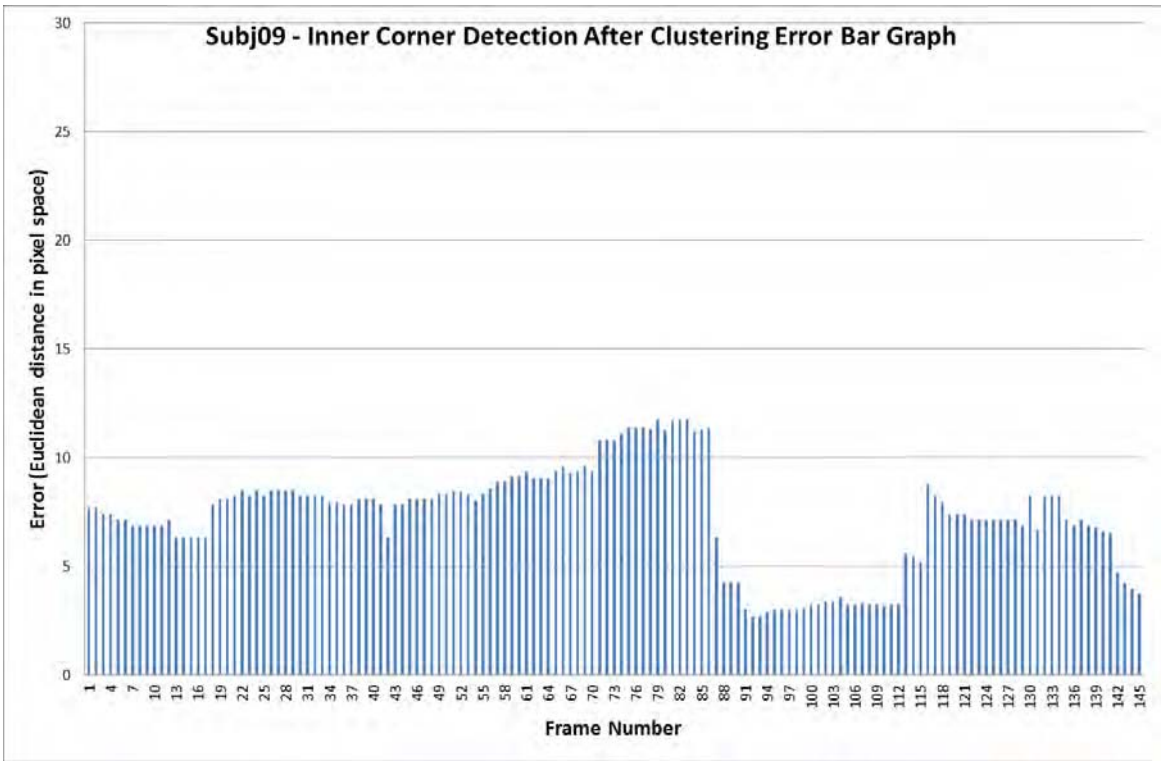

Subj06 - Outer Corner Detection After Clustering Error Bar Graph
Error (Euclidean distance in pixel space)
Frame Number

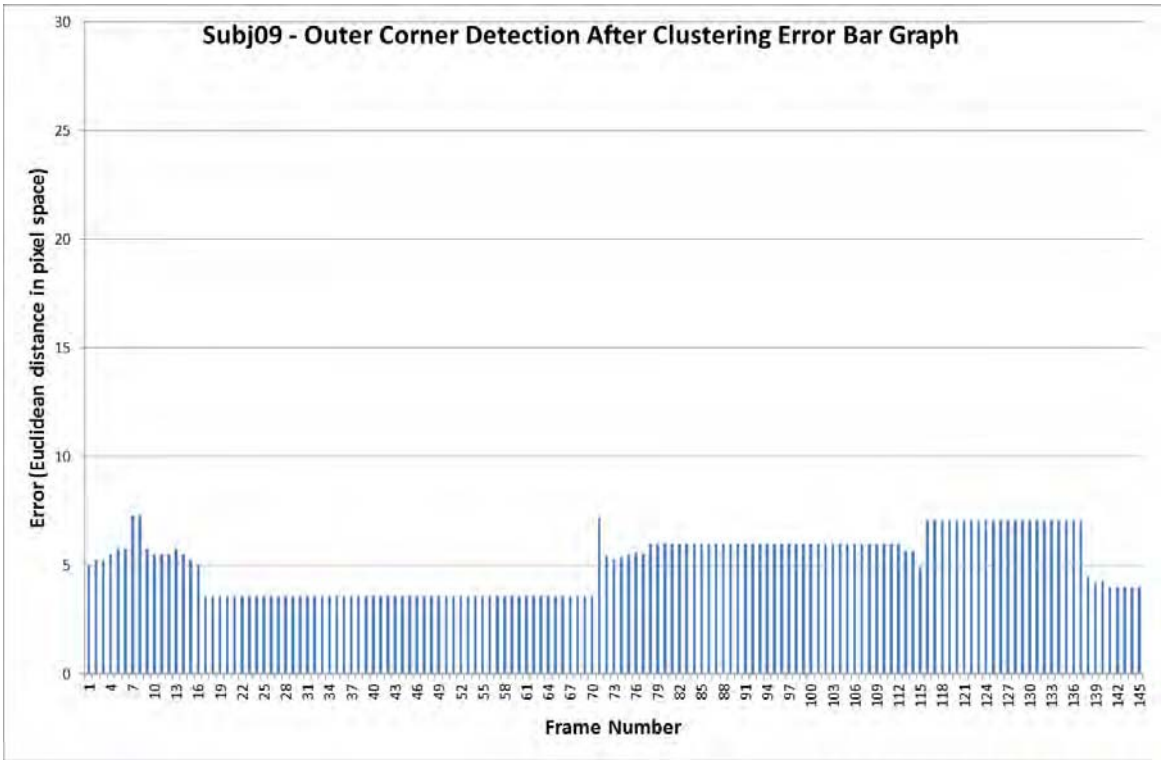

Subj07 - Inner Corner Detection After Clustering Error Bar Graph
Error (Euclidean distance in pixel space)
Frame Number


Subj07 - Outer Corner Detection After Clustering Error Bar Graph
Error (Euclidean distance in pixel space)
Frame Number


Subj08 - Inner Corner Detection After Clustering Error Bar Graph
Error (Euclidean distance in pixel space)
Frame Number


Subj08 - Outer Corner Detection After Clustering Error Bar Graph
Error (Euclidean distance in pixel space)
Frame Number


Subj09 - Inner Corner Detection After Clustering Error Bar Graph
Error (Euclidean distance in pixel space)
Frame Number


Subj09 - Outer Corner Detection After Clustering Error Bar Graph
Error (Euclidean distance in pixel space)
Frame Number

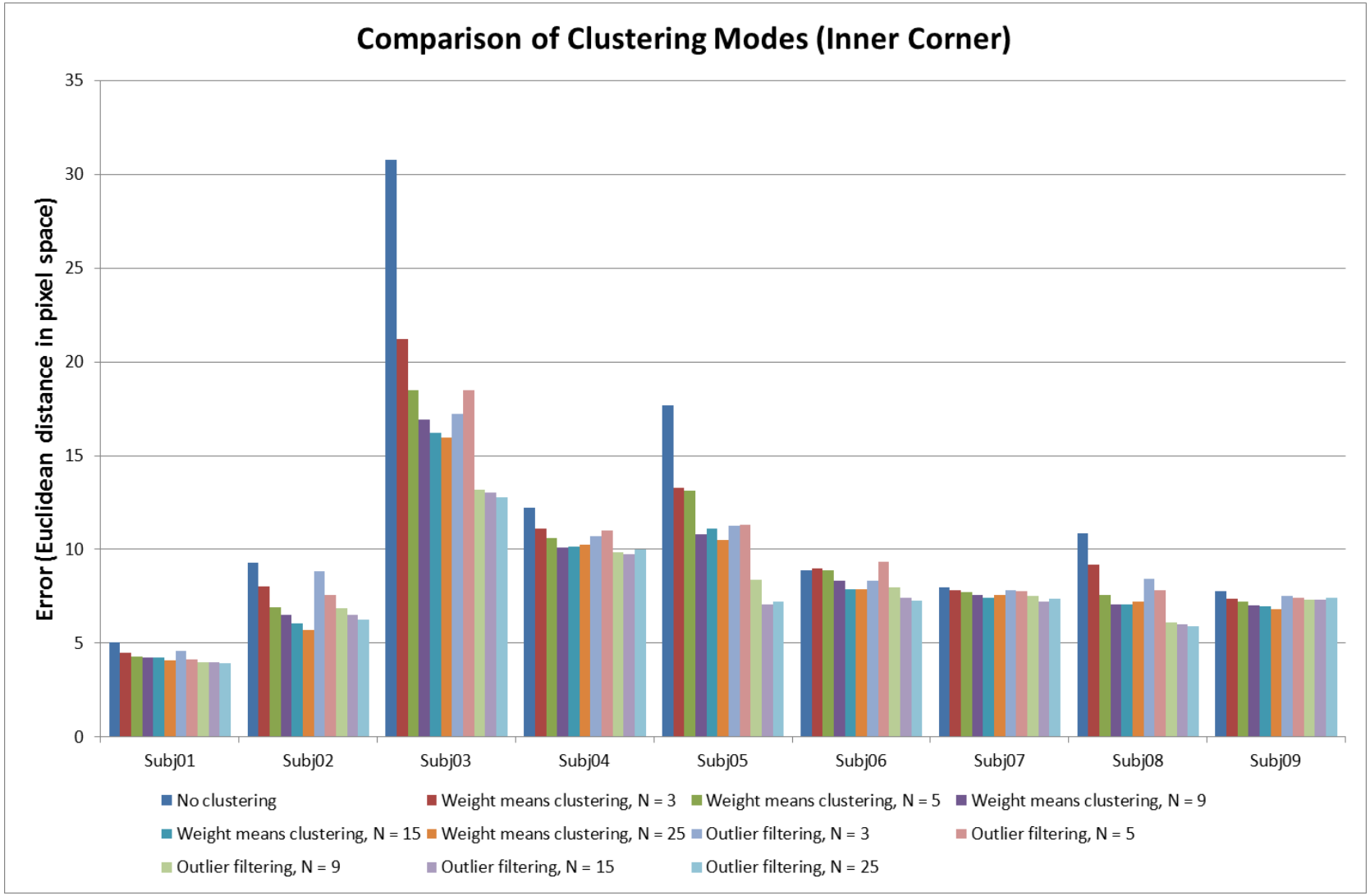


FIGURE 32: COMPARISON OF CLUSTERING MODES AND SETTINGS (INNER CORNER).

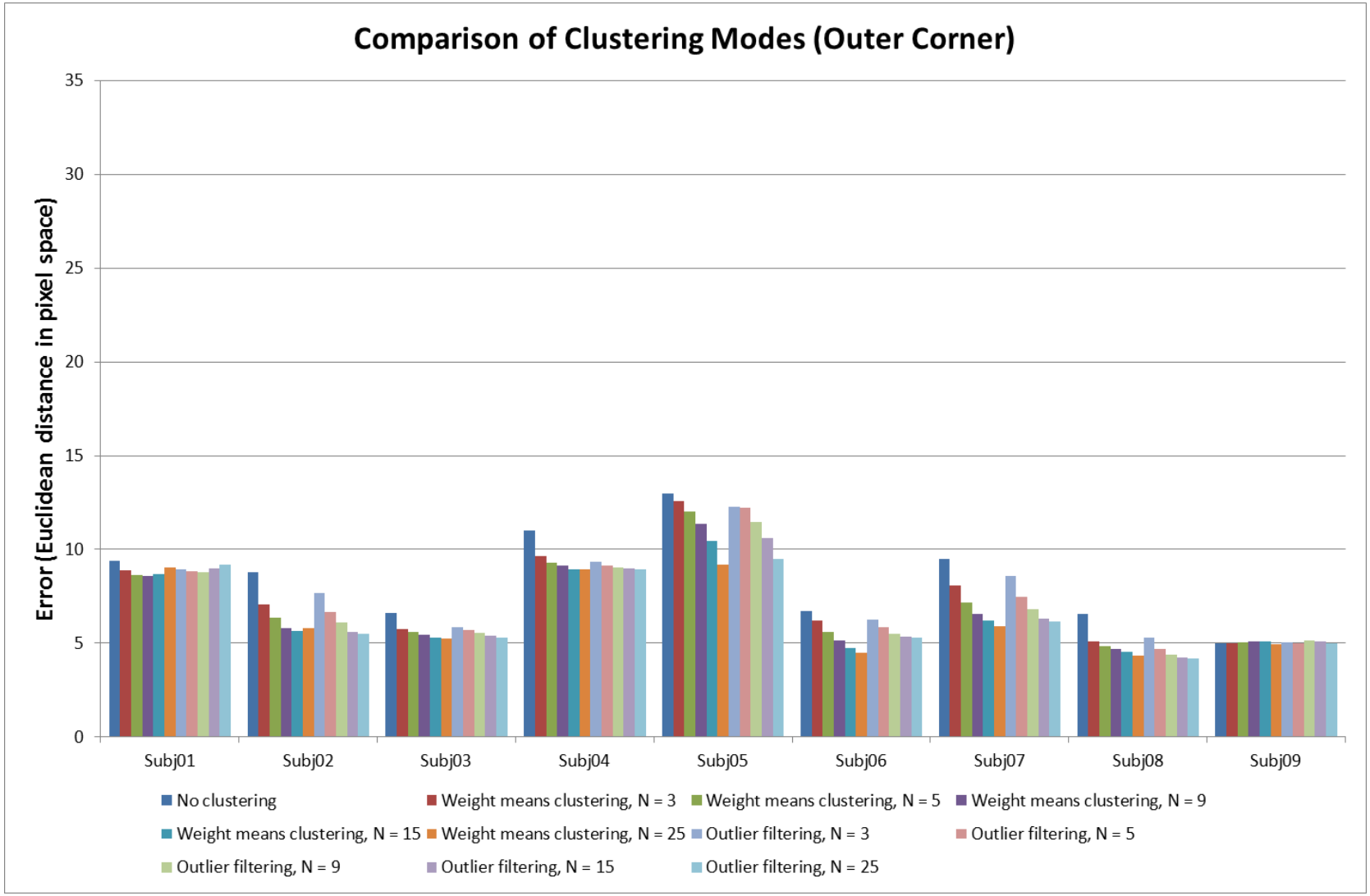


FIGURE 33: COMPARISON OF CLUSTERING MODES AND SETTINGS (OUTER CORNER).

EVALUATION OF THE 2D GAZE VECTOR CALCULATION ALGORITHM

The evaluation of the 2D gaze vector calculation algorithm is essentially an evaluation of the eye-tracker as a whole. In the previous subsections, the individual components of the algorithm were isolated and individually assessed without requiring input from any other components. For example, the iris boundary detection algorithm requires as input the pupil centre location and pupil contour; during the evaluation of the algorithm, this input was derived from the manually marked data such that the error assessed is purely based on the iris detection algorithm and not of the pupil detection and iris boundary detection algorithms in cooperation.

In addition, the actual calculation of the 2D gaze vector from its inputs (initial pupil centre, initial pupil contour and eye corner locations) is performed in exactly the same manner both in the tracker and the manually marked data set. Thus, the accuracy of the 2D gaze vector calculation depends solely on the accuracy of its individual inputs. Having considered this, the error measured in this evaluation is the difference between the angle calculated from the set of eye features extracted by the tracker and the angle calculated from the set of eye features that were manually marked.

The table below illustrates the average, standard deviation and maximum error values (degrees) for the complete test data set; for this test, the twelve (12) different positions recorded for each subject were used.

On average, the gaze angle error is 2.78° with standard deviation 1.99°, which is a range that renders the eye-tracker practical for the target applications. In the worst case, the error reached a total maximum or 9.51°. It is important here to consider that a) the 2D gaze angle is affected both by errors in the pupil detection and errors in eye corner detection b) errors in manual marking of the pupil and eye corners as explained earlier will have also affected these measurements c) like any eye-tracker which requires subject calibration, the accuracy of the output is directly proportional to the accuracy of the calibration.

A better benchmark may have been to measure the error after the 2D gaze angle has been classified into several distinct classes; this benchmark is offered as part of the case study, in Chapter 6. Further, a comparison of the trackable range and accuracy between the REACT eye-tracker and the SR Research EyeLink-II is presented in the next section of this chapter. While

these systems were designed for dissimilar applications, this evaluation is offered as a comparative benchmark if the REACT eye-tracker were extended to track gaze on a screen. The REACT eye-tracker is the first eye-tracker that a) is specifically designed for tracking extreme, usually non-visual, eye-movements and b) is head-mounted but uses the eye corners as reference points instead of the glint.

TABLE 26: 2D GAZE VECTOR CALCULATION EVALUATION - ERROR STATISTICS, IN DEGREES.

Sample	Frames	Average	σ	Maximum
Subj01	12	2.20	0.56	2.93
Subj02	12	2.39	1.30	4.74
Subj03	12	4.01	2.24	7.76
Subj04	12	1.95	1.41	4.72
Subj05	12	3.51	2.27	9.51
Subj06	12	2.98	2.66	9.12
Subj07	12	2.26	2.16	7.93
Subj08	12	2.66	1.40	4.50
Subj09	12	3.03	2.04	7.25
Overall	**108**	**2.78**	**1.99**	**9.51**

COMPARISON OF THE REACT EYE-TRACKER WITH SR RESEARCH EYELINK-II

For the evaluation of the eye-tracker to be complete, it is necessary to know how it performs compared to other systems and whether it satisfies the requirements identified earlier in this thesis. To this end, the REACT eye-tracker was compared to a commercial eye-tracker commonly used in psychology studies (e.g. Altmann and Kamide, 2007), the EyeLink-II by SR Research. The comparison was geared towards comparing the tracking ability and accuracy of the two eye-trackers over the full range of eye-movements possible.

The EyeLink-II has been primarily designed to track the subject's gaze onto a screen and an add-on is available to enable gaze tracking with a scene camera. On the other hand, as reviewed in this thesis, the REACT eye-tracker has been designed to track the subject's direction of gaze irrelevant to the world as it is concerned with non-visual eye-movements when the subject is not

focused on a visual target within the environment. Because of the different target applications, for this comparison to take place, the REACT eye-tracker had to be extended and a gaze-mapping scheme needed to be implemented that could map vectors from the image to locations on a screen.

The homographic mapping and calibration scheme presented by Li *et al.* (2005) was implemented in order to provide this mapping. Similar to EyeLink, at the beginning of each experiment, nine (9) points are displayed on the screen (centre, corners and mid-points of each side). When the subject has fixated at each point (synchronized with a key press), the pupil position in the image is detected and the vector from the reference point to the pupil position is calculated and recorded. In contrast to the eye-tracker by Li *et al.* (2005), where the reference point is the corneal reflection, the REACT eye-tracker uses the middle point of the line that connects the inner and outer eye corner as reference. In brief, the mapping H is a 3x3 matrix that has eight degrees of freedom and is calculated by generating a constraint matrix using measured point correspondences and determining the null space of the constraint matrix through singular value decomposition; for further details the interested reader is referred to Li *et al.* (2005).

The EyeLink is able to operate in two modes:

a) Pupil-only mode. In this mode, only the pupil is tracked and its location in the image is mapped to the screen coordinates. According to the EyeLink specification (SR Research, 2009) the tracker is able to operate within a range of ±30° horizontally, ±20° vertically and 0.5° of accuracy.
b) Pupil and corneal-reflection (PCR) mode. In this mode, both the pupil and the corneal reflection are tracked and the vector between the two features is used to map gaze onto the screen. By using the corneal reflection as the reference point, the EyeLink can prevent the introduction of tracking errors from the slippage of its heavy headband (approx. 420gr; SR Research, 2009). However, using the corneal reflection or glint also makes it vulnerable to loss of tracking when the corneal falls onto the sclera and is particularly hard to detect, as explained in Chapter 3. In PCR mode and with head-tracking enabled, the EyeLink specification indicates a conservative tracking range of ±20° horizontally, ±18° vertically.

If the subject is placed 50cm away from a 24" screen, the tracking range of the EyeLink is approximately exhausted. A typical 24" screen measures approximately 52cm wide and 32cm tall, which at 50cm away from it, corresponds to a range of ±27° horizontally and ±18° vertically. This presented a significant problem for this experiment; the EyeLink needs to be calibrated within this range[8] but the SDK that ships with EyeLink attempts to make full use of the screen resolution. Thus, if a large screen is used, the calibration points will be placed well beyond the tracking capabilities of the EyeLink and if a small screen is used, a limited range will be tested.

The 84" screen previously used in this chapter provided us with a potential range of approximately ±60° horizontally and ±52° vertically (when the subject is placed 50cm away from the screen). To circumvent the above limitation of the EyeLink SDK, a false screen resolution of a virtual screen (384x288 pixels) was reported to EyeLink by the controlling software that was developed for the experiment and the maximum possible resolution was chosen for the screen (1600x1200 pixels). In this configuration, the virtual screen spanned approximately ±22.2° horizontally and ±16.95° vertically – a range within EyeLink's tracking capabilities. Then, the calibration points dictated by the EyeLink SDK were translated such that they were correctly placed within the virtual screen, which is in turn placed in the centre of the actual screen as shown in Figure 34.

A total of four subjects (all male, Caucasian, 25-35 years old) took part in this experiment. The subjects stood in front of the screen at a distance of 50cm and a chinrest was used to fix their head such that head-tracking can be turned off for EyeLink (head-tracking in EyeLink depends on four markers that need to be placed at the corners of the screen whose bounds must be within the specified tracking range ±30°/±°20). Additionally, the chinrest serves to eliminate errors from slight head-movement due to swaying of the subjects while standing, for both eye-trackers.

The full screen was divided in increments of 100 pixels resulting in a maximum of $17x13 = 221$ points; the centre of the grid was manually adjusted according to the subject's height such that he could focus on it by gazing straight ahead. Before the experiment, each subject tested his

[8] This was found to be true experimentally. Outside of this range calibration failed too often making the test very cumbersome and tiring for both the experimenter and subjects. Of course, unless a successful calibration is performed, no eye-tracker would be able to operate.

maximum field of view by gazing at the full grid; rows or columns of points that were beyond their maximum field of view were removed from the grid resulting in a total of 195/195/195/165 points (REACT) and 150/150/150/120 points (EyeLink-II) for each subject respectively. Fewer points were used for EyeLink-II as the eye-tracker's visor blocked the top view beyond approximately 30°. During the trial the points were displayed in random order to avoid anticipatory eye-movements from the subject.

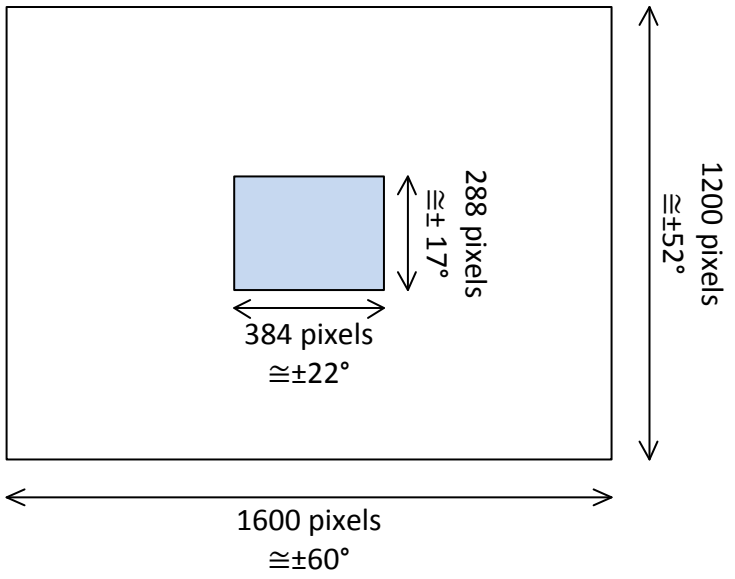


FIGURE 34: PHYSICAL AND VIRTUAL SCREEN FOR THE COMPARISON WITH EYELINK.

Calibration for the EyeLink involved displaying nine points on the virtual screen as described earlier. The REACT eye-tracker was calibrated using two different sets of nine points: a) the same set as the EyeLink and b) nine points that follow the same pattern but are spread across the complete field of view of the subject. Two calibration modes were used in order to perform a fair comparison to EyeLink as well as demonstrate the capability of REACT to operate and be calibrated beyond the range of the EyeLink-II.

Data was recorded for both EyeLink modes (pupil-only and PCR mode) and for each of the two different calibration modes of REACT thus resulting in four trials per subject. During calibration of the EyeLink, it was ensured that it performs optimally by adjusting the headband and position of the cameras as necessary such that a good calibration result (less than 1° average validation error) was obtained on each trial and for all subjects. Subjects were given ten minutes rest

between each trial in order to minimise fatigue that could influence their ability to fixate their eyes in such a large number of points.

Statistics for the error, in degrees of visual angle, of each mode across all subjects are shown in Table 27. The statistics were calculated for four different groups of ranges:

a) The closest approximation (±23°/±23°) of the conservative ±20°/±18° tracking range as per the EyeLink specification.
b) The closest approximation (±32.5°/±23°) of the pupil-only tracking range (±30°/±20°) as per the EyeLink specification.
c) Beyond ±23°/±23°; that is, the complete range minus the ±23°/±23° range.
d) Beyond ±32.5°/±23°; that is, the complete range minus the ±32.5°/±23° range.
e) The complete range.

The statistics were calculated in the above groupings such that the accuracy of the eye-trackers within the ranges in question can be analysed separately. The ranges were not exactly matched to the specification for two reasons: a) to have an evenly spread grid while minimising the number of points and consequently the fatigue of the subject and b) because the primary focus of this experiment is to test the maximum tracking range and associated accuracy of the eye-trackers of the complete range.

As expected, the EyeLink achieves a low error within the two constrained tracking ranges (2.44±3.56°, 3.30±5.53° for pupil-only mode and 4.09±7.63°, 5.23±8.33° for PCR mode) though higher than in the specification. This is not surprising given that within these ranges, only a small number of points were tested (25 and 35 for each range respectively) and some of these points were slightly beyond the specified maximum tracking range thus resulting in loss of tracking and high errors which significantly raise the average. The errors within the constrained field of view are below 1° if these outliers[9] were to be removed. Since the REACT tracker does not suffer from such losses of tracking, it displays significantly lower than EyeLink error for the limited range calibration (1.11±0.73°, 1.57±1.14). The same cannot be said for the full range calibration, where

[9] There is no reliable way to detect loss of tracking in the EyeLink. When tracking is lost, the eye-tracker will either output incorrect values or if the pupil is completely lost, it will be interpreted as a blink.

the error is 7.87±5.10°, 7.95±4.98. This is because the form of gaze mapping implemented in the REACT tracker for this experiment provides more accurate mappings near the calibration points.

The results for the two "difference" groups that examine eye-movements beyond the specified tracking ranges are quite different, with the lowest error achieved by the REACT tracker in full range calibration mode, followed by the EyeLink in pupil-only mode, then the REACT tracker in limited range calibration mode and finally the EyeLink in PCR mode. These results are coherent to the expectations that can be reasoned about the experiment.

The REACT tracker rarely suffers from loss of tracking as shown in the previous evaluation sections and thus its accuracy depends mostly on the accuracy of the pupil detection (reviewed previously) and the gaze mapping. In these groups, the gaze mapping algorithm performs best when calibrated with points that span the full range of targeted eye-movements and thus the REACT tracker in full range calibration mode comes first. The pupil-only mode of EyeLink is able to track over a larger range than the PCR mode and consequently suffers from loss of tracking less often (the pupil is easier to detect than the glint in large angle eye-movements), coming second. For the same reasons, the limited range calibration mode of the REACT tracker comes third while the EyeLink PCR mode last; for these groups, the glint will often fall onto the sclera thus resulting in loss of tracking for the EyeLink.

Having said all that, looking at the distribution of the error across the test grid is much more informative than the above statistics which are easily influenced by high errors within each group. The distributions for both eye-trackers and all four trial modes are shown in Figure 35 and Figure 36 respectively.

The EyeLink shows consistently fair performance within the middle of the grid (approximately ±32.5° horizontally and 23°/52° above and below the centre, vertically) in pupil-only mode. The highest errors are found outside this range, in all directions. In PCR mode, the map is much more inconsistent which is also not surprising given that it's very easy for the glint to disappear in the sclera as mentioned several times before. Consistently low results can only be observed in a small central range of approximately ±23° in both directions.

As explained before, the REACT tracker suffers from loss of tracking a lot less and thus the map in both modes is much smoother. In limited range calibration mode, the largest errors are observed

at the left, top and right edges of the map while in full range calibration mode, the errors on the edges of the map are small to medium and the highest errors are observed slightly more centrally than the limited range calibration mode. Once again, this is because the gaze mapping will be best near the calibration points.

Cases where the hardware obscured the view of the subject can be clearly seen on the maps. For example, in the pupil-only EyeLink map, two symmetrically placed that are fairly high relative to its neighbours are observed near the centre, most likely where the camera mounts or the cameras themselves were in the field of view of the users. Similarly, one point with high error relative to its neighbours is observed near the bottom and to slightly to the right side of both REACT maps.

One other artefact that needs to be explained is that the error progressively reaches high values towards the top left corner of both REACT maps. This increase is caused by the orientation of the camera in reference to the tracked eye. It is evident that the camera is pointed upwards and placed off-centre, to the one side thus distorting the uniformity of the feature points in the image. The resulting distortion is illustrated in Figure 37 where the pupil points that correspond to the calibration points for two subjects are shown. Of course, the camera can never be perfectly placed without a rigid setup and this problem exists with all other head-mounted eye-trackers including EyeLink (though because of the sophistication of the headband and the camera mounts, it is much easier to adjust the camera). Similarly higher error distributions are observed on the left side of the maps than on the right side for EyeLink though the pattern seems less severe than with REACT. This is due to two reasons:

- EyeLink appears to use a mapping function that uses a quadratic equation versus a linear function in the case of REACT. Hence, with a non-linear approach it is better able to cope with the non-linearity in question.
- EyeLink, being a mature product that has been in development for at least 15 years, appears to have developed some additional corrections mechanisms beyond the non-linear mapping function.

Both points above have been deduced from the debugging information that is written to the EyeLink output file.

In conclusion, the results of this experiment appear to support the specified tracking range of the EyeLink (±30°/±20° pupil-only, ±20°/±18° PCR mode). Even though the statistics calculated from the measurements taken in this experiment do not aid in precisely quantifying the range and accuracy within this field of view because of the high errors observed where there has been loss of tracking near the borders of this range, looking directly at the measurements for each subject reveals that when there was successful tracking the error is, in most cases, below 1°. This is not surprising as the EyeLink is a well-respected[10] eye-tracker within academia and it would not have gained its reputation if it were not for its high accuracy and high sampling rate (250 or 500Hz for EyeLink-II, other EyeLink models go up to 2000Hz).

However, this experiment was primarily focused on exploring what happens beyond the tracking range that EyeLink is known to operate well within. It was expected that especially in PCR mode, loss of tracking would occur for visual angles beyond the specified tracking range because the corneal reflection would be placed in the sclera, thus disabling EyeLink from being used to successfully track eye-movements beyond ±30°/±20°. Indeed, it was found that, outside of this range, the EyeLink performs poorly in terms of accuracy and also performs inconsistently, especially in the horizontal direction and especially in PCR mode. In contrast, REACT performs satisfactorily well (average error 5.74° when it's calibrated with points that span the full range) up to approximately ±56°/±52°, which is the full range of movements that were possible for the four subjects that participated in this experiment.

Finally, the quadratic gaze mapping function of EyeLink was found to be superior to the linear function used by REACT, which in fact causes high errors concentrated on the top left corner of the grid (though this will depend on the exact placement of the camera). This is not a concern for the thesis presented here as gaze mapping is not required functionality of the REACT eye-tracker for the target application but it was only implemented such that this comparison would be made possible.

[10] SR Research's website (http://www.sr-research.com/publications.html) mentions that EyeLink has been cited in over 1400 peer-reviewed publications.

TABLE 27: ERROR STATISTICS FOR BOTH EYE-TRACKERS, IN DEGREES. DIFFERENT SETS OF STATISTICS ARE CALCULATED FOR THE FULL TRACKING RANGE OF EYELINK ACCORDING TO ITS SPECIFICATION (±30°/±20°), THE CONSERVATIVE TRACKING RANGE FOR PCR MODE (±20°/±18°), OUTSIDE THE LATTER TWO RANGES AND OVER THE COMPLETE RANGE.

	Average	σ	Maximum
Within ±23°/±23°	2.44	3.56	19.81
Within ±32.5°/±23°	3.30	5.53	37.27
Beyond ±23°/±23°	7.38	8.29	46.97
Beyond ±32.5°/±23°	7.16	7.81	46.97
Overall	6.21	7.74	46.97

EYELINK-II, PUPIL ONLY MODE

	Average	σ	Maximum
Within ±23°/±23°	4.09	7.63	64.68
Within ±32.5°/±23°	5.23	8.33	64.68
Beyond ±23°/±23°	9.96	9.72	44.19
Beyond ±32.5°/±23°	9.97	9.71	44.19
Overall	8.55	9.59	64.68

EYELINK-II, PUPIL AND CORNEAL-REFLECTION MODE

	Average	σ	Maximum
Within ±23°/±23°	1.11	0.73	3.32
Within ±32.5°/±23°	1.57	1.14	5.33
Beyond ±23°/±23°	8.05	6.62	42.37
Beyond ±32.5°/±23°	8.52	6.71	42.37
Overall	6.88	6.61	42.37

REACT, LIMITED RANGE CALIBRATION

	Average	σ	Maximum
Within ±23°/±23°	7.87	5.10	23.53
Within ±32.5°/±23°	7.95	4.98	23.53
Beyond ±23°/±23°	5.94	5.46	26.01
Beyond ±32.5°/±23°	5.74	5.51	26.01
Overall	6.27	5.48	26.01

REACT, FULL RANGE CALIBRATION

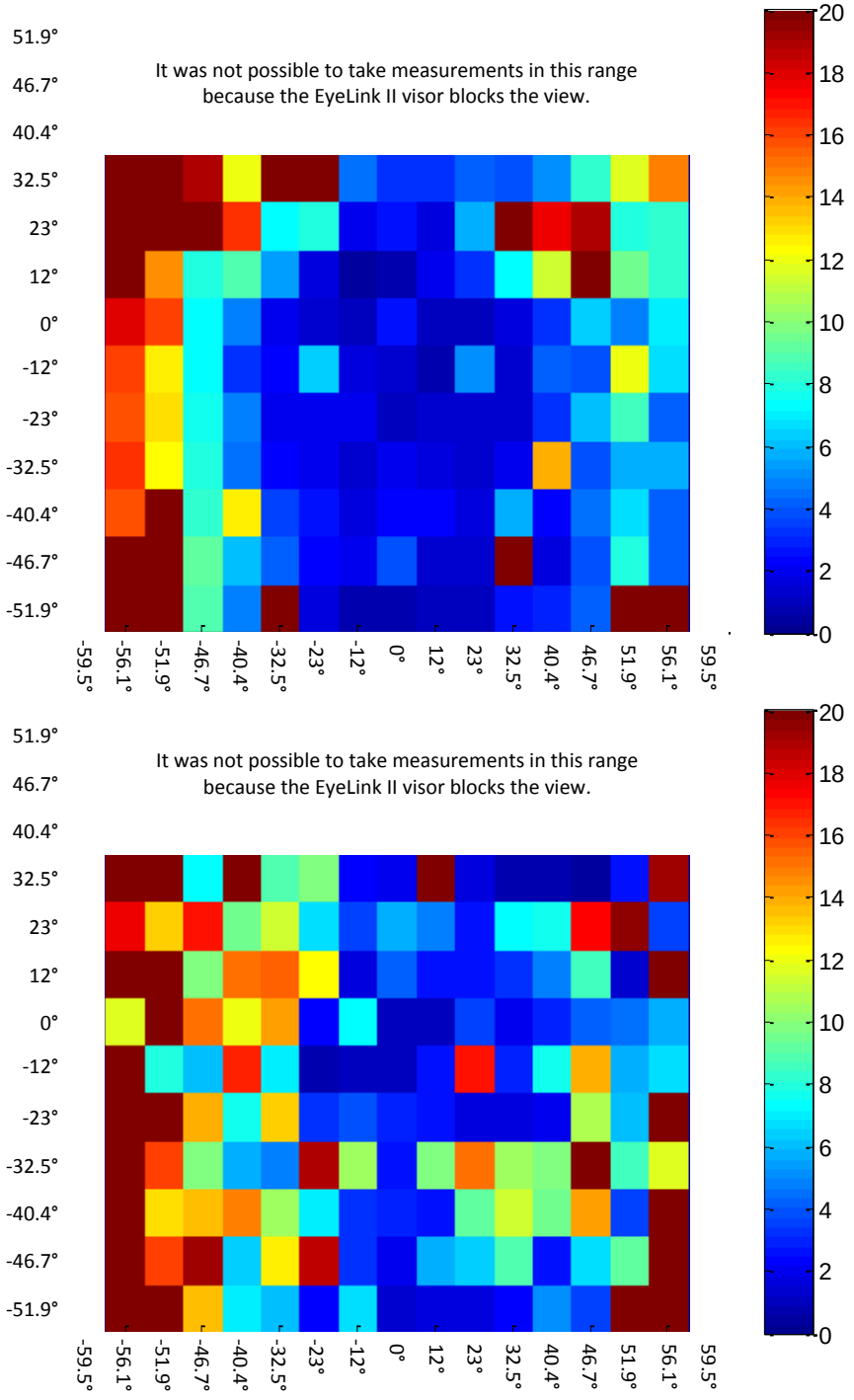


FIGURE 35: EYELINK-II ERROR (IN DEGREES, AVERAGE ACROSS ALL SUBJECTS) HEAT MAP FOR THE COMPLETE SET OF GRID POINTS BOTH FOR PUPIL-ONLY MODE (TOP) AND PUPIL AND CORNEAL-REFLECTION MODE (BOTTOM).

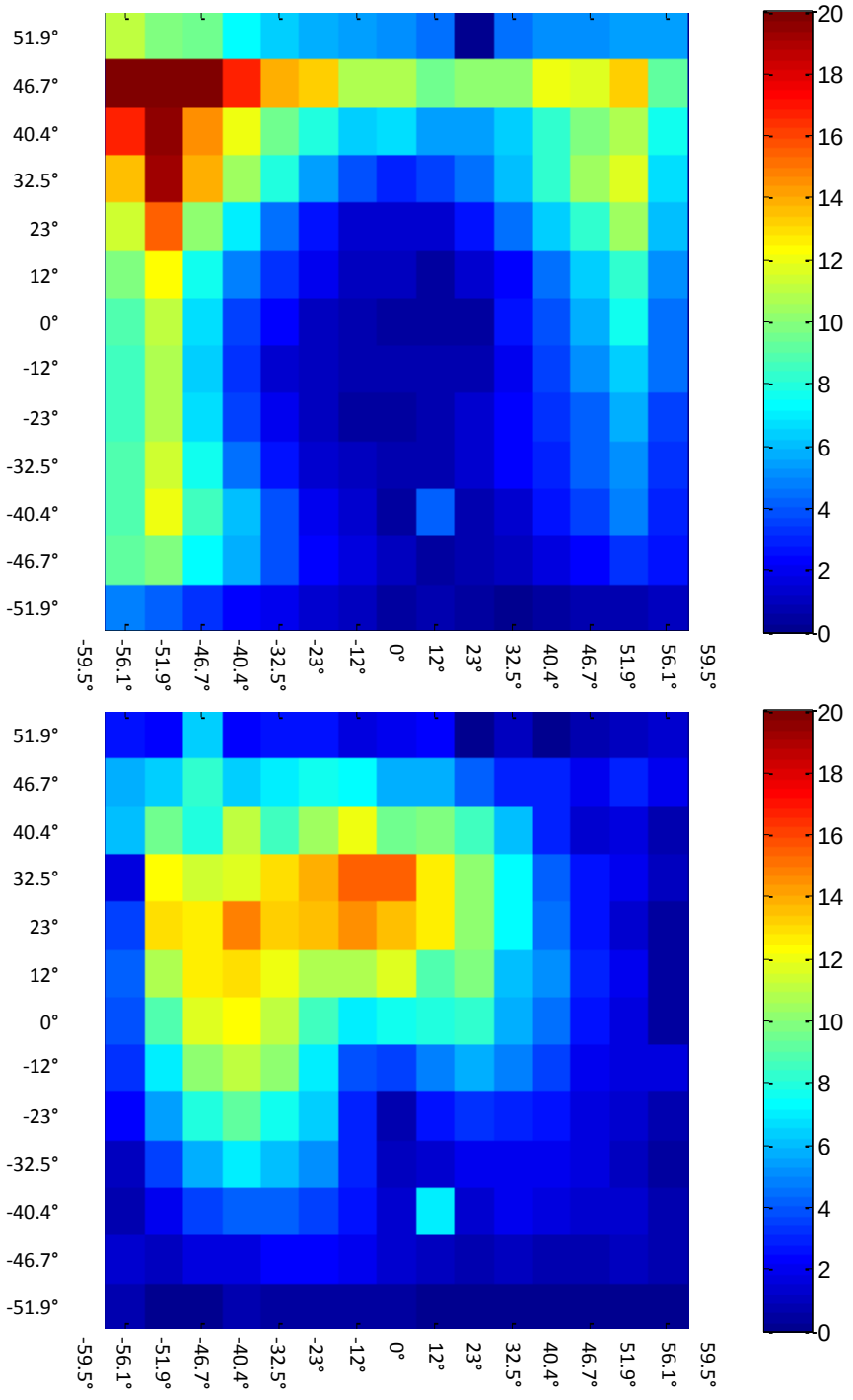


FIGURE 36: REACT EYE-TRACKER ERROR (IN DEGREES, AVERAGE ACROSS ALL SUBJECTS) HEAT MAP FOR THE COMPLETE SET OF GRID POINTS FOR BOTH CALIBRATION MODES; SAME NINE POINTS USED TO CALIBRATE EYELINK (TOP) AND NINE POINTS COVERING THE FULL FIELD-OF-VIEW (BOTTOM).

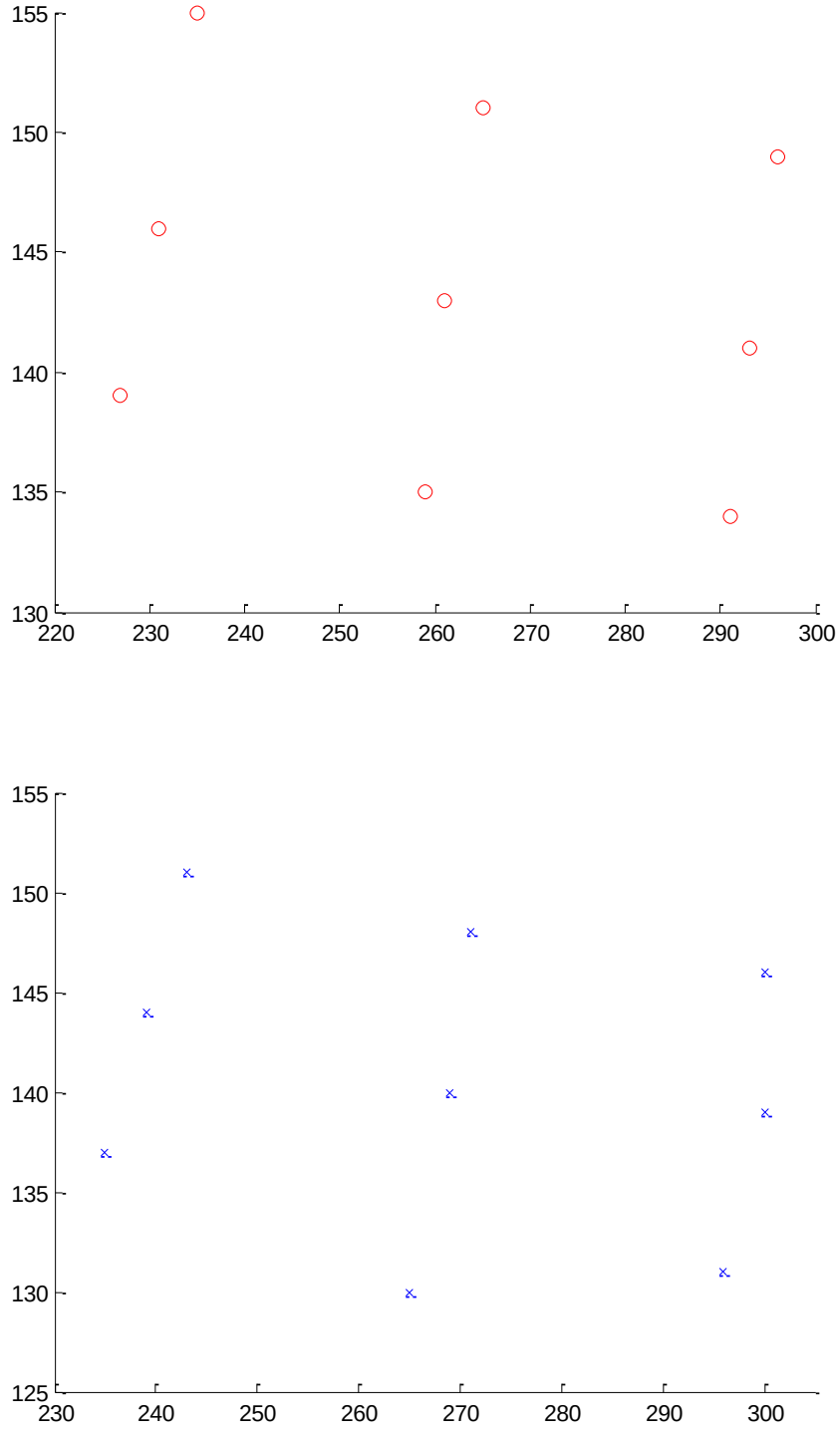


FIGURE 37: PLOTS OF THE PUPIL POSITIONS THAT CORRESPOND TO THE CALIBRATION POINTS DISPLAYED ON THE SCREEN FOR TWO SUBJECTS (TOP, BOTTOM).

EVALUATION OF THE REACT EYE-TRACKER HARDWARE USABILITY

In the eye-tracking literature, usability is rarely assessed and the eye-tracker which forms the basis for the hardware design (Babcock and Pelz, 2004) of the REACT eye-tracker is no exception. This may be because the comfort of the user is not as important for applications where the subject does not interact with another human being as is the case for applications targeted by the REACT eye-tracker. Another reason may be that engineers are much more concerned with the functional aspects of eye-tracking than with the non-functional requirements.

In this project, it was thought prudent to include a basic evaluation of the eye-tracker's usability that could highlight any major problems with its usability and the comfort of the subjects that would render it unusable for the applications in question. Thus, the basic questionnaire shown below () was distributed to the subjects that have taken part in all experiments; the results are shown in Table 29. Question one aimed to establish the overall level of comfort that subjects perceived themselves as experiencing during the experiments. Questions two and three were targeted towards establishing how aware the subjects were of the eye-tracker's presence on their head and its presence in their visual field respectively. Finally, question four asked the subjects whether they would hypothetically participate in a longer experiment (eight out of nine subjects made only occasional use of the eye-tracker with each session lasting less than five minutes) and in case of a negative answer, question five enquired as to whether this was relevant to a usability aspect of the eye-tracker (such as it being too heavy etc.).

As can be seen from Table 29, the majority of subjects (seven out of nine) answered they were generally comfortable wearing the eye-tracker. Subject five gave a medium rating (neither comfortable not uncomfortable) and only subject five and six gave a low rating on this question (2 – uncomfortable). The average across subjects is 3.6 points.

In this small sample, a rating of 4 (comfortable) on question one corresponded to a rating of 1-2 in question two (very unaware/unaware of the eye-tracked being placed on their head). Subjects one and nine who gave a rating of 4 on question one answered that they were neither aware nor unaware of the eye-tracker being placed on their head. In terms of invasiveness, a neutral rating can still be considered as favourable as it means that no discomfort that could distract the subject from the task at hand was caused. Of course, it also means that subjects that gave a neutral rating

are less likely to quickly forget about the existence of the eye-tracker than subjects that gave a positive rating. Finally, only subjects three and six gave a high (unfavourable) rating for question two.

TABLE 28: USABILITY QUESTIONNAIRE.

Q1. How comfortable was it to wear the glasses/eye-tracker?

Very uncomfortable	1	2	3	4	5	Very comfortable

Q2. How aware were you of the feeling of wearing the glasses on your head throughout the experiment?

Very unaware (not aware)	1	2	3	4	5	Very aware

Q3. How aware were you of the eye-tracker's camera in your vision throughout your experiment?

Very unaware (not aware)	1	2	3	4	5	Very aware

Q4. If you were asked and your schedule allowed, would you participate in an experiment where you would have to wear the eye-tracker for up to 20 minutes?

Yes	No

Q5. If you answered no in the above question, would that be because of any aspect of the eye-tracker's usability (e.g. feels too heavy on the head, is tiring to wear etc.)?

Yes	No

TABLE 29: USABILITY QUESTIONNAIRE RESULTS.

Subject	Q1	Q2	Q3	Q4	Q5
Subj01	4	3	2	Yes	-
Subj02	4	1	3	Yes	-
Subj03	4	4	2	Yes	-
Subj04	4	1	2	Yes	-
Subj05	3	3	1	No	No
Subj06	2	4	1	Yes	-
Subj07	4	2	2	Yes	-
Subj08	4	2	1	Yes	-
Subj09	4	3	1	Yes	-
Average	**3.6**	**2.5**	**1.6**	**-**	**-**

Question three received much more favourable answers than question two; four out of nine subjects gave a rating of 1 (very unaware), four out of nine subjects gave a rating of 1 (unaware) and subject two only gave a rating of 2 (neither aware nor unaware).

The results are certainly interesting; in question two, the rating was never higher than 4 (aware) but in question three the ratings did not exceed 3 (neither aware nor unaware). It is hypothesised that this is a favourable result; to be clear, it is hypothesised that unfavourable ratings in question three would have a more significant effect to the invasiveness of the eye-tracker than unfavourable ratings in question two. This hypothesis is made because:

- During a longer experiment where subjects would have time to become comfortable with wearing a foreign device on their head, they would slowly drift away from the consciousness of the eye-tracker and immerse themselves in the task or conversation central to the experiment.
- While movements of the head would not change the feeling of wearing the eye-tracker, if the eye-tracker significantly blocked their field of view, the subject would be reminded of its existence as their eyes moved.

The above reasoning, is reflected in the results as the subjects who gave an unfavourable rating on question one, gave an unfavourable rating in question two and a favourable rating in question three. It would thus appear that they judged the overall invasiveness of the eye-tracker based on the feeling of wearing it versus the eye-tracker camera being somewhat in their visual field.

In fact, subject five who gave a neutral rating on question one, at the end of the questionnaire, commented that he gave such a rating and answered "No" to a hypothetical future experiment because he found that his eyes hurt after each experimental session. This strain on the eyes must have been caused by trying to looking at points on the screen that spanned a larger area than comfortable to the subject. Thus, it can only be concluded that this was due to a miscommunication on the experimenter's part as the ability to reduce the radius of the circle that the points laid on was offered to each subject such that their eyes are not strained at any time.

Similarly, subject six commented that "it felt like the glasses wanted to slip to the end of the nose" and added that it may have been because he had never had to wear glasses before. Other than becoming familiar with the feeling of wearing glasses, from this statement, it can be concluded that it would be helpful to try several different frames for the eye-tracker, including metallic ones and find which frame or frames gathered the best responses from test subjects.

The overall performance of the eye-tracker in terms of invasiveness may be summarised by the comments of subject nine who stated that "overall, the eye-tracker was pretty unobtrusive although I was always aware that it was there". Even though all subjects reported that they were aware of the eye-tracker's existence, they found it was comfortable to wear it during the short experimental sessions and that they were rarely aware of the eye-tracker in their visual field. Thus, in longer sessions when the subject is engaged in the natural task of conversing with the interviewer, the invasiveness of the eye-tracker would be reduced from minimal to negligible.

CHAPTER 6: CASE STUDY

As explained in the introductory chapter and the literature review, our motivation in building the REACT eye-tracker has been to enable researchers interested in eye-movements that are not generated by outwards visual attention to e.g. tracking a visual target but rather by inwards attention (i.e. thinking) to perform experiments using eye-tracking technology.

Thus, in addition to the extensive evaluation of the eye-tracker in the previous chapter, it was decided to demonstrate the eye-tracker's applicability to a real-world scenario through a case study, at the same time illustrating the value of using an eye-tracker. For reasons that have been mentioned before and will be mentioned again throughout this chapter, the pilot experiment of this chapter does not attempt to support or disprove the NLP EAC model. The results from the experiment cannot support or disprove the model.

Once again bearing in mind our motivation through the NLP literature, a pilot study was designed where a subject was interviewed by the experimenter on various topics such as education and hobbies, meanwhile recording the subject's eye-movements, for a period of approximately twenty (20) minutes. The conversation was then manually transcribed and the position in time of each word uttered by both the subject and experimenter was also manually recorded. This process resulted in a rich data set that allowed the precise overlay of eye-movements, in time with speech. The video and audio were recorded simultaneously into the same file and thus synchronisation was taken care of.

In this chapter, the extension of the eye-tracker to continuously track the subject's eye-movements over time is described in detail. The pilot study experiment and its relation to the NLP eye-accessing cues model are briefly discussed with the aid of selected extracts. Finally, the full data set of the transcribed text with timestamps and overlaid eye-movement classifications are presented in Appendix B.

EYE-TRACKING OVER TIME

In Chapter 3, the algorithms involved in detecting the set of eye features necessary to calculate the 2D gaze of the subject were deconstructed in full detail. Regardless of the inherent complexity in calculating these low-level features, they provide little information to a simple user

of the eye-tracker who is only interested in what eye-movements the subject made during the experiment.

While the 2D gaze calculated using these features can be considered a higher-level output of the eye-tracker, it still serves poorly as the main output of the eye-tracker for several reasons discussed below.

First of all, it generates a large amount of data samples, far larger than what can be manually analysed by the average user. At 29.97 captured frames per second, 59.94 fields are analysed by the eye-tracker per second. That is almost 3596 data samples per minute.

Not only is the amount of data samples too large to manually process, but it may also be considered as unnecessary. The speed at which eye-movements are detected is limited by the capturing rate, which is dictated by the camera. For example, smooth pursuit[11] studies that require very high sampling rates could not be conducted because the camera is only capturing 29.97 frames/59.94 fields per second and would require much higher sampling rates (in the order of 250-500Hz) that only high-end commercial eye-trackers, such as the SR-Research EyeLink-II, can offer due to the inherent hardware costs. Thus, in the range of eye-movements that can be tracked by the REACT eye-tracker, the experimenter will most likely be interested in fixations, during which the eye is fixated for a certain amount of time at one location[12] and what will be referred to as "search paths" in this chapter.

Before even designing the REACT eye-tracker and while conceiving its requirements, a camcorder was used to record the eye-movements of a subject during short interviews. After transcribing the conversation and using the video time stamps to automatically overlay the eye-movements on the text, it was noticed that the subject would often perform several sequential eye-movements with small pauses in between before fixating at a final position and answering the question. For example, when asked about the nature of a particular experience, the subject may have looked right, paused for a very brief amount of time, then looked up and to the right

[11] Smooth pursuit is defined as the eye-movements that occur when the eyes closely follow a moving object.

[12] Normally, fixations refer to the point in the world that the subject is fixating upon. However, in this discussion, fixations simply refer to when the eye has moved off the centre and remains fixed for some time.

before finally answering the question. This sequence of eye-movements will be referred to as "search paths" from this point on.

Search paths although explicitly defined here are not a new observation. In NLP, they have been modelled through the notions of lead and representational system as introduced in the literature review of Chapter 2. However, in research related to the NLP EAC model, only a limited set of researchers have taken into account "search paths" or sequences of eye-movements such as Baddeley and Predebon (1991). For further information, see Chapter 2.

Thus, fixations and search paths are the eye-movements that the REACT eye-tracker is targeted at selecting. This selection is done as follows (illustrated in the state diagram of Figure 38):

1) The current pupil position is detected and the corresponding 2D gaze angle is calculated.
2) If the subject is looking straight ahead, the frame is *skipped*. Otherwise, it is classified into one of the eight categories listed below, the class is recorded and a count is maintained.
3) On the next frame, if the classified position is the same as the one recorded, the count is updated. If the subject is now looking straight ahead, the count is reset.
4) If the count reaches a threshold N_{EM}, the frame is *selected*. This allows fixations to be selected.
5) When a frame is selected, N_{EM} is set to a lower threshold N_{EM_SEARCH} and the algorithm continues onto the next frame. This allows for search paths to be detected and its component frames selected.

Eye-movements are classified into eight (8) classes as per the classes introduced in NLP (Bandler and Grinder, 1979; Figure 1); specifically up, up and to the left, up and to the right, left, down, down and to the left, down and to the right, right. For a 2D gaze angle θ and $\varphi = 20°$, a class C is assigned as follows (illustrated visually in Figure 39):

- $C = \text{R}$ for $-\frac{\varphi}{2} \leq \theta \leq \frac{\varphi}{4}$
- $C = UR$ for $\frac{\varphi}{4} \leq \theta \leq 90 - \varphi$
- $C = U$ for $90 - \varphi \leq \theta \leq 90 + \varphi$
- $C = UL$ for $90 + \varphi \leq \theta \leq 180 - \frac{\varphi}{4}$
- $C = L$ for $180 - \frac{\varphi}{4} \leq \theta \leq 180 + \frac{\varphi}{2}$

- $C = DL$ for $180 + \frac{\varphi}{2} \leq \theta \leq 270 - \varphi$
- $C = D$ for $270 - \varphi \leq \theta \leq 270 + \varphi$
- $C = DR$ for $270 + \varphi \leq \theta \leq 360 - \frac{\varphi}{4}$

As before, all processing takes place off-line.

A tolerance angle of $\frac{\varphi}{2}$ is used for the L and R classes on the bottom hemisphere (180° to 360°) to accommodate for the reduced vertical resolution that is a direct result of separating each frame into two fields (see de-interlacing in Chapter 3). The tolerance angle is further reduced to $\frac{\varphi}{4}$ for the top hemisphere of classes L and R (0° to 180°) to accommodate for the skew that is a direct result of the camera point upwards.

As analysed in Chapter 4, a calibration frame is required to initialize the eye-tracker to the subject. This is done by asking the subject to look straight ahead with his or her chin approximately parallel to the floor (posture is adjusted with the help of the experimenter). The eye-tracker automatically re-initializes itself to accommodate for changes in the location of the corners on every frame where the pupil position is within a contour twice as large as the initial pupil contour. In most sequences the latter contour is a little smaller than the iris. Re-initialization is not performed if the eye-tracker has re-initialized in the last 60 frames (approximately 10 seconds).

During re-initialization, the "initial pupil position" P used to calculate the 2D gaze angle is updated to P' from the new corner locations C_L' and C_R' such that:

$$\frac{C_L - P}{C_R - P} = \frac{C_L' - P'}{C_R' - P'}$$

This calculation is based on the assumption that the geometry acquired during the calibration of the eye-tracker as described by the distances between the eye corners and the initial pupil position is correct.

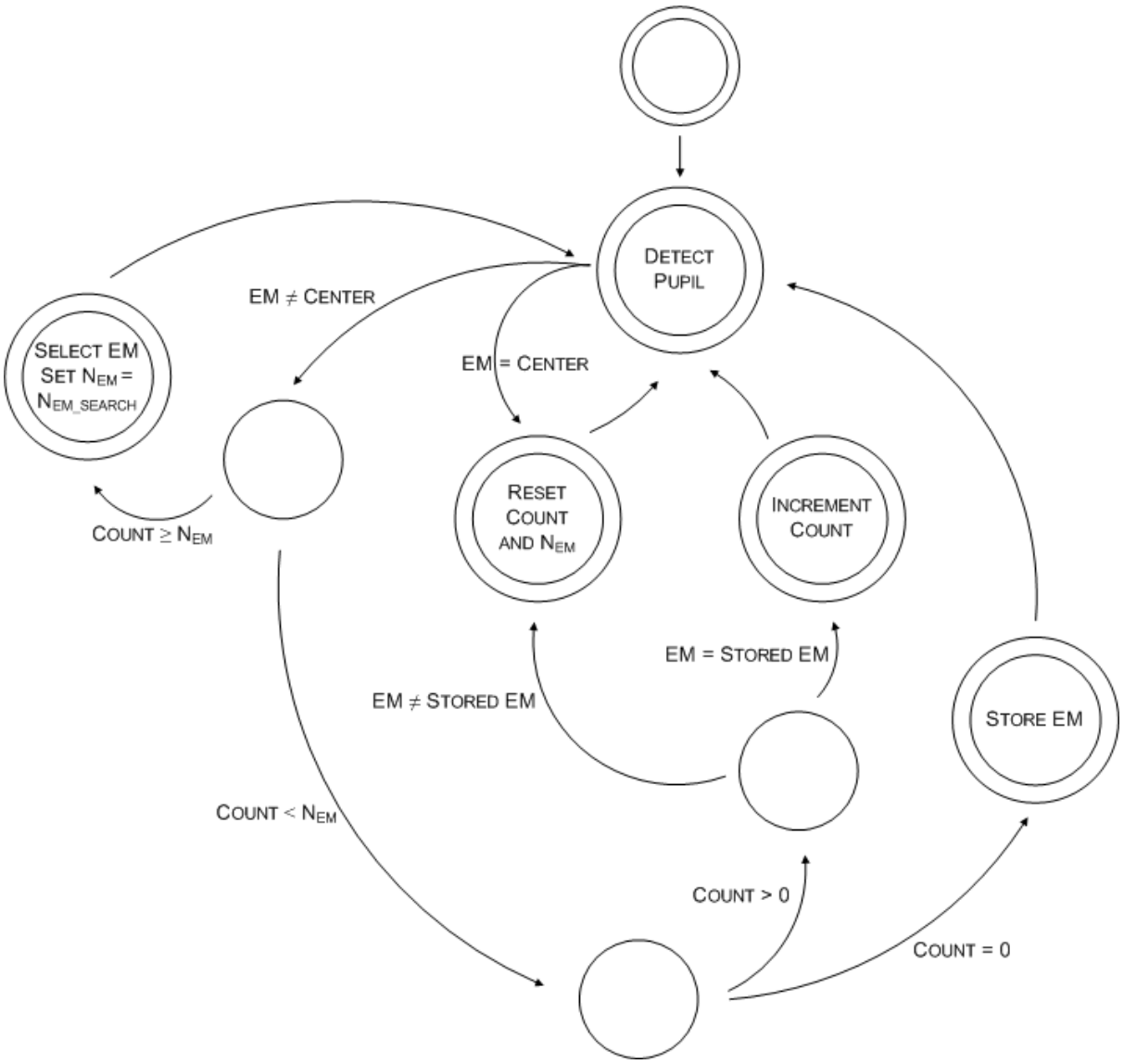


FIGURE 38: EYE-MOVEMENT SELECTION STATE DIAGRAM.

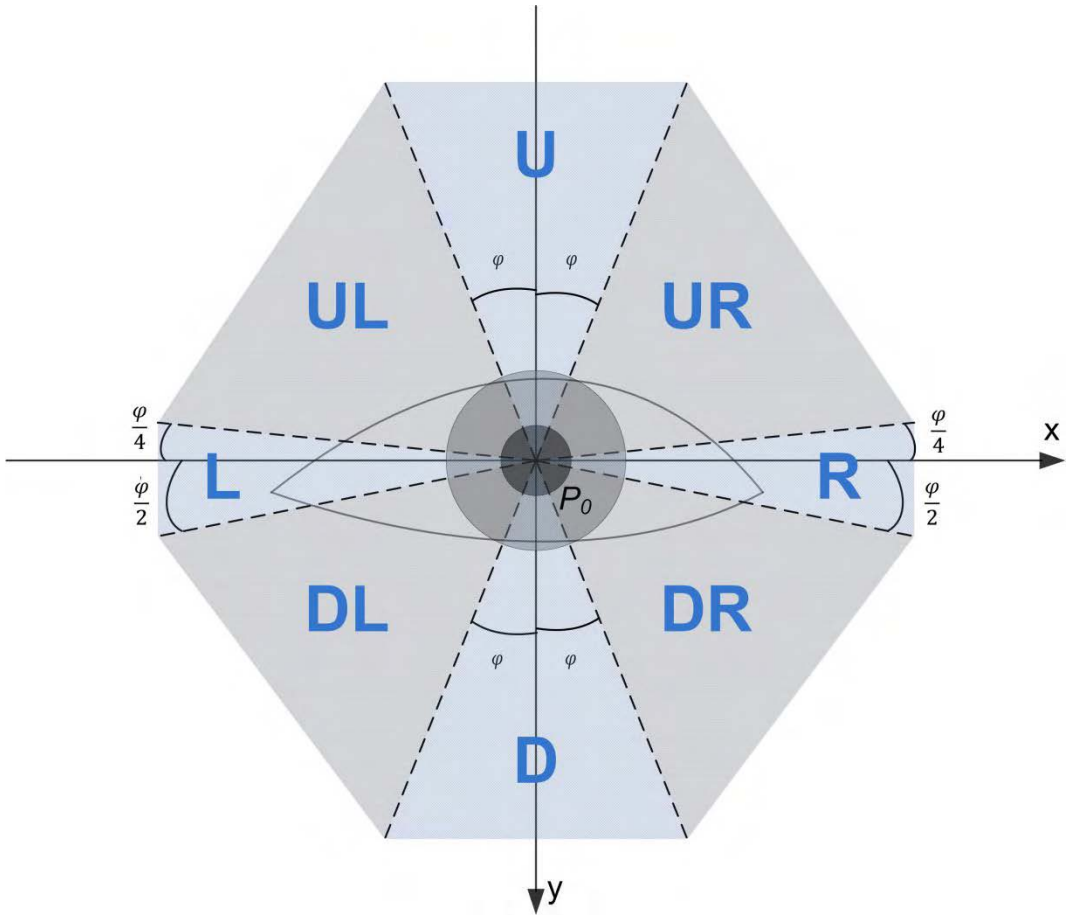


FIGURE 39: ILLUSTRATION OF THE CLASSIFICATION SCHEME.

CASE STUDY PILOT EXPERIMENT

The pilot experiment designed for this case study aimed to be a simplistic approximation of an experiment that would be performed to further investigate the correlation of eye-movements and thinking modalities or in other words, the NLP eye-accessing cues model.

The basic premise of such an experiment *would* be to investigate whether the eye-movement class is correlated to the question type or verbal predicates used by the subject when answering. Of course, because of the necessity to isolate and verify the subject's cognition discussed in Chapter 2, it would be appropriate to design an academically-sound questioning methodology based on the field of psycho-phenomenology (see Mathison and Tosey, 2008a; Mathison and

Tosey, 2008b). However, such a task is beyond the scope of this project and this experiment only serves to illustrate the value of an eye-tracker in its actualisation.

Additionally, as far as the analysis and technology involved is concerned, the structure of both experiments is essentially the same. Irrelevant of the questioning methodology, the task at hand is to select relevant eye-movements from the input video and accurately synchronize them with the transcribed text, which has been the focus of this case study. Normally, verbal predicates (such as "see" is visual, "hear" is auditory etc.) would be assigned to the transcribed text and this would allow the correlation with the eye-movement classes to be examined. This has not been done because a) lacking an academically-based questioning methodology the results would not have much meaning and b) it is not a technically difficult task beyond using the algorithms described in this thesis.

In this case study, a simple interview took place. The interviewer asked some basic questions about the subject's educational background and performance, his plans for the future as well as his hobbies, musical taste and travelling. The interviewer was deliberately sat on a chair with a white screen as the background; this was done to eliminate any interference caused by eye-movements to objects in the scene that may have attracted the subject's attention. Further, the chairs of both the subject and experimenter were adjusted such that their heads were on the same level and when the subject was looking at the experimenter, he would appear as looking straight ahead in the captured images. This was done to eliminate any interference that would have been caused the subject resting his/her eyes on the experimenter and the corresponding eye-movement. However, theoretically speaking, these eye-movements would not have been large enough to interfere in any way.

The discussion section below will briefly comment on the output of the case study and highlight the advantages of using the REACT eye-tracker in such a study. To avoid delving into complex matters that are beyond the scope of this thesis, the eye-movement-speech latency (Griffin and Bock, 2000) is not discussed or taken into account in this chapter.

DISCUSSION

The case study is a rich data set of speech from the subject and interviewer and the subject's eye-movements. In order to quantitatively examine such data, an appropriate visualisation method is

required to enable this (e.g. Figure 40). To generate this visualisation, the following process was followed:

a) The full conversation was manually transcribed in a text file; speech uttered from the subject was transcribed separately to that uttered by the interviewer and they are displayed on the top (blue) and middle (green) ribbons respectively.
b) Each word from the transcript was manually synchronized to the speech audio, precise to 0.1 seconds. To signify the start and finish of the utterance of each word, the ribbon is marked with the respective darker colour. Thus, when a word has been uttered by the interviewer, the ribbon is marked with dark blue and the word is drawn; an appropriate size font was used to make sure the word fits in its respective bounding box[13]. Each line (set of three ribbons, Figure 40) represents a maximum time length of 5 seconds and the beginning timestamp is displayed on the top-left hand-corner of each line. Both ribbons (subject and interviewer) use the same timeline, a tick is marked at the top of the interviewer's ribbon every 1 second.
c) Eye-movements that were selected by the eye-tracker were classified and a cropped thumbnail of the source image was drawn at the bottom ribbon, with the respective classification drawn below it, abbreviated (e.g. L is left, while DL is down and to the left). The thumbnails were placed at the exact time they occurred minus some thumbnails which were adjusted for presentation purposes (e.g. when part of the thumbnail would not appear because the eye-movement occurred at the end of the individual timeline). For each point in time that there is no eye-movement below the speech, it means that the subject was fixating at the same position as the previous eye-movement displayed, looking straight ahead, or moving their eyes.

[13] However, in some cases when the words were uttered in a short space of time (0.1-0.2 seconds), a minimum font size was enforced to ensure visibility, which has also created problems in some cases and words have been drawn on top of each other.

FIGURE 40: EXAMPLE OF VISUALISED CASE STUDY DATA SET.

As can be seen, the eye-tracker has successfully selected and classified relevant eye-movements which accurately exhibit the behaviour of the subject's eyes during the interview. But, before further commenting on the transcript, it is very important to make clear that this case study does not aim to prove, disprove or even provide evidence towards either case regarding the NLP EAC model. Since an academically-sound questioning methodology has not been developed, such attempt would be subject to the criticism of past research in Chapter 2. Without a solid questioning methodology, any study like this would constitute anecdotal evidence, at best.

Instead, the point of this case study is to highlight some of the eye-movement patterns that have previously been referred to in Chapter 2, where possible and, to demonstrate the benefits of using the REACT eye-tracker in non-visual eye-movement research. The full transcript is available for review in Appendix B.

First of all, obvious as it may be in an intuitive way, it is interesting to note that eye-movements have mostly taken place in time with or before relatively long pauses or while connective words such as "well" and "and" were uttered by the subject, in the example shown below (Figure 41). This is probably the first time this has been stated in the literature based on collected data versus the perceptive powers of the experimenter. In a full-scale study, it would be particularly easy to classify connective words and pauses in the subject's speech, correlate it with eye-movements in the vicinity and thus quantify this effect and find if it is statistically significant. This pattern of eye-movements will not have been acknowledged by model 1 of Figure 3.

Earlier, in Chapter 2, one of the main criticisms of past EAC model research was that in some studies, eye-movements were rated only after the experimenter has finished uttering the question (models 2-5, Figure 3). During the length of the interview, several eye-movements have followed a pattern contrary to this assumption: often, an eye-movement is made by the subject before the experimenter has finished uttering the question, such as shown below (Figure 42).

Once again, this statement is based on actual timed data for the first time in the literature and similarly, this pattern's occurrence could be quantified and labelled as statistically significant or not.

Another example of the two aforementioned patterns is displayed in Figure 43.

FIGURE 41: EYE-MOVEMENTS TAKING PLACE IN TIME WITH OR BEFORE PAUSES OR WHILE CONNECTIVE WORDS ARE UTTERED.

FIGURE 42: EYE-MOVEMENTS ARE SOMETIMES PERFORMED BEFORE THE END OF THE QUESTION.

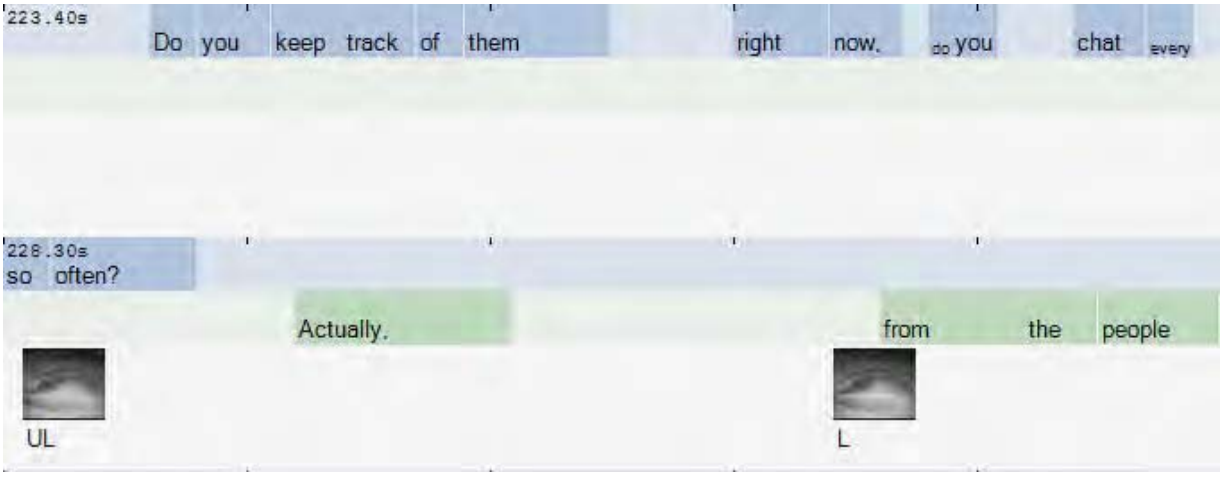
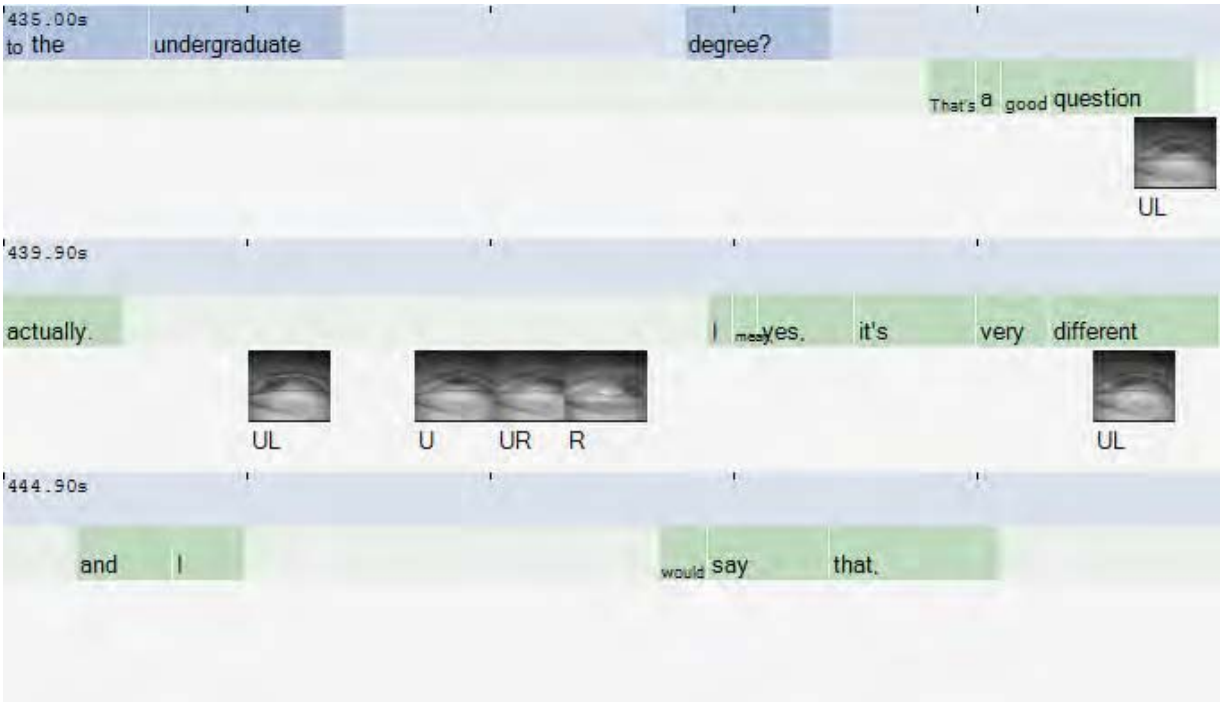


FIGURE 43: FURTHER EXAMPLE OF EYE-MOVEMENTS BEFORE THE END OF THE QUESTION AND DURING A PAUSE.

Multiple eye-movements in response to a question (models 4-5 and model 3 to a lesser extent, Figure 3) were recorded in very few past EAC model research studies and when they were, the analysis was somewhat rigid, only analysing the same number of occurrences across questions (see Chapter 2 for details). From the case study, it can be observed that particularly long pauses are often synchronised with a "search" or simply, multiple eye-movements, as in the two examples offered below (Figure 44). The following interpretations could intuitively be made:

- The long pause can be taken to mean that the requested information is not readily available and is searched for during the pauses. In the second example, "I can't remember" is perhaps then uttered as an unconscious attempt by the subject to keep the speech continuous and conversation interactive.
- Normally, a smaller number of eye-movements follows the end of a question and so, if such a pattern of eye-movements can be shown to be statistically significant and interpreted as an information-lookup strategy, a set of multiple eye-movements in response to a question may be taken to mean that the information is not readily available and not readily found during a search. Thus, the position of the eye keeps changing until the information is found or the search aborted.



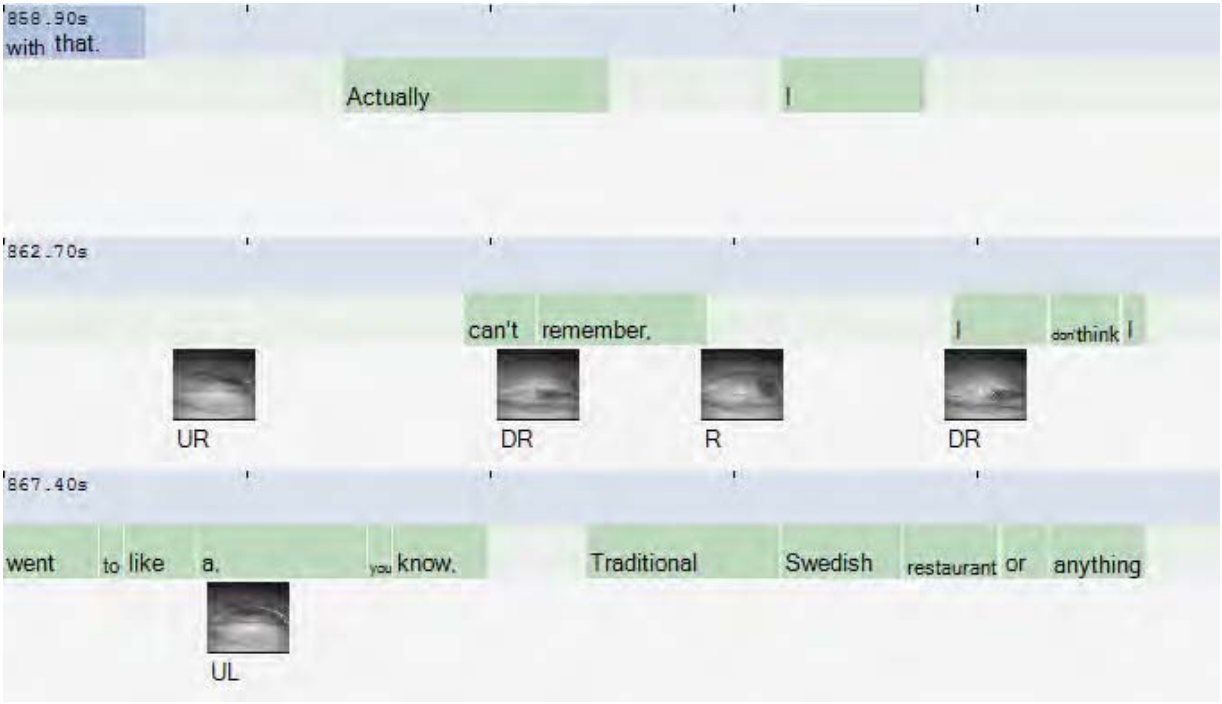


FIGURE 44: TWO EXAMPLES FROM THE TRANSCRIPT OF THE CASE STUDY WHERE MULTIPLE EYE-MOVEMENTS WERE PERFORMED IN ANSWER TO A QUESTION.

As before, with the use of the eye-tracker, this could be easily be verified in a full-scale study by classifying even longer pauses and speech patterns that demonstrate an inability to locate certain information and correlating them with the corresponding set of eye-movements.

With regards to correlating the question type or verbal predicates spoken by the subject to the corresponding eye-movement class as suggested by the NLP EAC model, limited relevance may be shown in the case study. This is because verbal predicates of any particular modality were rarely spoken by both the subject and interviewer i.e. predicates of unspecified modality, such as "think", were spoken instead of predicates that can be classified into one of the visual, auditory or kinaesthetic modalities, such as "see", "hear" or "feel".

This is most probably for three reasons:

a) The interview content was to a certain extent informational versus deeply subjective. That is, information about the past, present and future of the subject was collected but the interview did not delve deep into the experiences associated with each question.

b) According to NLP literature, highly intellectual people such as academics demonstrate a preference towards predicates of unspecified modality such as "think".

c) The particular communication style of the subject and/or interviewer may have limited the responses in this way.

Having said that, three examples of eye-movements consistent with the EAC model are displayed below[14]: a) "I'd say" corresponds to an auditory eye-movement, b) a sequence of eye-movements around "comfortable" begins and ends with a kinaesthetic eye-movement and c) "love", a verb associated with feeling is correlated to a kinaesthetic eye-movement.

FIGURE 45: THREE EXAMPLES OF EYE-MOVEMENTS CONSISTENT WITH THE EAC MODEL. THESE RESULTS ARE FOR ILLUSTRATION PURPOSES ONLY AND CANNOT PROVE OR DISPROVE THE MODEL.

[14] Please note that these examples are not sequential in time as the timestamp on the top-left hand corner shows. Also, the eye-movements are interpreted as if the subject adheres to the EAC generalization for normally-organized right-handed people from Figure 1.

It must also be noted that there are several eye-movements in the transcript that *cannot* be strictly interpreted as above. In the example show below, when the subject is speaking of being "stressed" and feeling "pretty calm" (kinaesthetic cues), an eye-movement relevant to visual construction is observed (Figure 46). Of course, this may be interpreted as the subject constructing a visual image of being stressed and/or feeling calm. Once again, this demonstrates the absolutely essential requirement for an elaborate and academically validated questioning methodology that is able to resolve such ambiguities.

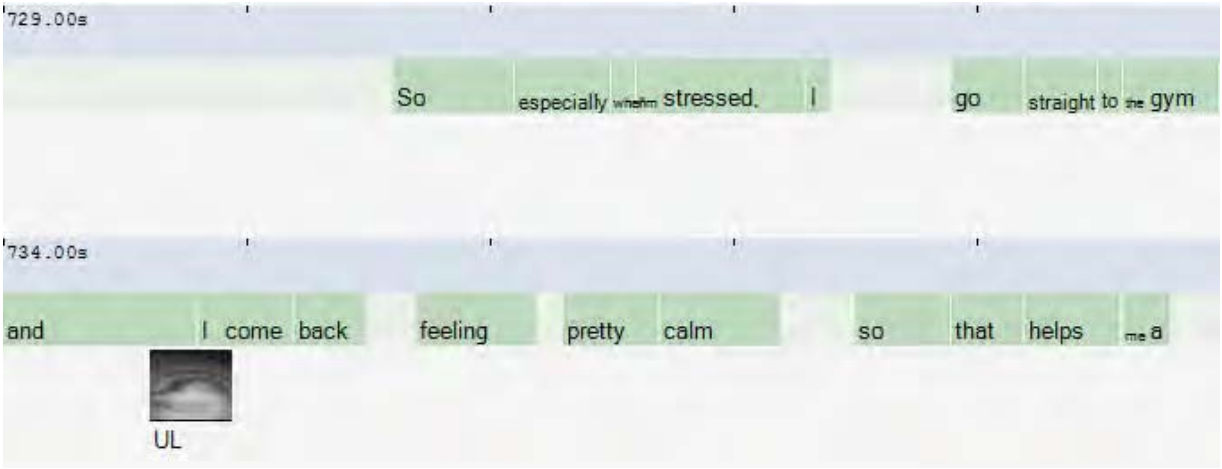


FIGURE 46: A VISUAL EYE-MOVEMENT IS OBSERVED NEAR THE UTTERANCE OF KINAESTHETIC CUES. AS PER THE EAC MODEL, THIS CAN BE INTERPRETED AS THE SUBJECT CONSTRUCTING A VISUAL IMAGE OF THEMSELVES BEING THAT WAY.

The only quantitative analysis that was performed on the case study is related to the classification of eye-movements, as shown in Table 30. The class output by the eye-tracker (labelled "Auto class") was compared to a manual classification performed by the experimenter (labelled "Manual class"). The manual classification proved to be a much harder task than anticipated as ambiguities were eminent in some cases when the eye-movement in question was on the borderline between two classes. In addition, in some cases, it was particularly hard to discriminate between upwards eye-movements and eye-movements on the baseline; this is both because of the reduced vertical resolution that is the result of de-interlacing and because the camera is pointed upwards thus reducing the apparent range of upwards eye-movements. Where ambiguous, the classification was marked as such and highlighted bright yellow whereas erroneous classification were marked red.

From the total 150 eye-movements, 7 received an ambiguous classification by the experimenter and 6 were erroneously classified by the eye-tracker (Table 30; Figure 47).

It is questionable whether ambiguous classifications can be avoided unless the subject's eyes are also captured from another camera placed on the same level and the video may be consulted to resolve ambiguities. Of course, while this would be feasible in an experimental, for the eye-tracker, setup, it would probably prove impractical for eye-tracker users conducting experiments.

All 6 classification errors were caused by the eye-movement being too close on the borderline between two classes. The classification algorithm, as described in Chapter 5, will determine the class solely on the 2D gaze angle calculated and based on pre-set thresholds. As with any other statically set threshold, it is bound to fail some of the time, when the thresholded value is very close to the threshold itself. In other words, when the gaze angle is on or close to the borderline between two classes, a human rater may be able to distinguish between the classes (though not always as proved by the 7 ambiguous ratings) but the algorithm cannot.

However, not only is the number of errors small but the classification algorithm can be improved to a) signal ambiguity in the classification and b) attempt to disambiguate between the two classes. The latter could be potentially be done by an algorithm based on the following concept:

- Creating an elliptical model E of the typical extreme eye-movements (U, UR, R, DR, D, DL, L, UL)
- Finding the closest point P_E to the current pupil position P, on ellipse E.
- Calculate the distance between P_E and the centre of ellipse E and compare it against the corresponding distance between the centre and the typical eye-movement point for each of the two candidate classes.
- Return the class that is closest to P_E.

In conclusion, in this chapter, a simplistic case study was successfully designed and performed in order to demonstrate the usefulness of using the eye-tracker in an experiment relating to non-visual eye-movements. Finally, the eye-tracker demonstrated satisfactory performance in selecting and rating the eye-movements throughout the case study.

TABLE 30: CASE STUDY CLASSIFICATION ERROR TABLE.

EM	Auto Class	Manual Class	Correct?
1	UL	UL	Yes
2	DL	DL	Yes
3	L	L	Yes
4	UL	UL	Yes
5	DL	DL	Yes
6	DL	DL	Yes
7	L	L / UL	Ambiguous
8	UL	UL	Yes
9	UL	UL	Yes
10	L	L	Yes
11	L	L	Yes
12	DL	DL	Yes
13	L	L	Yes
14	D	D	Yes
15	D	D	Yes
16	D	DR	No
17	L	L	Yes
18	L	L	Yes
19	UL	UL	Yes
20	UL	UL	Yes
21	U	U	Yes
22	L	L	Yes
23	UL	UL	Yes
24	U	U	Yes
25	UL	UL	Yes
26	UL	UL	Yes
27	L	L	Yes
28	UL	L / UL	Ambiguous
29	UL	L / UL	Ambiguous
30	L	L	Yes
31	L	L	Yes
32	L	L	Yes
33	UL	UL	Yes
34	R	R	Yes
35	DR	DR	Yes
36	UR	UR	Yes
37	UL	UL	Yes
38	UL	UL	Yes
39	DL	DL	Yes

40	L	L / UL	Ambiguous
41	DL	DL	Yes
42	L	L	Yes
43	UL	UL	Yes
44	UL	UL	Yes
45	DL	DL	Yes
46	DL	DL	Yes
47	UL	U	No
48	UL	UL	Yes
49	U	UR	No
50	UR	UR	Yes
51	R	R	Yes
52	UL	UL	Yes
53	L	L	Yes

54	UL	UL	Yes
55	UL	UL	Yes
56	UL	UL	Yes
57	UL	UL	Yes
58	UR	UR	Yes
59	L	L	Yes
60	UL	UL	Yes
61	UL	UL	Yes
62	L	L	Yes
63	L	L	Yes
64	UL	UL	Yes
65	UL	UL	Yes
66	UL	U	No
67	DR	DR	Yes

68	D	D	Yes
69	DL	DL	Yes
70	UR	UR	Yes
71	UR	UR	Yes
72	L	L	Yes
73	DL	DL	Yes
74	L	L	Yes
75	UL	UL	Yes
76	UR	UR	Yes
77	UR	UR	Yes
78	DR	DR	Yes
79	UL	UL	Yes
80	DL	DL	Yes
81	L	L	Yes

82	UL	UL	Yes
83	U	U	Yes
84	UR	UR	Yes
85	DR	DR	Yes
86	UL	UL	Yes
87	UL	UL	Yes
88	L	L	Yes
89	UL	L / UL	Ambiguous
90	L	L	Yes
91	UL	UL	Yes
92	DL	DL	Yes
93	UR	UR	Yes
94	DR	DR	Yes
95	UL	UL	Yes

96	L	L	Yes
97	DL	DL	Yes
98	R	R	Yes
99	UR	UR	Yes
100	UL	UL	Yes
101	U	U	Yes
102	UR	UR	Yes
103	R	R	Yes
104	UL	UL	Yes
105	UR	UR	Yes
106	DR	DR	Yes
107	UL	UL	Yes
108	UR	UR	Yes
109	D	D / DR	Ambiguous

110	DL	DL	Yes
111	DL	DL	Yes
112	UR	UR	Yes
113	DR	DR	Yes
114	R	R	Yes
115	DR	DR	Yes
116	UL	U	No
117	UL	UL	Yes
118	UR	UR	Yes
119	L	L	Yes
120	DL	DL	Yes
121	UL	UL	Yes
122	U	U	Yes
123	UR	UR	Yes

124	DR	DR	Yes
125	R	R	Yes
126	UL	UL	Yes
127	U	U	Yes
128	UR	UR	Yes
129	UL	UL	Yes
130	UR	UR	Yes
131	R	R	Yes
132	DL	DL	Yes
133	UL	UL	Yes

134	UL	UL	Yes
135	UL	UL	Yes
136	DL	DL	Yes
137	L	L	Yes
138	DL	DL	Yes
139	U	U	Yes
140	UL	UL	Yes
141	UL	UL	Yes
142	UR	U / UR	Ambiguous
143	DR	DR	Yes

144	UL	UL	Yes
145	UL	UL	Yes
146	R	R	Yes
147	R	R	Yes
148	UL	U	No
149	UL	UL	Yes
150	UL	UL	Yes

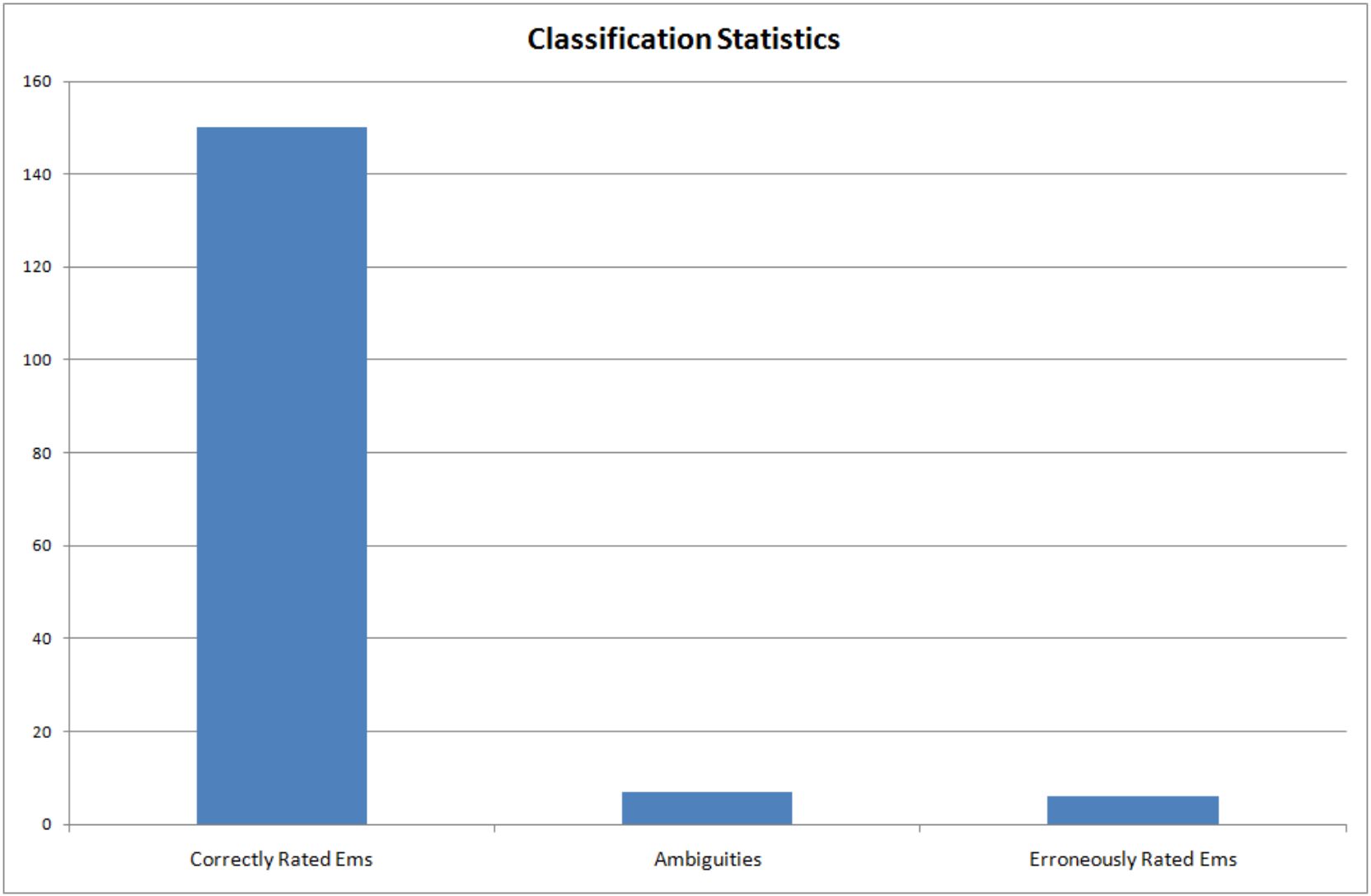


FIGURE 47: CLASSIFICATION STATISTICS BAR CHART.

CHAPTER 7: CONCLUSIONS AND FUTURE WORK

Neuro-Linguistic Programming and the Eye-Accessing Cues model have helped form the motivation behind this thesis; as discussed in Chapter 2, the EAC model has been the centre of research attention for almost as long as video-based eye-trackers have existed, with each study displaying one or more significant methodological errors, two of which have been pivotal.

The first is the lack of development of a questioning methodology with strong academic foundations, with the emphasis on "development". From the review of Chapter 2, one can be certain that eye-movements are directly linked to brain activity but we have yet to scientifically establish whether eye-movements in conversation can be consistently triggered and much more importantly, whether a questioning methodology that goes to depth, rather than breadth as all research studies have done to date, will consistently elicit the same or similar-enough eye-movements. These are very important research questions that need to be answered before researchers attempt to directly prove or disprove the EAC model itself and they are not trivial to answer. These limitations were also established through the presentation of this review at the First International NLP Research Conference and later through its publication in the conference proceedings (Diamantopoulos *et al.*, 2008).

Thus, in light of the inconsistent results and therefore tentative conclusions of past research as well as recent aforementioned developments in academia, it was established that the EAC model is in fact a very complex model which requires further research attention and rigorous examination if it is going to be proved or disproved.

Research in the EAC model has stopped for almost two decades and now is a good time to continue these efforts, as eye-tracking technology has matured and become more affordable too, which brings us to the second pivotal methodological error: the lack of use of a reliable and precise method to record, select and rate eye-movements.

As discussed in Chapter 3, to date, no eye-tracker had fulfilled all the requirements of tracking non-visual eye-movements[15].

First of all, existing eye-trackers have, without exception, been designed to track visual eye-movements (within 30° of angular range) and are severely limited when faced with the task of tracking non-visual eye-movements which tend to occur outside the regular field-of-view (30°-55°). A formal explanation of this empirical conclusion cannot be found in the literature though an intuitive conclusion is easily deduced; in order to minimize strain of the eyes and maximize visual acuity, people tend to turn their heads in the general direction of the observed object and accurately shift their eyes to bring it in focus.

Secondly, invasiveness is a major concern in this particular application as it may "break" the rapport between the subject and experimenter. Such a link seems particularly plausible in light of recent research; in their experiments, Tognoli *et al.* (2007) found that the phi complex is a brain rhythm that serves social functions of the brain (both independent and those requiring coordination between people) and produces wave patterns similar to those of mirror neurons. Thus, the invasiveness of the eye-tracker can significantly affect the results of the study. Traditionally, the invasiveness of an eye-tracker is related to physiological variables such as whether it is mounted on the subject's body or whether the subject's movement is restrained through a device like a chin rest or a bite bar. In this thesis, it was updated to include a major psychological variable: whether the subject is aware of being watched or not. This awareness was suggested to be mainly affected by three factors: a) the feeling of a foreign device on the subject's head, b) its existence in the subject's visual field and c) by the calibration procedure which inevitably highlights the non-natural conditions of an eye-tracking experiment. Taking all these variables into consideration, a remote eye-tracker which is the least invasive in the old definition may not be the least invasive in the updated form of invasiveness.

Additionally, requirements of practical significance were determined to be important for this research. Specifically, maintaining low-assembly cost, ease of use and transparency in calibrating the eye-tracker to the subject is vital to making eye-tracking technology more accessible and ensuring its adoption in such research. Also an important requirement of practical significance is

[15] Eye-movements not concerned with viewing or tracking an object in the external environment.

the viability of tracking the eyes of both the subject and interviewer. As the technology becomes commonplace and experimenters become more ambitious in terms of how technology can facilitate their work, it is essential to minimise the required effort to setup such experiments. Finally, in light of the aforementioned development in requirements by researchers, the ability to extend the eye-tracker to 3D may prove useful.

Thus, in Chapter 2, the need for a specific type of eye-tracker was established and in Chapter 3, current eye-trackers were surveyed in search of one that would fulfil all of the above requirements. Failing to find one, the main objective of the current research work was established and in Chapter 4, a novel set of feature extraction algorithms were presented for extracting the location of the pupil, the iris radius and location of eye corners from images taken from an actively-illuminated head-mounted eye-tracker. This eye-tracker was based on a low-cost hardware design previously published by Babcock and Pelz (2004) thus satisfying the cost-related requirement early on.

First, the pupil is converted to binary using simple global thresholding. Connected components in the image are found using the common labelling algorithm and an ellipse is fitted on each component to find the one with the best fit and thus select the pupil. The contour is then refined using an active contour (also known as snake), before the final pupil location is calculated. Then, the iris radius is calculated using an adaptation of edge strength by Zhou and Pycock (1997), originally developed to segment images of cells. The eye corners are located in the image by making use of the partial x- and y- derivatives and initiating a search in groupings of local maxima. The accuracy of both the iris radius and eye corners search is increased by collecting several candidates and using a novel statistical algorithm to filter outliers that is simple and yet effective. Finally, the 2D gaze direction is calculated using only one calibration point which can be extracted without explicit instructions to the subject and thus the required ease of use and subject calibration transparency mentioned earlier is achieved.

The REACT eye-tracker is one of very few eye-trackers to detect and use the eye corners as reference points to calculate the gaze direction in 2D and the only one to do so from images taken with an actively-illuminated head-mounted eye-tracker; as extensively discussed throughout Chapter 4, the appearance of the eye and the inner eye corner in particular is significantly different close-up than it is from far and the eye corners cease to appear as symmetric. This is a

significant development in eye-tracking as the eye corners are always visible, in contrast with the glint which can fall onto the sclera and be very hard or impossible to detect.

Moreover, accurately calculating the iris radius means that by incorporating a 3D model of the eye and providing more calibration points, should future research require it, the eye-tracker can be extended to calculate 3D gaze in a similar fashion to Wang *et al.* (2005). The use of features versus an appearance-based model also means that the eye-tracker can be adapted to other potential mainstream or sub- applications without much effort.

In Chapter 5, the evaluation of the eye-tracker took place. Specifically, the accuracy of the feature extraction was assessed both independently and as a whole. The eye-tracker achieved a practical level of performance that renders it acceptable for use in the target research application(s). In a comprehensive test of range and accuracy, the REACT eye-tracker was found able to track eye-movements in the complete viewing range of the participants (approx. ±56°/±52°) with an average error of approx. 5° over the whole range and an average error of approx. 1.5° within a narrow field of view (±30°/±20°).

The practicality of the eye-tracker for the target application was further demonstrated in Chapter 6, where a pilot study was designed and served as a case study of a real-world application. Algorithms to select and classify the subject's eye-movements that are relevant to the experimenter were introduced; the classification was based on the eight (8) classes of the EAC model. Last but not least, a basic evaluation of the eye-tracker's usability and impact on the perceived comfort of the subjects was performed and yielded satisfactory results.

While a formal study of the improvement over a human rater was not performed, it should follow that the eye-tracker is able to more accurately track eye-movements, especially when performed quickly and in complex sequences. Thus, the REACT eye-tracker is able to facilitate extensive and in-depth research of non-visual eye-movements.

The implications for the computer vision components are several. First, it was demonstrated that reference-based eye-tracking is possible over a large field of view by using the eye corners are reference instead of the glint. Second, the edge strength algorithm used originally for cell segmentation was applied in a completely new context and problem. As these algorithms are further improved, or with better hardware, higher levels of accuracy can potentially be achieved.

FUTURE WORK

Future work can be divided into work related to enhancing the eye-tracker and work related to the investigation of the EAC model.

In terms of further enhancing the eye-tracker, the accuracy of the eye corner detection algorithm can be further improved by developing an algorithm to refine the eye corner locations using the full scale image, by performing a local search within a small window centred at the eye-corner location found at the reduced scale images.

Also, accuracy would be further increased by placing the camera straight in front of the eye, with the lens coplanar to the front surface of the eyes (i.e. if we assume a reference system at the eyeball centre, the camera would be placed at a location $[-x, 0,0]$, with the x-axis going into the person). However, since this would significantly block the subject's field of view and thus make the eye-tracker significantly invasive, an alternative solution would be to estimate a three-dimensional transform that would transform the feature-points to the location they would appear to be at *if* the camera was placed that way. The latter transform could perhaps be estimated by creating a model of extreme eye-movements and collecting several subject calibration points. As the extra calibration would come at the cost of added invasiveness, one would need to be careful in designing such an algorithm. In addition, such an algorithm would most likely help resolve ambiguities in the classification of eye-movements.

The usability assessment of the eye-tracker (Chapter 5) is a rare occurrence in eye-tracking literature; usually, usability is taken for granted. That is not to say that care is not taken by researchers while designing the eye-trackers to minimise interference. However, no *formal* study has taken place to assess just how invasive eye-trackers are and how they may affect the person's behaviour. Thus, eye-tracking system designs have to be no longer limited to functional requirements and pay more attention to non-functional requirements. In the case of the REACT eye-tracker, different frames (including metallic ones) and designs must be tested with a concurrent assessment of their impact in perceived and actual subject invasiveness (perceived being that reported by the subject on questionnaires and actual being any change in behaviour found through scientific means).

One possible direction in actually reducing the invasiveness of the eye-tracker is to explore the uses of a wireless technology to transmit the video to a recording device. With the development of low-powered wireless protocols with relatively high bandwidth such as Bluetooth 3.0, this is certainly a possibility that appears feasible and must be explored.

It was also noticed that while the eye-tracker is not dependant on the exact positioning of the camera in relation to the subject's eye, results tend to be more accurate when the camera is better placed. This is also the case with most, if not all, commercial eye-trackers and two have been tested by the author. It would be useful to develop feedback mechanisms that help the experimenter to adjust the eye-tracker such that the best possible results are achieved.

Of course, the most obvious future work is a full-scale NLP experiment. As discussed earlier, before attempting to directly examine the EAC model again, a questioning methodology needs to be designed, most probably based on procedures from the field of psycho-phenomenology, which is concerned with the exploration of subjective experience. Further, the eye-tracker enables past research with regards to the behaviour of eyes during conversational tasks to be made concrete and accurate conclusions to be drawn.

With the continuous development of eye-tracking, a transparent technological solution where a speech recognition engine is combined with automated eye-tracking to enable fully automated analysis of an experiment similar to that of the case study of Chapter 6 can be sought for. Such a development would further encourage researchers to perform the required experiments and not be burdened by the manual work involved in transcribing the speech, selecting and rating the eye-movements and synchronising them to the text – not to mention that the result of such a task would be at best crude if done manually.

If any truth is found in the EAC model, numerous doors will open for human-computer interaction or perhaps human-computer *training*. For example, in a study by Malloy (1987) it was found that spelling performance could be significantly be improved by visualising the word at the same time as directing the person to look up and to their left. One cannot help but to envisage a computer system, which with the aid of the REACT eye-tracker and through the use of speech synthesis can consistently enforce this behaviour in pupils while at the same time providing them with interesting spelling tasks and giving feedback in terms of their spelling score – a system not that different to an educational game.

APPENDIX A: EYE-TRACKING HARDWARE

A recent attempt to build a low-cost, lightweight eye-tracker was made by Babcock and Pelz (2004). The latter eye-tracker comprised one eye camera (monocular) and one scene camera which are mounted on the subjects' head on a glass frame and whose images are multiplexed into one video stream and recorded onto videotape for offline processing. In combination with the Starburst algorithm (Li *et al.*, 2005), it enables the eye-tracker user to track the subject's eye-movements in any natural environment. The entire system is also lightweight enough to package into a backpack and hence allows the subject to be mobile. After performing off-line processing, the eye-movements could be overlaid over the scene camera video.

The REACT eye-tracker presented in this thesis is a modified, scaled-down version of the eye-tracker presented by Babcock and Pelz (2004). Why this design was chosen is explained in detail in chapters 2 and 3. This appendix briefly presents both eye-trackers as well as their similarities and differences; Table 31 summarizes. Figure 48 shows a photograph of the experimenter wearing the eye-tracker.

Both eye-trackers are monocular but they can be easily extended for binocular eye-tracking but this would add to their invasiveness to the subject's field of view as well as complicate post-processing if the two eye-movement data sources were to be combined. Thus, a close-up video of the eye and its movement is recorded in both cases through a Supercircuits PC206XP monochrome pinhole camera; the relevant camera's properties are summarized in Table 32.

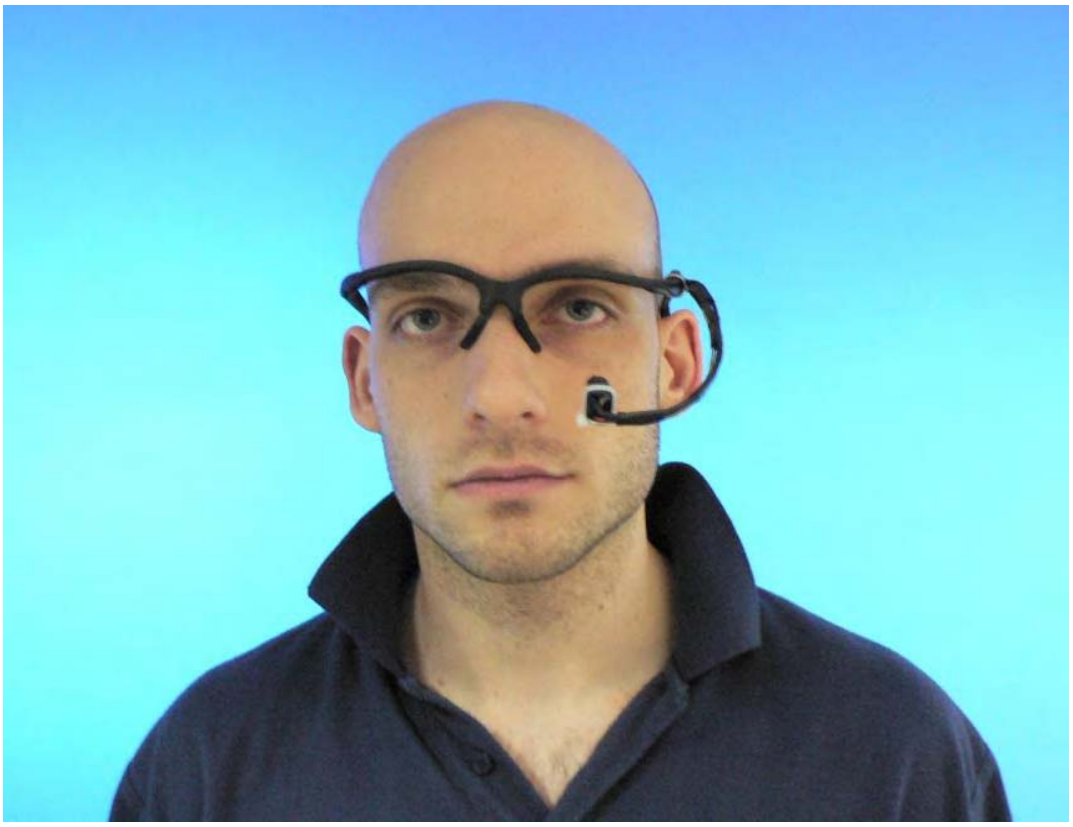


FIGURE 48: PHOTOGRAPH OF THE EXPERIMENTER WEARING THE EYE-TRACKER.

TABLE 31: SUMMARY HARDWARE COMPARISON OF THE EYE-TRACKER PRESENTED HERE AND THE EYE-TRACKER DEVELOPED BY BABCOCK AND PELZ (2004)

Hardware aspect	Babcock and Pelz (2004)	REACT Eye-tracker
Number of Eye Cameras	Monocular	
Eye Camera Type	Supercircuits PC206XP monochrome pinhole camera	
Scene Camera	Supercircuits PC53XS colour camera	None
Scene Reference Pointer	Laser diode projection system	None
Infrared Sources	One	
System Mount	Safety glasses	
Power Source	12V DC – unknown	12V DC power supply

TABLE 32: SUPERCIRCUITS PC206XP CAMERA SPECIFICATION

Image Sensor	¼" BIWCMOS
Video System	EIA/CCIR
Effective Pixels	510x492 (EIA) / 500x582 (CCIR)
Scanning System	2:1 Interlace
Resolution	640x480
Synchronization	Internal
Horizontal Sync Freq.	15.734 kHz (EIA) / 15.626 kHz (CCIR)
Vertical Sync Freq.	60 Hz (EIA) / 50 Hz (CCIR)
Video Output	1Vp-p. 75 Ohms composite
Lens	3.5mm/F2.6 (Standard), 3.1mm/F3.4(3T)
Power Supply/Current	DC 12V / 20mA (max)
Dimensions (WxH)	8.5mm x 8.5mm
Weight	10gr

The PC206XP camera encompasses some of the most desirable properties for a head-mounted eye-tracking system.

- At only 10 grams, it is a very lightweight camera and thus decreases or even eliminates any subject discomfort. The well-respected commercial EyeLink-II eye-tracker which is also head-mounted weighs an enormous 420 grams (SR Research, 2009). In our preliminary tests with the EyeLink-II, this weight disadvantage often caused subjects to complain, ask for a break after a short time (10-15 minutes) of wearing the eye-tracker or even ask to discontinue the experiment. The total weight of the REACT eye-tracker is approximately 60 grams which allows for prolonged experiments with little to no user discomfort. In fact, as discussed in the hardware usability evaluation section of Chapter 5, most subjects answered that they would be open to taking part to a 20-minute experiment where they would have to wear the eye-tracker at all times.

- Its small weight also allows it to be mounted on a lightweight, non-metal frame, such as safety glasses. This also contributes to the total weight of the eye-tracker headgear as well as to a positive user attitude towards the eye-tracker.
- The camera's small size is also reflected by its dimensions; at a square 8.5mm, it can easily become part of the subject's field of view and it will rarely obscure any objects in the environment from the subject. In combination with the camera being off-centre and thus not blocking the subject's central field of view, subjects are barely aware of the existence of the eye-trackers existence in their visual field. As discussed in the usability evaluation section, in general, subjects answered that they were unaware of the tracker's existence in their visual field.
- Despite its small size, the camera is able to capture at a relatively high-resolution: 640x480 pixels interlaced video at 29.97 frames per second. After de-interlacing, this becomes 640x240 at 59.98 fields per second. Such resolution efficiency can be said to be ideal for a head-mounted eye-tracker that is not targeted to analyse high-speed eye-movements.
- The camera itself requires a power source capable of delivering 12V DC and consumes a maximum current of 20mA. As before, such low requirements allow the minimization of the head-mounted hardware.

Both the REACT eye-tracker and the eye-tracker by Babcock and Pelz (2004) use a combination of an infrared light source pointing towards the subject's eye and a non-infrared filter mounted on the camera lens to introduce the dark pupil effect as explained in past chapters. The effect of capturing the eye using the infrared spectrum is described in extensive detail in the next section. On the hardware side:

- A standard infrared (IR) LED that transmits light at a wavelength of 940nm is used as the illumination source. It requires 1.2V to power it and provides a safe, for the subject's eyes, amount of infrared light. When illumination regulations come into effect (Hansen and Ji, 2010), it is almost certain that the REACT eye-tracker will require no modifications in order to pass.
- The non-infrared filter attached to the camera lens is a Kodak Wratten 87c equivalent (Babcock and Pelz, 2004), which blocks most (400-700 nm) of the natural light spectrum

(380-750 nm; Starr, 2005) and thus allows only the infrared light emitted by the IR LED to pass.

The only two components that require power in the REACT eye-tracker are the PC206XP camera (12V) and the IR LED (1.2V). A standard 220V AC input, 12V DC output power supply was used, in conjunction with a power transformer that converts 12V to 1.2V DC to power the LED. If our application required portability, this power supply could have easily been replaced by a series of 8x1.5V batteries. The circuit diagram for the power transformer is shown in Figure 49 below.

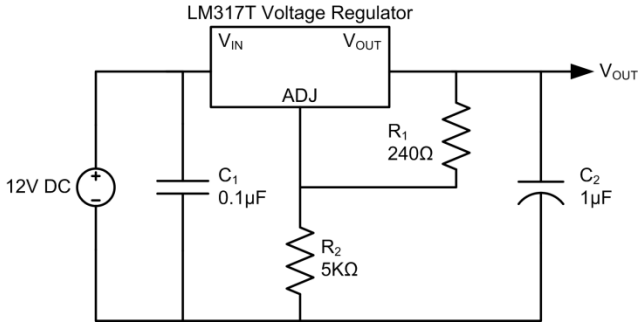


FIGURE 49: POWER TRANSFORMER CIRCUIT DIAGRAM.

Mounting the eye-tracker and cabling

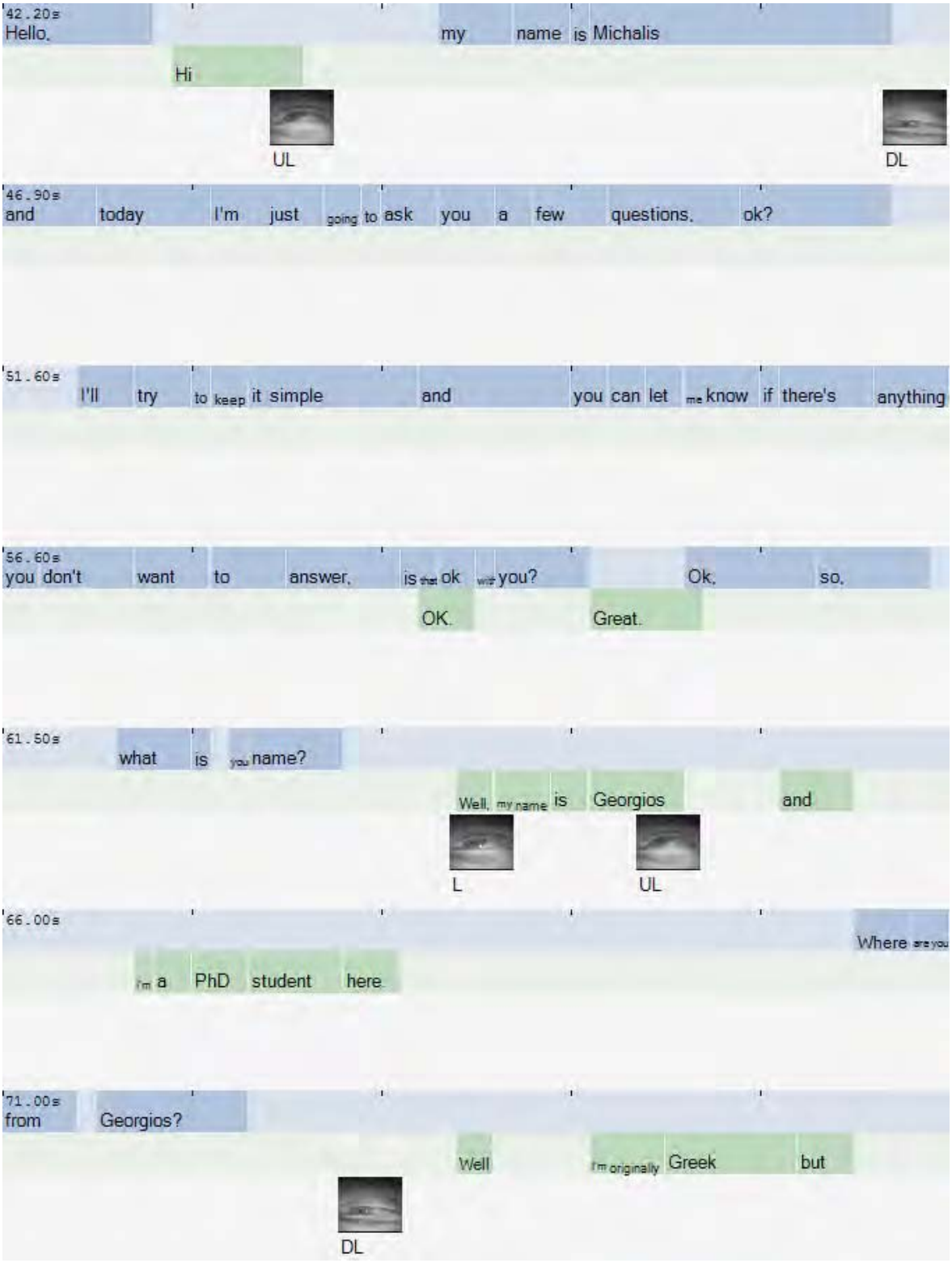
Since the REACT eye-tracker is so lightweight, it can be safely and easily mounted on a pair of plastic safety glasses. In fact, with the absence of the scene camera and laser diode (Babcock and Pelz, 2004), it is even possible to remove the plastic glass from the front of the safety glasses and just use their frame. This significantly reduces the obstruction of the subject's field of view; even though the plastic glass is clear, it adds to the experience of "wearing". In other words, with the plastic glass removed, some subjects may even forget they are wearing the eye-tracker after quickly acquiring a level of comfort with them (as discussed in the usability evaluation).

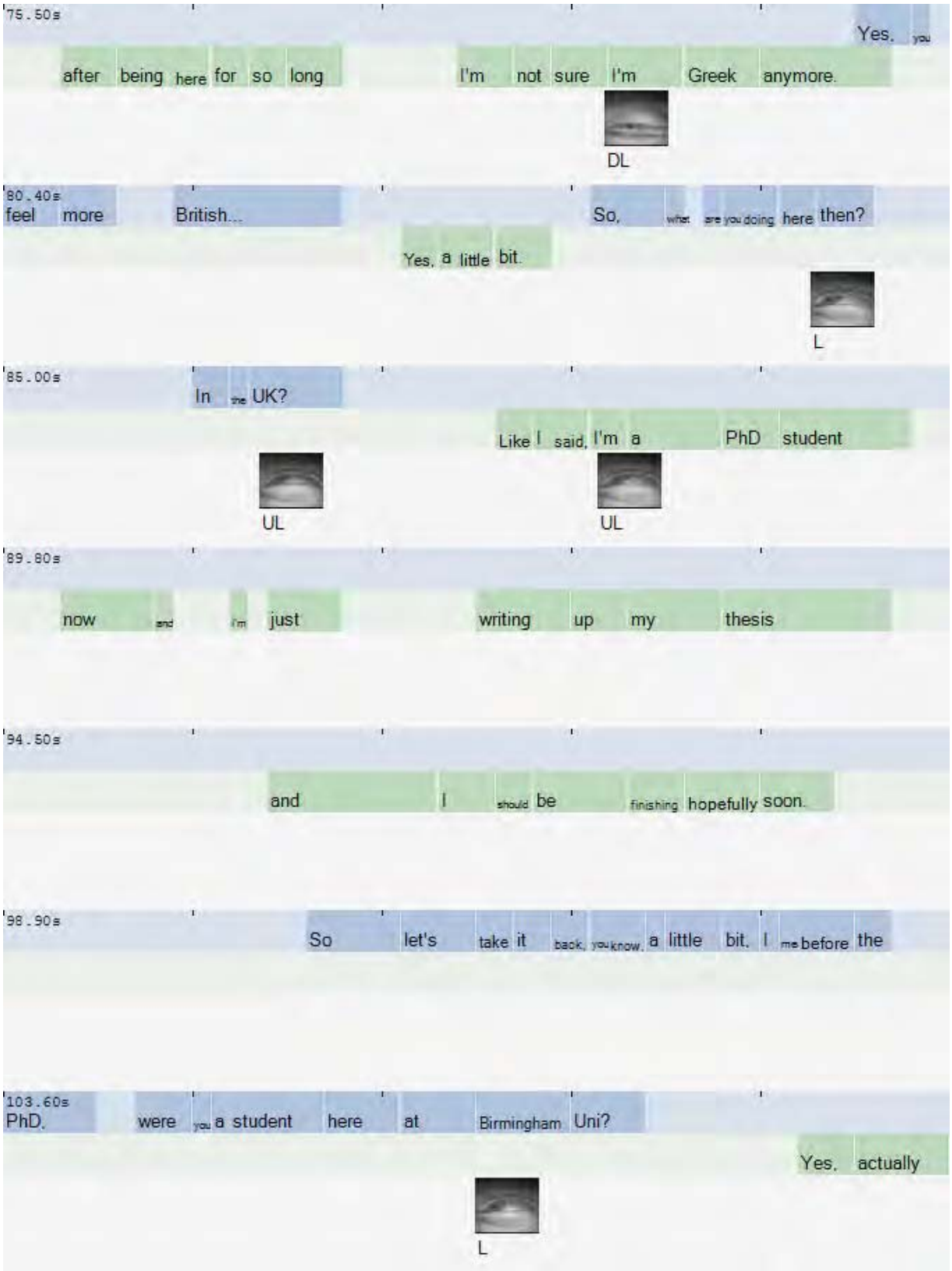
The cabling between the eye-tracker headset and the ground components, if used inappropriately, can cause discomfort to the user. Four cables are required:

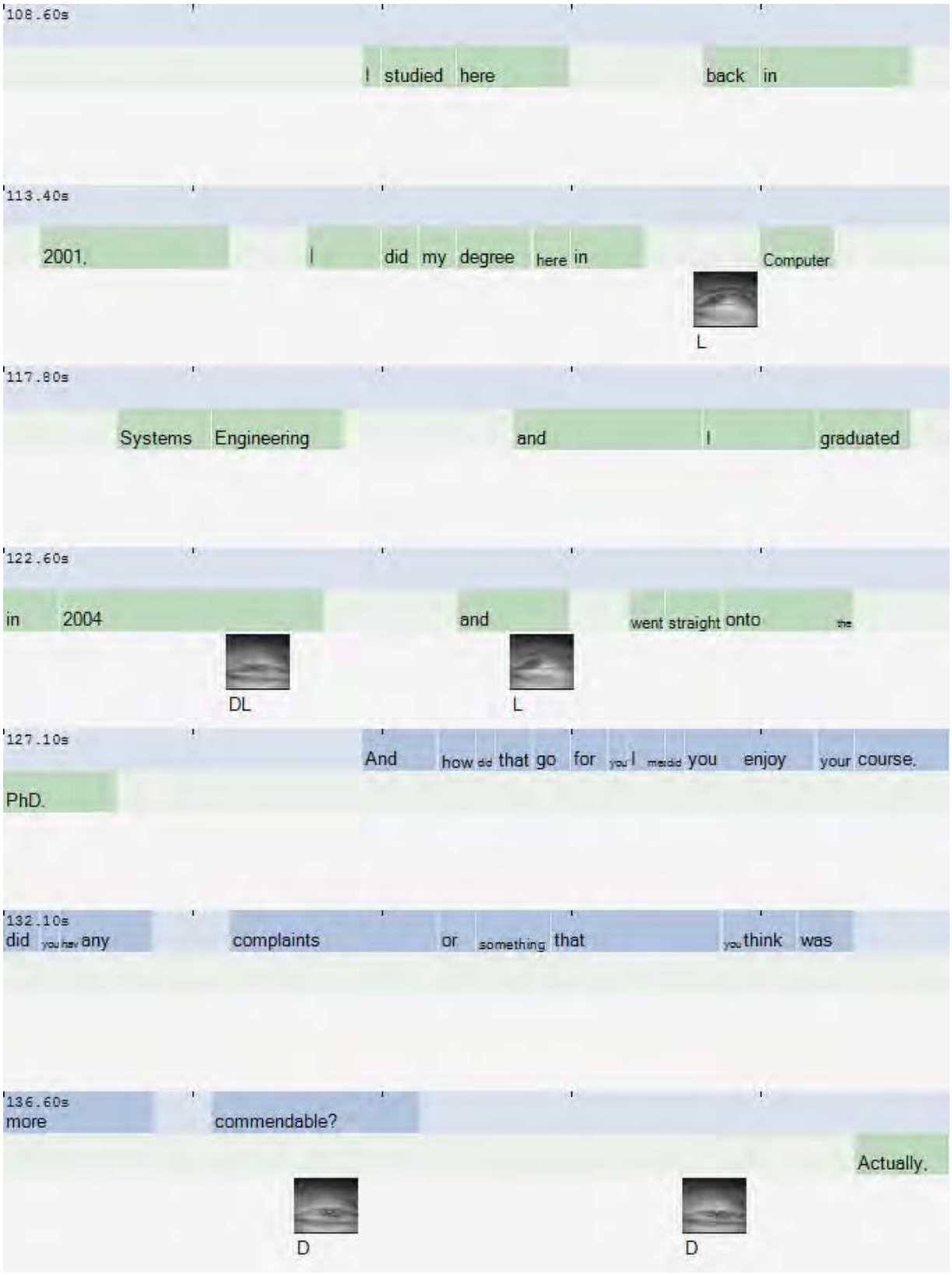
- Ground (common)
- 12V+ (power source for the camera)
- 1.2V+ (power source for the IR LED)
- Video signal carrier

For example, if the cables are too heavy, they can tilt the safety glasses as the cables can often weigh more than the frame itself. Similarly, if the cables are too thick, they can be a constant reminder to the subjects that they are wearing the eye-tracker. On the other hand, if the wrong types of cables are used, the eye-tracker will not work; specifically, if the video is carried in a non-shielded wire, the video signal does not get across intact due to electrical interference. Thus, a shielded 4-core wire is used.

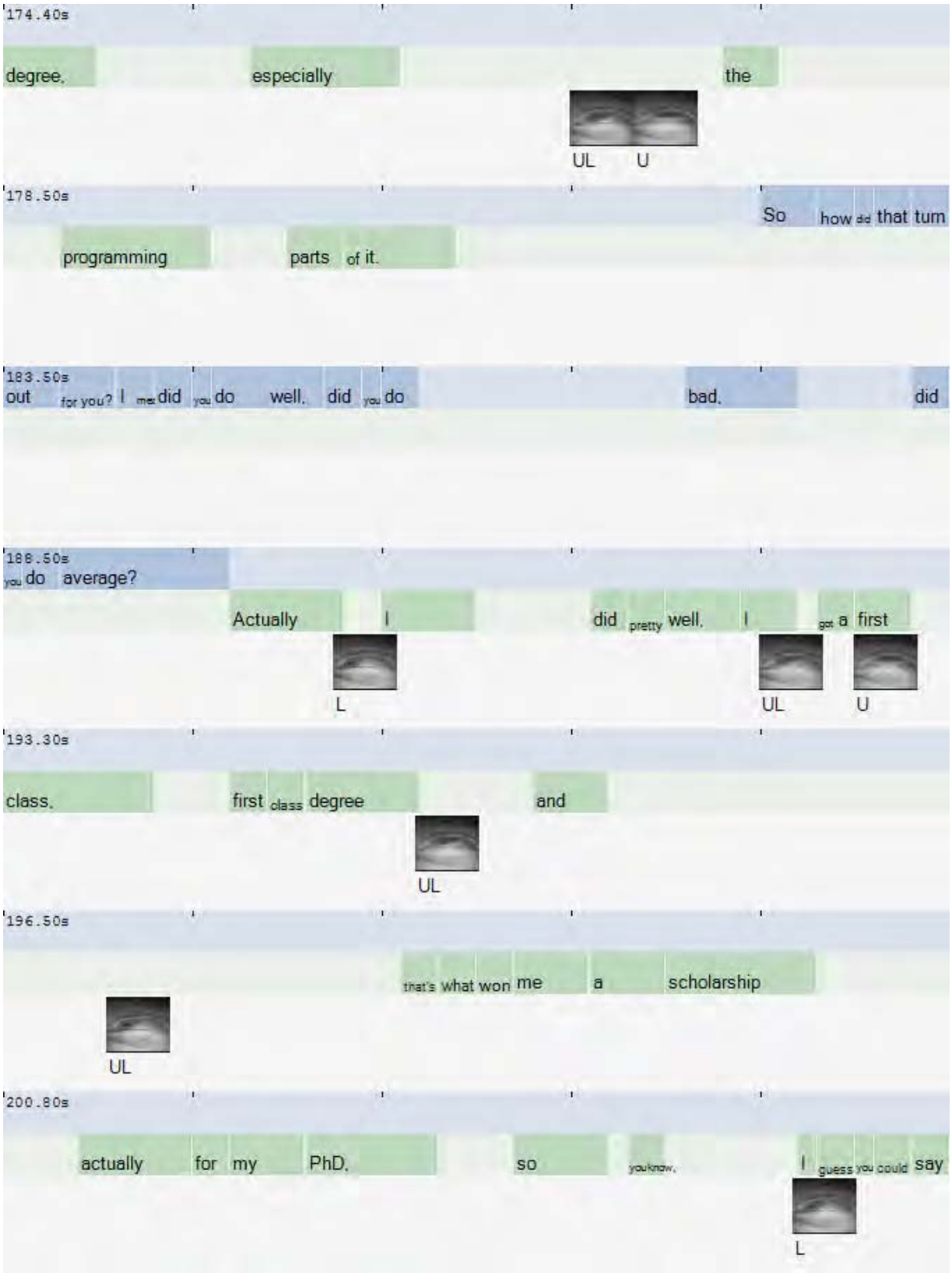
APPENDIX B: CASE STUDY FULL TRANSCRIPT




75.50s
Yes, you
after being here for so long
I'm not sure I'm Greek anymore.
DL
80.40s
feel more British...
So, what are you doing here then?
Yes, a little bit.
L
85.00s
In the UK?
Like I said, I'm a PhD student
UL
UL
89.80s
now and I'm just
writing up my thesis
94.50s
and I should be finishing hopefully soon.
98.90s
So let's take it back, you know, a little bit, I me before the
103.60s
PhD, were you a student here at Birmingham Uni?
Yes, actually
L


108.60s
I studied here back in
113.40s
2001, I did my degree here in Computer
L
117.80s
Systems Engineering and I graduated
122.60s
in 2004 and went straight onto the
DL
L
127.10s
And how did that go for you I mean you enjoy your course.
PhD.
132.10s
did you have any complaints or something that you think was
136.60s
more commendable?
Actually.
D
D

141.60s
yes, I had a very good time during my degree, I mean, it was very
D
146.10s
busy. we had a lot of work
150.80s
always so I didn't do probably a lot
L
155.40s
outside sort of studying. I mean, you know, I
160.30s
was going out every so often but I wasn't partying every night sort of
L
165.30s
thing and I really enjoyed actually
169.70s
the courses and everything we had to do for our
UL


174.40s
degree, especially the
UL U
178.50s
So how did that turn
programming parts of it.
183.50s
out for you? I me did you do well, did you do bad, did
188.50s
you do average?
Actually I did pretty well. I got a first
L UL U
193.30s
class, first class degree and
UL
196.50s
that's what won me a scholarship
UL
200.80s
actually for my PhD, so youknow, I guess you could say
L

205.80s
So it wasn't too bad then?
I did fairly well. No, it was OK.

UL

210.50s
Right, so what about, you know let's stay a little bit on you know, this

214.40s
student lifestyle. I mean obviously, you had

218.60s
different classmates, different people you met.

223.40s
Do you keep track of them right now, do you chat every

228.30s
so often?
Actually. from the people

UL

L

233.30s
who I was with during my degree there's only

L

237.30s
probably one person that is still here and he's only here
241.70s
because he's doing a PhD himself. Everybody else
L
246.30s
has left and
248.90s
I do sort of keep in touch with them.
253.80s
Thank God for Facebook really. Keeps us connected
UL
258.70s
So are you a Facebook
and we sort of catch up with each other every so often.
263.60s
addict then?
Well, I guess you could say that.
R DR
UR

268.40s

When I'm working all the time it's a good way to sort of

273.10s

relax and tune out a little bit so you know, check what everybody is doing and

277.90s

Right, right. Is there make some comments and you know, play around a bit.

282.70s

some particular (Facebook) application that you like to play with?

286.80s

Actually I'm not really into the Facebook games, I just

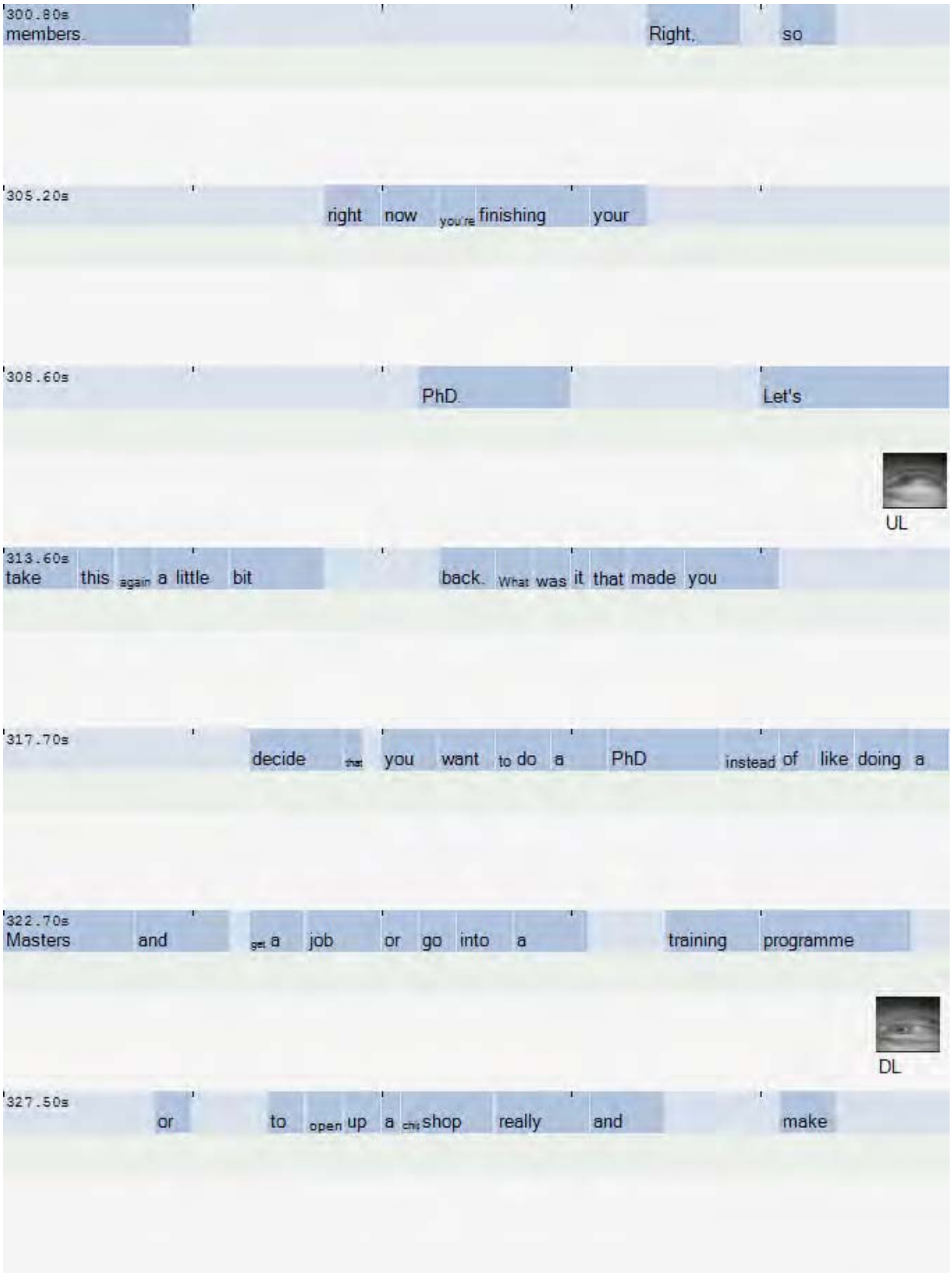
291.40s

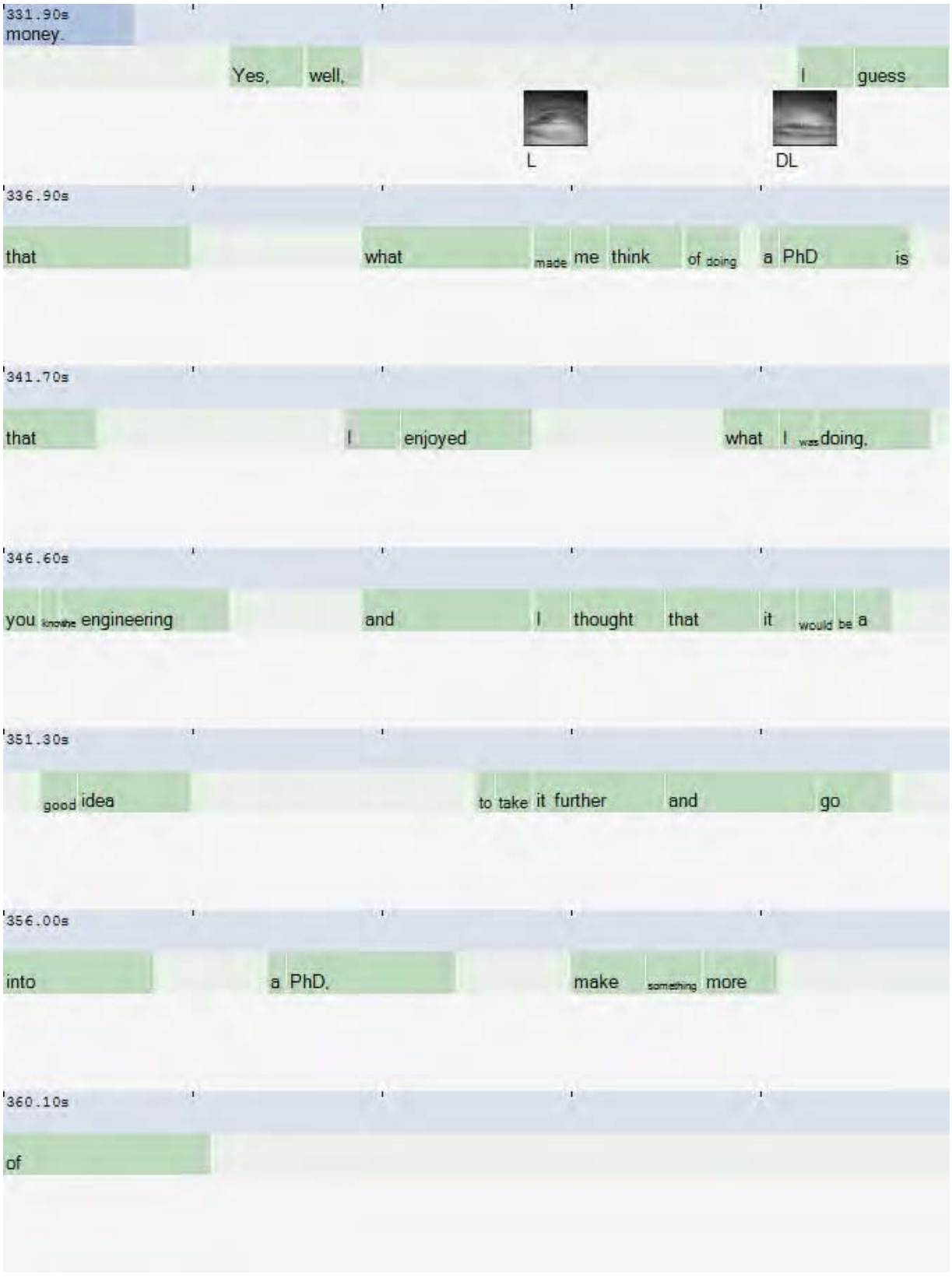
can't get into them. they just seem like a bit of a waste of time to

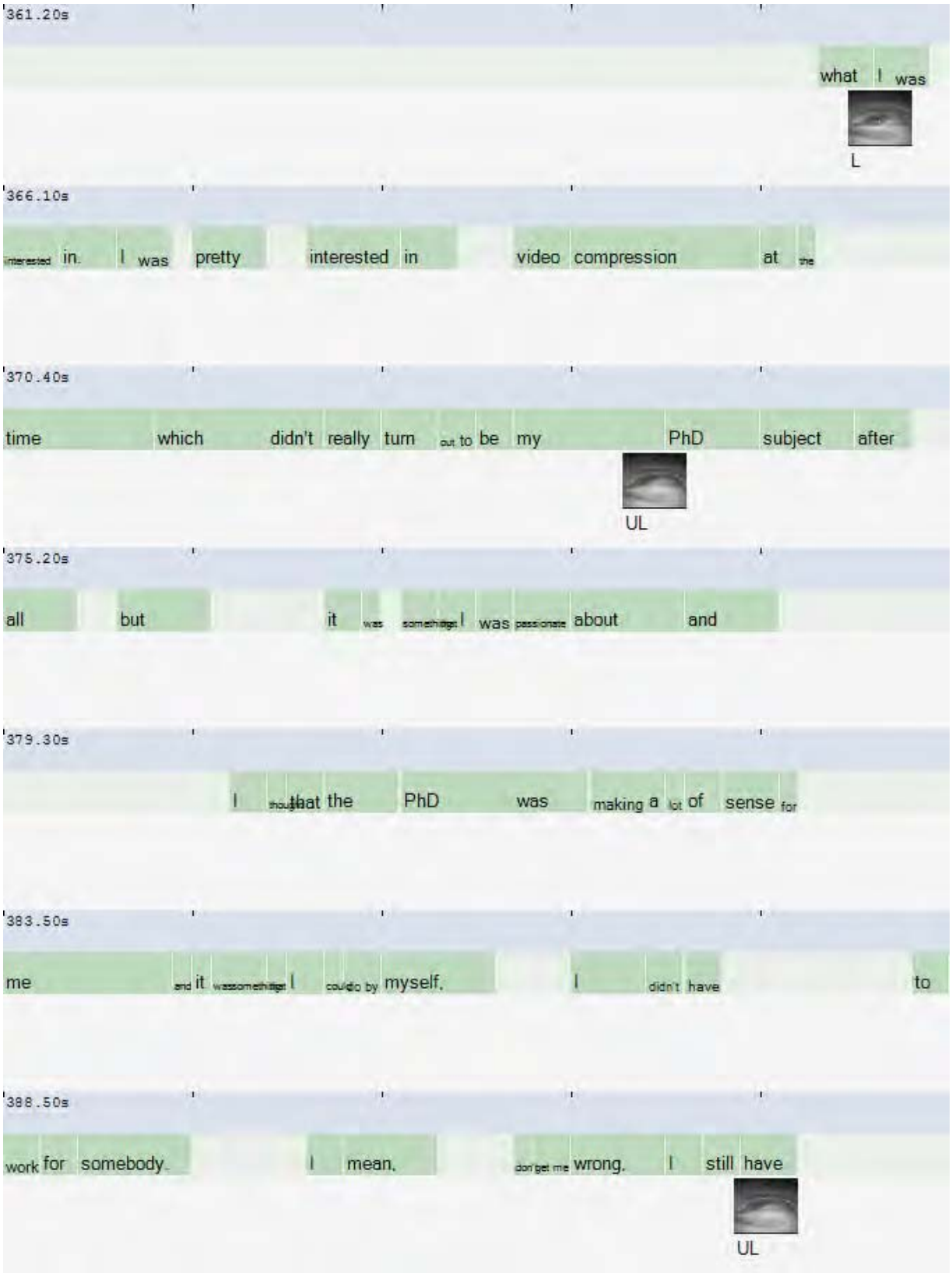
296.20s

So you not a Mafia Wars fan then? Because I'm looking to recruit "family" me. No.

UL


300.80s
members.
Right,
so
305.20s
right
now
you're
finishing
your
308.60s
PhD.
Let's
UL
313.60s
take
this
again
a
little
bit
back.
What
was
it
that
made
you
317.70s
decide
that
you
want
to
do
a
PhD
instead
of
like
doing
a
322.70s
Masters
and
get
a
job
or
go
into
a
training
programme
DL
327.50s
or
to
open
up
a
chip
shop
really
and
make


331.90s
money.
Yes, well, I guess
L
DL
336.90s
that what made me think of doing a PhD is
341.70s
that I enjoyed what I was doing.
346.60s
you knowthe engineering and I thought that it would be a
351.30s
good idea to take it further and go
356.00s
into a PhD. make something more
360.10s
of


361.20s
what I was
L
366.10s
interested in. I was pretty interested in video compression at the
370.40s
time which didn't really turn out to be my PhD subject after
UL
375.20s
all but it was somethingI was passionate about and
379.30s
I thought that the PhD was making a lot of sense for
383.50s
me and it wassomethingI could do by myself. I didn't have to
388.50s
work for somebody. I mean, don'get me wrong. I still have
UL

392.70s

had to be directed by my supervisors and

DL

397.70s

so on but it was something that I have chosen to do

401.90s

and the subject and everything. So it was something that I was interested in,

DL

406.70s

that sort of made sense in terms of my

411.30s

I me ofcourse there is

future plans.

416.20s

a difference between say the PhD and the

420.70s

undergraduate degree. One is a

425.40s
taught course and the other you're pretty much on your own, so
430.00s
how did this feel? The PhD process, compared
435.00s
to the undergraduate degree?
That's a good question
UL
439.90s
actually. I mean yes, it's very different
UL U UR R UL
444.90s
and I would say that,
449.00s
I mean both are very challenging but
453.90s
in very different ways. When I started university
L UL

458.90s
and I started doing my degree, it was a lot about
462.60s
getting used to,
467.10s
how to say this,
470.90s
getting used to a
UL
UL
475.90s
environment a different way of doing things compared to high school, you know, it was much more
UL
UR
480.60s
specialized even though we were doing a lot of different subjects, it was something that I was
485.50s
finally interested in rather than in high school where I do all the subjects and

490.50s

then from that going into a PhD. it was

494.90s

definitely even more of a funnel.

499.80s

you know going further specifically into.

504.80s

you know, what it is

509.80s

that I'm doing and I'd say that one of the

L

514.80s

major differences is that in the undergraduate degree.

519.50s

the sort of path if you like. you know.

524.50s

whatare you goingto do and how are you goingto do it, is laid out for you, you just have to do it and

UL

529.20s

the how and, you knowhow well is the main

UL

533.50s

challenge and that's still in the PhD but you also have to set yourown path.

538.30s

You knowthe supervisors are there to guide you a little bit

541.90s

but really you have to do everything by yourself

L

546.80s

and you have to understand how you goingto go from A to B to C

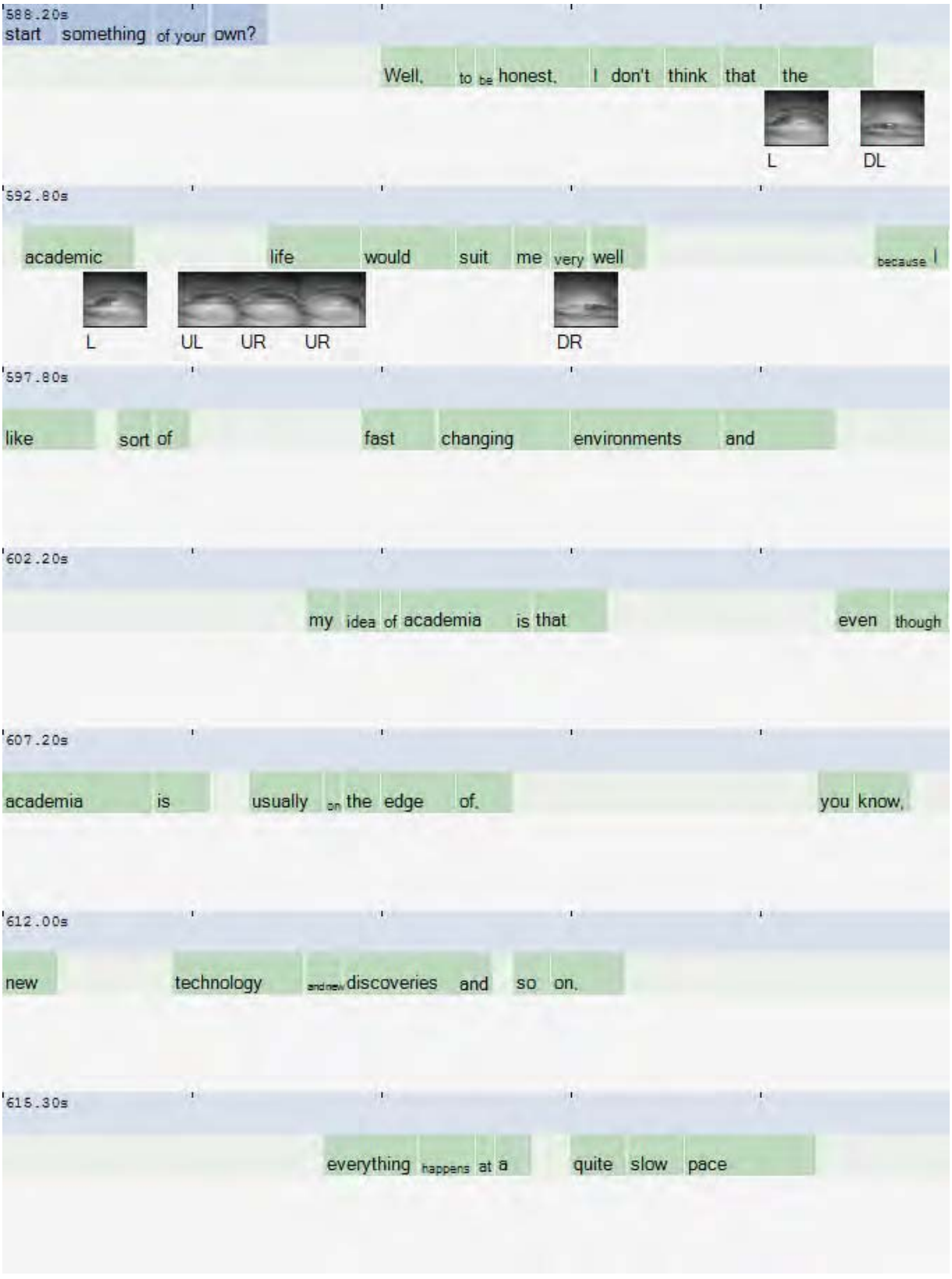
551.70s

and theneventually onto hopefully getting yourPhD

L

UL

556.40s
so, yeah, overall, very different I'd say. You have to be
UL
561.20s
very motivated, self-motivated and
UL
564.40s
definitely interested in what you are doing.
DR
D
DL
568.40s
Right, so ok, I mean we've looked at the past now
UR
573.40s
and we've looked at the present but I just want to ask you now about the
578.20s
future. I mean do you have any plans for after you finish? I mean, do
UR
583.20s
you want to be an academic or do you want to go to a company or


588.20s
start something of your own?
Well, to be honest, I don't think that the
L
DL
592.80s
academic life would suit me very well because I
L
UL
UR
UR
DR
597.80s
like sort of fast changing environments and
602.20s
my idea of academia is that even though
607.20s
academia is usually on the edge of, you know,
612.00s
new technology and new discoveries and so on,
615.30s
everything happens at a quite slow pace

619.60s
and that's something that I'm sort of not comfortable with.
UL DL L UL U UR DR
623.90s
So I don't really know
628.30s
exactly what'm going to be doing. there's no sort of firm plans
632.40s
but I definitely like to start something of my own, start my own
637.20s
business, obviously something to do with technology. I
UL
642.10s
don't know what exactly yet but there's a few options on the table
UL
647.10s
and I guess time will tell which one of these options I'll be
L

651.60s
pursuing. But I'm just taking, you know one thing at a time at the moment and just
656.50s
focusing on the PhD trying to finish it as soonas possible
UL
661.40s
Alright.
and then take it from there.
674.60s
So, ok then, let's skip the work bit
679.40s
and talk a little bit more about yourself. I mean, is there something that you like to do in
684.40s
your free time?
Actually I
L
UL
689.20s
used to have a lot more free
DL

694.20s

time and a lot more hobbies in the past well,

698.50s

because I had more free time, but I still

702.00s

keep it at a minimum I suppose

706.00s

and well first of all I really enjoy working out and going to the

711.00s

gym, I probably do that about three to five

716.00s

times a week when I can and you know it's just my time of the

UR

DR

721.00s

day when I just take time out to work out and relax

725.40s

afterwards, it can be very destressing for me.

729.00s

So especially when stressed. I go straight to the gym

734.00s

and I come back feeling pretty calm so that helps me a

UL

738.80s

lot and also I dance

742.80s

tango which I've been doing for

L DL

747.50s

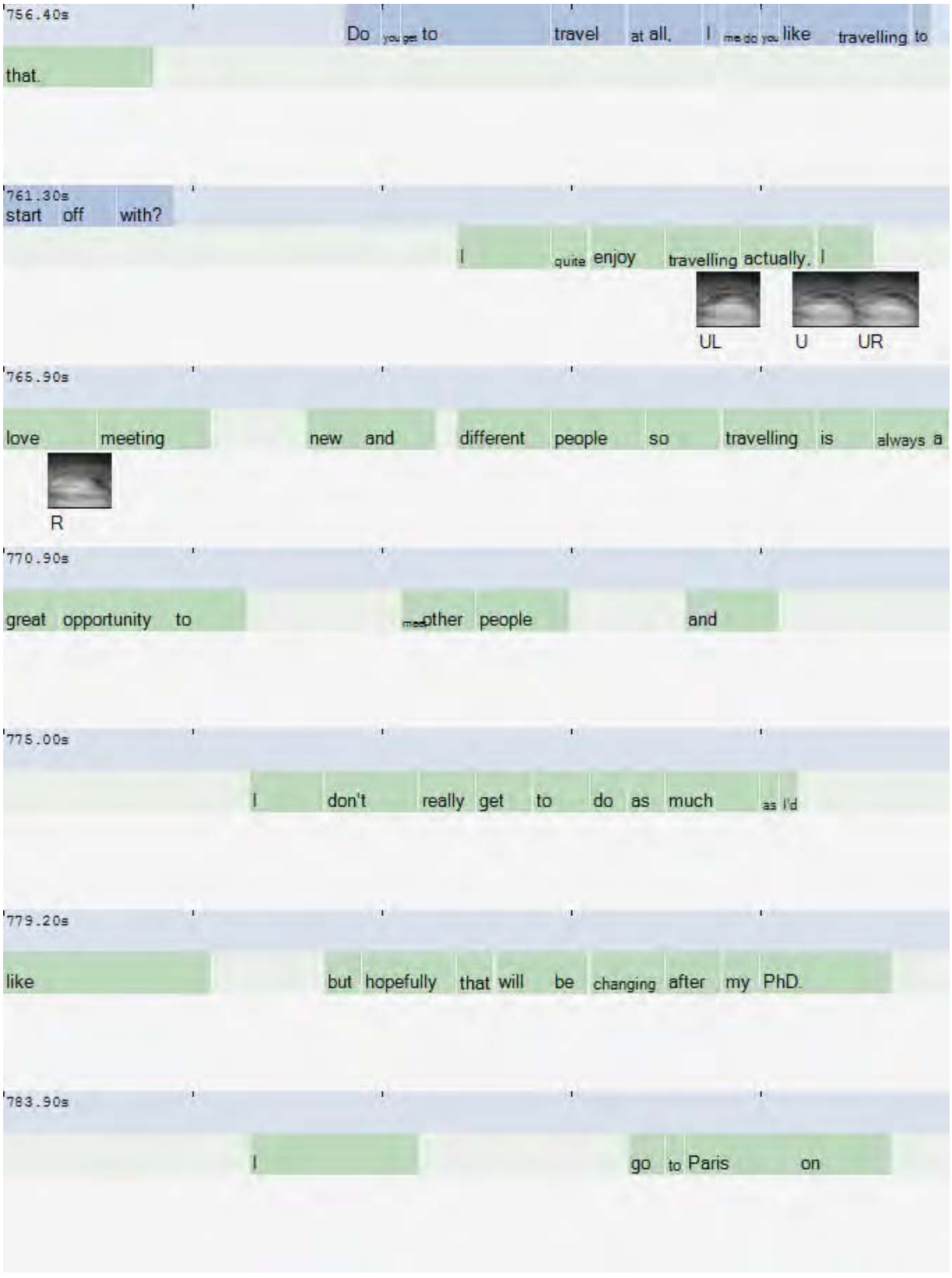
about five years now so it's a great great hobby of

R

UR

752.00s

mine and I really quite enjoy doing


756.40s
Do you get to travel at all. I me do you like travelling to that.
761.30s
start off with?
I quite enjoy travelling actually. I
UL
U
UR
765.90s
love meeting new and different people so travelling is always a
R
770.90s
great opportunity to meet other people and
775.00s
I don't really get to do as much as I'd
779.20s
like but hopefully that will be changing after my PhD.
783.90s
I go to Paris on

788.60s
every chance I have because the tango community there
UL
793.50s
is very big and I have a great time there.
798.20s
Is there any place that you've been to lately?
UR
DR
803.00s
Because I'm thinking of making some travelling of my own. I was
807.80s
looking for ideas.
Actually, what was the last place that I've been
UL
UR
812.60s
to? I think the last place I went to was Stockholm.
D
DL
817.30s
last Christmas actually and actually it's very

822.10s
Right, put my skis
snowy at this time of the year so,

826.90s
on.
yes, if you like snow that's definitely a good place to go and

DL

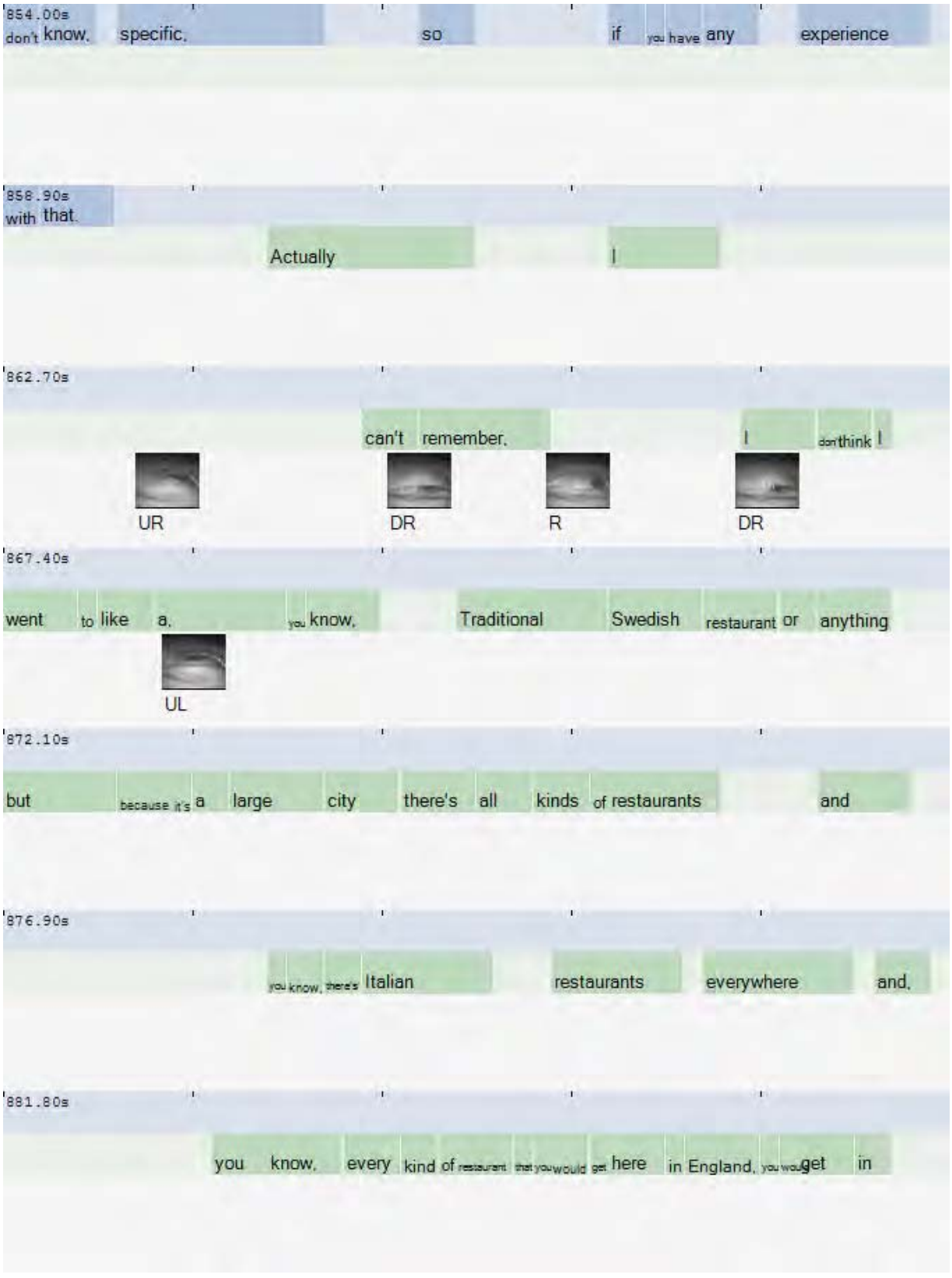
831.50s
the Swedish people are very friendly

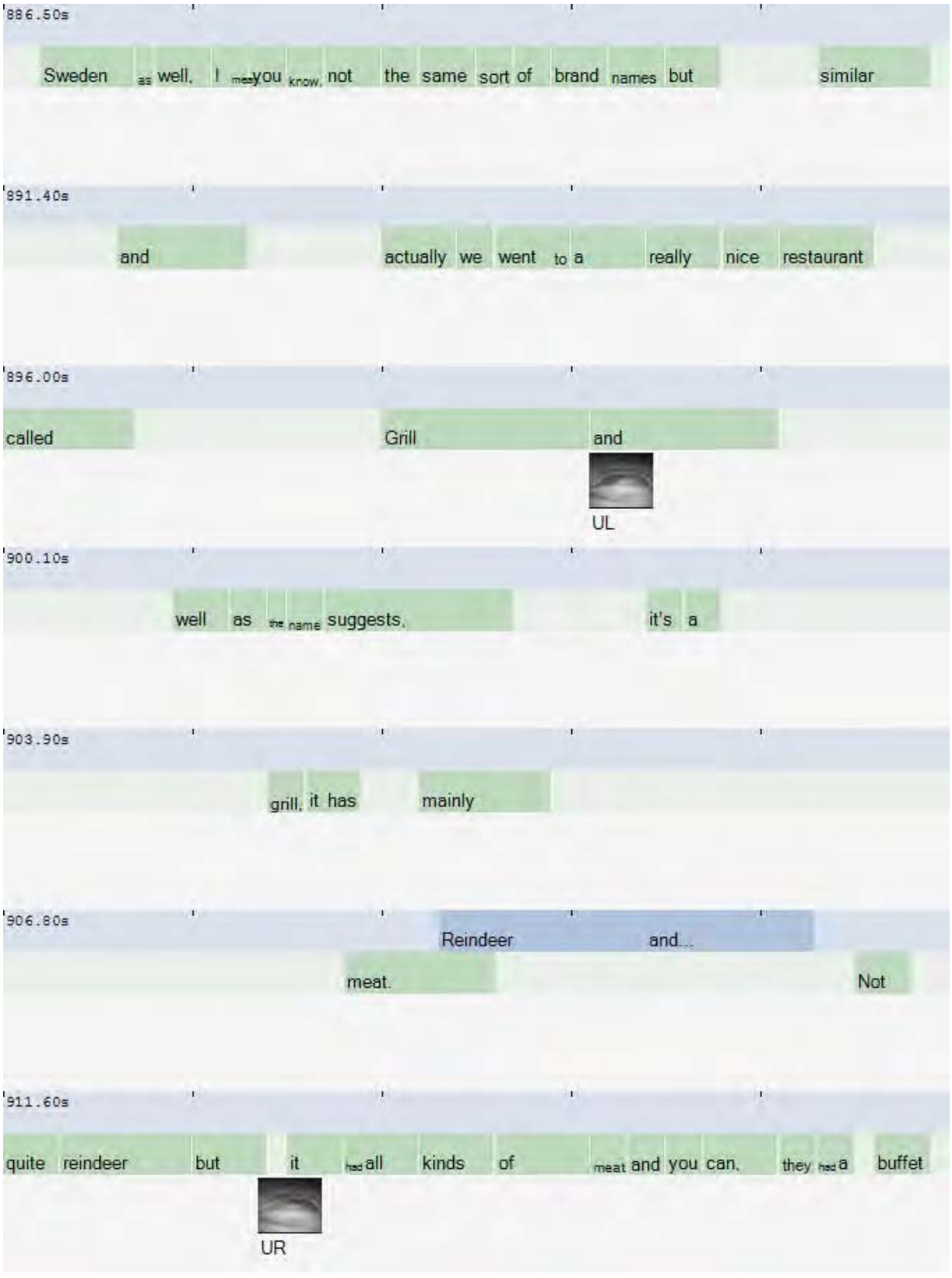
835.90s
so I had a good time and you can get some

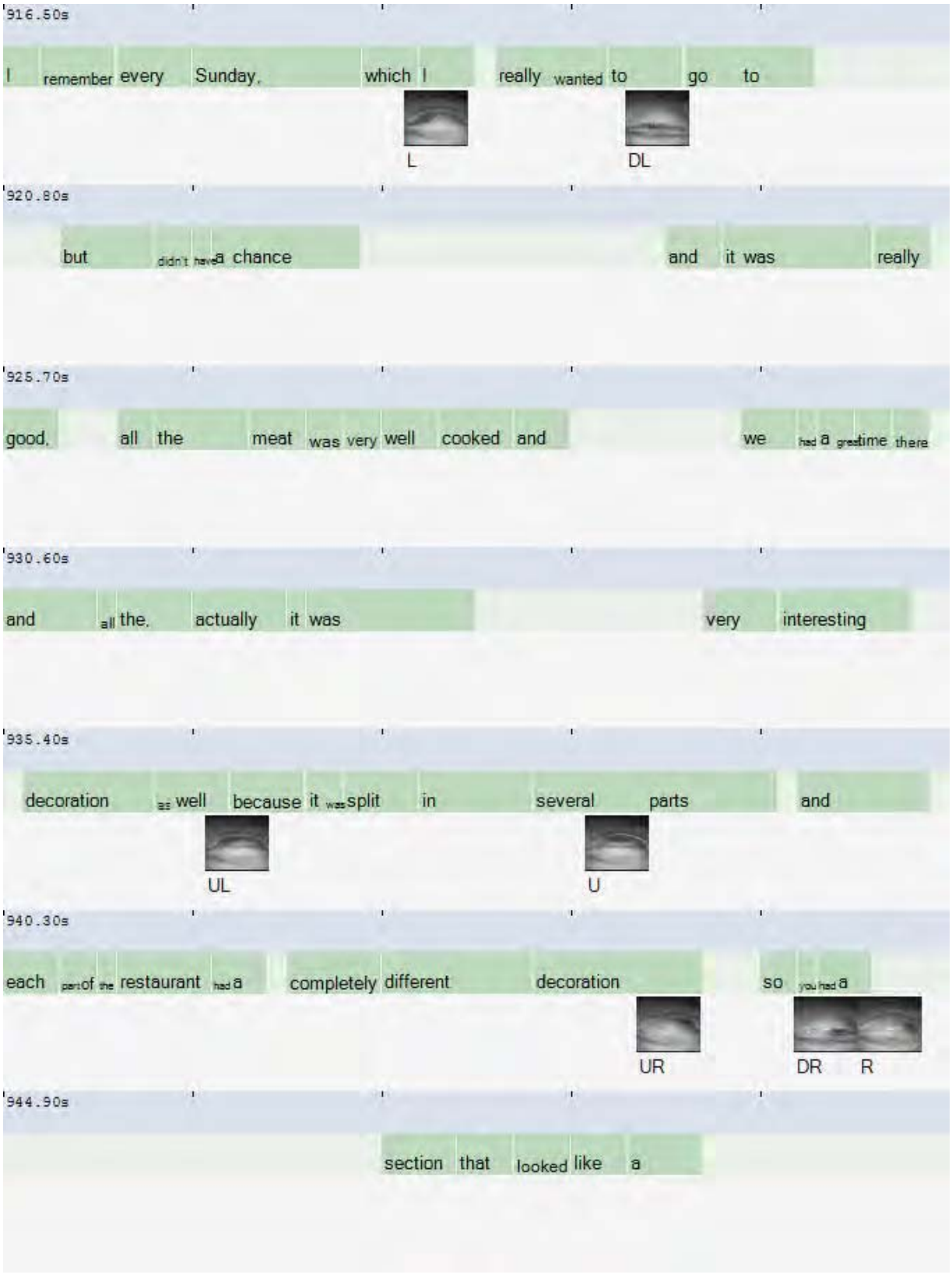
839.70s
fairly decent accomodation for fair price

844.20s
I've alw been wondering about these guys because
so it's a pretty good deal.

849.10s
they say it's a very nice country and all but the food is a little bit, I


854.00s
don't know, specific, so if you have any experience
858.90s
with that.
Actually I
862.70s
can't remember, I don think I
UR DR R DR
867.40s
went to like a, you know, Traditional Swedish restaurant or anything
UL
872.10s
but because it's a large city there's all kinds of restaurants and
876.90s
you know, there's Italian restaurants everywhere and,
881.80s
you know, every kind of restaurant that you would get here in England, you would get in


886.50s
Sweden as well, I mean you know, not the same sort of brand names but similar
891.40s
and actually we went to a really nice restaurant
896.00s
called Grill and
UL
900.10s
well as the name suggests, it's a
903.90s
grill, it has mainly
906.80s
Reindeer and...
meat. Not
911.60s
quite reindeer but it had all kinds of meat and you can, they had a buffet
UR


916.50s
I remember every Sunday, which I really wanted to go to
L
DL
920.80s
but didn't have a chance and it was really
925.70s
good, all the meat was very well cooked and we had a greatime there
930.60s
and all the, actually it was very interesting
935.40s
decoration as well because it was split in several parts and
UL
U
940.30s
each part of the restaurant had a completely different decoration so you had a
UR
DR
R
944.90s
section that looked like a

948.60s

French, sort of boudoir, kind of room and then

953.30s

another one that looked like a palace kind of

957.80s

thing and I can't remember what the other ones were but it was very very interesting to be

UL

962.80s

So really was an excuse for you to go from room to room and there.

U

UR

967.50s

just eat meat in different places. That's cool.

UL

971.80s

And do you listen to music, at all?

975.80s

Well, you know, everybody does I suppose to a

UR

R

980.50s

certain extent. I love listening to

DL

985.40s

music. I listen to the radio a lot because I like that. you know it sort of

UL

990.40s

changes and it suprises you and you anticipate what the next

995.40s

song is going to be an you wish that they play your favourite song and so on and so

1000.00s

forth so that is a

1004.30s

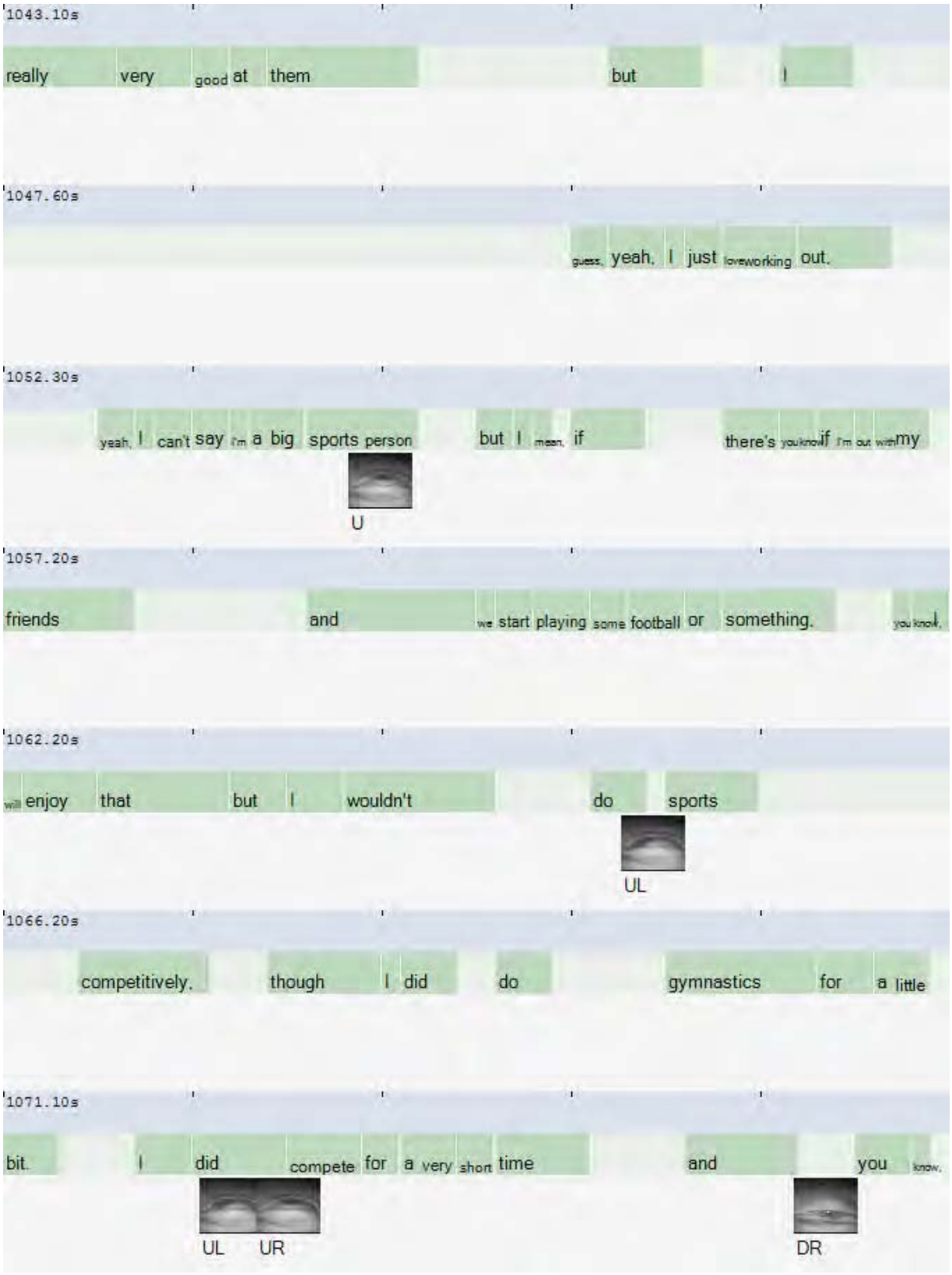
good music source for me I suppose but

UL

1009.30s

I also like to listen to, you know own music

1013.80s
collection
and
just find different
1017.40s
artists.
1018.90s
Ok,
and
1023.50s
what's you relationship
with
sport?
You just
UL
1028.50s
said that
you go
to the
gym.
Do
1033.50s
you like
sports?
I'm not a very
DL
L
1038.30s
big
team sports
person,
I
was
never
DL


1043.10s
really very good at them but I
1047.60s
guess, yeah, I just loveworking out.
1052.30s
yeah, I can't say I'm a big sports person but I mean, if there's you know if I'm out with my
U
1057.20s
friends and we start playing some football or something, you know,
1062.20s
will enjoy that but I wouldn't do sports
UL
1066.20s
competitively, though I did do gymnastics for a little
1071.10s
bit. I did compete for a very short time and you know,
UL UR
DR

1076.00s
that was competitive which I guess, you know against what I've just said but again
UL
1080.80s
gymnastics is a very individual sport, there's no sort
1085.80s
of teamwork in
UL
1089.90s
Right, and what about some time for yourself, I mean when you're
gymnastics as such.
1094.90s
at home, do you read books?
1097.50s
Well, I mean, like I
R
R
1102.30s
said, I don't really have the time anymore but when I do, I
UL

1107.00s
do like to read books. I haven't read one in a very long
1111.50s
time but again, hopefully after my PhD I have more free time and
1116.20s
Well, ok, that's all then,
read a bit more.
UL
1120.80s
thank you very much.
Thank you.
UL

APPENDIX C: PUBLISHED AND SUBMITTED PAPERS

Published papers include:

Diamantopoulos, G., Woolley, S. I., and Spann, M., 2008. A critical review of past research into the Neuro-Linguistic Programming Eye-Accessing Cues model. In: *Proceedings of the First International NLP Research Conference.* Surrey, UK.

Diamantopoulos, G., Spann, M., 2005. Event Detection for Intelligent Car Park Video Surveillance. *Elsevier Journal of Real-Time Imaging, Special Issue on Video Object Processing for Surveillance Applications*, 11 (3), pp. 233.

The paper that follows is for submission to the Special Issue of the Signal, Image and Video Processing Journal entitled "Unconstrained Biometrics: Advances and Trends".

Novel eye feature extraction and tracking for non-visual eye-movement applications

Georgios Diamantopoulos

School of Electronic, Electrical and Computer Engineering

University of Birmingham

Birmingham

B15 2TT

United Kingdom

+44 77 324 903 85

gxd186@bham.ac.uk

Sandra I. Woolley

Michael Spann

Abstract

The Neuro-Linguistic Programming Eye-Accessing Cues model suggests that there is a correlation between eye-movements and the internal processing mode that people employ when accessing their subjective experience. Past research on the model used direct viewing or at best video recording of the subject to select and rate eye-movements. The aforementioned methods are not only error-prone but also have a low-detail output compared to a modern eye-tracker (e.g. time information is missing) thus negatively influencing the reliability of the results. While a plethora of eye-tracker designs is available to date, none of them has been designed to track non-visual eye-movements (eye-movements that are a result of neuro-physiological events and are not associated with vision), which tend to range outside the normal visual field and thus perform poorly in such cases. Therefore, this paper introduces a set of novel algorithms for the extraction of relevant eye features (pupil position, iris radius and eye corners) that are combined to calculate the 2D gaze direction and to classify each eye-movement to one of eight classes from the model. The applicability of the eye-tracker is demonstrated through a pilot study that serves as a real-world

application case study. The performance of the eye-tracker is found to be practical for the intended purpose.

Keywords

Eye-movements, eye-tracking, neuro-linguistic programming eye-accessing cues, eye feature extraction, gaze

1 Introduction

Eye-tracking is a field that has been actively research for the past few decades and while eye trackers have thus changed dramatically over the last few decades, eyes are still mainly studied as a functional organism of vision. With the exception of studies in rapid eye-movements during sleep, it was only recently that eyes were studied as a part of the brain and its function (in the waking state). While there is a reasonable amount of studies relating eye-movements to speech activities such as reading text or maps, very few studies have involved the recording and tracking of eye-movements with *non-visual* task performance.

To be clear, the term *non-visual tasks* refers to those tasks which do not explicitly rely on vision to be performed [13]. Similarly, the term *non-visual eye-movements* refers to eye-movements concerned with non-visual tasks.

The Neuro-Linguistic Programming (NLP) Eye-Accessing Cues (EAC) model was introduced in [2] and suggests that the direction of non-visual eye-movements indicates the modality (visual, auditory, kinaesthetic) of their subjective experience a person is currently accessing. Simply said, it is said that when a person[16] is looking down and to their right, they are accessing a feeling associated with the experience they are talking about or examining internally.

While it cannot be denied that eye-movements are hard-wired to brain function, the NLP EAC model was not scientifically validated by its authors and in a critical review of EAC research that appeared at later dates and attempted to (dis-)prove the model, it was shown that the results

[16] This is a generalization offered by Bandler and Grinder (1979) for a cerebrally normally-organized right-handed person.

suffered from severe methodological and experimental flaws [4]. The main flaw that is relevant to this discussion is the use of direct viewing to record, select and rate the eye-movements, which has been the genesis of very significant limitations.

Studies relevant to the NLP EAC model and other such models would have benefited by the use of eye-tracking systems. However, selecting a suitable eye-tracking system for this task does not prove as easy. This is for several reasons, the most important one being that eye-trackers to date were designed to track visual eye-movements [7] [12] [15]. Whilst the classification of visual versus non-visual eye-movements is not significant in itself, visual eye-movements are normally bound by a much smaller field of view. By contrast, when a person is thinking his or her eyes will usually shift to one extremity of the eye socket (irrelevant of the direction). Thus, if the person was asked to look at the location indicated by this shift, he or she would have turned his or her head and performed a much smaller eye-movement to reach the target; this behaviour is largely undocumented. What is, however, documented is the tendency of subjects to shift their eyes when asked to answer a question (not related to a visual task) and before they return to looking at the interviewer [8].

Therefore, it is no surprise that eye-trackers designed to track visual eye-movements fail to track non-visual eye-movements of such extremities and the need for development of an eye-tracker that is able to track such eye-movements is introduced.

This paper is concerned with the development of the Robust Eye-Accessing Cues Tracker (REACT) whose main requirements are to maintain a high level of accuracy while tracking non-visual eye-movements as well as minimize invasiveness and cost. The REACT eye-tracker is head-mounted but very lightweight (approx. 60 grams) and is thus minimally invasive. A head-mounted approach was chosen over a remote camera one as to minimize cost and increase the resolution of the captured images and consequently the accuracy viable by the eye-tracker. Moreover, a remote camera requires that the head is tracked and not only does that increase the computational complexity but it also decreases the ability of the eye-tracker to track extreme eye-movements as discussed earlier. The hardware design is based on the low-cost eye-tracker presented in [1], with the scene camera and associated hardware removed. As such, no more time will be spent discussing the hardware design itself.

On the hardware-level, the eye-tracker works by illuminating the eye with infrared light while blocking most of the visible light spectrum with an infrared filter imposed over the camera lens. This produces the dark-pupil effect which allows for the easy detection of the pupil and, in a sense, offloads some of the processing to the hardware. The captured images are then processed in software and a set of three eye features are detected: the pupil centre and contour, the iris radius and the location of the eye corners. Finally, features are combined in order to calculate the 2D gaze angle and classify each eye-movement to one of eight eye-movement classes extracted from the EAC model [2]: up, up and left, left, down and left, down, down and right, right and up and right.

This paper is organised as follows. Section 2 describes the extraction of the eye features (pupil centre and contour, iris radius, and eye corners) in detail. Section 3 describes the calculation of 2D gaze and its classification to one of eight classes. Section 4 is concerned with the performance evaluation of the eye-tracker and Section 5 describes a pilot study experiment that was designed to serve as a real-world case study. Some concluding remarks are offered in Section 6.

2 Eye feature extraction

Active illumination from one or more infrared sources is a common method used in eye-tracking [7] [12] [15] to produce the dark- or bright-pupil effect. The bright-pupil effect is produced when the light source is coaxial to the camera and the dark-pupil effect otherwise [6]. Either effect is particularly useful in locating the pupil as it is highly-contrasted to the iris which has different reflection properties and thus appears as grey, as shown in Figure 1 (in this particular setup, it is the dark-pupil effect).

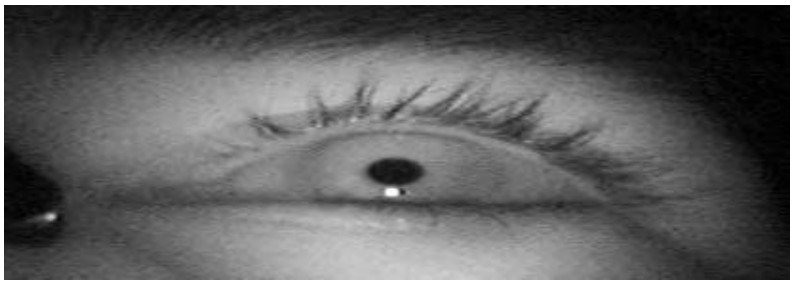


Figure 1: Example input image.

Pupil contour estimation

The first step in detecting the pupil is thresholding the input image I_{input} to a binary image I_t with a threshold value T such that:

$$I_t(x,y) = \begin{cases} 0, when\ I_{input(x,y)} > T \\ 255, otherwise \end{cases}$$

Using connected components labelling, the binary image can be converted to a higher-level description of the image content. Then, a minimum and maximum blob area is defined (A_{min} and A_{max} respectively) and used to filter blobs resulting from random noise (too small) and blobs that are formed in the darker areas of the image (where the infrared illumination fades rapidly).

S_{bf} is now the set of pupil candidates. If the set is empty, the default threshold value T is adjusted to $T-10, T-20, T+10$ and $T+20$ and the thresholding and labelling is applied again. This adjustment is necessary because the threshold is not chosen adaptively and in some cases may include too much or too little of the pupil.

By taking into account the elliptical shape of the pupil, the algorithm to select the pupil blob from the set of candidate blobs $S_{bf} = \{B_0, B_1, \dots, B_n\}$ is as follows. For each blob B_x in S_{bf}, the outmost pixels $outmost(B_x)$ are selected by scanning each line within the blob's bounding rectangle and selecting the first and last pixel that matches the blob's label. The set of pixels $outmost(B_x)_{0..m}$ is then used to fit an ellipse $E(B_x)$ using the algorithm described in [10]. For each ellipse $E(B_x)$, a measure of the fit error $e_{E(B_x)}$ is calculated as the average of the Euclidean distance between the detected points to the ellipse contour (see Figure 2). Finally, the pupil contour estimate blob B_{pupil} is found by selecting the blob in set S_{bf} which has the minimum error.

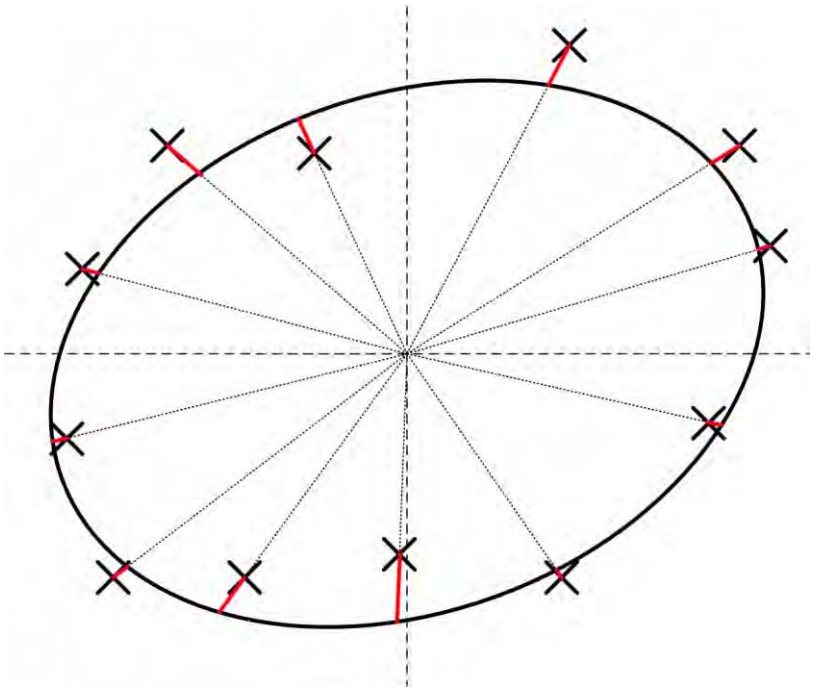


FIGURE 2: ILLUSTRATION OF THE DISTANCES USED TO CALCULATE THE ELLIPSE FIT ERROR ON A HYPOTHETICAL PUPIL ELLIPSE (RED LINES DESIGNATE THE DISTANCES WHOSE AVERAGE IS THE ERROR ESTIMATE).

Pupil contour refinement

After performing an initial estimation of the pupil contour from the thresholded image, it is necessary to further refine it; since global thresholding is used, the results are very much dependent on the threshold value used. Whilst the pupil is *always* dark in the images, just how dark it is will depend on the camera topology (distance from the eyeball and angle of positioning i.e. exactly how the infrared light falls on the pupil), its pupil reflection properties and their variance between subjects. The aforementioned refinement is done in the REACT eye-tracker through the use of active contours [3], also known as snakes. The greedy snake algorithm implemented in the REACT eye-tracker is derived from [19].

The snake is iterated a predefined number of times and the new spline points define the refined pupil contour $C_{pupil} = \{p_1, \dots, p_n\}$. The final output of this component comprises C_{pupil} and the pupil centre

$$\bar{C}_{pupil} = average(C_{pupil})$$

Iris radius calculation

Locating the iris is a problem that has been tackled several times before but usually in remote eye-tracker setups with significantly different image formations (e.g. [14] [17]). Thus a novel approach is proposed here.

Even though it may have been possible to design the eye-tracker without locating the iris boundaries, doing so provides two significant advantages: a) it provides a robust starting point for the challenging task of locating the eye corners which is a very challenging task and b) it allows the eye-tracker to be extended to calculate the 3D gaze provided the camera is fully calibrated; one such approach is found in the remote eye-tracker system developed in [17].

A similar task to the detection of the iris boundaries, is cell segmentation [22]. Cells, much like the iris, are fairly uniform in terms of the pixels' intensity levels on the inside. In a similar vein, cell image background is also uniform, like the sclera. Thus, the edge is not necessarily defined by the change in grey-level intensity, which is the basis of most edge-detectors, but rather by a change in the uniformity over a range of pixels.

For a population of size M, the edge strength at a division m, is defined as [22]:

$$mlr = \frac{\hat{\sigma}_M}{\hat{\sigma}_m \, \hat{\sigma}_{M-m}}$$

Where $\hat{\sigma}$ is the standard deviation of the grey-level pixels.

In the original edge strength calculation equation shown above [22], the whole population is considered; however, it was empirically found that with eye image data, edges attributed to eyelashes and eyelids can severely alter the standard deviation of each population and thus make the algorithm fail or return erroneous (in this context) results. Thus, the original formula was modified to work with a fixed window such that for a line of pixels L, given a constant window size W, the edge strength at a index i is equal to:

$$mlr_i = \frac{\hat{\sigma}_{[i-W,i+W]}}{\hat{\sigma}_{[i-W,i]} \, \hat{\sigma}_{[i,i+W]}}$$

The iris is detected on frames where the person is looking approximately straight ahead and several boundary candidates are generated using the above approach. A filtering algorithm

discards any candidate points whose x-coordinate falls outside the specified confidence interval. If the resulting set is empty after the filtering, the interval is progressively reduced using values $\sigma = 2.576$ (99% interval), $\sigma = 2$ (95.5% interval), $\sigma = 1.645$ (90% interval) and $\sigma = 1$ (68.27% interval).

Eye corner detection

Locating the eye corners is probably the most significant challenge for the set of input images taken with the REACT eye-tracker. The problem of locating the eye-corners has been tackled before [9] [11] [16] [18] [20] [21] but the systems in question operated, without exception, on a full-face, sometimes colour, image.

Detecting the location of the corners is also a significant task as it provides two static points of reference. In the current configuration, these two reference points are useful in two ways:

(c) To calculate the principal axis by which to calculate the 2D gaze angle (next section). The camera may be rotated as a result of misplacement by the experimenter or slippage of the frame.
(d) In long sequences where the absolute position of the eyeball centre is bound to change over time (e.g. frame slippage etc.), the eye corners can be used to detect whether a re-initialization of the eye tracker is required.

Additionally, if the eye-tracker was to be configured to detect the 3D gaze (using a fully calibrated camera), the eye corners are essential to disambiguating the 3D vector solution. For more details, the interested reader is referred to [17].

In order to ease the task of finding the eye corners, some detail as well as noise is removed; the input image is scaled down to ¼ of its original size using Gaussian Pyramid Decomposition. Then, the partial x- and y-derivative of the scaled image f are calculated:

$$dx(x, y) = |f(x, y) - f(x - 1, y)|$$

$$dy(x, y) = |f(x, y) - f(x, y - 1)|$$

For the y-derivative, non-maxima are suppressed locally using a 1x3 window. Whilst this is an irregular window (usually square windows are used for computer vision operations), it has been

empirically found that it preserves the vertical edges better than a 3x3 window. This is most likely because the derivative is a one-column operation too.

As mentioned earlier, slightly different algorithms are used to detect the inner and outer corner due to the different eye morphological structure evident at this image resolution. Both algorithms are however based on the same principle:

5. It is assumed that the edges formed between the eyelids and the sclera are within the top N local maxima for a restricted window ($N = 2$). This assumption was empirically tested.
6. A grouping process begins near the iris boundary previously found and continues outwards, grouping all local maxima that are connected, using an 8-connectivity criterion.
7. The groups are searched for a set of two predefined patterns (shown in Figure 3) and if found, the grouping is terminated at that point. These patterns have been empirically found to occur when the lower eyelid edge is joined with another face line edge and thus the purpose of this step is to separate the two edges.
8. The final corner is selected from the group (outer corner) or pair of groups (inner corner) that demonstrate the maximum derivative energy. The energy of a group of points g is calculated as:

$$e(g) = sum(dy(g))$$

For a pair of groups $g1, g2$:

$$e(g1, g2) = e(g1) * e(g2)$$

The added steps and differences between the two algorithms are briefly summarized here. In Step 1, the search window includes both the upper and lower eyelid edges for the inner corner but only the lower eyelid edges for the outer corner. This is done for several reasons:

c) For the inner corner: typically, the lower eyelid edge does not meet the upper eyelid edge. Thus, a distance criterion between the two edges has to be applied to find the corner. Further, often, the lower eyelid edge will be joined to a face line. The pattern detection offered in the main algorithm will in some cases alleviate this problem but only in combination with the distance constraint is the algorithm robust.

d) For the outer corner: typically, the camera is rotated around the vertical axis towards the outer corner thus making the eyelid-half on the inner corner side appear longer and the eyelid-half on the outer corner side appear shorter. Thus, the upper eyelid on the outer corner side is sloped several degrees more than the inner corner side. For this reason, grouping local maxima points on the top eyelid results in several disjointed groups. In order to robustly find the outer corner, pattern matching is combined with a refinement based on the partial x- derivative of the image. Since the upper eyelid edge and the lower eyelid edge always meet on the outer corner side, a strong maxima is created in the partial x-derivative image. This maxima is used to refine the corner in the last step and is found by searching a 10x5 window.

FIGURE 3: PREDEFINED SEARCH PATTERNS FOR THE OUTER CORNER SEARCH. THE PATTERNS ARE REVERSED FOR THE INNER CORNER.

Finally, to increase the robustness of the corner detection results which can be sensitive to noise, corner detection is applied over a group of *N* consecutive frames centered on the target frame and outliers in the resulting eye corners are calculated by the same statistical process used to filter iris boundary candidates. During development, it was found that this algorithm gave superior performance versus no clustering and a weighted means scheme where candidates are inversely weighted according to their distance to the mean and the weight sum is calculated.

3 Calculation and classification of 2D gaze

Calculating a 3D gaze vector would have required a fully calibrated camera (Wang *et al*., 2000). Even though there are modern means of camera calibration that greatly simplify the process, it is still too involved to be performed by the user of a system (versus the developer) like the REACT

eye-tracker. For this reason and given that when investigating non-visual eye-movements 3D gaze offers little or none additional information over 2D gaze, it was decided that 2D gaze calculation would suffice.

The eye-tracker requires only one calibration point – that of the subject looking approximately straight ahead. This calibration point can be provided on-line (in real-time while recording and tracking at the same time) or off-line (after the data sample has been recorded) and initializes the tracker by calculating the initial pupil position P_0 and contour C, the iris radius R and the eye corners locations P_{Co} and P_{C1}.

At each point in time, the 2D gaze vector is calculated as follows:

- If the current pupil position P_t falls within the initial pupil contour C, it is assumed that the subject is looking straight ahead.
- Otherwise:
    - A reference x-axis and corresponding y-axis are established using the line that connects the two corners P_{Co} and P_{C1}.
    - The centre of the reference axis's is translated to coincide with the initial pupil position P_0.
    - The 2D gaze vector between P_0 and P_t is calculated as well as the gaze angle θ:

$$\theta = \tan^{-1} \frac{y_t - y_0}{x_t - x_0}$$

The above process is visually illustrated in Figure.

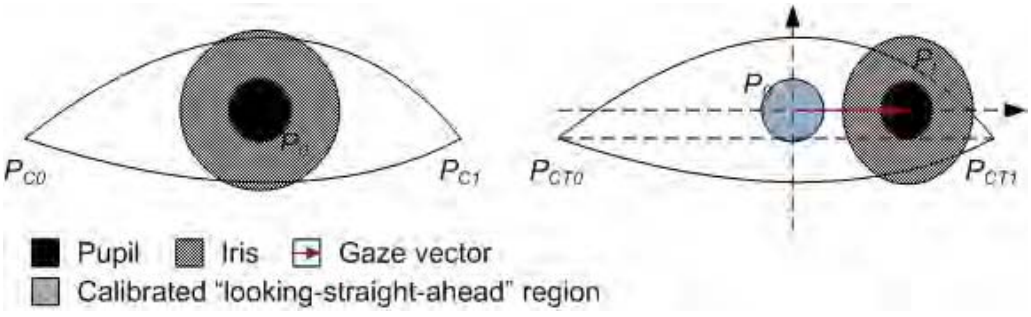


FIGURE 4: CALCULATION OF THE 2D GAZE VECTOR.

4 Case Study

In addition to the evaluation of the feature extraction and 2D gaze calculation algorithms, it was decided to demonstrate the eye-tracker's applicability to a real-world scenario through a case study. Bearing in mind our motivation through the NLP literature, a pilot study was designed where a subject was interviewed by the experimenter on various topics such as education and hobbies, meanwhile recording the subject's eye-movements, for a period of approximately twenty (20) minutes. The conversation was then manually transcribed and the position in time of each word uttered by both the subject and experimenter was also manually recorded. This process resulted in a rich data set that allowed the precise overlay of eye-movements, in time with speech.

Fixations from the data sequence are selected as follows:

6) The current pupil position is detected and the corresponding 2D gaze angle is calculated.
7) If the subject is looking straight ahead, the frame is *skipped*. Otherwise, it is classified into one of the eight categories listed below, the class is recorded and a count is maintained.
8) On the next frame, if the classified position is the same as the one recorded, the count is updated. If the subject is now looking straight ahead, the count is reset.
9) If the count reaches a threshold N_{EM}, the frame is *selected*. This allows fixations to be selected.
10) When a frame is selected, N_{EM} is set to a lower threshold N_{EM_SEARCH} and the algorithm continues onto the next frame. This allows for search paths to be detected and its component frames selected.

Eye-movements are classified into eight (8) classes as per the classes introduced in NLP [2]; specifically up, up and to the left, up and to the right, left, down, down and to the left, down and to the right, right. For a 2D gaze angle θ and $\varphi = 20°$, a class C is assigned as follows:

- $C = \text{R}$ for $-\frac{\varphi}{2} \leq \theta \leq \frac{\varphi}{4}$
- $C = UR$ for $\frac{\varphi}{4} \leq \theta \leq 90 - \varphi$
- $C = U$ for $90 - \varphi \leq \theta \leq 90 + \varphi$
- $C = UL$ for $90 + \varphi \leq \theta \leq 180 - \frac{\varphi}{4}$

- $C = L$ for $180 - \frac{\varphi}{4} \leq \theta \leq 180 + \frac{\varphi}{2}$
- $C = DL$ for $180 + \frac{\varphi}{2} \leq \theta \leq 270 - \varphi$
- $C = D$ for $270 - \varphi \leq \theta \leq 270 + \varphi$
- $C = DR$ for $270 + \varphi \leq \theta \leq 360 - \frac{\varphi}{4}$

A tolerance angle of $\frac{\varphi}{2}$ is used for the L and R classes on the bottom hemisphere (180° to 360°) to accommodate for the reduced vertical resolution that is a direct result of separating each frame into two fields. The tolerance angle is further reduced to $\frac{\varphi}{4}$ for the bottom hemisphere of classes L and R (0° to 180°) to accommodate for the skew that is a direct result of the camera pointing upwards.

A calibration frame is required to initialize the eye-tracker to the subject. This is done by asking the subject to look straight ahead with his or her chin approximately parallel to the floor (posture is adjusted with the guidance of the experimenter). The eye-tracker automatically re-initializes itself to accommodate for changes in the location of the corners on every frame where the pupil position is within a contour twice as large as the initial pupil contour. In most sequences the latter contour is a little smaller than the iris. Re-initialization is not performed if the eye-tracker has re-initialized in the last 60 frames (approximately 10 seconds).

During re-initialization, the "initial pupil position" P used to calculate the 2D gaze angle is updated to P' from the new corner locations C_L' and C_R' such that:

$$\frac{C_L - P}{C_R - P} = \frac{C_L' - P'}{C_R' - P'}$$

An example of a visualised segment from the case study data with the speaker and subject speech, thumbnail of the selected frame and output classification (abbreviated, e.g. up-left is UL) is showing in Figure 5.

FIGURE 5: EXAMPLE OF VISUALISED SEGMENT FROM THE CASE STUDY DATA.

5 Evaluation and results

Given that the REACT eye-tracked is feature-based, it makes sense to evaluate its performance in extracting these features by calculating the Euclidean distance between each feature point as extracted by the eye-tracker and the feature point as manually marked by the author.

Thus, on each intermediate step of the 2D gaze calculation (detecting the pupil, calculating the iris radius and locating the corners), the appropriate set of frames were selected (the selection process will be described in detail) from the test video database and the errors were measured. The set of manually marked frames will also be referred to as the validation data set.

It is desirable to assess the performance of each component separately and thus, for the iris radius and corners extraction algorithms that depend on previous outputs (pupil location and pupil location, iris radius respectively), they were taken from the validation data set such that there is no interference from errors from other components. The 2D gaze angle calculation algorithm was evaluated by comparing the 2D gaze angle calculated using inputs (pupil position, iris radius, corners locations) from the eye-tracker versus using inputs from the validated data set.

The pupil detection algorithm was evaluated over 12,334 frames and showed an average accuracy of 2.04 ± 3.32 pixels. Over 1856 test frames, the iris radius was on average calculated with an accuracy of 2.11 ± 1.42 pixels. Over the same test set as the iris radius, the eye corners were on average calculated to an accuracy of 8.32 ± 5.78 and 8.41 ± 5.40 pixels for the inner and outer corner respectively. The 2D gaze direction angle was on average calculated with an accuracy of 2.78 ± 1.99 degrees, a range considered practical for the target applications.

Finally, the class output by the eye-tracker was compared to a manual classification performed by the experimenter. The manual classification proved to be a much harder task than anticipated

as ambiguities were eminent in some cases when the eye-movement in question was on the borderline between two classes. From the total 150 eye-movements, 7 received an ambiguous classification by the experimenter and 6 were erroneously classified by the eye-tracker.

It is questionable whether ambiguous classifications can be avoided unless the subject's eyes are also captured from another camera placed on the same level and the video may be consulted to resolve ambiguities. Of course, while this would be feasible in an experimental, for the eye-tracker, setup, it would probably prove impractical for eye-tracker users conducting experiments.

All 6 classification errors were caused by the eye-movement being too close on the borderline between two classes. The classification algorithm will determine the class solely on the 2D gaze angle calculated and based on pre-set thresholds. As any other statically set threshold, it is bound to fail some of the time, when the thresholded value is very close to the threshold itself. In other words, when the gaze angle is on or close to the borderline between two classes, a human rater may be able to distinguish between the classes (though not always as proved by the 7 ambiguous ratings) but the algorithm cannot.

Example output images of the extraction of the complete set of features during calibration are shown in Figure 6.

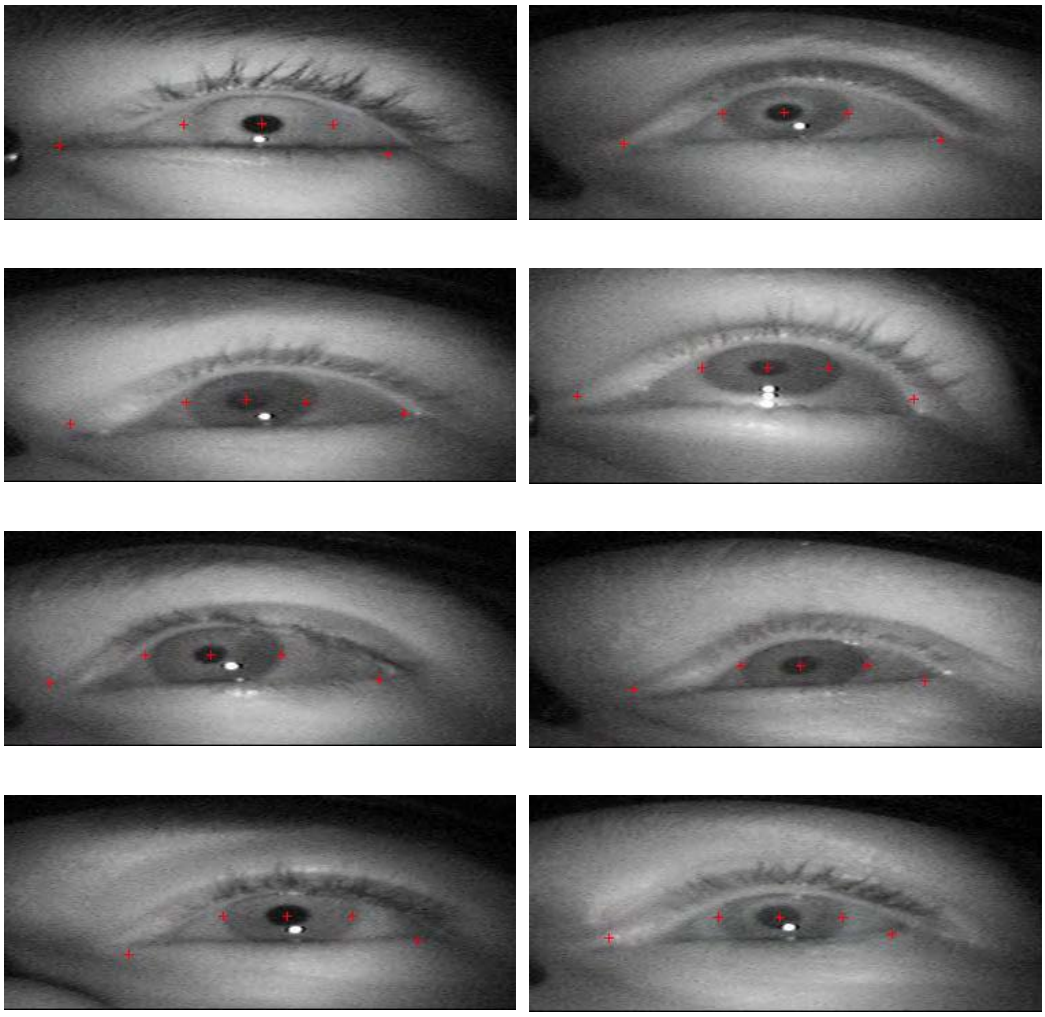


FIGURE 6: EXAMPLE FEATURE EXTRACTION IMAGES.

6 Conclusion

The main objective of the current research work was to develop an eye-tracker that is able to track extreme eye-movements and calculate their gaze direction is 2D. A set of novel feature extraction algorithms were presented for extracting the location of the pupil, the iris radius and location of eye corners and calculating the gaze direction from images taken from an actively-illuminated head-mounted eye-tracker. The accuracy of the feature extraction was assessed both independently and as a whole; the eye-tracker achieved a practical level of performance that renders it acceptable for use in the target research application(s). This was further demonstrated through the pilot study that was designed and served as a case study of a real-world application.

References

[71] Babcock, J. S. and Pelz, J., 2004. Building a lightweight eye-tracking headgear. In: *Eye Tracking Research and Applications (ETRA) Symposium*, 109-113.

[2] Bandler, R. and Grinder, J., 1979. Frogs into Princes: neuro-linguistic programming. Moab, UT: Real People Press.

[3] Blake, A. And Isard, M., 1998. *Active contours*. London, New York: Springer.

[4] Diamantopoulos, G., Woolley, S. I., and Spann, M., 2008. A critical review of past research into the Neuro-Linguistic Programming Eye-Accessing Cues model. In: Proceedings of the First International NLP Research Conference. Surrey, UK.

[5] Ehrlichman, H., Micic, D., Sousa, A., & Zhu, J. (2007). Looking for answers : Eye movements in non-visual cognitive tasks. Brain and cognition , 64 (1), 7-20.

[6] Ebisawa, Y., 1998. Improved video-based eye-gaze detection method. IEEE Transactions on instrumentation and measurement, 47 (4), 948-955.

[7] Ebisawa, Y., Tsukahara, S., and Ishima, D., 2002. Detection of feature points in video-based eye-gaze direction. In: Proceedings of the 24th annual conference and the annual fall meeting of the Biomedical Engineering Society.

[8] Ehrlichman, H., Micic, D., Sousa, A. and Zhu, J., 2007. Looking for answers: eye movements in non-visual cognitive tasks. *Brain and Cognition*, 64 (1), pp. 7–20.

[9] Feng, G. C., and Yuen, P. C., 1998. Variance projection function and its application to eye detection for human face recognition. *International Journal of Computer Vision*, 19, 899-906.

[10] Halir, R. and Flusser, J., 1998. Numerically stable direct least squares fitting of ellipses. In: *Proceedings of the 6th International Conference in Central Europe on Computer Graphics and Visualization*. Plzen, CZ, 123-132.

[11] Lam, K., and Yan, H., 1996. Locating and extracting the eye in human face images. *Pattern recognition*, 29, 771-779.

[12] Li, D., Winfield, D., Parkhurst, D. J. (2005). Starburst: A hybrid algorithm for video-based eye tracking combining feature-based and model-based approaches. In: Proceedings of the IEEE Vision for Human-Computer Interaction Workshop at CVPR, 1-8.

[13] Richardson, D.C. and Spivey, M.J., 2004. Eye-tracking: characteristics and methods. In: Wnek, G. and Bowlin, G. (eds.) Encyclopedia of Biomaterials and Biomedical Engineering. New York: Marcel Dekker.

[14] Sirohey, S., Rosenfeld, A., and Zuric, Z., 2002. A method of detecting and tracking irises and eyelids in video. *Pattern Recognition*, 35, 1389-1401.

[15] Takegami, T., Gotoh, T., and Ohyama, G., 2002. An algorithm for an eye-tracking system with self-calibration. Systems and computers in Japan, 33 (10), 1580-1588.

[16] Tian, Y., Kanade, T., and Cohn, J. F., 2000. Dual-state parametric eye tracking. In: *Proceedings of the 4th IEEE International Conference on Automatic Face and Gesture Recognition*.

[17] Wang J-G., Sung E., and Venkateswarlu, R., 2000. On eye gaze determination via iris contour. *IAPR Workshop on Machine Vision Applications.* Tokyo, Japan.

[18] Wang, J. G., Sung, E., and Venkateswarlu, R., 2005. Estimating the eye gaze from one eye. *Computer Vision and Image Understanding*, 98 (1), 83-103.

[19] Williams, D. J. and Shah, M., 1992. A fast algorithm for active contours and curvature estimation. *Graphical Models and Image Processing: Image Understanding*, 55 (1), 14-26.

[20] Xu, C., Zheng, Y., and Wang, Z., 2008. Semantic feature extraction for accurate eye corner detection. In: *Proceedings of the 19th International Conference on Pattern Recognition.*

[21] Zhang, L., 1996. Estimation of eye and mouth corner point positions in a knowledge-based coding system. In: *Proceedings of the SPIE*, vol. 2952. Berlin, Germany.

[22] Zhou, P. and Pycock, D., 1997. Robust statistical models for cell image interpretation, *Image and Vision Computing*, 15, 307-316.

LIST OF REFERENCES

Atlmann, G. T. M., and Kamide, Y., 2007. The real-time mediation of visual attention by language and world knowledge: Linking anticipatory (and other) eye movements to linguistic processing. *Journal of Memory and Language*, 57, pp. 502-518.

Andiel, M., Hentschke, S., Elle, T., and Fuchs, E., 2002. Eye tracking for autostereoscopic displays using web cams. In: *Procedings of the SPIE, 4660,* pp. 200-206.

Babcock, J.S., Pelz, J.B., and Peak, J.F. (2003). The Wearable Eyetracker: A Tool for the Study of High-level Visual Tasks. In: *Proceedings of the Military Sensing Symposia Specialty Group on Camouflage, Concealment, and Deception*, Tucson, Arizona.

Babcock, J. S. and Pelz, J., 2004. Building a lightweight eye-tracking headgear. In: *Eye Tracking Research and Applications (ETRA) Symposium*, 109-113.

Baddeley, M. and Predebon, J., 1991. Do the eyes have it? A test of neurolinguistic programming's eye movement hypothesis. *Australian Journal of Clinical Hypnotherapy and Hypnosis*, 12 (1), pp. 1–23.

Bakan, P. and Strayer, F. F., 1973. On reliability of conjugate lateral eye movements. *Perceptual Motor Skills*, 36 (2), pp. 429–30.

Bandler, R. and Grinder, J., 1975. *The Structure of Magic: a book about language and therapy*. Palo Alto, CA: Science and Behavior Books.

Bandler, R. and Grinder, J., 1979. *Frogs into Princes: neuro-linguistic programming*. Moab, UT: Real People Press.

Baron-Cohen, S. and Cross, P., 1992. Reading the eyes: evidence for the role of perception in the development of a theory of mind. *Mind and Language*, 2, pp. 173–86.

Beck, C.E. and Beck, E. A., 1984. Test of the eye-movement hypothesis of neurolinguistic programming: a rebuttal of conclusions. *Perceptual and Motor Skills*, 58 (1), pp. 175–76.

Benoit, A., Caplier, A., and Bonnaud, L., 2005. Gaze direction estimation tool based on head motion analysis or iris position estimation. In: *Proceedings of the 13th European Signal Processing Conference*.

Blake, A. And Isard, M., 1998. *Active contours*. London, New York: Springer.

Brandt, S.A., and Stark, L.W., 1997. Spontaneous eye movements during visual imagery reflect the content of the visual scene. *Journal of Cognitive Neuroscience*, 9 (1), pp. 27–38.

Baymer, D., and Flickner, M., 2003. Eye gaze tracking using an active stereo head. *Proceedings of the 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition.*

Buckner, M. and Reese, M., 1987. Eye movement as an indicator of sensory components in thought. *Journal of Counseling Psychology*, 34 (3), pp. 283–87.

Bouguet, J-Y., 2008. *Camera calibration toolkit for Matlab.* Available from http://www.vision.caltech.edu/bouguetj/calib_doc/ [Accessed 28/06/2008].

Buckner, R.L. and Wheeler, M.E., 2001. The cognitive neuroscience of remembering. *Nature Reviews Neuroscience*, 2 (9), pp. 624–34.

Burke, D., Meleger, A., Schneider, J., Snyder, J., Dorvlo, A. and Al-Adawi, S., 2003. Eye movements and ongoing task processing. *Perceptual Motor Skills*, 96 (3), pp. 1330–38.

Canny, J., 1986. A computational approach for edge detection, *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 8 (6), 679-698.

Carpenter, R. H. S., 1988. *Movements of the Eyes,* 2nd ed. London: Pion.

Chen, J., and Ji, Q., 2008. 3D gaze estimation with a single camera without IR illumination. In: *Proceedings of the 19th International Conference on Patter Recognition*, pp. 1-4.

Chen, J., Tong, Y., Gray, W., and Ji, Q., 2008. A robust 3D eye gaze tracking system using noise reduction. In: *Proceedings of the 2008 Conference on Eye Tracking Research and Applications*.

Cheney, S., Miller, L. and Rees, R., 1982. Imagery and eye movements. *Journal of Mental Imagery*, 6, pp. 113–24.

Christman, S.D., Garvey, K.J., Propper, R.E. and Phaneuf, K.A., 2003. Bilateral eye movements enhance the retrieval of episodic memories. *Neuropsychology*, 17 (2), pp. 221–29.

Clarke, A. H., Ditterich, J., Druen, K., Schonfeld, U., Steineke, C., 2002. Using high frame rate CMOS sensors for three-dimensional eye-tracking. *Behaviour Research Methods, Instruments and Computers,* 34 (4), 549-560.

Chomsky, N., 1965. *Aspects of the Theory of Syntax*. Cambridge, MA: MIT Press.

Collet, C., Finkel, A., and Gherbir, R., 1998. CapRe: A gaze tracking system in man-machine interaction. *Journal of Advanced Computational Intelligence and Intelligence Informatics,* 2 (3), 77-81.

Collewijn, H., Erkelens, C. J., and Steinman, R. M., 1988. Binocular co-ordination of human vertical saccadic eye movements. *Journal of Physiology,* 404, pp. 183-197.

Colombo, C., Comanducci, D., Del Bimbo, A., 2007. Robust iris localization and tracking based on constrained visual fitting. In: *Proceedings of the 14th International Conference on Image Analysis and Processing*.

Coutinho, F. L. Z., and Morimoto, C. H., 2006. Free head motion eye gaze tracking using a single camera and multiple light sources. In: *IEEE Brazilian Symposium on Computer Graphics and Image Processing.*

Day, M., 1964. An eye movement phenomenon relating to attention, thought and anxiety. *Perceptual Motor Skills*, 19, pp. 443–46.

Demarais, A.M. and Cohen, B.H., 1998. Evidence for image scanning eye movements during transitive inference. *Biological Psychology*, 49, pp. 229–47.

Diamantopoulos, G., Woolley, S. I., and Spann, M., 2008. A critical review of past research into the Neuro-Linguistic Programming Eye-Accessing Cues model. In: *Proceedings of the First International NLP Research Conference.* Surrey, UK.

Dilts, R. and DeLozier, J., 2000. *Encyclopedia of Systemic NLP and NLP New Coding*. Capitola, CA: Meta Publications.

Dooley, K.O. and Farmer, A., 1988. Comparison for aphasic and control subjects of eye movements hypothesized in neurolinguistic programming. *Perceptual and Motor Skills*, 67 (1), pp. 233–34.

Dorn, F.J., Atwater, M., Jereb, R. and Russell, R., 1983. Determining the reliability of the NLP eye-movement procedure. *American Mental Health Counselors Association Journal*, 5 (3), pp. 105–10.

Duke, J.D., 1968. Lateral eye movement behavior. *Journal of General Psychology*, 78, pp. 189–95.

Ebisawa, Y., 1998. Improved video-based eye-gaze detection method. *IEEE Transactions on instrumentation and measurement*, 47 (4), 948-955.

Ebisawa, Y., Tsukahara, S., and Ishima, D., 2002. Detection of feature points in video-based eye-gaze direction. In: *Proceedings of the 24th annual conference and the annual fall meeting of the Biomedical Engineering Society*.

Ehrlichman, H., Weiner, S. and Baker, H., 1974. Effects of verbal and spatial questions on initial gaze shifts. *Neuropsychologia*, 12, pp. 265–77.

Ehrlichman, H. and Weinberger, A., 1978. Lateral eye movements and hemispheric asymmetry: a critical review. *Psychological Bulletin*, 85, pp. 1080–101.

Ehrlichman, H., 1981. From gaze aversion to eye-movement suppression: An investigation of the cognitive interference explanation of gaze patterns during conversation. *British Journal of Social Psychology*, 20 (4), pp. 233–41.

Ehrlichman, H. and Barrett, J., 1983. Right hemispheric specialization for mental imagery: a review of the evidence. *Brain and Cognition*, 2 (1), pp. 55–76.

Ehrlichman, H., Micic, D., Sousa, A. and Zhu, J., 2007. Looking for answers: eye movements in non-visual cognitive tasks. *Brain and Cognition*, 64 (1), pp. 7–20.

Einspruch, E. and Forman, D., 1985. Observations concerning research literature on neuro-linguistic programming, *Journal of Counseling Psychology*, 32 (4), pp. 589–96.

Elich, M., Thompson, R.W. and Miller, L., 1985. Mental Imagery as revealed by eye movements and spoken predicates: a test of neurolinguistic programming. *Journal of Counseling Psychology*, 32 (4), pp. 622–25.

Ellickson, J.L., 1983. Representational systems and eye movements in an interview. *Journal of Counseling Psychology*, 30 (3), pp. 339–45.

Falzett, W., 1981. Matched versus unmatched primary representational systems and their relationship to perceived trustworthiness in a counseling analogue. *Journal of Counseling Psychology*, 28 (4), pp. 305–8.

Farmer, A., Rooney, R. and Cunningham, J.R., 1985. Hypothesized eye movements of neurolinguistic programming: a statistical artifact. *Perceptual Motor Skills*, 61 (3), pp. 717–18.

Feng, G. C., and Yuen, P. C., 1998. Variance projection function and its application to eye detection for human face recognition. *International Journal of Computer Vision*, 19, 899-906.

Gonzalez, R. C. and Woods, R. E, 2002. *Digital Image Processing, 2nd Edition*. Upper Saddle River, N. J.: Prentice Hall.

Griffin, Z. M., and Bock, K., 2000. What the eyes say about speaking. *Psychological science,* 11 (4), pp. 274-279.

Grinder, J. and Bandler, R., 1976. *The Structure of Magic II: a book about communication and change*. Palo Alto, CA: Science and Behavior Books.

Grinder, J., Delozier, J., and Bandler, R., 1977. *Patterns of the Hypnotic Techniques of Milton H. Erickson, M.D. Volume II*. Cupertino, CA: Meta Publications.

Guestrin, E. D., and Eizenman, M., 2006. General theory of remote gaze estimation using the pupil center and corneal reflections. *IEEE Transactions on Biomedical Engineering*, 53 (6), pp. 1124.

Guestrin, E. D., and Eizenman, M., 2008. Remote point-of-gaze estimation requiring a single-point calibration for application with infants. In: *Proceedings of the 2008 Conference on Eye Tracking Research and Applications*, pp. 267.

Guitton, D., and Volle, M., 1987. Gaze control in humans: eye-head coordination during orienting movements to targets within and beyond the oculomotor range. *Journal of neurophysiology*, 58 (3), pp. 427-459.

Gumm, W.B., Walker, M.K. and Day, H.D., 1982. Neurolinguistic programming: method or myth? *Journal of Counseling Psychology*, 29 (3), pp. 327–30.

Halir, R. and Flusser, J., 1998. Numerically stable direct least squares fitting of ellipses. In: *Proceedings of the 6th International Conference in Central Europe on Computer Graphics and Visualization*. Plzen, CZ, 123-132.

Handy, T.C., Miller, M.B., Schott, B., Shroff, N.M., Janata, P., Van Horn, J.D., Inati, S., Grafton, S.T. and Gazzaniga, M.S., 2004. Visual imagery and memory: do retrieval strategies affect what the mind's eye sees? *European Journal of Cognitive Psychology*, 16 (5), pp. 631–52.

Hansen, D. W., and Pece, A. E. C., 2005. Eye tracking in the wild. *Computer Vision and Image Understanding, Special Issue on Eye Detection and Tracking*, 98 (1), 155-181.

Hansen, D. W., and Ji., Q, 2010. In the eye of the beholder: a survey of models for eyes and gaze. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 32 (3), 478-500.*

Harris C. and Stephens M., 1988. A combined corner and edge detector. In: *Proceedings of the 4th Alvey Vision Conference*, pp. 147-151.

Heap, M., 1988. Neurolinguistic programming: an interim verdict. In: Heap, M. (ed.) *Hypnosis: current clinical, experimental and forensic practices*. London: Croom Helm.

Heinzmann, J., and Zelinsky, A., 1998. 3-D facial pose and gaze point estimation using a robust real-time tracking paradigm. In: *Proceedings of the IEEE International Conference on Automatic Face and Gesture Recognition*, pp. 142-147.

Hennessey, C., Noureddin, B., and Lawrence, P., 2006. A single camera eye-gaze tracking system with free heead motion. In: *Eye-Tracking Research and Applications 2006, San Diego.*

Hennessey, C., and Lawrence, P., 2009. Noncontact binocular eye-gaze tracking for point-of-gaze estimation in three dimensions. *IEEE Transactions on Biomedical Engineering,* 56 (3), pp. 790.

Horowitz, M.J., 1983. *Image Formation and Psychotherapy*. Northvale, NJ: Aronson.

Ishima, D., and Ebisawa, Y., 2003. Eye tracking based on ultrasonic position measurement in head free video-based eye-gaze detection. In: *Proceedings of the IEEE EMBS Asian-Pacific Conference on Biomedical Engieering*, pp. 258-259.

Ji, Q., and Yang, X., 2002. Real-tie eye, gaze, and face pose tracking for monitoring driver vigilance. *Real-Time Imaging,* 8, pp. 357-377.

Kass, M., Witkin, A., and Terzopoulos, D., 1988. Snakes: Active contour models. *International Journal of Computer Vision*, 1(4), pp. 321-331.

Kim, K-N., and Ramakrishna, R.S., 1999. Vision-Based Eye-Gaze Tracking for Human Computer Interface. In: *IEEE International Conference on Systems, Man, and Cybernetics, Tokyo, Japan*.

Kinsbourne, M., 1972. Eye and head turning indicates cerebral lateralization. *Science*, 176 (4034), pp. 539–41.

Kocel, K., Galin, D., Ornstein, R.E. and Merrin, E.L., 1972. Lateral eye movement and cognitive mode. *Psychonomic Science*, 27, pp. 223–24.

Kohlbecher, S., and Poitschke, T., 2008. Calibration-free eye tracking by reconstruction of the pupil ellipse in 3D space. In: *Proceedings of the 2008 Conference on Eye Tracking Research and Applications*, pp. 135.

Kosslyn, S., 2005. Mental images and the brain. *Cognitive Neuropsychology*, 22 (3), pp. 333–47.

Kwok, C., Fox, D., and Melia, M., 2004. Real-time particle filters. In: *Proceedings of the IEEE*, 92, pp. 469–484.

Laeng, B. and Teodorescu, D.S., 2002. Eye scanpaths during visual imagery reenact those of perception of the same visual scene. *Cognitive Science: a multidisciplinary journal*, 26 (2), pp. 207–31.

Lam, K., and Yan, H., 1996. Locating and extracting the eye in human face images. *Pattern recognition*, 29, 771-779.

Li, D., Winfield, D., Parkhurst, D. J., 2005. Starburst: A hybrid algorithm for video-based eye tracking combining feature-based and model-based approaches. In: *Proceedings of the IEEE Vision for Human-Computer Interaction Workshop at CVPR*, 1-8.

Li, F., Kolakowski, S., and Pelz, J., 2007. Using structured Illumination to enhance video-based eye tracking. In: *International Conference on Image Processing*,

Liang, D. B. B., and Houi, L. K., 2007. Non-intrusive eye gaze direction tracking using color segmentation and Hough transform. In: *International Symposium in Communications and Information Technologies*, pp. 602-607.

Luther, A. C. and Inglis, A. F., 1999. *Video engineering, 3rd edition*. USA: McGraw-Hill Professional.

MacDonald, B.H. and Hiscock, M., 1985. Effects of induced anxiety and question content on the direction and frequency of lateral eye movements. *Neuropsychologia*, 23 (6), pp. 757–63.

Mathison, J., 2006. *Phenomenology*. Available from: http://www.som.surrey.ac.uk/NLP/Resources/Phenomenology.pdf [Accessed 08/10/2007].

Mathison, J. and Tosey, P. , 2008a. Exploring Inner Landscapes: NLP and Psycho-phenomenology as innovations in researching first-person experience. *Qualitative Research in Management and Organization Conference*, New Mexico, March 11-13th.

Mathison, J. and Tosey, P. , 2008b. The Potential of NLP as a Methodology for Investigating First Person Experience. *The First International NLP Research Conference*, University of Surrey, 5th July 2008

Matsumoto, Y., and Zelinsky, A., 2000. An algorithm for real-time stereo vision implementation of head pose and gaze direction measurement. In: *Proceedings of the 4th IEEE International Conference on Automatic Face and Gesture Recognition*, pp. 499.

McCarthy, A., Lee, K., Itakura, S. and Muir, D., 2006. Cultural display rules drive eye gaze during thinking. *Journal of Cross-cultural Psychology*, 37 (6), pp. 717–22.

Merad, D., Metz, S., and Miguet, S., 2006. Eye gaze estimation from a video. In: *Proceedings of the Spring conference on Computer Graphics*.

Meyer, A., Bohme, M., Martinetz, T., and Barth, E., 2006. A single-camera remote eye tracker. In: *Perception and Interactive Technologies, volume 4021 of Lecture Notes in Artificial Intelligence,* pp. 208-211.

Morimoto, C.H., Koons, D., Amir. A, and Flickner, M., 2000. Pupil detection and tracking using multiple light sources. *Image and Vision Understanding*, 18, pp. 331-335.

Morimoto, C.H., Amir, A., and Flicker, M., 2002. Detecting eye position and gaze from a single camera and 2 light sources. In: *Proceedings of the 16th International Conference on Pattern Recognition*, pp. 314-317.

Nagamatsu, T., Kamahara, J., Tanaka, N., 2009. Calibration-free gaze tracking using a bionocular 3D eye model. In: *Proceedings of the 27th International Conference on Human Factors in Computing Systems.*

Newman, R., Matsumoto, Y., Rougeaux, S., and Zelinsky, A., 2000. Real-time stereo tracking for head pose and gaze estimation. In: *IEEE International Conference on Automatic Face Recognition,* pp. 122-128.

Noureddin, B., Lawrence, P.D., and Man, C.F., 2005. A non-contact device for tracking gaze in a human computer interface. *Computer Vision and Image Understanding*, 98, pp. 52-82.

Ohno, T., Mukawa, N., and Yoshikawa, A., 2002. FreeGaze: a gaze tracking system for everyday gaze interaction. In: *Proceedings of the 2002 symposium on Eye Tracking Research and Applications (ETRA),* 125-132.

Ohno, T., and Mukawa, N., 2004. A free-head, simple calibration, gaze tracking system that enables gaze-based interaction. In: *Proceedings of the Eye Tracking Research and Applications Symposium, 2004*, pp. 115-122.

Park, K.R., and Kim, J., 2005. Real-time facial and eye gaze tracking system. *IEICE Transactions on Information and Systems,* 88 (6), pp. 1231.

Park, K. R., 2007. A real-time gaze position estimation method based on a 3-D eye model. *IEEE Transactions on Systems, Man and Cybernetics,* 37 (1), pp. 199.

Poffel, S.A. and Cross, H.J., 1985. Neurolinguistic programming: a test of the eye-movement hypothesis. *Perceptual Motor Skills*, 61 (3), p. 1262.

Ramdane-Cherif, Z., and Nait-Ali, A., 2008. An adaptive algorithm for eye-gaze-tracking-device calibration. *IEEE Transactions on Instrumentation and Measurement*, 57 (4), pp. 716.

Richardson, D.C. and Spivey, M.J., 2000. Representation, space and Hollywood Squares: looking at things that aren't there anymore. *Cognition*, 76 (3), pp. 269–95.

Richardson, D.C. and Spivey, M.J., 2004. Eye-tracking: characteristics and methods. In: Wnek, G. and Bowlin, G. (eds.) *Encyclopedia of Biomaterials and Biomedical Engineering*. New York: Marcel Dekker.

Richardson, D.C. and Dale, R., 2005. Looking to understand: the coupling between speakers' and listeners' eye movements and its relationship to discourse comprehension. *Cognitive Science*, 29, pp. 1045–60.

Sandhu, D.S., 1991. Validation of eye movements model of NLP through stressed recalls. *Annual meeting of the American association for counseling and development*. Reno, NV.

Sharpley, C., 1987. Research findings on neurolinguistic programming: non-supportive data or an untestable theory? *Journal of Counseling Psychology*, 34 (1), pp. 103–7.

Sheehan, P.W., 1967. A shortened form of Betts' questionnaire upon mental imagery. *Journal of Clinical Psychology*, 23 (3), pp. 386–89.

Shi J. and Tomasi C., 1994. Good features to track. In: *IEEE Conference on Computer Vision and Pattern Recognition,* pp. 593-600*.*

Shih, S-W., Wu, Y-T., and Liu, J., 2000. A calibration-free gaze tracking technique. In: *Proceedings of the International Conference for Pattern Recognition*, pp. 201-204.

Shih, S-W., and Liu, J., 2004. A novel approach to 3D gaze tracking using stereo cameras. *IEEE Transactions on Systems, Man, and Cybernetics,* 34 (1), pp. 234.

Sirohey, S., Rosenfeld, A., and Zuric, Z., 2002. A method of detecting and tracking irises and eyelids in video. *Pattern Recognition*, 35, 1389-1401.

Spivey, M.J., and Geng, J.J., 2001. Oculomotor mechanisms activated by imagery and memory: eye movements to absent objects. *Psychological Research*, 65 (4), pp. 235–41.

SR Research, 2009. *EyeLink II head mounted user manual*. Ontario, Canada: SR Research, ver. 2.14.

Starr C., 2005. *Biology: Concepts and Applications*. Thomson Brooks/Cole.

Stiefelhagen, R., Yang, J., and Waibel, A., 1997. Tracking eyes and monitoring eye gaze. In: *Proceedings of the 1997 Workshop on Perceptual User Interfaces*, pp. 98-100.

Sun, X., Chen, G., Zhao, C., and Yang, J., 2006. Gaze estimation of human eye. In: *Proceedings of 6th International Conference on ITS Telecommunications,* pp. 310.

Takegami, T., Gotoh, T., and Ohyama, G., 2002. An algorithm for an eye-tracking system with self-calibration. *Systems and computers in Japan,* 33 (10), 1580-1588.

Tan, K-H., Kriegman, D. J., and Ahuja N., 2002. Apperance-based eye gaze estimation. In: *Proceedings of the 6th IEEE Workshop on Applications of Computer Vision,* pp. 191.

Templer, D.I., Goldstein, R. and Pennick, S.B., 1972. Stability and interrating reliability of lateral eye movement. *Perceptual and Motor Skills*, 34, pp. 469–70.

Thomason, T.C., Arbuckle, T. and Cady, D., 1980. Test of the eye-movement hypothesis of neurolinguistic programming. *Perceptual and Motor Skills*, 51, p. 230.

Tian, Y., Kanade, T., and Cohn, J. F., 2000. Dual-state parametric eye tracking. In: *Proceedings of the 4th IEEE International Conference on Automatic Face and Gesture Recognition*.

Tognoli, E., Lagarde, J., DeGuzman, G. C., and Kelso, J.A., 2007. The phi complex as a neuromarker of human social coordination. In: *Proceedgins of the National Academy of Sciences of the United States of America,* 104 (19), pp. 8190-5.

Tosey, P., 2006. *Annotated index of NLP terms in Bandler and Grinder's publications*. Available from: http://www.som.surrey.ac.uk/NLP/Resources/NLPannotatedindex2006.pdf [Accessed 08/10/2007].

Tsuji, K., and Ayoagi, M., 2006. Eye direction by stereo image processing using reflection on an iris. In: *Proceedings of the IASTED International Conference on Computational Intelligence,* pp. 355.

Tweed, D., and Vilis, T., 1990. Geometric relations of eye position and velocity vectors during saccades. *Vision Research*, 30 (1), pp. 111-127.

Valenti, R., Sebe, N., and Gevers, T., 2008. Simple and efficient visual gaze estimation. In: *Workshop on Multimodal Interactions Analysis of Users in a Controllect Environment*.

Varela, F.J. and Shear, J., 1999. *The View From Within: first-person approaches to the study of consciousness*. Exeter: Imprint Academic.

Villanueva, A., and Cabeza, R., 2007. Models for gaze tracking systems. *EURASIP Journal on Image and Video Processing*, 2007 (23570).

Villanueava, A., Daunys, G., Hansen, D. W., Bohme, M., Cabeza, R., Meyer, A., and Barth, E., 2007. A geometric approach to remot eye tracking. *Universal Acces in the Information Society*, 8 (4), pp. 241-257.

Villanueva, A., and Cabeza, R., 2008. A novel gaze estimation system with one calibration point. *IEEE Transactions on Systems, Man and Cybernetics*, 38 (4), pp. 1123.

Wade, N. J., and Tatler, B. W., 2005. *The moving tablet of the eye: The origins of modern eye movement research*. New York: Oxford University Press.

Wallhoff, F., Ablabmeier, M., and Rigoll, G., 2006. Multimodal face detection, head orientation and eye gaze tracking. In: *Proceedings of the IEEE Internantional conference on Multisensor Fusion and Integration for Intelligent Systems*, pp. 13.

Wang J. G., Sung E., and Venkateswarlu, R., 2000. On eye gaze determination via iris contour. *IAPR Workshop on Machine Vision Applications.* Tokyo, Japan.

Wang, J. G., Sung, E., and Venkateswarlu, R., 2005. Estimating the eye gaze from one eye. *Computer Vision and Image Understanding*, 98 (1), 83-103.

Wertheim, E.H., Habib, C. and Cumming, G., 1986. Test of the neurolinguistic programming hypothesis that eye-movements relate to processing imagery. *Perceptual and Motor Skills*, 62 (2), pp. 523–29.

White, K. P., Hutchinson, T. E., and Carley, J. M., 1993. Spatially dynamic calibration of an eye-tracking system. *IEEE Transaction on Systems, Man, and Cybernetics,* 23 (4).

Wikipedia, 2009. *De-interlacing.* Available from: http://en.wikipedia.org/wiki/Deinterlacing [Accessed 12/09/2008].Williams, M., and Hoekstra, E., 1994. Comparison of Five On-Head, Eye-Movement Recording Systems, *Technical Report UTMTRI-94-11*. Ann Arbor, Michigan: The University of Michigan Transportation Research Institute.

Williams, D. J. and Shah, M., 1992. A fast algorithm for active contours and curvature estimation. *Graphical Models and Image Processing: Image Understanding*, 55 (1), 14-26.

Xu, L-Q., Machin, D., and Sheppard, P., 1998. A novel approach to real-time non-intrusive gaze finding. In: *Proceedings of the 1998 British Machine Vision Conference*, pp. 428.

Xu, C., Zheng, Y., and Wang, Z., 2008. Semantic feature extraction for accurate eye corner detection. In: *Proceedings of the 19th International Conference on Pattern Recognition.*

Yamazoe, H., Utsumi, A., Yonezawa, T., and Abe, S., 2008. Remote and head-motion-free gaze tracking for real environments with automated head-eye model calibrations. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*.

Yarbus, A., 1967. *Eye movements and vision*. New York: Plenum Press.

Yoo, D. H., and Chung, M. J., 2005. A novel non-intrusive eye gaze estimation using cross-ratio under large head motion. *Computer Vision and Image Understanding*, 98, pp. 25-51.

Yu, L. H., and Eizenmann, M., 2004. A new methodology for determining point-of-gaze in head-mouted eye tracking systems. *IEEE Transactions on Biomedical Engineering*, 10 (51), pp. 1765.

Zhang, L., 1996. Estimation of eye and mouth corner point positions in a knowledge-based coding system. In: *Proceedings of the SPIE*, vol. 2952. Berlin, Germany.

Zhou, P. and Pycock, D., 1997. Robust statistical models for cell image interpretation, *Image and Vision Computing*, 15, 307-316.

Zhu, J., and Yang, J., 2002. Subpixel eye gaze tracking. In: *Proceedings of the 5th IEEE International Conference on Automatic Face and Gesture Recognition*.