Automated Usability Analysis and Visualisation of Eye Tracking Data

by

Jhani André de Bruin

Submitted in partial fulfilment of the requirements for the degree
Master of Science (Computer Science)
in the Faculty of Engineering, Built Environment and Information Technology
University of Pretoria, Pretoria

April 2014

Automated Usability Analysis and Visualisation of Eye Tracking Data

by

Jhani Adré de Bruin
E–mail: jhani.de.bruin@sap.com

Abstract

Usability is a critical aspect of the success of any application. It can be the deciding factor for which an application is chosen and can have a dramatic effect on the productivity of users. Eye tracking has been successfully utilised as a usability evaluation tool, because of the strong link between where a person is looking and their cognitive activity. Currently, eye tracking usability evaluation is a time–intensive process, requiring extensive human expert analysis. It is therefore only feasible for small–scale usability testing.

This study developed a method to reduce the time expert analysts spend interpreting eye tracking results, by automating part of the analysis process. This was accomplished by comparing the visual strategy of a benchmark user against the visual strategies of the remaining participants. A comparative study demonstrates how the resulting metrics highlight the same tasks with usability issues, as identified by an expert analyst. The method also produces visualisations to assist the expert in identifying problem areas on the user interface.

Eye trackers are now available for various mobile devices, providing the opportunity to perform large–scale, remote eye tracking usability studies. The proposed approach makes it feasible to analyse these extensive eye tracking datasets and improve the usability of an application.

Keywords: Eye tracking, Automated usability testing, Benchmark user.
Supervisors : Dr. KM Malan,
Prof. JHP Eloff
Department : Department of Computer Science
Degree : Master of Science

Acknowledgements

I would first like to thank my Heavenly Father for giving me the potential and means to reach my goals. I would also like to express my gratitude towards the following people, without whom this work would not have been possible:

- My supervisor Dr. Katherine Malan, for the continuous guidance, innovative ideas, patience, support and insightful feedback;
- My co–supervisor Prof. Jan Eloff, for the support and perceptive comments;
- My mother Estelle Snyman, for the editing, understanding, always believing in me and assisting me in every way during the completion of this dissertation (and every other day);
- My husband Chris Coetzee, for your love, encouragement, brainstorming and all the coffee;
- The rest of my family, friends and colleagues, for all your interest and advice;
- Prof. Helene Gelderblom, for providing the eye tracking equipment, knowledge and assistance;
- Dr. Marek Zielinski, for all the editing assistance and suggestions.

The financial assistance of the National Research Foundation (NRF) towards this research is hereby acknowledged. This study also benefited from the support of SAP P&I BIT Mobile Empowerment. Opinions expressed and conclusions arrived at, are those of the author and are not necessarily to be attributed to the companies mentioned in this acknowledgement.

Contents

List of Figures

List of Algorithms

List of Tables

Chapter 1

Introduction

Usability in software is no longer a luxury – it is a requirement. The software development world is extremely competitive and usability plays a major role in the success of software [7, 22]. Most often, for any given problem there are a number of software solutions available, providing the same functionality (especially in the mobile application space). User friendly software improves the chances that people will accept and use a specific solution over another, that has the same features. It should be easy to complete a given task (effectively) in a reasonable time frame (efficiently) and without frustrating the user (satisfactorily) [211].

In order to identify possible issues with the design of an application, usability evaluations should be performed. The selected usability evaluation methods, phases of usability testing and evaluation tools are highly dependent on the available time, money and expertise. Other factors that could also influence this decision is the product to be evaluated, which data needs to be captured, whether users should be involved and many more. Heuristic evaluation methods involve an expert who scrutinizes the usability of an application, whereas simulation methods mimic user behaviour to identify problems in the user interface. A widely used method is user testing, when a group of users are asked to complete a number of tasks using a given application. From this method, data metrics, observations and/or interviews can capture usability issues [99, 102]. Input from the effected user group, as captured by usability testing, provides important usability information and knowing what the user is focusing on provides even greater insight into

possible issues with the application design [148]. The interesting field of eye tracking supplies this enlightening, rich information because of the eye–mind hypothesis, which suggests a link between what a person is looking at and their cognitive activities [110].

Eye tracking is a technique of capturing and recording where a person is looking; this has been used for more than 100 years and applied to numerous fields. Recording eye movements has improved significantly over the years, up to a point where the technology is advanced enough to allow eye tracking by means of a mobile phone [60, 129, 138]. The advancement in this field allows for large scale, remote eye tracking usability studies to be performed, capturing large quantities of eye movement data, while people are using the application. This provides great insight into the usability of an application as the eye movements of the target user group are captured in the usual environment. An expert analyst will typically interpret the eye tracking data captured during a usability test to determine if there are any usability issues and where the issues lie. Experts make use of eye tracking visualisations, mapping out where the users fixated to export statistics, and the complete replay of the usability tests to analyse the eye tracking data.

The basic data captured by eye tracking are fixations, the points where the user focussed, and saccades, the movement from one fixation to another [108, 134]. A sequence of fixations and saccades are called a scan path. Analysts usually make use of eye tracking metrics, heatmaps, scan paths and areas of interest, to analyse data [152, 162].

1.1 Problem statement

Analysis of eye tracking data is a time consuming activity [73, 183], especially when done on a large scale, due to the high dependence on human expertise. The need for analysing large eye tracking datasets is growing as the availability of inexpensive, accurate eye trackers in various mobile devices increases [15]. Even with sophisticated tools, it still takes a lot of time to generate information from the data, such as heat maps, scan paths, fixation clusters and even basic statistics for a specific task [101].

For a typical expert analysis, the expert analyst maps out areas of interest on the user interface. This requires additional knowledge about the user interface, the available functionality and process flow. From the area of interests, the statistical data can

be exported about the usability, such as the time to the first fixation on the relevant component or the percentage of fixation on a specific component [55, 73]. There are also a number of visualisations that can be generated to provide an overview of the eye tracking data. Visualisation of spatial data allows analysts to put the complex eye movement data into context, but visualisations can be cluttered and information can easily be overlooked [17, 162, 163]. To overcome this, the expert will have to investigate each of the replays of the eye tracking data individually, which results in even more time spent on the analysis.

The problem addressed by this research is the immense amount of time it takes to analyse eye tracking data as well as the extensive knowledge required about the relevant application. The proposed solution should provide insight into the usability of an application from the eye tracking data, for the expert to interpret. This approach can be used, in an automated manner, on large–scale eye tracking studies and can be applied without prior knowledge of the user interface, such as which components are areas of interest.

1.2 Research objectives

In this study, a method is proposed to assist expert analysts to filter through eye tracking data and spend less time scrutinising all the eye tracking information, by only focusing on the data indicating a high probability of usability issue occurrences. The method also includes visualisations that simplify the eye tracking data by highlighting areas where possible usability issues lie.

To accomplish this, the following objectives were set:

- To develop a method for deriving a number of comparative metrics from the eye tracking data of a pilot usability study. The metrics suggest potential usability issues, by highlighting high deviation from a benchmark user.

- To establish a method for selecting and using a benchmark user as input to the proposed approach to automatically extract comparative eye tracking usability metrics.

- To map the relevant data back onto a user interface, in order to visualise the location of possible usability issues.
- To validate the proposed method by applying the techniques to a larger usability study and comparing the results to an independent expert–based usability study, conducted on the same data.

1.3 Methodology

In previous work [67] an investigation was conducted on different usability methods, one of which involved eye tracking data. Analysis of the eye tracking data was extremely time consuming. Based on these experiences it was decided to investigate ways of minimising the time an expert would have to spend analysing eye tracking data results. This exploratory research led to the study at hand.

This empirical study made use of the data collected for the usability study (referred to as the Pilot study [67]) which triggered this investigation. The Pilot study data was used because there was already extensive knowledge accumulated on the data from the previous investigation. This allowed validation of the results when different methods were applied to the data.

The collected data was investigated by means of an experimental process. A number of techniques were designed and applied to possibly shorten the eye tracking data analysis process. After a few iterations, a method was identified that automatically produces quantitative results, which allows data filtering and therefore reduces the amount of data an expert analyst will have to spend examining eye tracking data. Further analysis of the data could indicate additional information in the fixation and saccade data. A visualisation technique was designed to communicate these results. A comparative study was conducted with the expert–based usability pilot study, to support these findings.

The larger the dataset, the longer the eye tracking data will take to analyse. Thus, a usability study was conducted with more participants and more tasks, to which the proposed method was applied (referred to as the Validation study [39]). The Validation study was used to ensure that if the proposed method is applied to a larger dataset, then analysis time will be reduced. Another comparative study was done to compare

the results from a proposed approach with the findings from an expert–based usability validation study.

The apparatus, tasks, applications, participants and setting [48] are explained in detail in Chapter 3, as well as the expert–based usability findings. The data collected and analysis using the proposed method are discussed in Chapters 4 and 5.

1.3.1 Ethical clearance

This study complies with the University of Pretoria's Code of Ethics for Research. Ethical clearance was obtained from the Faculty Committee for Research Ethics & Integrity, Faculty of Engineering, Built Environment and IT, University of Pretoria and the College of Graduate Studies Research Ethics Committee of the University of South Africa for the Pilot and Validation study respectively. For compliance, all participants completed a consent form before usability tests commenced. The names of the participants were never saved with demographic, personal or testing data information; all records were delimited by means of unique numbers.

1.4 Contributions

By following the experimental design discussed in the methodology section, the objectives of this study were met. The main contributions of this research are:

- The idea of a benchmark user was introduced in the proposed method to extract comparative, quantitative eye tracking results (Section 4.3). The implication of this approach is that part of the usability analysis can be fully automated, without the need for prior mapping out areas of interest on the user interface.
- A method was developed to produce new indices for automatically highlighting tasks with high deviation from the benchmark user. The following indices were defined:
  - FDI (Fixation Deviation Index): quantifies user fixation points distribution around a benchmark user fixation, to highlight how much the user fixations differed from the benchmark user fixations (Section 4.4).

  - $SDI_{eliminate}$ (Saccade Deviation Index Eliminated): the percentage of user saccades that aligns to that of the benchmark user's saccades, which is then eliminated, quantifying how many similar paths were followed between components (Section 4.5).
  - $SDI_{remainder}$ (Saccade Deviation Index Remainder): the number of user saccades not eliminated, indicating the number of user saccades that differ from the saccades of the benchmark user (Section 4.5).
  - SLI_{total} (Saccade Length Index Total): the total user saccade length of the remaining saccades, to quantify by how much the user saccades differ from the benchmark saccades (Section 4.5).
  - $SLI_{average}$ (Saccade Length Index Average): the average saccade length of the remaining saccades, indicating the type of eye movement that occurred during a task (Section 4.5).

- A method for automatically visualising problem areas on the user interface was developed. The visualisations are based on the deviation from the benchmark user. Two visualisations were produced:
  - BDA (Benchmark Deviation Areas): use the fixation points to highlight the areas where high deviation occurred (Section 4.4.5).
  - BDV (Benchmark Deviation Vectors): indicate repetitive deviation saccades that occur between components (Section 4.5.6).

- Two usability studies were conducted and expert analysis was performed on the data. The Pilot study was used to develop the proposed method and the Validation study was used to show the feasibility of the proposed approach. These two usability tests are discussed in Chapter 3.

The introduction of a benchmark user and deviation indices can also be applied to eye tacking data in a situation where it is required to determine how much the eye tracking data of a novice differs from an expert user. The amount of eye tracking deviation can even be extended and applied in the security field to identify users, detecting if the eye

movements of the authenticated person differs from the current user. This demonstrates some of the potential fields of study where the proposed method can be applied.

1.4.1 Publications produced by this study

The following publications were produced from the research in this study:

- H. Gelderblom, J. de Bruin, and A. Singh. *Three methods for evaluating mobile phone applications aimed at users in a developing environment: a comparative case study.* In Proceedings of the Mobile Communication for Development, volume 3, pages 321–334, New Delhi, 2012. [67]
  Dissertation: The Pilot usability study and expert findings in Section 3.5

- J.A. de Bruin, K.M. Malan, and J.H.P. Eloff. *Saccade Deviation Indicators for Automated Eye Tracking Analysis.* In Proceedings of Eye Tracking South Africa, volume 1, pages 47–54, Cape Town, 2013. ACM. [38]
  Dissertation: Saccade Deviation Index as discussed in Section 4.5

- J.A. de Bruin, K.M. Malan, J.H.P. Eloff, and M.P. Zielinski. *The Use of a Benchmark Fixation Deviation Index to Automate Usability Testing.* In P.S.P. Gamito and P.J. Rosa, editors, I see me, you see me: inferring cognitive and emotional processes from gazing behavior, chapter six, pages 104–124. Cambridge Scholars Publishing, Lisboa, first edition, 2014.[39]
  Dissertation: Fixation Deviation Index as discussed in Section 4.4

- J.A. de Bruin, K.M. Malan, and J.H.P. Eloff. *Benchmark User–based Eye Tracking Indicators for Automated Analysis of User Interfaces.* Submitted to the International Journal of Human–Computer Studies.
  Dissertation: The application of the proposed approach to the Validation usability study as discussed in Section 3.6 and Chapter 5

1.5 Dissertation overview

The following is an outline of the remaining chapters of the dissertation:

Chapter 2 presents the necessary background information to this study. A summary is provided of eye tracking and usability testing. The background research continues to focus on the use of eye tracking in usability and related research on automated eye tracking analysis and visualisation.

Chapter 3 describes the two usability tests conducted for this study referred to as the Pilot and Validation studies. The necessary empirical planning, including apparatus, system, user, locale and tasks for each study is discussed in detail. The chapter also includes the findings from an expert–based analysis for each of the usability tests.

Chapter 4 contains the main contribution of this study, discussing how a benchmark user can be used to automate eye tracking analysis. The resulting metrics and visualisations are discussed; each step in the proposed method is explained and applied to the data from the Pilot study. These findings are then compared to the expert–based findings from the same data.

Chapter 5 presents the results from the proposed method as applied to the Validation study. The effect of the method when applied to a larger dataset is investigated. The findings are once again compared to the findings of an expert analyst, considering the same data.

Chapter 6 brings the study to a conclusion. The overall findings and relevance of the proposed method are discussed. In ending, some suggestions are made as how this work can be used in future research.

Chapter 2

Background to Eye Tracking and Usability Testing

Knowledge is power. For more than a century, immense knowledge has been built up in the field of eye tracking. Eye tracking has been applied in various disciplines and has grown from a research topic to being adapted for commercial use. Usability is one of the fields that successfully adopted eye tracking for evaluation. This chapter provides information on eye tracking and usability and relevant research studies that applied these technologies in various ways.

2.1 Introduction

During the 100 years of eye tracking research, eye tracking has been used in many facets of research. Eye trackers allow for increasingly accurate readings and the capturing devices are less invasive to the users than they used to be [156, 182]. Since the 1980s eye tracking has been used extensively in the field of usability testing. The father of design, Vitruvius, defined three categories of architecture to determine if a building is well designed: *Firmitas*, *Utilitas* and *Venustas* [210]. *Firmitas* requires that the building should be durable and long–lasting. *Utilitas* determines if it meets the needs that it was designed for and makes it convenient for the users. Lastly, *Venustas* refers to the aesthetics. These design factors have been applied to the usability of products since the

first century when it was defined.

This chapter covers what eye tracking is, where it comes from, techniques to capture eye movements and also the quality of eye trackers. This leads to more information on the type of data that is captured and a description of how to interpret the different types of data. The last part of the eye tracking section focusses on how eye tracking is applied in different fields of study, which includes usability testing. The focus then shifts to software usability to provide background information on what usability is and the different methods available for usability testing. The qualitative and quantitative results of usability testing are discussed for these methods. Lastly, research relating to the automation and visualisation of eye tracking in usability testing is covered.

2.2 Eye tracking

This section provides background information on the field of eye tracking.

2.2.1 What is eye tracking?

Our eyes continuously move from one position to another when observing any given visual stimuli; these movements are called saccades. After a saccade our eyes become stationary on a position where you focus; this is referred to as a fixation. Eye tracking is the action of recording these eye movements by any of the various available eye tracking techniques [21, 55, 70, 156, 166, 228].

The eye–mind hypothesis, as tested by Just and Carpenter, shows a strong link between the cognitive activity of a person and what they are looking at [110]. This premise makes eye tracking a very interesting tool, applicable to a vast variety of fields. Eye tracking is divided into two main categories: interaction and diagnostics [48]. Studies have shown that a person looks at an object before pointing at it; therefore eye tracking can be used as an interaction tool, where an interface will react to the gaze of the person [229]. Diagnostics, on the other hand, record the eye movements of a person, which can then be viewed, analysed or exported.

There are various ways in which eye trackers record the gaze direction at a specific time. These technologies can be head mounted or stationary, allowing different degrees

of head movement [44, 140]. The eye tracking evolution is described in the next section.

2.2.2 Where does eye tracking come from?

During 1878 Émile Javal noticed rapid movement of the eye during reading [16, 101]. Lamare, who worked with Javal, continued the study and tried to develop a method to record eye movement. Lamare was able to determine the number of eye movements by enabling each movement to result in a sound, but the overall experiment did not result in accurate results [44, 217]. Research continued into mechanical eye movement recording. Aherns [204, 215] attempted to record eye movement by attaching bristles to an eye to trace eye movement on a smoke drum, but this was also unsuccessful. In 1898 to 1900, Delabarre and Edmund Huey both managed to successfully record eye movements mechanically, building on Ahrens' design [41, 93]. A small plaster cup was constructed with a hole in the center to enable the user to see, which was then placed on the cornea. The cup was attached to a wire and a lever, the lever recorded the eye movements on a smoke drum. Two main disadvantages of the design were that the eye movements were too fast for the mechanical device and some discomfort was caused in the eye. To overcome the discomfort, cocaine was applied to the cornea, which affected the natural movement of the eye [215, 218].

Shortly after this Dodge and Cline designed a non–intrusive eye tracking method with the use of photography [16, 204]. This technique was first introduced by Stratton [215], but it could only record the eye path and number of movements. Dodge and Cline improved the method in 1901 and were able to additionally record the horizontal velocity of the eye. This formed the foundation for light reflections used in modern eye trackers. A camera captured a vertical light reflection from the cornea in order to determine the horizontal movement of the eye [44, 204, 220]. A couple of improvements were made with regard to this method, but a noticeable improvement only occurred in 1905 when Judd, McAllister and Steel used motion pictures instead of single photos [101, 215]. Additional improvement was made by placing a reflective material in the cornea of the eye, resulting in two dimensional tracking and the experiment no longer needed to be conducted in dark lighting [219]. Enhancements of the photographic corneal reflection eye tracking technique continued, even though the processing was extremely resource intensive. Some

key researchers were Koch (1908), Weiss (1911), Grey (1925), and Miles and Shen (1925). They all contributed to improve this eye tracking method [215].

With the eye tracking techniques in place, the research method could be applied in other fields of study. Eye tracking was mostly applied in reading [16, 166]. Tinker extended the normal reading experiment to study the effect of font and page size, font type and layout on eye movements. Buswell was one of the first researchers to apply eye tracking differently, recording how people look at images [21]. Buswell made use of 200 participants who viewed 55 images [82]. This was a large study in the context of eye tracking and even more impressive since this study was published in 1935. The study extended many aspects of eye tracking including metrics, eye movement, the effect of instructions on eye movements, as well as expert–novice eye movement comparison with regard to art work. Later studies were even more creative in applying eye tracking. In 1947 Fitts, Jones and Milton studied the eye movement of pilots over the controls in the cockpit [62]. This was also one of the first studies where eye tracking was applied to the field of usability. In the 1950s, Yarbus investigated how scan paths and fixations were influenced by different instructions, using the same images. In his studies he realised that eye movements will not only move over images, but focus on relevant and necessary areas [10, 48, 218].

Improvement of eye trackers continued throughout history, as it does today. In 1936 Mowrer, Ruch, and Miller discovered that the position of an eye can be determined by means of electrodes attached to the face, around the eye region [228]. Jung used this technology, an electro–oculogram, to measure vertical and horizontal eye movement concurrently [44, 204]. Another milestone in eye tracking history was the first head mounted eye tracker invented in 1948 by Hartridge and Thompson. This eye tracker was the start of eye tracking experiments outside, away from a confining laboratory. Researchers like Land and co–researchers studied eye movement of people driving (1994) [123] and making tea (1999) [124].

Up until the 1970s raw eye movement data recording had been time consuming, but with computers being adopted in different fields, the eye movement could automatically be recorded in real time. From this point on, a lot of focus has been placed on improving mobility and accuracy in eye tracking. By the time personal computers became more

widely available in the 1980s, human–computer interaction became important and eye tracking could be applied in this field. In 1981 this led to the investigation of interaction with a computer by means of an eye tracker by James Levine. The application was initially developed for human–computer interaction for people with disabilities [128, 137]. The use of eye trackers for interaction with a computer has grown and is tending towards a standard input modality, available in computers and mobile devices [107, 28].

2.2.3 How does the eye work?

This section briefly describes the anatomy of the human eye as a reference for the sections that follow. The anatomy of the eye is illustrated in Figure 2.1.

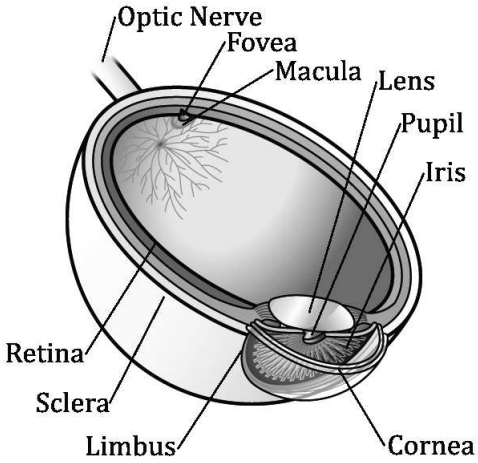


Figure 2.1: The anatomy of the human eye.

The outside of the eye consists of the white part of the eye, known as the sclera and the coloured part, the iris, which has a black center known as the pupil. The iris and pupil are covered by the cornea, which is more curved than the sclera. The limbus is the edge of the cornea that links to the sclera. The light is passed through the cornea, that bends the light and it then moves through the pupil and the lens of the eye that is directly behind the pupil. The lens helps to focus the light on the fovea that is inside the macula, the center part of the retina, which is the lining of the eye. The retina is a light–sensitive inner layer of the eye that converts light into electro–chemical signals

sent through the optic nerve. The focal point of the light entering the eye falls on the macula and fovea that have the most colour receptors [176, 200].

2.2.4 How are eye movements captured?

In 1928, Vernon already summarized methods of eye movement recordings [215], covering 8 mechanical methods and 11 photography methods. Captured eye gaze data can be used to analyse the eye movements of the user without a reference to where the person is looking, just relative to the head of the person. Most eye tracking methods, however, make use of a point of regard when recording eye movements, to know where a person is looking relative to a specific scene.

Over the years, researchers have used intrusive and non–intrusive methods to record eye tracking data. Eye movements can be observed by a person, but eye tracking technology has evolved to allow eye movements to be recorded automatically. Two of the four high level eye tracking techniques are intrusive, using scleral contact lenses [218, 228] or Electro–oculography, involving sensors placed on the skin [149, 180]. The Scleral contact lens evolved from the use of plaster–of–Paris contact lenses, to the use of a lens with a coil inside while positioning the user's head in the center of a magnetic field to produce very accurate eye tracking data [48, 140]. Electro–oculography is a technique that measures the cornea–retinal potential on the surface of the skin around the eyes [194, 204]. The other two high level eye tracking techniques make use of optical methods, making it a non–invasive technique [140]. The two non–invasive high–level eye movement measuring techniques are discussed in this section, namely: video–oculography and video–based corneal reflection.

Photo– or video–oculography

Various forms of image–processing have been used to extract eye movements from images, either photos or videos captured by a camera. The quality of the available camera and the rate at which the images are captured affect the quality of the data, but also the cost of the eye tracker. The eye tracking images used to be analysed manually, but as technology progressed, the image processing is now done automatically [42, 180]. Image

processing makes use of distinct physical eye characteristics, such as the shape of the eye, the pupil and the limbus to determine the eye movement [48].

A problem with tracking the limbus is that the eyelids can cover the top and/or bottom part of the limbus, hindering vertical eye tracking. Horizontal eye movements are easier to track, because of the high contrast between the iris and sclera, making a photo–diode a fitting device to additionally use.

The pupil is a well–defined feature of the eye with a distinct contour. To determine the position of the pupil is in many cases easier than tracking the limbus – unless the contrast between the pupil and the iris is very low. The chances are also smaller that the eyelids cover a part of the pupil. By using an infrared light, the pupil is even more clearly defined in the image and the colour of the iris does not affect the perceptibility of the pupil [129, 140, 228]. By using a combination of limbus and iris tracking, it is possible to produce better eye gaze tracking, as shown by Duchowski [48].

Video–based corneal reflection

Most of the current eye trackers [107, 28] make use of one or more near infra–red light sources to illuminate the pupil for accurate, real time eye tracking, even in poor lighting conditions. Infra–red light is outside the visible range for humans, therefore it is neither visible nor harmful to the human eye. Thus, making it a sufficient component to use for eye tracking as it will not cause any distraction for the user and provides a consistent light source for eye tracking [48, 129, 169].

The infra–red light also causes four reflections from different surfaces of the eye, called Purkinje images [16, 137]. Two of the reflections are from the front and back surfaces of the cornea and the remaining two reflections are from the front and back surface of the lens of the eye [74]. Techniques make use of different reflections to estimate eye movement. The first Purkinje image is known as the corneal reflection and is the most widely used of all four reflections [140]. This is used to determine the spatial vector between the center of the pupil and the center of the corneal reflection. This combination allows for a small amount of head movement without affecting the data accuracy. For more head movement, additional facial features need to be traced to correctly record the eye movement in three dimensions [92]. The dual–Purkinje, for example uses the first

and the fourth (reflected from the back of the eye lens) reflections to determine eye gaze [37]. This method, even though it is very accurate, does not allow any head movement.

By placing the infra–red light source at the same position as the visual stimuli, the point of regard can be determined. For this method calibration is required prior to data recording, which allows for much more head movement [180, 228].

2.2.5 What makes a good quality eye tracker?

The quality of data that is recorded by an eye tracker is affected by a number of factors. Depending on the requirements of the needed data and the available resources, the appropriate eye tracker should be selected. Factors influencing the quality of eye tracking can be related to the participants, experimental set–up or the equipment [16].

For the eye tracker user, some physical characteristics can influence the data quality, depending on the type of eye tracker. The shape of the eye and the colour of the iris can hinder eye gaze recordings if the pupil and/or iris cannot be determined. How deep and close together the eyes are situated in their sockets, ethnicity as well as the curve of the cornea are all factors that could effect eye movement recording quality [74]. Some external factors can also impact the data quality if there are reflections on the glasses that hinder the recording of the eye. The environment where the eye trackers will be utilised should be considered when the type of eye tracker is selected. The availability of light, position and mobility of the user could affect the data accuracy and should all be taken into consideration and adjusted before using the eye tracker [16].

Spatial and temporal measurements are used to determine the robustness, accuracy and precision of the eye tracker data. Spatial accuracy is the most referred to measurement and is the difference between where a user looked and the actual point that was recorded. A standard of < 0.5 degrees difference is regarded as a good quality eye tracker [44, 119, 140]. A recent study done by Holmqvist [89] does show that a 0.5° difference can significantly affect the outcome of a study. Latency or temporal measures show the difference between the exact time and the recorded time of the eye movement. Precision measures can be calculated to determine the spatial and temporal variance of the data [44, 88], which also reflects the quality of the data. Spatial resolution measures the smallest eye movement that can be recorded. This is relevant when micro–saccades

should be recorded. Lastly, the overall robustness of the data of a study indicates how well the eye tracker performed in recording all the data without data loss due to external factors [16, 89, 198].

2.2.6 What quantitative data is available?

Six basic types of eye movement have been identified over the years: smooth pursuit, optokinetic reflex (nystagmus), vestibular, vergence and the two most used types, fixations and saccades. Smooth pursuits allow a person to follow a moving object from a certain distance, moving at a certain velocity by adjusting the eye gaze accordingly. Optokinetic nystagmus is also smooth pursuit, but involves the eyes repeatedly jumping back to a specific position, with a fast saccadic movement [48, 218, 228]. The eyes will counter–act head movement to a degree, by means of vestibular eye movements, to have the same view of an object. Allowing a person to focus on an object close to their eyes, vergence eye movements shift both eyes inward [119, 127, 166]. These eye movements, together with pupil dilation, blink rate and eye lens measurements are used in very specific studies. Most eye tracking studies that focus on information processing and the intentions of the user, however, make use of fixation and saccade data. This data can be applied in diagnostic or interactive systems.

Gaze data

The gaze data refers to all the data points captured by the eye tracker. The data is captured at a certain frequency, depending on the capabilities of the eye tracker being used. Typically the sampling rate of eye tracking devices range between 50Hz and 250Hz (between 50 and 250 samples per second), this results in voluminous amounts of raw data [101] and analysis is necessary to extract different types of eye movement. After the gaze data has been captured, one of many algorithms can be used to separate the data into fixations and saccades [185]. The first order eye gaze data captures the position (x– and y–coordinates) and the time of the eye gaze [164].

Fixations

A fixation occurs when the eye movement is kept steady over an object in order to visually observe an area [55, 69, 72, 228]. Stratton's work in 1906 moved the focus of eye tracking away from just the saccades towards the fixation location of the viewed scene [219]. A fixation usually lasts between 100 to 300 milliseconds, during which information processing occurs of the content being viewed [13].

A fixation is not completely motionless on one point, minuscule movements occur during a fixation, called tremors or micro–saccades. These micro–movements are necessary to keep the visual stimulation alive when staring at the same location [16, 166, 198]. There are numerous algorithms available to define a cluster of gaze points and micro–saccades as a fixation [185]. Different algorithms are available to identify saccades and fixations: by identifying gaze points with low moving speed (velocity–based), with eye gaze positions that are contained in a certain area for a specified time (area–based) or lastly, with gaze positions clustered together (dispersion–based) [23, 164, 185]. The velocity–threshold identification (I–VT) algorithm is a well–known algorithm and available in commercial software like Tobii Studio. This fixation identification algorithm calculates the velocity between each gaze point. If the velocity of a movement is higher than the set threshold, then the movement will be classified as a saccade. All consecutive fixation points within a certain threshold are grouped together in a cluster and the centroid of that cluster presents a fixation [16, 183]. Each algorithm and threshold set has a strong influence on how sensitive the algorithm is for classifying fixations.

Saccades

A saccade is the gaze relocation movement between fixations [44, 55, 72, 101]. The word saccades originates from a French word meaning 'jerk' to describe rapid eye movements; the term was defined by Javal in the 1970s in Paris [215, 218]. Saccades are measured as the angular distance that the eye moves from one fixation to another, known as amplitude. A saccade lasts between 10 and 100 milliseconds and can reach speeds of up to 800° per second [209]. Saccade latency occurs just before every saccade when the user subconsciously determines where and when the gaze will move [110]. Saccades can either be voluntary, when the eyes are directed to a specific area, or reflexive if moving

to a stimuli in the peripheral vision [97].

Saccadic suppression occurs during a saccade leading to no visual information processing (decoding) and the eyes are blind to a certain degree [13, 26, 169]. Some studies state that should a target be moved or deleted from a scene during a saccade then the saccade will still end in the original, intended position, then relocate to find the target [161]. Whether or not cognitive activities are suppressed during saccades is still debated [166]. Liversedge and Findlay [131] argue that cognitive processing occurs not only during fixations but also during saccades. A study by Henderson and Hollingworth [83] shows quantitative data on how saccades are effected by moving, changing or deleting the target object.

Related data

A scan path is a set of consecutive fixations and saccades. The scan path length can be the fixations and movement between two objects or all the fixations and saccades during task completion. A number of metrics can also be derived from the scan path data to provide additional information on search patterns, cognitive load and thought processes [69, 156, 215].

Other eye movement data output that can be captured are pupil dilation and blink rate. A decrease in blink rate and/or wider pupil dilation can indicate high cognitive load. Unfortunately, these metrics are easily affected by external factors like changes in the light and eye dryness [86, 153, 169, 215].

Metrics and related cognitive processes

From the different eye movement data, additional metrics can be derived. The metric should be considered within the context of the given task. A study by Just and Carpenter demonstrates the link between the fixation behaviour and the cognitive load of a person completing a given task [110].

Jacob and Karn [101] surveyed 21 eye tracking studies and counted the number of times each eye tracking metric was used in these studies; the top six metrics were all related to fixation. Ehmke and Wilson [55] summarised the different eye movement metrics (mainly from:[36, 72, 101, 110, 156]) such as: fixations, saccades, scan path and

gaze data used in usability eye tracking studies that relates to the cognitive activities of participants. The following metrics were gathered from the Ehmke summary and other resources:

Fixations:

Number of fixations indicates the total number of fixations that occurred in a given time frame, task or during a scan path. A higher number of fixations per task can indicate inefficient searching [72].

Fixation duration can show more than just how long a person focuses on an object – when a person fixates on an area, information from that object is decoded. In some cases fixation will remain on an object for a while longer while the user awaits system response [110]. Depending on the context of the task, longer fixations can indicate difficulty to obtain information or that it captures more interest [55].

Fixation spatial density is a measure representing search efficiency during a task. If a user is tasked to find an object, then high fixation spatial density indicates a inefficient and distributed searching [71]. Studies done by Cowen et al. [36] and Goldberg and Kotval [72], use a grid to determine the fixation spatial density by calculating the number of fixations per cell.

Time to first fixation is usually the time from the start of the task to the first fixation of a specific object or area of interest [23]. Longer time to first fixation indicates that relevant information is difficult to locate.

Fixation duration mean for a task is not related to an element in a visual stimuli, but shows the average interpretation time of the task [36, 72]. A longer average duration indicates more time required to interpret the visual stimuli while completing a task.

Saccades:

Number of saccades is the number of times that a user re–orientates where he/she is looking. This does not include micro–saccades and can be calculated as the total

number of fixations minus one [72, 101]. A greater number of saccades can also indicate inefficient searching.

Saccade length (amplitude) is usually measured in degrees that represent the angular distance between sequential saccades [72] or in pixels to measure the distance moved across a visual stimuli. A larger saccade amplitude indicates a more directed and planned eye movement, whereas shorter saccades can indicate more searching [73].

Average saccade length provides an overall indication if the user intently searched or scanned the visual stimuli [70]. Excessive searching can be identified if this metric is above a specified threshold [117].

Regressive saccades also known as backtracks, are saccades moving in the opposite direction to the general eye movement. For example if a user is reading from left to right then a regressive saccade would move from right to left [166]. Regressive saccades can show if the user's objective changed, if other information was expected or if a user misread something [86, 131].

Scan paths:

Scan path length is the total length of all the saccades in the scan path, measured in pixels. Longer total scan paths indicates poor scanning behaviour, when a person could not locate the necessary information efficiently [72].

Scan path duration is the total time of all the fixations in the scan path while completing a task. As fixations are used to calculate this metric, it is related to cognitive processing. A longer scan path indicates that more cognitive processing is required for the task [72].

Transition matrix is the back–and–forth eye movements that occur between two objects rather than just scanning the one object completely before moving on to the next object. This could also relate to decision making [72, 73].

2.2.7 What qualitative data is available?

Eye tracking data is location–based and humans can better interpret two dimensional data when seen visually. For this reason, a number of standard visual outputs are used in eye tracking. Data should also be superimposed on the relative visual stimuli, to avoid incorrect results if the data is interpreted out of context.

Figure 2.2 shows the different eye tracking visualisations discussed in this section. The visualisations are of a single user completing a task during a usability study on a mobile user interface.

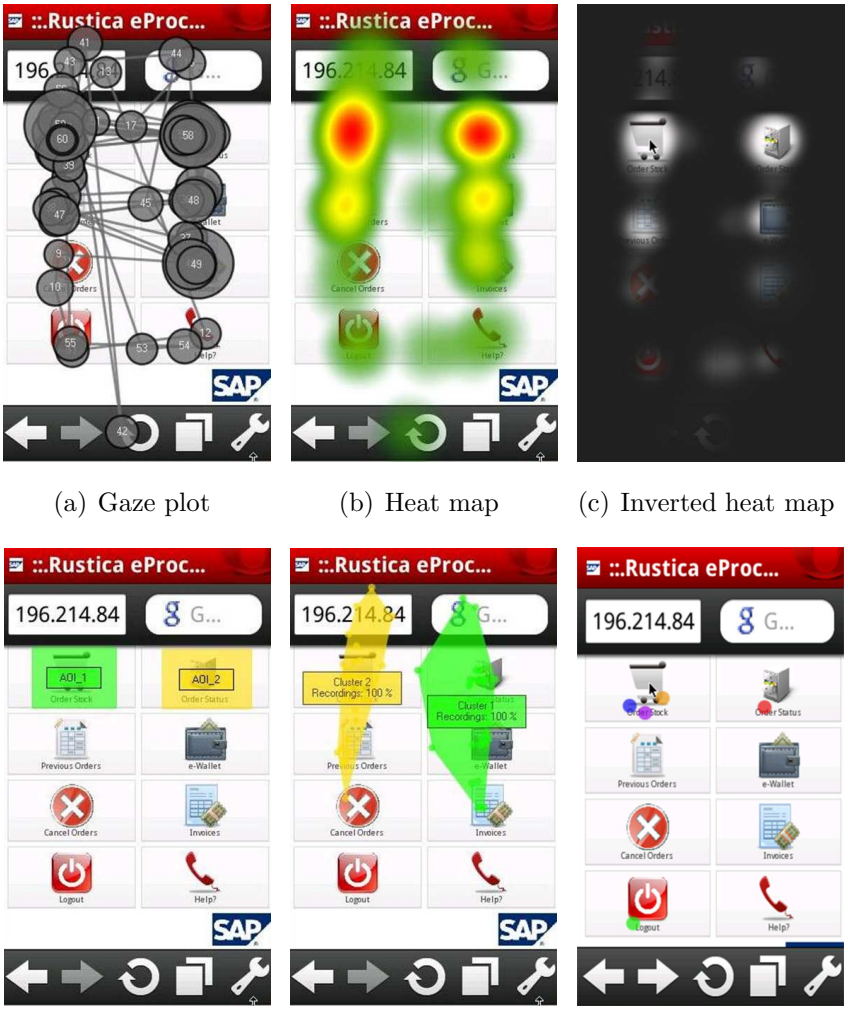
Gaze plot

The first visualisation is the gaze plot of the scan path of a participant while completing a task. As shown in Figure 2.2(a), fixations are represented by circles and lines represent the saccades and the path followed between fixations. The order of the fixation points are indicated by numbers in the circle and the size of the circle can be a fixed size or directly related to the fixation time. The gaze plot data of one or multiple participants can be superimposed onto the scene, depending on the needs of the study [134, 169].

Heat map

A heat map refers to a visualisation, utilising colours usually associated with representation of heat, to highlight areas on a target that received a lot of attention. Heat maps are superimposed over the target scene and represent aggregated data over a time span. Different colours can be used for the heat maps, but shades of colour including green, yellow and red are mostly used. Colours are assigned according to the fixation intensity and a radius is set to fade the colour around the specific point, ensuring a smooth flow of colours. The visualisation in Figure 2.2(b) is a heat map representation, where the darker colours show higher fixation.

Heat maps can be used to visualise different aspects of fixation data, such as fixation count, absolute fixation duration, or relative fixation duration (relative to the total viewing time for multiple participants or tasks) [17]. In addition, participant percentage heat maps can be used to visualise the number of participants who fixated on the same



(a) Gaze plot (b) Heat map (c) Inverted heat map

(d) Areas of interest (e) Clusters (f) Bee swarm

Figure 2.2: Eye tracking visualisations exported from Tobii Studio.

point [17, 208].

An inverted heat map (see Figure 2.2(c)) uses the same calculation as a normal heat map, but the whole scene is superimposed by a solid colour and the areas with fixation are highlighted by changing the opacity of the solid overlay [169].

Seeing that there are many variations of heat maps, all the necessary data should be made available to give context to the visualisation, such as the type of fixation data visualised and the maximum values represented by the most intense values [70].

Areas of interest

The areas that an expert expects a user to find significant are defined as the areas of interest (AOI). These areas are defined on the scene by geometric shapes available in eye tracking software [193, 208]. Figure 2.2(d) shows two AOIs that are highlighted on the user interface.

Quantitative eye tracking data, such as the percentage of fixations on an AOI, relative to all fixations, the number of participants who fixated on that AOI or the order in which the AOIs were fixated on, can be exported for each AOI [4, 53, 73]. There are even methods to create AOI for dynamic scenes as shown in an article by Papenmeier and Huff [152].

Clusters

Clusters are areas with a high fixation or gaze point density [169, 208]. As shown in Figure 2.2(e), two areas were identified where high fixation counts occurred. Clusters can be used to automatically highlight AOI on the target scene, although this could be misleading if an insignificant object draws the attention of all the participants. A mean shift clustering algorithm, designed by Santella and DeCarlo [187], is used to cluster the fixation points. Unlike the k–means algorithm, the clustering algorithm does not require a number of clusters to be provided, only a distance threshold.

Bee swarms

Bee swarms do not show summarized data, but are a frame–by–frame replay of the fixation data of one or more participants. This is especially useful to see how more than one participant react to scene changes or other information displayed. Figure 2.2(f) shows the fixations of each selected participant in different coloured dots at the specific point in time [16, 208]. This analysis method is time consuming, because the expert will have to watch the replay, sometimes in slow motion and multiple times to extract the needed information.

Playback

Each fixation point and saccade of a participant can be replayed to show the scan path that the participant followed over time. In some applications a moving window can be set, which will show a trail of the scan path followed over the window time span. This method is the most time consuming, as an expert will have to watch all the necessary eye tracking playbacks of one or more participants for the data analysis, but it is particularly useful for dynamic scenes [193, 208].

2.2.8 How can eye tracking data be used?

Eye trackers have been applied in different fields and in many interesting ways. In a comprehensive survey, Duchowski [47] distinguishes between two main eye tracking applications: interactive and diagnostic. Interactive eye tracking allows the user to use eye movements as an input module. Diagnostic eye tracking applications record eye movements without the user being effected by the recording, and then analysing the eye movement data accordingly. Some eye tracking application fields are discussed in this section for both interactive and diagnostic, with special focus on eye tracking in usability.

Eye tracking in interaction

The initial research for eye tracking as an input method was first introduced as an assistive technology for people with severe disabilities, like quadriplegics, as an interaction and communication method. Jacob [100] investigated interaction methods to allow natural interaction using eye gaze.

Eye control has been used in combination with mouse (MAGIC tool [229]), keyboard (EyePoint tool [121]), speech [139] and wink [198] interaction in attempts to improve the input method. Users can use an on–screen keyboard or eye gestures for writing [133, 226] or tools, such as EagleEyes [68] or EyeDraw [91] to draw images. Gaze–contingency also forms part of user interaction, where displays are rendered or updated according to the gaze position of the user [49, 74]. The eye tracking interaction field is growing as the accessibility of accurate affordable eye trackers is increasing.

Eye tracking in diagnostics

For the diagnostic application of eye tracking, the data is captured and then analysed for different purposes. This application can range from neuroscience to marketing.

When eye movement such as saccades and fixations were first defined, the focus of the studies was mostly on how people read and perceive images. During picture viewing a person will get an overview within the first few fixations and for the remaining viewing time, detail about the image will be gathered [132]. For reading, standard eye tracking measurements and reading patterns have been defined for different age groups [166] and significant deviation from these metrics could indicate reading disorders. A reading disorder such as dyslexia, can be identified by a high frequency of regressive saccades [165, 196]. There is a clear difference in eye movements when a person is reading out loud or silently [171], showing that given the same visual stimuli, eye movements can differ depending on the task at hand. Buswell [21] conducted extensive studies consisting of over 2000 eye movement records; some of his investigations included studies on areas with high fixation density, fixation duration over time as well as the effect of instructions given prior to viewing an image on the fixation positions [218]. Rayner et al. [167] also investigated the effect of instructions on eye movements, but on an advertisement. Participants fixated on pictures for task related to the aesthetics of the advertisements, but then fixated more on the text when asked if they would buy the product [168, 189]. The marketing field further extends to websites to determine if viewers pay attention to advertisements and to shopping centres to examine which products draw the user's attention [19, 84, 94].

Eye tracking research of humans completing everyday tasks reflect our cognitive activities. Land et al. [124] investigated the eye movements of a person making tea and Hayhoe [78] investigated the task of making a sandwich. In a comparative study, using the eye movement resulting from these two studies, it was found that similar eye movements are present, such as a high fixation ratio on the object used for the current task at hand [122]. Other every–day tasks, such as map reading, model building and hand washing show more advanced planning, while performing the task. For example, users will fixate on the object they need to use long before moving to that object [154]. Eye tracking diagnostics can have very practical applications, such as identifying driver

fatigue [106], which was built on research that monitors eye movements in certain driving situations [135]. To simplify the analysis of eye movements during natural tasks, a virtual environment can be used and the exact eye movements of the user on virtual objects can be determined [79].

The link between cognitive activity and gaze points allows for interesting research in the field of neuroscience. Eye tracking has been successfully applied to identify schizophrenia by monitoring the smooth pursuit [177] and autism by analysing where a person fixates on a scene of people communicating [18, 201]. Attention–deficit/hyperactivity disorder (ADHD) and fetal alcohol syndrome disorder (FASD) can also be identified and distinguished [213] by means of eye tracking diagnostics.

Eye tracking in usability testing

Eye tracking usability testing forms part of diagnostic eye tracking applications. This section is dedicated to usability testing, as the focus of this study is on the use of eye tracking in usability testing.

Usability does not only refer to the ease–of–use of websites or applications, it extends to human machine interfaces (HMI). Some studies have investigated the use of eye tracking to evaluate the usability of HMI, some of which include cockpit design [62], air traffic control [116], e–book readers [197] and medical equipment [224]. Interaction between human and machine should be intuitive and effective. By analysing cognitive activity, the sequence of interaction as well as where the user expects an interaction module to be, the machine usability can be improved for optimal utilisation.

A large variety of eye tracking usability studies focus on websites. Goldberg and Kotval [72] did a comprehensive study on what eye tracking data can expose about usability of systems and how users search on websites to optimise the placement of headings [73]. Ehmke and Wilson [55] summarised previous research on how eye gaze metrics can be applied to identify certain usability issues on websites. The benefits of using eye tracking as a web usability analysis tool include, but are not limited to: areas highlighted that draw the attention of the user, determining if crucial information is overlooked by the users, highlighting the strategy of the user when completing a given task and outputting quantitative comparative data for unbiased analysis [141, 173]. Some

studies have investigated and confirmed the value of eye tracking data in usability testing by investigating the correlations between the eye tracking results and known usability issues [56, 179].

The same techniques and metrics of eye tracking usability analysis of web pages can be applied to desktop applications [58, 150]. Byrne et al. [23] investigate existing usability models and incorporate eye tracking data to the models to investigate user interaction with drop–down menus. Video–based corneal reflection eye trackers that are built into a desktop screen are ideal to test website or desktop applications. Studies have been successfully conducted on a mobile device, but data analysis tend to be more qualitative than quantitative [31, 192, 207]. Eye tracking tests on mobile applications were usually conducted using emulators on desktop computers, but newly available eye trackers in a mobile device overcomes this problem [107].

Unfortunately, eye tracking analysis of usability studies can often be very time consuming [73]. With the availability of various cost effective eye tracking devices that are available for any computer or mobile device, the possibility exists to perform large scale, remote eye tracking usability studies [60]. This will allow usability studies to reach the exact target user group, in the relevant environment, without the logistics currently associated with these evaluations. Even with sophisticated tools it is still time consuming to extract information from the eye tracking data, such as: heat maps, scan paths, fixation clusters and even basic statistics for a specified task [101]. More detail on usability and usability testing methods are provided in the next section.

2.3 Usability

This section covers usability of software applications as well as usability testing. The different usability testing methodologies are discussed with focus on user–based automated testing.

2.3.1 What is usability?

Usability, as defined by the standards board ANSI [190] and ISO [211], is the ability for specific users to achieve specified goals efficiently, effectively and satisfyingly within

the given context. The user interface is the communication layer between the user and the application logic. A user should be able to interact with the system with ease, irrespective of the level of complexity of the application logic [22]. A usable application allows a user from a defined user group to complete a given task (effectively) within an acceptable time frame (efficiency). The given application should also provide a positive experience (satisfaction) for the user during task completion without issues like errors, latency or unexpected behaviour [11, 223].

Learnability, memorability and errors are criteria defined by Nielsen [143] that should be considered, over and above the criteria as defined by the standards board. Learnability shows how quickly users can figure out how to complete a task, without extensive training. After a period of not using the system, a user should be able to return and remember how to use the application; this refers to the memorability of an application. Lastly, errors in a system should be avoided, but the number of mistakes that a user makes and the recovery rate has a direct effect on the usability. Other usability criteria have also been defined by researchers as summarized in these studies: [85, 105].

2.3.2 Why is usability testing important?

Before the 1980s a very limited group of users had access to computers. For a new system user there was time for training and the time spent on a task was not a main concern. In the 1980s when computers became more widely available, the number of users to train and systems available increased significantly. This resulted in high costs for businesses in terms of time and resources. A great need for more user friendly applications arose. Users should be able to complete a task without hindrance or additional assistance [145].

Usability can have a direct effect on a business, whether it is to improve the internal systems or to improve the product that the company sells. Increased productivity, less post–sales support and training are all benefits of better usability, as discussed by Dumas and Redish [51].

2.3.3 How can usability be determined?

Computer interaction is part of our daily lives while using mobile phones, desktop computers, ATMs and our television sets – all these are computers systems that should be user friendly and improve our daily lives. In order to determine if there are usability issues or to extract usability metrics, a usability evaluation should be performed on the system. Usability evaluations are formative, if they are applied during the development process or summative, when applied to a complete system [75, 153, 174]. For each of the two usability evaluation types, there are a number of phases that should be applied. These phases are, capture, analysis and critique. Usability data is collected during the capturing phase to use in the other phases. Secondly there is the analysis phase; observations are made about captured data of the given system. In the critique phase, recommendations are made on how to improve on the problems identified during the analysis phase [58, 99].

There are a number of formative and summative usability evaluation methods (UEM). The evaluation methods are divided into classes, but the number of UEM classes varies in the available research. Nielsen [143] defines three UEM classes: inspection, testing and inquiry. Ivory and Hearst [99] classify the different usability evaluation into five categories: inspection, testing, inquiry, analytical and simulation. Other studies include classes such as expert–based, user evaluation, automated, empirical, heuristic, observational and model–based evaluations [46, 90, 102, 153, 203]. Fitzpatrick [63] grouped UEMs in one of four quadrants, real or representational users and real or representational systems. For the purpose of the study, the following four usability classification groups are used:

Inspection: Refers to expert–based UEMs that involve one or more expert user, specialist, developer or designer.

Testing: Applies a user–based approach involving participants from the defined user group during an empirical evaluation.

Inquiry: Also involves users, but on a more qualitative basis, to capture user opinions about the system, such as likes, dislikes and preferences.

Model: Utilises models, mock–ups and simulations to analyse the usability of a system by means of predefined criteria.

Usability evaluation methods are categorised in the above mentioned classes and can be automated to different levels. Ivory and Hearst [99] discuss different automated usability methods and separate automation into four different types, one for each of the phases of usability evaluation (capture, analyse and critique) and one case where there is no automation.

The following section discusses different types of UEMs in each of the listed classes to give an overview of the methods and their applications.

Usability inspection methods

One or more of the following people are involved in inspection methods: usability experts, software developers, user interface designers, expert users and the client. The involved parties evaluate the usability of the system during different stages of the product's development [144]. These methods can be used to identify usability issues, comment on the overall usability of the system and highlight pressing design matters. Inspection methods remove the need to get the appropriate user group and allow for empirical testing and only require a few experts for the testing [87].

Cognitive walk–through: The cognitive walk–through method is based on the idea that a user will explore a new system in order to figure out what to do, rather than undergoing formal training. The involved evaluators are required to have extensive knowledge on the cognitive processes of the user group, tasks and the system in order to correctly evaluate each task [102, 155]. The experts step through the tasks and follow a dialogue based on the predicted user actions. This process can be introduced early in the development of the product [6].

Heuristic evaluation: The method of heuristic evaluation involves a couple of usability experts, familiar with the usability guidelines to which software should conform, that compare the user interface with the predefined criteria. The outcome of this method is a list of identified usability issues [142, 147]. This cost effective method

usually requires more than one expert to identify a variety of usability issues and to avoid biased views [203].

Pluralistic walk–through: This method involves a wide spectrum of users such as the user group, developers, usability experts and other members with knowledge of the system. Users identify the usability issues by stepping through a task and noting their thought processes and actions [158, 203]. It could be difficult to find users to iteratively participate in the pluralistic walk–through [87].

Formal usability inspection: There are six steps involved in this method, which a moderator facilitates to capture usability issues and solutions. First, the moderator plans and organises the session and then secondly provides the participants with the needed user profiles and system information during the kick-off meeting. The participants are usually from different backgrounds, such as developers, system users, product owners and usability experts. For the third step, each participant is asked to review the system by stepping through a number of tasks on their own and then provide feedback about the usability of the system. During step four, the feedback (issues and possible solutions) is logged by the moderator. For step five, the moderator documents the suggested changes, which will be implemented once approved. The last step involves a follow-up session with the participants on the reworked system [6, 87].

Other inspection methods: Additional inspection can be conducted by an expert by going through a check list of features to determine if the system has all the necessary steps to complete the task easily. The designers can also check the design to make sure that it is consistent with other systems from the same company. Lastly, an expert can determine if the usability conforms to a set standard, whether international or defined by the company [144].

Usability testing methods

Participants representing the user group partake in the usability testing methods by completing tasks on a prototype or complete system. These tests are user–centered and the tests should be set up to identify usability issues on the user interface as well as in

the processes of the system [174, 203]. These methods are effective in identifying most of the usability issues that relate to the users of the system. A disadvantage of this method is to find actual users or participants who can represent the intended user group and to take time to complete the usability test [9]. The following are categories of usability testing methods:

Think–aloud: The think–aloud process involves the users verbalising what they are thinking or wondering while performing a task and an expert will record the output. Usually a video or audio recording is made for reference and the expert will record the tasks and cognitive processes of the user in a predefined coding scheme. This method provides great insight into why certain users have difficulties completing a task. The unnatural task of verbalising everything that the user is thinking might be distracting [3, 104, 153].

A variation of the think–aloud method, is the question asking method; where the usability tester asks the participant a number of questions relative to the task at hand. By prompting the user for specific information, the responses can deliver more comparative results [174].

Performance measurement: The participants perform a number of predefined tasks on a prototype or complete system, while predetermined usability performance measures are recorded. Performance measures include, but are not limited to: time spent on a task, number of errors, number of events, number of times assistance is needed, time to first event, error recovery rate and percentage of tasks completed [2, 178, 202]. The quantitative data is usually captured automatically by an additional tool available or by manually coding the session by means of a video replay [99]. These tests can be performed in a controlled usability lab or remotely, depending on the availability of travel funds or remote testing tools [77].

Co–discovery: During the co–discovery method the two participants discuss what they think the next step in the task should be and motivate why. This allows an expert tester to automatically or manually record natural communication of the thought processes of the users in a conversation. This method can easily be conducted either in a laboratory or in the field [174, 178].

Coaching: The number of times a user asks for assistance is used as a performance measure. This method usually requires one expert of the system to assist and coach the user and another usability expert to facilitate the study [143]. By recording the questions that the users ask the expert during this time, provides rich information about what instructions should be added to the system or documented in the manuals [46].

Eye tracking: Recording where the user is looking while completing a usability test can give insight into the cognitive activity of the user. The eye tracking usability testing method provides quantitative and qualitative data. This can be interpreted by an expert analyst to identify usability issues and reasons why a user struggled to complete a task. Eye tracking can be recorded without distracting the user and some cognitive processes can be identified from the eye tacking results without the think–aloud method [52, 146]. Measures, such as average fixation duration and number of saccades are recorded and provide insight into the usage patterns of the participant [34].

Usability inquiry methods

The inquiry methods, like usability testing methods, involve users of the system, but these methods just focus on the overall system usage and user opinions, likes and dislikes. The inquiry method can be applied during any development stage: before the design starts to identify the needs of the users, during prototyping to get an overview of the users' opinions and also after the system is completed to determine the overall satisfaction with the system [12, 102]. There are two main activities of inquiry methods: observing the user in the natural environment and getting the views of the user.

Field observation: Both usability studies conducted in a lab and in the field provide insight into the usability of a system. A system used in an everyday setting allows the detection of performance issues, environmental effects and practical inefficiencies [102]. Prior to the development of a system, a field study can be of great value to observe the users in the field and collect information on how the systems should work in the relevant environment [90].

Question–asking protocol: Questionnaires, interviews and focus groups are different methods of inquiry to get the requirements and user views of a system [2, 6, 46]. Users answer questionnaires independently and with the amount of detail which they see fit. Interviews on the other hand, allow the analyst to gather more detailed answers from the user [203]. Lastly, focus groups gather the views of multiple users at once, but users can influence each other's opinions [94].

Usability model–based methods

In 1980 Card et al. [25] developed a model that takes the human perception, memory, cognitive activity and movement into consideration in order to accurately predict the interaction with a given system. To complete a task, a user performs a number of activities, such as perceptual and cognitive processing cycles, eye movements, visual and auditory capacity and movement processing. Each of these activities is allocated a timespan and occurs either in serial or parallel. The model human processor (MHP) is used to identify efficient layouts and compare user interfaces quantitatively and even suggests a more efficient layout [102]. The following describes two modelling approaches: analytical models and simulation [58].

Analytical modelling methods: Analytical models utilise a user and/or system models to predict and analyse the performance of the user's interaction with the system. The models need to be created very specifically for a task, interface and user.

Card et al. [24] created the GOMS (Goals, Operators, Methods, Selection rules) model based on an expert user's performance to complete a task. The GOMS method includes the *Goals* that the users have to complete by means of *Operators* available in the system. The user will follow a *Method* which involves sub–goals and operators to complete the given task. *Selection rules* are defined if there is more than one method that can be followed to complete a task [22, 24]. All of the human activities required to complete a goal should be defined in the user model [115]. The time assigned to each human activity is also dependent on the skill level of the represented user. The GOMS provides predictions on the execution time, learnability and error frequency [174].

Simulation methods: Using the MHP model of the user or interface, interaction is mimicked by the simulation in order to capture performance and analyse the interaction with the system. Simulations can be executed with a variety of parameters to provide insight into different skilled users as well as various combinations of interface layouts and components used in a system. From this data, informed decisions can be made as to which design will be sufficient for a specific user group [99, 155].

Various tools have been developed to simulate tasks, processing and component interaction of a system such as ACT–R, CCT, Soar and GLEAN. The tools require user profiles and steps to complete the task. Some tools, like the Genetic Algorithm Model, can simulate how different users will learn how to complete a task. The Automatic Mental Model Evaluator (AMME) generates Petri–nets to represent the simulated steps that users will follow to complete a task. These simulation methods use similar input to the analytic models, such as eye and mouse movement, to measure complexity, performance, predictions, problem solving and searching [33, 182].

2.3.4 What data results from usability testing?

Usability can be measured by performance metrics, captured while a participant interacts with a system as well as from user feedback on the system. Performance metrics, such as mouse clicks and task success, can be used to measure two of the three key aspects of usability: effectiveness and efficiency. This type of data can only indicate where and what the problem is, but it cannot say why there is a problem [5, 175, 214]. Feedback from the participant, either from a questionnaire or observation, can provide information on the satisfaction and ease of use of the system. The following section describes a number of quantitative metrics captured automatically or manually during a usability test:

Time on task: The total time spent on a task directly reflects the efficiency of a task [214, 223]. Time can also be divided into sections such as the amount of time spent on errors, time spent on reading and time the user was not assisted (productive period) [14, 43, 125].

Number of errors: User interfaces should be designed to prevent the users from running into errors. Errors can include unsuccessful tasks, incorrect actions and the wrong sequence of events performed [190]. Errors are task dependent and the error probability or error frequency should be calculated per task [157, 188].

Task completion: Task completion is a binary value, indicating if the user completed the task or not. The objectives of a task should be clear so the participants know when they have completed the task at hand [64, 214]. The number of tasks completed within a certain period and the ratio of task completed against task failures can be derived from this metric.

Task success: The level success at which a participant managed to complete a task can be determined by criteria, such as: number of attempts, number of regressive actions performed, method of completing the task and percentage of subtasks completed [214]. These levels are specific to a test and should be predefined [155, 157].

Events: Events are significant user interactions and differ depending on the input device. The relevant actions and sequence of events should be predefined and captured [85, 214]. The number of events that was triggered or not triggered can be counted to show which actions the user missed.

Assistance: Should an interface have insufficient information or an illogical flow of events, then the participant could ask the test facilitator for assistance or use the manuals, if available. The frequency or the time spent on the task can be used as performance metrics to identify if sufficient information is available on the interface [43].

Lostness: Lostness is a metric that indicates how far the user was from the perfect navigation sequence. Smith [199] used lostness in website navigation, but this can also be calculated for desktop and mobile applications if the navigation is recorded.

User feedback: User feedback can be quantified to a certain degree, with the use of Likert scales, the participants can specify how satisfied or frustrated they were with a task [223]. Positive and negative comments can be counted, as well as the number

of participants that mention similar aspects of the system and user experience that they viewed as significant [5, 105].

Depending on the usability study, different metrics can be used to determine the efficiency, effectiveness and satisfaction of the participant with regard to the given system.

2.4 Automated eye tracking analysis in usability testing

Automating eye tracking usability testing keeps the evaluation user–centered and assists in reducing the time and resources spent on usability testing. Usability tests have been automated by analysing quantitative performance metrics, extracting patterns from user interaction with the system, deviation from expected task completion behaviour and to deduce findings from statistical or visual results [99].

2.4.1 Automated usability analysis

Automated usability analysis tools provide the ability to quickly analyse an interface and provide quantitative and visual results, providing a good foundation on which other automated analysis can build. Automated usability tools, such as the Automated Website Usability Analysis (AWUSA) [205], WebRemUSINE [151], the Hand–held device User Interface Analysis (HUIA) [9] and the Metrics Comparison tool [120] all make use of usability performance metrics captured to automate the usability analysis process to different degrees. There are over 40 different available usability metrics, relating to interface design, navigation paths, statistics, quality, user data and contextual data, as used by the Metrics Comparison Tool.

The WebRemUSINE and HUIA tools both make use of expected user behaviour (as defined by an expert) to compare to the actual user behaviour, captured during the usability study. The results are used to highlight if a user deviated from the expected behaviour, where the deviation occurred and the amount of deviation. The WebRemUSINE provides the functionality to add the behaviour of a user to the expected behaviour,

if the user did not follow the predefined steps, but still completed the task effectively and efficiently.

The Metrics Comparison Tool provides one score for a website; these results can be used to compare the usability score of a system relative to other usability scores. The HUIA tool outputs more than one metric and thresholds are set for the user interface and event data. Should a metric not be within the threshold, then possible usability issues are highlighted.

Applied in this study: This study uses metrics applied in the mentioned studies, but incorporates eye tracking data, which is not used in any of the mentioned automated analysis tools. The proposed approach also produces performance metrics to highlight usability issues. These metrics provide comparative results which indicate possible usability issues if above a specified threshold. The study also builds upon the idea of utilising the user behaviour data as the expected behaviour if the user completed a task efficiently and effectively.

2.4.2 Automated eye tracking analysis

Eye tracking provides great insight into the usability of systems and the thought process of the users while performing tasks on a user interface, see Section 2.2.8. Ehmke and Wilson [55] define eye tracking measures that can be used to identify usability problems and suggest the automation of the analysis process. Automation of eye tracking analysis in usability testing is a complex task, but studies have made progress towards finding methods to achieve automation. A tool recently created by Fabo and Durikovic [58] introduces the use of eye tracking in an automated usability analysis tool. The tool visualises the difference between the fixation data position and where the user interacted with the system when an error occurred on the user interface.

Consider a study by Iqbal and Bailey [96], designed to identify if a user is reading, manipulating objects, searching or doing equations from eye tracking data and areas of interest. The x– and y– coordinates of the gaze data were plotted over time on separate graphs to investigate the patterns that emerge from different tasks. The plotted patterns verify that different tasks can be identified solely by using eye tracking data. The work-

load can also be identified by the amount of eye movements that occurred in an area of interest. These principles can be applied to assist the automation of usability testing. Knowing which task a user was busy with, can feed into other automated eye tracking analysis. This could assist in eye tracking metrics interpretation and comparison, minimising the time an expert should spend on the analysis.

The Gaze–based Usability Inspector Tool (GUIT) [4] automatically generates visualisation on a user interface to provide an overview of the usability of a system, relative to the recorded eye tracking data. The system is provided with fixation data of all the participants and screen shots of the tested application and either areas of interest or a grid size. The system makes use of icons to summarize reading, scanning, viewing order and viewing rank of the eye tracking data of each region. The frequency of each eye movement in a region differs and different colours are assigned to the icons to represent these frequencies. Three icon colours are used for high, medium or low values, which are separated by defined thresholds. This tool also considers the scan path and navigation data, by visualising similar scan paths followed by different participants. The similar scan paths are identified by the string–edit comparison. The GUIT tool does extensive analysis on the eye tracking data and provides the information to the analyst in an intuitive manner, reducing the time that the expert analyst has to spend on interpreting the eye tracking data of all the participants and different interfaces.

A study by Holland et al. [86] also addresses the excessive amount of time expert analysts spend analysing eye tracking data, stating that one minute of eye tracking data can take up to an hour to analyse manually. This method uses the approach created by Komogortsev and Holland [117] to automatically detect excessive visual search. The eye tracking data segmentation is applied in five different ways. For each of the segments, seven different excessive visual search indices are calculated: fixation count, saccade length, pupil size, scan path length, scan path area, scan path area times the number of inflictions, and lastly, a combination of the scan path area, number of inflictions and interval duration. Thresholds (usually the average value) are set for each index and if the indices of a segment are above the set threshold, then possible usability issues are highlighted. By automatically identifying the different segments in task completion and applying automated excessive visual search identification, the study can recognise

specific data that the expert analyst should investigate further. This method is user interface independent, as it does not require areas of interest to be mapped out and can reduce the analysis time by up to 40%.

Applied in this study: All these methods attempt to minimise the time that an expert needs to spend on interpreting and analysing the eye tracking data. These methods provide the expert with the ability to filter through the data and only focus on the eye tracking data above a certain threshold that could potentially contain usability issues. Areas on the user interface or time segments (which are user interface independent) are identified and the expert can analyse these subsets further. The method proposed by this study applies similar concepts by automatically highlighting tasks or subtasks with usability issues with the use of newly introduced deviation indices. This approach attempts to accomplish this without the need to map out areas of interest.

2.4.3 Automatic generation of gaze similarity metrics

For an analyst it is possible to manually identify similarities in eye tracking data, such as scan paths or fixation patterns. The problem with this approach is that the process is time–consuming and the analyst can easily overlook similarities if the dataset is too big. Some methods have been developed to automatically generate quantitative and visual results to highlight similarities in eye tracking data. Wooding [227] proposes an approach for highlighting similarities in fixation data by producing a three dimensional visualisation, namely a fixation map. The visualisation uses the coordinates of the fixations as two dimensions and the number of fixations as the third dimension. The method produces a coverage metric that indicates the percentage of points on a visual stimuli that was viewed above a certain threshold (height). Another metric produced by the method is a similarity metric that is obtained by subtracting the height of two different fixation maps and calculating the average height of the newly generated fixation map. This method is also applied to identify areas of interest where the height of the fixation map is higher than a given threshold.

Another study investigating similarity of scan paths, by Jarodzka et al. [103], view saccades as geographical vectors and compares the eye tracking data accordingly. String–

based scan path comparison requires areas of interest to be mapped out and other point–based scan path comparison methods does not consider the scan path sequence. The method proposed by Jarodzka et al. [103] overcomes these limitations by using a saccade as a mathematical entity to produce five measures of similarity: shape, length, direction, position and fixation duration. The method can only compare two scan paths at a time and simplifies the scan path by means of amplitude–based and direction–based clustering, prior to comparison. A matrix is created that compares each of the saccades in the one scan path to every other saccade in the other scan path. This data is then used to find the shortest path between the start and end point of the scan paths and these scan paths are then compared to determine the similarity of the two scan paths.

Applied in this study: The proposed approach does not consider similarity as much as it quantifies dissimilarity in eye tracking data. There are, however, some steps that are alike in these methods. Similar to the Wooding study [227], the fixation deviation also just considers the fixation position and not the fixation duration or even the fixation sequence. For the saccade or scan path data, the proposed method, like the Jarodzka et al. study [103], utilises saccades as vectors to compare the data in terms of amplitude, direction and position and keep the saccade sequence in consideration. The proposed method also clusters saccades, but not prior to comparison, but after the comparison to visualise the data. Similar to [103], the proposed approach only compares two scan paths at a time, of the participant and benchmark user. The proposed approach builds on this research, but adjusts the view by determining how much the data of the participants differ from that of the benchmark user.

2.4.4 Automatic identification of areas of interest

Many automated eye tracking analysis tools and methods require areas of interest to be mapped out manually, before the automated analysis can commence. The WebEyeMapper tool [170] provides some functionality in this regard, by automatically mapping the point where a person was looking to an element on a website. The tool captures fixation data together with any event and context data on the website that can change the view of the dynamic user interface. The mapper aligns each fixation to the element on which

the user focused and saves the fixation duration, element, and the text that was read in a database. Even though this tool does not identify areas of interest, it does provide the functionality to extract statistical information, such as: most viewed elements, viewing order, the amount of time spent on the element and the text that was read. Some of the eye tracking tools, like Tobii Studio [208], provides the functionality to use clusters of fixations as areas of interest. The clustering algorithm, defined by Santella and DeCarlo [187], is used by Tobii Studio to cluster the fixation data by means of a mean shift algorithm. The number of clusters is not defined for this algorithm, because a predefined distance threshold influences the number of clusters that are formed. For each of the fixations, the fixation is moved to the average position of all the fixations within the distance threshold of that fixation. This is repeated until all the distances between all the clusters are larger than the distance threshold. The fixations will eventually converge to one point and each fixation that was used to create this centroid is then assigned to that cluster. Building upon the mean shift clustering algorithm, a scan path comparison tool (iComp) was developed by Heminghous and Duchowski [81]. The fixations of all the participants are automatically clustered and labelled to form areas of interest. The labels are used to then implement the string–editing scan path comparison as implemented by these studies [160, 222]. Even though this tool was not directly developed for usability studies, it could be applied successfully in this field.

Applied in this study: Methods can identify areas of interest in various ways and then use these areas to automate eye tracking analysis by extracting eye tracking data relative to the area of interest or components. The method proposed in this study automates eye tracking analysis without the need to map out areas of interest. Relevant areas are derived from the visual strategy of the selected benchmark user. The proposed method also clusters eye tracking data with the use of a set threshold instead of a number of clusters that should be defined.

2.5 Eye tracking visualisations

Eye tracking is very rich, spatial and temporal data. By representing the eye tracking data visually, rather than just as the raw numerical data, assists analysts to easily extract information from the data. A number of commonly used eye tracking visualisations were discussed in Section 2.2.7. This section investigates different aspects of eye tracking visualisation as well as alternative visualisations of eye tracking data, developed for a specific field or application.

2.5.1 Adoption of standard eye tracking visualisations

Heat maps have been adopted in various fields to represent eye tracking data and is an effective way to summarise eye tracking data. A study by Bojko [17] discusses the importance of having all the relevant information available when displaying heat maps, otherwise the data could be misinterpreted. The study highlights the effect that parameters (like the minimum fixation duration or count, time segment, colour threshold and the usage of raw or fixation data) can have on the visualisation. Visualisations can be adjusted for specific needs, such as the inverted heat maps [216] to only highlight significant areas or the rendering of painterly abstracts from pictures by adopting the heat map concept [186]. The basic visualisation techniques can be re–used in various ways, depending on the needs of the study.

2.5.2 Eye tracking visualisations for specific research

Most of the commercial eye tracking technologies provide standard eye tracking visualisations, but some studies might need custom visualisations to investigate a specific hypothesis. TAUPE is an open source eye tracking visualisation tool that can be used by researchers to write extensions for their own needs. Most of the standard eye tracking visualisations are available in this tool as well as the functionality to draw a convex hull around a cluster of fixations. A convex hull is a polygon drawn with the fixation points as the corner points of the polygon. A study by Ramloll and Trepagnier [164] focuses on the process of creating custom eye tracking visualisations and the different design aspects that should be considered when designing visualisations. These aspects include,

but are not limited to: length, position, scale, colour, orientation, transparency, data granularity, labelling and data interaction. With these guidelines, the study continues to design a visualisation tool to show fixation clusters and the general direction of the scan path by means of arrows. The availability of visualisation tools to create custom visualisations, like the one proposed by Ramloll and Trepagnier [164], are still very limited and specific visualisations have to be generated by the researchers themselves.

2.5.3 Areas of interest and grids for eye tracking visualisations

A number of studies [50, 70, 109, 184] map out areas of interest on a user interface and associate each area with a string character. The character representing the area where the user fixated on is appended to the fixation sequence string. String comparison is then performed on the fixation sequence strings of different users to determine the similarity between the scan paths. The resulting datasets are often large and all these studies can benefit from a visualisation tool, like eyePatterns [222], to address this problem. The eyePatterns tool generates a hierarchical graph and clusters similar paths together. The more similar the sequences are the closer they will be to one another in a two dimensional space. The problem with single character labelling is that the pattern followed on the screen is not always intuitive in the visualisation. To overcome this problem, the eSeeTrack applications [212] allow analysts to assign a descriptive keyword to each area of interest. The eSeeTrack tool generates a WordTree [221] of the patterns followed between areas of interest on the screen with the size of the font representative of the fixation frequency in the area. Another tool that requires a set of areas of interest as well as a cognitive process model, is the EyeTracer tool, developed by Salvucci [183]. This tool primarily predicts fixation areas with respect to a specified cognitive process model. The tool visualises the implications of different fixation algorithms and prediction models to simplify the analysis. One of the visualisations provided by this tool is the ability to highlight actual fixations that occurred outside of the predicted fixation areas.

To avoid mapping out areas of interest, grid regions are generated over the visual stimuli, such as user interfaces. As an example, over 360 websites viewed by 20 users was divided into 9 regions to determine which regions of websites are prominent to users

[20]. In each of the grid regions a singular metric was visualised by means of a value and a circle representing the eye movement metric. Another study created the GUIT tool [4], allowing either the use of grids or areas of interest. The eye movement data is then represented in terms of icons in each of these regions to provide the analyst with insight into the usability of the user interface. Granularity of grid regions are often a concern, as the components on a user interface can stretch over multiple grid regions or one grid region can contain multiple components. This limits the use of grid regions if the grid size is not relevant to the average component size of the user interface.

2.5.4 Temporal eye tracking visualisation

Eye tracking data is temporal and most visualisations summarise the eye tracking data over time, in order to visualise data in a two dimensional (2D) space. In studies where temporal data is more important, then time could replace the x– or y–coordinates of the data in a 2D graph, as done by the Räihä et al. [162]. To avoid omission of one of the eye tracking data dimensions, a three dimensional (3D) space should be used. Li et al. [130] made use of a Space–Time–Cube (STC) to represent eye movement data in a 3D space. The x– and y–coordinates are represented on two of the axes, and time is added on the z–axis. Time and expertise are required by both these methods, to map out linear areas of interest or to understand how to interpret the visualisation.

Applied in this study: The visualisations produced by the proposed approach in this study build upon the visualisation techniques and tools, discussed in this section. The mentioned design factors were considered to ensure that the data is represented correctly in terms of the position, size, orientation and scale. The visualisation is also semi–transparent to make sure that the user interface components can still be seen. When reporting on visualisations, any parameters that could influence the output should be declared. The benchmark deviation vectors produced by the proposed approach utilise triangles as arrows to indicate the direction of the saccade clusters. Fixation points in each cluster represent points in polygons to draw benchmark deviation areas, but the convex hull algorithm was not used, as some of the fixation points were not highlighted by this technique. As the proposed approach highlights where participants deviated and

not when the deviation occurred, the visual outputs do not reflect the temporal data. This proposed approach highlights deviation from the expected behaviour by means of a benchmark user, which is assumed include relevant areas where a user should fixate in order to efficiently and effectively complete a task, not areas of interest.

2.6 Expert–novice eye movements

As a person gains experience in a field, they perceive visual data differently and are able to deduce information more effectively [29]. Studies have investigated how the visual strategies of experts, intermediates and novices differ. These studies have been applied in fields such as radiology, classification of fish, arts and sports. A study by Gegenfurtner et al. [65] considers the three theories on why the eye movement of various levels of expertise differ. The study shows that an expert will have shorter fixations, longer saccades, focus more on relevant areas and find the relevant areas in less time.

Initial studies investigated how chess masters perceived chess boards. The studies showed that an expert is able to reproduce the positions of chess pieces more accurately than less skilled players, but only if the chess pieces are not in random places on the board [40, 30]. A study by Reingold et al. [172] extended this research and demonstrated how experts are better at extracting information with less fixations on the individual pieces and rather between the pieces – indicating a larger visual span of the experts. In another study, eye movement metrics of expert and intermediate chess players were recorded while they had to find the best move on a pre–set chess board. This results in more meaningful eye movement, from the longer saccades, and less fixations by the master players. This study also emphasises the enhanced encoding and peripheral processing of the expert players from fixation data [29].

Experts are more proficient in encoding visual stimuli in their specialisation fields. By providing a novice with information on the visual strategies of an expert, the novice can adopt these strategies early on and become more effective and efficient in completing tasks. A study investigating the difference between expert and novice pilots while conducting simulated landing, found significant differences in the number of fixations, where they fixate and less time spent in the areas where they fixated [112]. The study suggests

that the visual scanning strategies of expert pilots should become part of the flight training methods, to improve student pilot development. In the medical field, eye tracking data was recorded during the use of laparoscopic instruments, by surgeons and students during training. The differences in the visual strategies were recorded to investigate how to apply eye tracking to track the training of a student or to determine the skill level of a surgeon [126]. A study by Sadasivan et al. [181] visualised the eye movements of an expert aircraft inspector as training material for novices. The eye tracking data of the expert was grouped into areas of interest, the sequence followed between these areas and how much time was spent on each area. This extracted data was superimposed onto the visual stimuli and used to train a group of novices. In comparison with a control group, the group trained with the expert eye tracking visualisations showed significant improvement, supporting the feasibility of training novices to use the same visual strategies as experts.

Applied in this study: These studies indicate that it would be valuable to know how much the eye tracking data of a novice differs from the visual strategy of an expert. This can be used to track the progress of the novice during training or determine the skill level of the novice during evaluation. By visualising the areas on the visual stimuli where the novice deviated most from the expert, could also highlight inefficiencies in the visual strategies of the novice.

2.7 Conclusion

This chapter investigated the fundamentals of eye tracking and usability methods as these are the two core elements of the current study. The last part of the chapter investigated research more closely related to the method proposed by this study. The existing automated usability testing methods provide much needed functionality, but do not allow additional data input, such as eye tracking data. Then existing automated eye tracking evaluation methods and tools were discussed. The proposed approach extends these tools by introducing the use of a benchmark user instead of highlighting areas of interest, providing new indices that can be used for the statistical analysis. Different

visualisations of eye tracking data were then discussed. The visualisations produced by the proposed approach is based on similar principles of past research. Lastly, this investigation considered eye tracking studies that make use of experts in order to train, classify or evaluate novices. The proposed method draws from past research to provide a way to extend the automation of eye tracking in usability, but also investigates the possibility to apply the approach in other fields, such as expert–novice eye tracking.

Chapter 3

Expert–based Usability Studies

Knowing where a user is looking on a user interface can provide great insight into the usability of a system. For this reason, eye tracking has become a widely used tool in usability testing. This chapter describes two usability studies: a Pilot study which was used to develop the proposed techniques and a Validation study used to test the proposed techniques.

3.1 Introduction

Usability studies are adapting all the time to align themselves with the latest technologies. There is a wide spectrum of devices, other than personal computers, that require user friendly interfaces, such as television sets, tablets, mobile devices and even cars to name but a few.

In order to verify the feasibility of the proposed approach for automating eye tracking analysis, two different mobile on–line procurement applications were evaluated. The initial study, referred to as the Pilot study, consisted of only 5 participants. This provided the needed data to develop the proposed approach for automating eye tracking analysis. The Pilot study required participants to make use of a mobile procurement application, namely Rustica [80], to order a number of products from a wholesaler. The second study, the Validation study, involved more participants and the data from this study was used to validate the proposed approach when applied to a larger dataset of 33 participants.

The Validation study evaluated an application called BiYP (Business in Your Pocket) [27]. Participants were also tasked to order a number of products from a wholesaler.

Initial background on usability testing is provided in Section 3.2. Both usability studies made use of the Tobii eye trackers [206] to capture eye movements, as is discussed in more detail in Section 3.3. The Pilot and Validation studies are discussed in Sections 3.5 and 3.6, respectively. For each of these usability studies, the application tested is discussed, followed by a description of the participants, the tasks they had to complete, and the findings of an expert usability analysis.

3.2 Usability testing considerations

This study consists of two usability tests, utilising eye trackers to identify usability issues. A number of things should be taken into consideration in order to conduct a usability test and be able to capture the necessary data, such as: when in the development process to perform a usability test, who should be tested and where to carry out the test.

3.2.1 What?

Before commencing a usability test, the goal of the test should be outlined. There are many external factors that have an impact on the usability test, such as the available resources, equipment and time [144]. The goal of the usability test should be defined within these constraints. The usability goals are not always limited to the definition criteria (efficiency, effectiveness and satisfaction), but should sometimes align to additional criteria, such as business goals, safety and severity of results [136, 178].

3.2.2 When?

Usability tests can be applied iteratively throughout the product development life cycle, to one or multiple phases [51]. The first stage is to explore, before the design process begins, when existing or similar software can be tested to identify if there is any room for improvement and how users interact with these systems. The initial system design, even if they are paper based, can be assessed and improved upon if necessary. During

the development from low to high fidelity prototypes the interfaces can be evaluated iteratively. Lastly, the final design could be tested to verify that not only the complete solution is working, but also to identify errors that occur that could affect the usability of the system [145].

3.2.3 Who?

A user profile should be available with demographic or experience information describing the typical user of the system. If it is too difficult or expensive to perform the usability test on real users, then the selected participants should fit the profile of the real users as closely as possible. The system should be designed for the users and not the customer, but these two are not always mutually exclusive [51]. There are various sources of participants, like students, associates and volunteers who could be selected for the study, but screening these participants to ensure that they have the required profile and skill set is crucial to the study [178]. The number of participants and analysts required for the study should be specified. The number of analysts depends on the setup of the study. Nielsen has demonstrated that five participants are sufficient for usability studies [143], while other studies argue that more participants are needed [59, 95].

3.2.4 Where?

Different environments are available to perform usability tests: laboratories, in the field and remote evaluations. In the field the participants are at ease and familiar with their surroundings. This allows the users to perform the task naturally and in context of the environment. The downside of the field tests are that not all the equipment needed for recording the user interactions are available. Thus, a laboratory test makes it easier for the evaluators to capture all the needed usability information with the available equipment and provides a controlled environment without distractions during the test. Lastly, usability tests can be performed at remote locations, utilising technology to communicate with the participant and capture the data automatically. This allow users all over the world to participate in a usability study without the need for a usability lab. Unfortunately, the tests are limited to the required usability data to be captured and

access the participants have to the system [52, 67, 178].

3.2.5 Tasks

For some usability tests a scenario is sketched for the participants, followed by a number of tasks they have to complete. The participants should complete the tasks on the system that the real users will make use of. It is not always necessary to test all the tasks – some complex tasks, time consuming tasks or tasks that are outside the scope of the usability study can be left out of the study [51]. The following tasks should be tested: most frequently used, critical to other operations, newly added and basic tasks of the system. In some cases the tasks can be complex or consist of a number of different user interfaces to be evaluated; in order to simplify the analysis, the tasks could be divided into subtasks. Subtasks are a number of logical steps that are followed to complete a bigger overall task [51, 114, 205].

3.2.6 Data

The data captured during a usability study depends on the needs of the study. Questionnaires can be completed by the participants to capture personal information and knowledge prior to the usability test. During the usability test, qualitative and/or quantitative data can be captured either automatically and/or manually. After the study the participants can also answer questions on their experience while using the system or make any recommendations they have. The usability facilitator should decide which data is needed and ensure that all the tools are in place to capture the required information [178].

3.2.7 Report

As stated by Rubin and Chisnell [178], there are usually two different phases of reporting after a usability test. An initial report will be given shortly after the test that will highlight the most prominent issues that should be focused on. Thereafter, additional time will be spent to provide a more extensive report with additional usability issues and statistical data extracted from the quantitative data [51, 190]. The usability report

should be in an appropriate format for the readers to understand and extract the required information.

3.3 Eye tracking apparatus

There are numerous companies manufacturing eye trackers [88, 173], varying in price and quality. Research in developing more cost effective eye trackers is improving the availability of eye trackers. An appropriate eye tracker is selected for a study depending on the available funds, the accuracy of the data required, and access to equipment. The Usability Lab at UNISA, the University of South Africa, provided the necessary facilities and equipment, which included Tobii eye trackers, to conduct the usability studies for this work.

Tobii Technology [206] was established in 2001 and sold their first eye tracker in 2002. Since then, the range of eye trackers has evolved and expanded. Tobii hardware for eye tracking research include, but are not limited to, eye tracking glasses, mobile device eye trackers, and eye trackers for personal computers with and without screens.

In 2011, when the Pilot study was done, the UNISA usability lab had the Tobii 1750 eye tracker, which was used for the usability test. Two years later, in 2013 the usability lab acquired a newer model, the T120, also a Tobii eye tracker. This model was used to capture the eye tracking data in the Validation study. Some technical specifications of these two eye trackers are listed in Table 3.1 and the screens are shown in Figure 3.1.

Specification	**Tobii 1750**	**T120**
Frame Rate	50Hz	120Hz
Screen Size	17” TFT LCD	17” TFT LCD
Screen Resolution	1024px × 768px	1280px × 1024px
Binocular Tracking	Yes	Yes
Freedom of Head Movement	30 × 16 × 20 cm	30 × 22 × 30 cm
Software	Tobii Studio 2.3	Tobii Studio 3.2

Table 3.1: Technical specification of the two Tobii eye trackers used in this study.

Both eye trackers are built into a 17 inch LCD screen and track the movement of both eyes, making use of one or more infra–red cameras. The T120 (Figure 3.1(b)) captures

data at a much higher rate, 120 gaze points per second, compared to the Tobii 1750 (Figure 3.1(a)), which only captures 50 gaze points per second.

(a) Tobii 1750 (b) Tobii T120

Figure 3.1: Photos of the two Tobii eye trackers used in these usability studies.

3.4 Experimental setup

A number of aspects concerning the apparatus and environment should be set up before the usability study can commence.

The Tobii Studio software was used to set up the flow of the experiments. The instructions were displayed to the participants, followed by the task they were required to complete; this was repeated until the usability test was completed. The data captured by the Tobii Studio was exported as fixations, using the Velocity–Threshold Identification (I–VT) fixation classification algorithm, provided by the Tobii Studio application. Parameters, such as a fixation radius of 35 pixels and minimum fixation duration of 100 milliseconds were specified for exporting fixation data. With these parameters a fixation was recorded if the eye movement threshold was greater than 0.35 pixels/ms or if the continuous fixation duration was greater than 100 milliseconds.

Even though mobile applications were used, the tasks were completed on a computer screen, accessing the applications through mobile emulators with a set size as indicated in pixels (px). The Rustica application is a web–based system and was accessed through

the Opera Mobile 10 emulator, set to a 500px × 310px size. For BiYP, a native Windows Phone 7 application, the Windows Phone 7 emulator, was used with a screen size of 510px × 430px.

An additional factor taken into consideration during the set up of the study was to ensure that the participants sat on a chair with no wheels, to minimize movement out of the head boxes. The freedom of head movement is specified in Table 3.1. Lastly, the reflection of light on the reading glasses of some participants were also noted and minimized to get optimal results. With the apparatus in place and the testing environment set up, the usability tests were carried out.

3.5 Pilot study

This section describes the expert based usability test of the Rustica application, giving some context about where the application originated from and a short description of the application used. Details about the participants and the tasks they were required to complete for the usability study are given. The section ends with the findings collected from the expert analysis of the eye tracking data.

3.5.1 Context

For small convenience shop (Spaza shop) owners in the rural areas of South Africa, the Rustica application provided a technology to overcome a daily problem. Spaza shops are situated in remote areas where there are poor road conditions, limited cash flow and no access to global supply chains. Owners often run the shops by themselves and when they have to replenish stock, the shop has to be closed. The owner would have to travel all the way to a city, by minibus taxi and buy the needed stock with limited cash.

A project, involving SAP (Africa) and the Council for Scientific and Industrial Research in South Africa (CSIR), was launched in 2006 in the Kgautswane area to support Spaza shop owners. The pilot project involved two socio–preneurs (intermediaries) and one supplier, Sasko Bakeries (Pioneer Food). It allowed Spaza owners to order bread by sending an SMS to a socio–preneur who then bundled and submitted the orders to the supplier and delivered the products. The project provided a great service and grew into

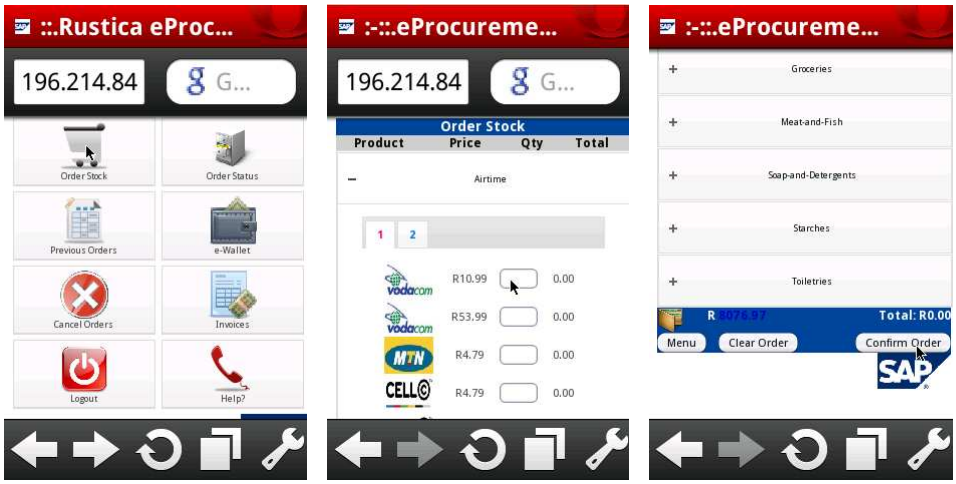
Project Rustica, developed by SAP Research, to provide additional functionality and a wider range of products and suppliers. Rustica evolved from an SMS ordering system to an ICT solution providing a web–based procurement system designed for multiple mobile devices and platforms.

The original usability study, that recorded the data used in the Pilot study, was conducted to investigate which data can be captured with different usability testing methods and which usability issues can be identified with each method. The study compared usability tests, firstly with the use of an eye tracker with the application running in an emulator and secondly with the application running on a mobile device and making use of a video to record interactions and lastly with the users using the application in the field with an expert observing. This study is described in [67] and will be referred to as the expert based pilot usability study. The following section provides more detail about the relevant data of this usability study.

3.5.2 Rustica application

Rustica [66, 80] mainly allows users to order stock, view the current status of an order, view previous orders and manage their eWallet (i.e. their account balance). Figure 3.2 shows the user interfaces for the Rustica application. Figure 3.2(a) is the main menu of the application, allowing navigation to the main functions provided by the application. When 'Order Stock' is selected, the screen from Figure 3.2(b) is loaded where the products can be selected from different sub–categories. If the owner wants to order a product, the desired quantity should be entered into the text box, next to the product. Once all the products are added, the order can be confirmed as shown in Figure 3.2(c), located at the bottom of the product page.

Ordering products is the main function of Rustica, thus the usability study involved only this component. These were the three screens the participants interacted with and for which the eye tracking data was captured.



(a) Menu (b) Products (c) Confirm order

Figure 3.2: User interfaces for the Rustica mobile procurement application.

3.5.3 Participants

The Rustica project was rolled out in the Kgautswane area of South Africa, which has a population of 120 000 people and consists of 19 villages. Spaza shop owners typically conform to the demographics of an African, female adult with a schooling level between completing primary school and graduating from secondary school. They are proficient in English, although it is not their first language. The user group is also mostly computer illiterate but makes use of mobile devices on a daily basis. The participants who partook in the Pilot study fitted the same demographic profile as that of the Rustica users in the rural areas. Actual Spaza shop owners were not used for this study, as they were too far away from the usability lab and it would be too costly for both parties to get the Spaza shop owners to travel to UNISA. The expert based study included 10 participants in total, but the eye tracking of only 5 participants were recorded. This sample size was sufficient to develop the proposed approach for the Pilot study. With a small sample size the data and results could easily be checked to see if there were any errors in the process logic.

Table 3.2 gives more detail about the participants. The adults were mostly female and all African adults, with English as a second language and another African language

as their first language. The highest level of schooling was secondary school and only one of the participants had used a computer prior to the study. All of the participants made frequent use of their mobile devices throughout the day. Some basic computer training was given to the participants before the usability testing started. This was to ensure that the participants could make use of a computer mouse, seeing that they were mostly computer illiterate. The tasks were also explained to the participants before the usability test commenced.

User	Gender	1st Language	Schooling	Computer Usage	Mobile Usage
1	F	Tswana	Grade 12	Never	2–5 × per day
2	F	Tsonga	Grade 12	3–10 times in total	>5 × per day
3	F	Ndebele	Grade 11	Never	>5 × per day
4	F	Tswana	Grade 8	Never	2–5 × per day
5	M	Tsonga	Grade 12	Never	2–5 × per day

Table 3.2: Demographic information of the participants for the Pilot study.

3.5.4 Tasks

The key feature of Rustica is product procurement, thus the participants were tasked to complete three small tasks: navigate to the order page, order a number of products and confirm the order. The additional functionality in Rustica was not tested as part of the usability study. After providing the necessary demographic information, the participants were given a scenario and a number of tasks they had to complete. A scenario was sketched for each participant and the tasks were explained, but the participants did not see the application or screens before the usability test started.

Scenario: The following scenario was explained to each participant:

"You are the owner of a Spaza shop in a small, remote town in South Africa. You have access to the Rustica application on your smart phone allowing you to order stock from a wholesaler. You are already registered on the Rustica application, thus you don't have to provide any additional information concerning delivery address or payment. Make use

of Rustica to order the needed stock for your shop and have it delivered conveniently to your shop."

Task 1: Figure 3.2(a) was the first screen the participants saw. They were tasked to select the button which allowed them to order stock from a wholesaler. To complete the task, the 'Order Stock' button in the first column had to be selected.

Task 2: Once the participants clicked the 'Order Stock' button, the web–based application navigated to the products page. To complete this task, the participants had to order:

- 3 × R10.99 Vodacom Recharge Vouchers
- 2 × R4.79 MTN Recharge Vouchers

Both these products were on the initial screen when the page was loaded, so no additional navigation was needed. The required quantity had to be entered into the text box next to the product, as seen in Figure 3.2(b).

Task 3: After the required products were ordered, the participants had to send the order to the wholesaler by confirming the order. For this task, the users had to scroll to the bottom of the orders page and select the 'Confirm Order' button, as shown in Figure 3.2(c).

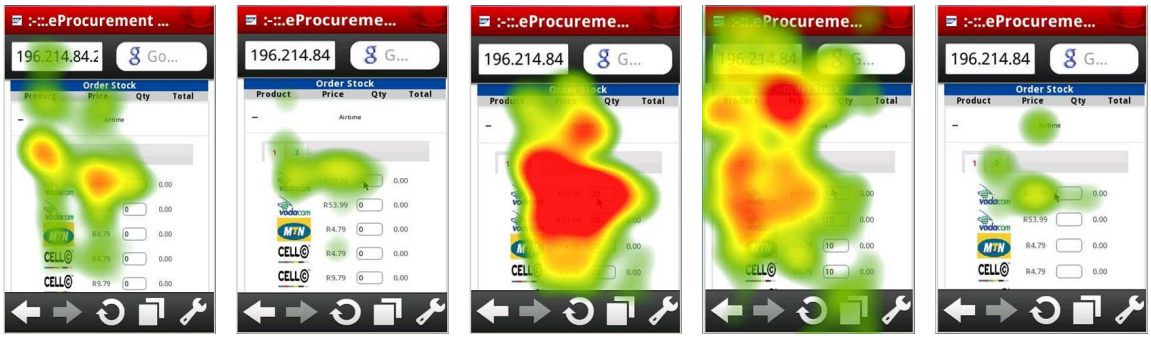
The five participants completed all the tasks, using the available equipment while the data was captured by the Tobii eye tracker. A usability analysis was then completed by an expert analyst.

3.5.5 Findings

An expert–based study involves a usability expert who scrutinises the results and discovers usability issues in the system. In the expert based usability study, involving the Rustica application, three methods of mobile application usability evaluation were compared. One of the methods involved the analysis of Rustica with the use of an eye tracker

in a lab. The results were then analysed by an expert analyst. Some of the results were published in a paper [67]. The usability issues found in this paper along with additional usability issues discovered will be covered in this section.

The expert based pilot study paper discussed Task 2 of the Rustica application, as this covered the main functionality of the application. Figure 3.3 represents the eye tracking data as heat maps for the participants while completing Task 2. Figure 3.3(a) – 3.3(c) are from the expert based study paper and Figure 3.3(d) – 3.3(e) were exported independently from the eye tracking data captured during the Pilot study.



(a) Participant 1 (b) Participant 2 (c) Participant 3 (d) Participant 4 (e) Participant 5

Figure 3.3: Heat maps for participants completing Task 2 of the Pilot study.

The heat maps in Figure 3.3 illustrate all the fixation points and fixation duration on a specific area, from the moment the screen was loaded until the user clicked on the correct text box. The eye tracking data replays for each participant were investigated at least once to identify additional usability issues from the scan paths. This process was time–consuming, since the total duration of footage was at least 90 minutes. The following were observed from an expert analysis:

Product images: Images of the products were the first things that drew the attention of the participants, as seen from the replays. They looked at the images and could identify each product, but in some cases all the images were fixated upon before moving back up to the price of each product. The images were appealing to the participants and got a lot of attention.

Product price: In Task 2 the price was used to determine which of the two Vodacom recharge vouchers to order. Even though the participants only needed to pay attention to the Vodacom prices, most of them searched through all the images, then all the prices and then they backtracked up the prices back to the correct item. This was also observed from the replays.

Text boxes: The text box was expected to be the significant focus of attention for the participants, because that was where the product quantity should have been specified to complete the task. The participants did not, however, pay much attention to the text boxes. A reason for this behaviour could be that the participants were not computer literate and were unfamiliar with a text box.

Product tabs: The tabs at the top of each category delimited the products for faster loading. Some of the participants focussed on the tabs after identifying the product, as seen from the replays. Seeing that the participants were not familiar with text boxes, they were considering the tabs to enter the quantity of 3 products as specified by Task 2.

Table headings: As soon as the participants realised that the tabs were not the components they had to interact with, they moved on to the table headings and read the description of each column, one of which was 'Qty' to indicate the segment where the quantity could be entered.

Emulator menu: After fixating on most of the elements on the screen, and still not sure where to enter the quantity, the participants moved to the menu items of the emulator. The emulator should not even have been considered, as it was not part of the Rustica interface and would differ, depending on the device.

Even though usability issues could be discovered from eye tracking data only, by means of the current practices, the process was still time–consuming, especially analysing scan path data by means of the eye tracking data replays. The existing visualisations could get cluttered and information could easily get lost. Thus, it was not always feasible

to analyse large numbers of participants using expert–based approaches. The following section discusses a larger usability study used to validate the proposed approach for automated eye tracking analysis described in Chapter 4.

3.6 Validation study

The following section describes a usability study for the BiYP (Business in Your Pocket) application [27, 57], conducted by means of an eye tracker. Context about the application is given describing the features, user group and purpose of the usability study. The participants selected for the usability test are described as well as the tasks they were asked to complete. Lastly, usability issues identified in the BiYP application by an expert analyst are provided.

3.6.1 Context

The BiYP application provided many different business services that a small business could use. One of the services includes on–line procurement capabilities, similar to the services provided by Rustica. The usability issues discovered by the Pilot study were addressed during the design of BiYP to provide more user–friendly interfaces.

The BiYP application is focussed on slightly bigger businesses than Spaza shops, located closer to the city. In collaboration with Metro Hyper, a nationwide food and goods wholesaler, the application was launched in a pilot study to provide smaller shops with the ability to order products with the use of a mobile device. The largest number of products sold by Metro Hyper is from sales to small shops and not walk–in sales. The current system used for sales to small shops is, however, very inefficient. Shops will phone the wholesaler and order the wanted products, the manager will write them down and give the list to a packer to collect from the Metro Hyper floor. The small shop owner will then pick up the products and pay the required amount. BiYP addressed this problem by providing the small shops with the capability to place orders using a mobile device. Metro Hyper receives the orders by means of an interactive web–interface and can send updates to the client regarding the orders. This automates the process, making it more efficient.

The main purpose of this usability study was to capture the eye tracking data of a larger group of participants while using BiYP. An expert analysis was conducted using the eye tracking data to identify usability issues. The Validation study was performed to validate that the proposed method in this dissertation can be applied to a large dataset to show the full potential of the method.

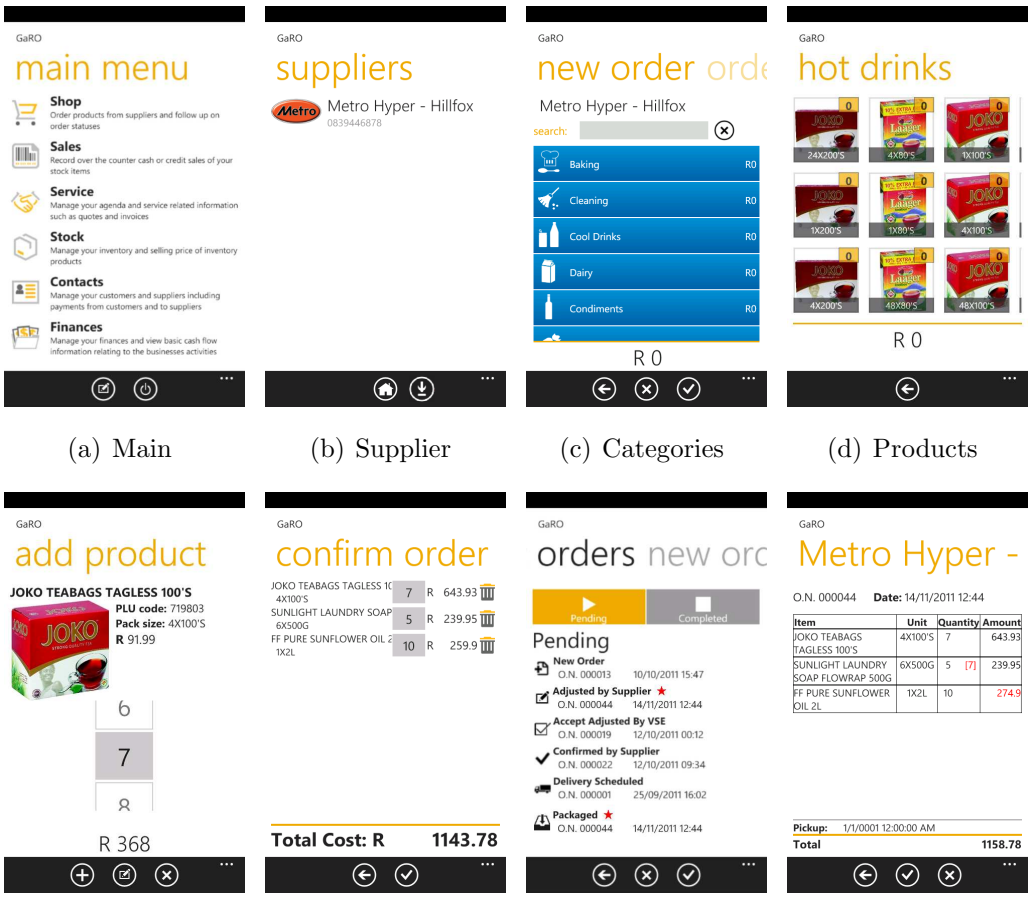
3.6.2 Application

The main menu of the BiYP mobile application, Figure 3.4(a), contains a list of the services provided by the application. When 'Shop' is selected, the available wholesalers appear and the user can select the desired wholesaler to view their catalogue, as shown in Figure 3.4(b) and 3.4(c). In Figure 3.4(c), some improvements from the Pilot study include the images next to the category name to assist users in finding the right category with ease. Once a category is selected a page full of products, Figure 3.4(d), is shown as images as well as the quantity ordered – another usability issue improved.

To order a product, the user will select the required product, navigating to the product details and select a quantity. Figure 3.4(e) shows the quantity selection of a product. This replaces the hindering text boxes used in Rustica. Once the user is satisfied, the order can be confirmed and sent to the wholesaler, Figure 3.4(f). There are a larger number of screens in BiYP than in Rustica, presenting only the essential functionality on a page and removing confusion and unnecessary information. The last two screens, Figure 3.4(g) and 3.4(h), provide real–time updated information about the orders for the user from the wholesaler. The functionality mentioned here was tested during the Validation usability study and eye tracking data was captured on all of the above–mentioned screens.

3.6.3 Participants

BiYP was designed to cater for a wider range of users than Rustica. VSE owners spanned a wide demographic, including people from different age and ethnic groups and levels of schooling. The typical VSE owners who will make use of BiYP, are likely to use technology in other facets of their business and personal life. For this study 33 participants



(a) Main (b) Supplier (c) Categories (d) Products

(e) Product (f) Confirm (g) Orders (h) Invoice

Figure 3.4: User interfaces for the BiYP mobile procurement application.

were selected. These participants were recruited from the SAP Research Pretoria offices as well as external volunteers. The minimum requirements were that the participants should be able to read and write, but could be any gender and off any age and from any ethnic group.

Each of the participants was asked to complete the questionnaire, shown in Appendix B.1, to capture the demographic information they were willing to disclose. Table 3.3 contains all the participants who joined in the usability study and the available information. The unique identification number was used to keep the data anonymous. The gender, age and highest level of schooling were the relevant demographic information that was

captured. All of the users had used a personal computer and smart phone device at the time or in the past. The mobile usage captured the amount of time a user spent on their mobile device daily. The e–Shop column displays the number of on–line shops the participants made use of on a regular basis. This was to capture how familiar the participants were with on–line shopping. Lastly, the BiYP column indicated if the users had used or seen the BiYP application prior to the study or not. The 33 participants all completed the tasks as described in the following section.

3.6.4 Tasks

Even though the design of the procurement functionality of BiYP was based on the recommendations from the Pilot usability study, new usability issues could still be introduced by the new design. There could also be usability issues resulting from the varying user groups in the two studies; a user group with more exposure to technology would expect different components on the screen. A user group with more exposure to technology could have a different expectation of a user interface. For this usability study a scenario was sketched for the participants and all the tasks were explained to them. The participants were informed that their eye movements would be tracked by a non–invasive eye tracker while completing the usability evaluation.

Scenario: The following scenario was explained to each participant:

"You are the shop owner of a small grocery store in South Africa. For your convenience, the BiYP mobile application provides a procurement service, which allows you to order products from wholesalers such as Metro Hyper. This service also provides real time updates, from the supplier, regarding your orders. You are already registered and logged into the BiYP application. Replenish the needed stock for your shop by means of placing an order with Metro Hyper Hillfox, using the BiYP application."

Task 1: The participants were requested to order products from Metro Hyper Hillfox with the shop service in BiYP. As shown in Figure 3.4(a) – 3.4(d), from the landing screen the users had to navigate to the list of product categories for Metro Hyper Hillfox and select the category containing the following products:

User	Gender	Age Group	Schooling	Mobile Usage	e–Shop	BiYP?
1001	M	46–55	Honours	3× per day	1	No
1002	F	26–35	Honours	Hourly	1	Demo
1003	M	36–45	Master's	Bi–hourly	3	Demo
1004	F	26–35	Grade 10	Bi–hourly	2	Demo
1005	F	46–55	Matric	3× per day	9	Demo
1006	F	36–45	Matric	Bi–hourly	1	No
1007	F	15–25	Honours	Hourly	8	Demo
1008	M	26–35	Master's	3× per day	0	No
1009	M	15–25	Honours	Bi–hourly	7	Demo
1010	F	15–25	Honours	3× per day	5	Demo
1011	F	46–55	Bachelor's	Bi–hourly	3	No
1012	F	26–35	Master's	Bi–hourly	0	Used
1013	M	26–35	Honours	Hourly	4	Dev
1014	F	15–25	Honours	Hourly	1	Demo
1015	M	26–35	Master's	<1× per day	0	Used
1016	F	46–55	Master's	Hourly	5	Used
1017	F	26–35	Master's	Bi–hourly	2	Used
1018	F	36–45	Master's	Bi–hourly	1	Demo
1019	F	46–55	Honours	Bi–hourly	1	No
1020	F	15–25	Bachelor's	Hourly	1	No
1021	F	15–25	Master's	3× per day	1	Demo
1022	F	26–35	Honours	Hourly	1	No
1023	M	36–45	Bachelor's	Hourly	4	Demo
1024	F	26–35	Honours	Hourly	3	Used
1025	F	26–35	Master's	Bi–hourly	0	Demo
1026	F	26–35	Doctorate	Bi–hourly	1	Demo
1027	M	26–35	Master's	Hourly	4	Demo
1028	M	55–65	Doctorate	Hourly	3	Used
1029	M	36–45	Doctorate	Hourly	0	Dev
1030	F	15–25	Bachelor's	Hourly	2	No
1031	M	15–25	Matric	Bi–hourly	2	No
1032	M	15–25	Matric	Bi–hourly	2	No
1033	F	15–25	Honours	Bi–hourly	4	Dev

Table 3.3: Demographic and technology usage information about participants of the Validation study.

- 24 × 750g Ricoffy Instant Coffee, Regular
- 12 × 100's Glen Teabags, Tag–less Pouches

On the product page the item had to be added by selecting the required quantity, as can be seen in Figure 3.4(e). Once all the needed products were added to the list, the order had to be confirmed and was sent to the wholesaler, as shown in Figure 3.4(f).

Task 2: To test the learnability, the participants were asked to repeat the steps of Task 1 and order the following products:

- 5 × 2L Clover Fresh Milk, Full Cream
- 3 × 2L LiquiFruit Juice, Mango

Task 3: To view orders placed with Metro Hyper Hillfox, the participants had to navigate to the order page of the wholesaler and find the previous orders. From here the order had to be selected to view the invoice and/or any changes made to the order by the wholesaler. This is depicted in Figure 3.4(g) – 3.4(h).

The eye tracking data was captured for all 33 participants while completing the above mentioned tasks. An expert analyst then analysed the eye tracking data to identify usability issues in the system.

3.6.5 Findings

This section covers the usability analysis performed by an independent expert analyst, on the eye tracking data captured during the Validation study. An expert can make use of different eye tracking data outputs such as: heat maps, data replay, mapping out areas of interest and inspecting the data output. For this study the expert did not have any knowledge on how the system worked, thus it was not possible to map out areas of interest. The expert was also not involved in the usability test when the data was captured. For this reason the best way for the expert to investigate the data was to watch the replays of the recorded eye tracking data.

Participant	Time on Task 1	Time on Task 2	Time on Task 3
1014	66	40	24
1021	61	67	22
1024	126	60	64
1026	207	60	125
1027	138	55	64
1028	159	71	109
1029	104	67	65
1030	171	82	83
1031	140	79	71
1032	96	57	88

Table 3.4: Expert review of the time spent on a task, in seconds, for the Validation study.

The expert analyst selected 10 eye tracking recordings from the 33 available recordings. While watching the replays, the analyst noted observations from the usability of the application. The analyst watched the videos repeatedly in order to extract the usability issues, until the analyst could not identify any new observations from the eye tracking data. This process was time–consuming even though only the data of a third of all the participants of the study was analysed. The full report can be found in Appendix B.3.

The expert analyst made use of the time–on–task metric for the initial study. From this data the learnability was noted in the time difference between the similar Tasks 1 and 2. There was high variance in the task completion time in Task 3, where the participants spent much more time on the task than expected by the analyst. The following usability problems were identified by the expert analyst from watching the replays of the usability study:

Main menu: In order to view previous orders, the users were expected to select the 'Shop' item, which was counter–intuitive. The description of the menu item did not provide the necessary information to the users that the orders could be viewed by navigating to the 'Shop' service. The participants scanned up and down the main menu to find the correct item. One of the participants selected a help icon from the bottom menu, other participants just looked at the menu for assistance.

Orders header: The participants had to navigate to the page by selecting 'orders' at the top, right of the screen. All of the participants eventually navigated to this page either on purpose or by accident. The 'order' title was very light and the word was cut–off. This made it very difficult for the participants to determine how they should navigate to the order history.

Item quantity: The participants struggled to enter the quantity of the product they had to order. This was mostly because an emulator was used and not a touch screen. This issue related more to the setup of the usability test than the usability of the overall system. A touch screen should be used to determine if the input component was also confusing to the participants when not using a computer mouse.

3.7 Conclusion

Two expert–based usability studies were described in this chapter, both executed on a mobile procurement system, conducted in a usability lab with user groups that matched the demographics of the system's end users. Each step of the usability test was described in detail, followed by an expert analysis of the eye tracking data. The expert analyst identified usability issues concerning each of the applications, making use of conventional usability evaluation methods. This process was time–consuming, yet effective. The role of expert analysts was crucial to the usability evaluation process, and still played a vital role in the proposed method discussed in this study. The data captured from the usability tests, as discussed in this chapter, is used in the remainder of this study to develop and validate the proposed automated eye tracking analysis method.

The following chapter describes the proposed approach for shortening the time an expert analyst will have to spend investigating the usability of eye tracking data.

Chapter 4

Proposed Automated Usability Analysis

Time is money... and to minimize the time spent on a project saves money. Eye tracking analysis takes up a lot of valuable time of an expert analyst. In this chapter an automated usability analysis method is proposed to minimize the time spent on usability eye tracking analysis and the proposed approach is tested on the eye tracking data of the Pilot study. The automated results are compared to the usability issues identified by an expert analyst to determine the feasibility of using this proposed method in future eye tracking usability studies.

4.1 Introduction

There are numerous metrics and methods for analysing eye tracking data. Examining the eye tracking data of each participant individually can be time–consuming and extensive knowledge of the user interface is required, because the expert will have to watch the replay of each participant completing every task and know where they should look and what events should occur. To address this problem, this chapter proposes an automated method to extract comparable, user interface independent results. The approach combines concepts from usability testing and eye tracking and introduces the use of a benchmark user to automate the analysis.

Multiple fields of study make use of benchmarks to assess relative performance. Consider the scenario where a number of users interact with a system to achieve a given task. The basic premise of the proposed method is that if a participant is the most successful in achieving the given task, then that participant can be said to use the most efficient visual strategies required to complete the task, compared to the other participants. That user is then selected as the benchmark user. Subsequently, the proposed method suggests that the difference between the eye movements of the benchmark user and the other participants, while performing the same task, can be used as the basis for automatically quantifying and highlighting the usability issues of a system. The proposed method is a supplementary method for usability testing analysis with the use of eye tracking data, reducing the time spent by an expert analyst to reduce the time spent on analysing all the data of a usability study.

The aim of this chapter is to describe the concept of analysing eye tracking data automatically by means of a benchmark user. Figure 4.1 shows the overall process of the proposed approach. The datasets resulting from the *Pilot usability tests* that are relevant to this study are discussed in the *data pre–processing*, Section 4.2. The *benchmark user identification*, Section 4.3, is needed for both the automated eye tracking analysis processes, known as the fixation deviation index (FDI) and the saccade deviation index (SDI) processes, discussed in Section 4.4 and 4.5, respectively. The data processing methods for fixations and saccades are described and the processes were applied to the Pilot usability dataset. The results were compared to the usability issues discovered by an expert in the Rustica application, to determine the feasibility of extracting similar usability issues by applying the proposed automated method.

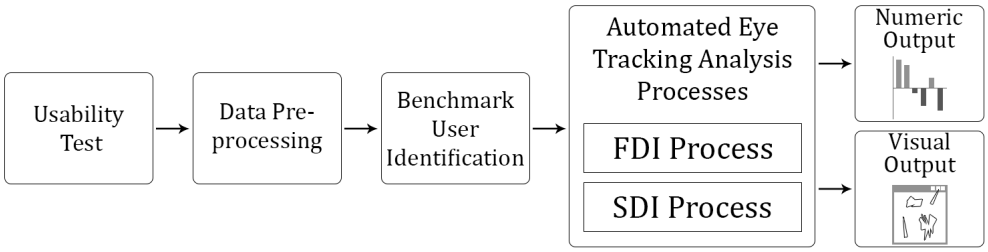


Figure 4.1: The process of the proposed approach, including data processing prior to the FDI and SDI processes and data output.

4.2 Data pre–processing

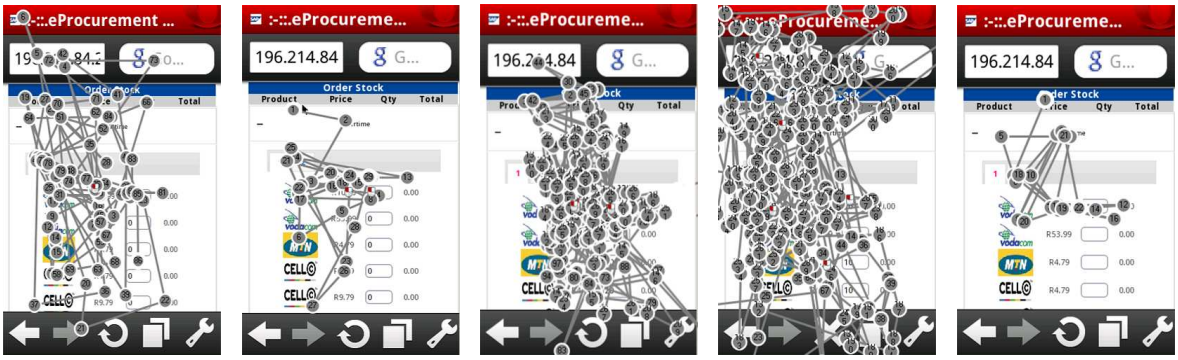
During the usability tests, the eye tracking data was recorded per participant for every task. The data should be in the correct format before it can be applied in the proposed approach: fixation and saccade data are required and should be in datasets for each participant and each task or subtask. This section describes the raw eye tracking data that was captured with reference to the data from the Pilot usability study. The data from the usability studies were used in the initial investigation discussed in this section, seeing that it was a relatively small study, involving only five participants. This provided a suitable dataset to test the feasibility of the proposed method. A smaller dataset also allowed for uncovering potential errors and checking the reliability and validity of the results. A greater number of participants are used later in Chapter 5 for validating the process and the effectiveness when applied to a larger dataset.

4.2.1 Exported fixation data

The eye movement (gaze) was captured by the Tobii eye tracking device together with the Tobii Studio software to save the eye gaze points. Tobii Studio has many different features for analysing the captured eye tracking data, such as replaying, visualising and exporting data. The raw eye gaze data was recorded at a specific frequency and could then be saved as is or exported as fixations, as discussed in Section 3.3.

Fixations and saccades were used as the basis for analysis in the proposed method. There is one saccade between every two fixations; thus, in total there will always be one saccade less than the total number of fixations. The images in Figure 4.2(a) – 4.2(e) represent the fixation and saccade data for participants 1–5, while completing task. The circles on the figure represent the fixations and the lines represent the saccades. The numbers inside the fixation circles indicate the order in which the fixations of the participants took place on the user interface. These visualisations were exported by means of Tobii Studio.

Figure 4.2 gives a clear view of where the participant focussed. Figure 4.2(a), 4.2(b) and 4.2(e) were still usable representations of where the participants fixated and in which order. This visualisation is no longer sufficient if there are too many fixations, as the



(a) Participant 1 (b) Participant 2 (c) Participant 3 (d) Participant 4 (e) Participant 5

Figure 4.2: Fixations of each participant for completing Task 2 of the Pilot study.

screen data becomes cluttered, resulting in extended analysis time. Considering Figure 4.2(c) and 4.2(d), it was difficult to identify the order and exact elements the participant fixated on were not clear. These visualisations can easily become cluttered and possible loss of information can occur.

To be analysed further, the fixation data was exported from the Tobii Studio software as described in the following section.

Index	Timestamp	Duration	Point X	Point Y
1	8165	341	471	314
2	8182	225	489	499
3	8407	674	531	559
4	9081	383	513	293
5	9464	691	436	254
6	10156	191	404	447
7	10347	366	404	494
8	10714	308	419	390
9	11021	200	416	533
10	11221	624	406	388

Figure 4.3: A screen-shot of sample data of the fixation data exported by Tobii Studio.

4.2.2 Processed eye tracking data

The fixation data was exported as a text file with a time stamp as to when the fixation occurred, the duration as well as the x– and y–coordinates; see a data snippet in Figure

4.3. Tobii Studio has the ability to identify fixations from raw gaze data, using one of various algorithms available. The I-VT algorithm was used to export the fixation data for this study.

Tobii Studio does not export saccade information, so the saccades had to be derived from the fixations. Each saccade has a start and end point fixation and is logically equivalent to a Euclidean vector, with a position, magnitude and direction. From the fixation points the saccade can be derived, as the magnitude is calculated using Euclidean distance between the fixation points.

Algorithm 1 Derive saccades from fixation data

```
1: for all participants : p do
2:     for all fixations : f_{i-1} do          ▷ Derive saccades from fixations
3:         if f_{i-1} ≠ null then
4:             saccade_i ← vector(f_i, f_{i+1})
5:             add saccade_i to saccades
6:         end if
7:     end for
8: end for
```

As shown in Algorithm 1, all the saccades were exported as vectors for each participant. By means of two consecutive fixations (f_i and f_{i+1}), a saccade ($saccade_i$) was derived as a vector. Each $saccade_i$ was then added to the set of saccades for that participant. It should be noted that the fixations used are centroids of the raw gaze points captured by the eye tracker, using the Velocity-Threshold identification fixation classification algorithm, as discussed in Section 2.2.6.

4.2.3 Event data

Tobii Studio records a wide range of data over–and–above the fixation data. Some of the data captured includes the screen content, keystrokes, mouse clicks and web page navigation. These events can be used to extract usability behaviour from usage data. For the data to be comparable in using the proposed method, it was a requirement that the eye tracking data should be analysed per user interface screen (comparing apples with

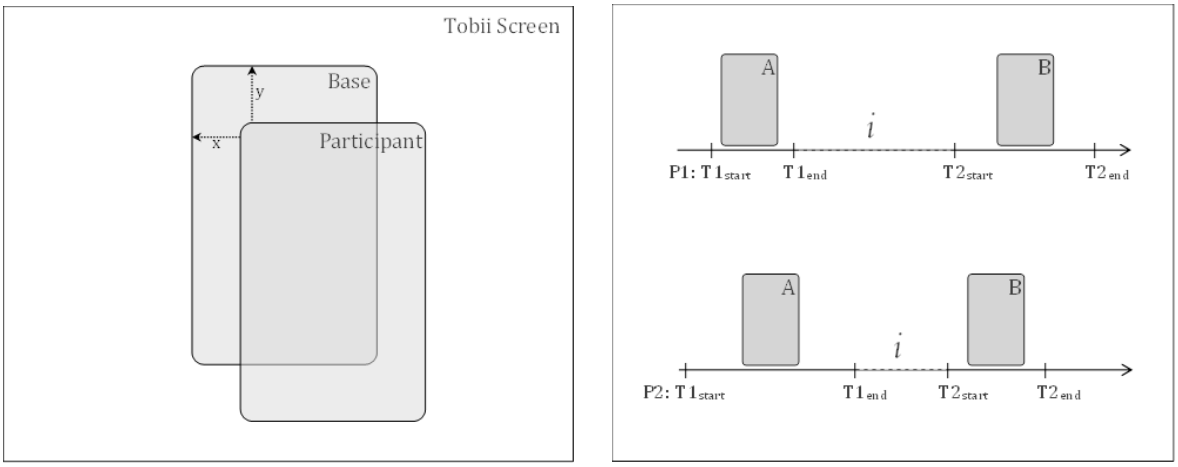
apples). For both the applications tested in this study, a screen change was triggered by an event. Thus, events had to be captured either using Tobii Studio or another method. Seeing that the Pilot study was a web–based mobile application, running in a mobile emulator, Tobii Studio could capture the event data as it does in any browser. The event clicks captured with Tobii Studio were used to define the screen changes. For the Validation study, the application ran in the Windows Phone emulator and the same event data could not be captured by Tobii Studio. Additional data capturing was therefore added to the mobile application. For each event, the mobile application captured when that event occurred, on which element the event occurred and if a screen change occurred because of that event. The event data was used during data pre–processing to divide the data into subsets corresponding to the user tasks.

4.2.4 Eye tracking data pre–processing for proposed approach

During the pre–processing stage, the fixation data is imported, the fixations are aligned by means of affine transformation to the correct positions, see Figure 4.4(a), saccades are derived from the fixations and data subsets are created according to the defined tasks and subtasks.

Affine fixation transformation

Affine data point transformation was necessary, because not all the emulators were in the same position on the screen when the eye tracking data was captured. Tobii Studio captured everything on the screen with a set size of 1024px × 768px, but the emulators were much smaller. Thus, the first step was to align the eye tracking points relatively for all the participants. To align these screens, the position of the emulator was captured per participant. A base position was specified, see Figure 4.4(a), and all the data points of the eye tracking data were then shifted, using affine transformation, in a two–dimensional plane, to that position, using the difference between the base position and the participant emulator position for the x and y values.



(a) Translate the data points to the set base position.

(b) Data subsets: participants P1 and P2, completed two tasks T1 and T2 on UI A and B.

Figure 4.4: Data points translation and subset division.

Data subset partitioning

Each screen of the evaluated application should be analysed separately. This was achieved by separating the fixations and saccades into subsets, with each set containing the eye tracking data captured on a specified screen. The screen that should be evaluated can be selected by an expert or the analysis can be applied to all of the available screens to get an overall view of the performance. The specific event data was used to know where to split the data, as a screen change occurred on some events. Figure 4.4(b) illustrates the time participants $P1$ and $P2$ took to complete tasks $T1$ and $T2$ involving user interface screens A and B. Before each task commences, instructions are shown to the user on the screen, the middle time segment (i) denotes the period when the instructions were displayed. To get comparable results, data for interface A and B of participant $P1$ will be compared to interface A and B of participant $P2$ respectively, even if the execution duration differed between the participants. The data was broken into segments between the *start* and *end* of event $T1$ and $T2$. No fixation data was exported for the time segment while the instructions (i) were displayed. The number of screens should be predefined and are dependent on the usability study.

After the pre–processing stage, the data from the usability study was in a usable format for the benchmark user identification, the FDI and the SDI processes.

4.3 Benchmark user

Every user perceives a user interface differently, but in the end the users have to accomplish the same task. Thus, there are a few crucial parts of the user interface that the user will have to focus on to make informed decisions. Some users will accomplish the task more efficiently than others. For this reason, this study introduces the concept of a benchmark user, to compare other participants to, what represents the best available visual approach for task completion, by fixating on the necessary user interface elements.

The benchmark user would have focussed (fixations) on the necessary areas of the user interface and moved from one element to the other in an efficient way (saccades), compared to the rest of the participants. Thus, the analysis is user interface independent, as the areas of interest would automatically be derived from where the benchmark user focussed. The goal of the benchmark user is to form a basis for comparison for the fixation and saccade data of the other participants.

4.3.1 Related work using benchmark tasks and users

Benchmarks are used as a point of reference for comparison in evaluations [159] and are applied in numerous ways across a wide spectrum of fields [45, 195, 225]. Performance can be calculated, with the use of metrics, relative to a set benchmark.

Usability testing has incorporated benchmark tasks to use as a baseline for performance comparison. In an attempt to improve automated usability evaluations, Ivory and Hearst [98] describe a system that allows performance measures to be automatically generated. To accomplish this, benchmark tasks should be defined along with a sequence of events and comparative performance metric [8]. These benchmark tasks have also been generated by means of a genetic algorithm that evolves from user interaction with a system as input and generates realistic benchmark task variations [113]. The GLEAN tool [115], automates part of the GOMS model [24] by introducing benchmark tasks as an input into the model to reduce time spent on usability analysis. Benchmark tasks are

not only used in model–based usability methods, but are also applied in usability testing methods. Usability testing uses benchmark tasks to define the tasks that users have to perform [76, 182]. Benchmark performance metrics can be set for each task, such as the expected time to complete a task or the number of errors that could occur [1, 54, 178]. These benchmark tasks and benchmark performance measures can also be generated to be re–used in future usability tests for comparative outputs [61, 111, 223].

Scholtz et al. [191] developed a tool that uses the average performance of two expert users to establish benchmark (or baseline user) for how well the users are expected to perform. The baseline users are tasked to perform card sorting [35] and the results are compared to the card sorting results of other participants by means of predefined metrics. Baseline or benchmark users can also be used for quantitative comparative studies, as done by Comber and Maltby [32], who drew a metaphor between language construction and task execution on software and applied Shannon's language entropy equation to the recorded tasks. The results of all the users are then compared to the results of a benchmark user, also referred to as an expert user, to determine the feasibility of user interface analysis.

Benchmarks are applied in various ways in the usability field, drawing from these applications. This study proposes the introduction of a benchmark user to automate and speed–up the eye tracking usability analysis process. In 1967 a study was conducted by Yarbus to demonstrate that eye movements are not just random, but task driven [10]. This proposed method, selects the user who performed a task efficiently and effectively as the benchmark user. The method further assumes that the benchmark user will focus on the essential components of a user interface to complete a task. The eye tracking data of the other participants are then compared with regard to the visual strategy of the benchmark user.

4.3.2 Selecting a benchmark user

For this proposed approach, a benchmark user should be selected depending on the requirements of the usability study. A benchmark user can either have extensive knowledge of the application used in the usability study or fit the required demographic of the targeted user group. For the first option, a developer, designer or even a project

manager of a system can be selected as the benchmark user, someone who is familiar with the application and will be able to complete tasks efficiently and successfully. The behaviour of such a benchmark user is used as the expected behaviour, which the system design was based on, to the actual behaviour from participating users. The other option is to select the benchmark user as the participant, fitting specific demographics, who completed the task the best in the pool of participants who partook in the usability test. This would allow comparison against a person from the user group of the system who managed to complete the tasks efficiently. Multiple benchmark users can also be selected for comparison, for example by selecting a different benchmark user for different user groups.

Some pre–defined criteria should be set to identify the benchmark user. The criteria for selecting a benchmark user are dependent on the usability study; one study might consider the least number of events to complete a task as an important success factor, whereas in another study it might be important to ensure that the exact data was entered.

4.3.3 Pilot study benchmark user identification

To select a benchmark user from the Pilot study, a number of attributes were considered. The first criterion was whether or not the user completed the task successfully. A task is completed successfully if all the task requirements have been met. This was to ensure that the participant understood what was required and could execute the task. Next, the lowest number of fixations and saccades were highly important, this was to identify the participant who could extract information from the screen efficiently and respond effectively. Lastly, time to complete a task was considered, if there were users with the same number of fixations. This would indicate a higher overall fixation duration, which indicates that more time was spent interpreting the user interface components and thus the participant was less efficient. A different benchmark user can be selected for every task.

Consider Table 4.1 for selecting the benchmark user. According to the pre–defined criteria, participant 5 was the selected benchmark user for Task 1, 2 and 3 and thus served as the benchmark user for the Pilot study.

Participant	Task Success?	Fixations	Saccades	Time on Task	Benchmark User?
Task 1					
1	Yes	52	51	26.88s	×
2	Yes	42	41	19.28s	×
3	Yes	38	37	21.55s	×
4	Yes	33	32	32.70s	×
5	Yes	17	16	13.33s	✓
Task 2					
1	Yes	91	90	32.60s	×
2	Yes	28	27	13.38s	×
3	Yes	262	261	118.90s	×
4	Yes	250	249	161.32s	×
5	Yes	20	19	17.23s	✓
Task 3					
1	Yes	95	94	56.64s	×
2	Yes	18	17	11.78s	×
3	Yes	119	118	52.67s	×
4	Yes	41	40	52.32s	×
5	Yes	10	9	13.52s	✓

Table 4.1: Benchmark user selection for the Pilot study, Tasks 1–3.

In the expert based usability study of the Rustica application (section 3.5.5), Task 2 was analysed in detail and usability issues were listed for this user interface. In order to compare the expert findings to the findings from the proposed approach, the analysis focussed on Task 2. The gaze data in Figure 4.2(e) shows the fixations and saccades for the benchmark user (participant 5) while completing Task 2. If the scan path is viewed more closely it can be seen that to complete Task 2, the benchmark user focussed on the category title at the top of the page as well as the page delimiting tabs to establish if the necessary data is shown. The participant identified both the product in the category and the product price. The eye movements then moved to the right where the text box was and then the current quantity of zero products. The user clicked on the text box to complete the requirements for Task 2.

This eye tracking data is used in the FDI (Section 4.4) and the SDI (Section 4.5) processes, to identify where and by how much the other participants deviated in order to automate the analysis process.

4.4 Fixation deviation index

Fixations as data points hold a lot of information, because of the link between the cognitive activity of users and what they are looking at. There are different aspects of a fixation that can be recorded: the position, starting time, duration and the fixation sequence. This information can be applied and visualised in various ways, depending on the desired information to be extracted. For the requirements of this study, it is important to note on which elements of the user interface a participant focussed, thus the fixation position is relevant.

To automate the process, a benchmark user is selected to make the analysis independent of the user interface layout as the benchmark fixations indicate the areas of interest. A metric is introduced to express the fixations variance between the participants in the study and the benchmark user; this metric is referred to as the fixation deviation index (FDI). This index automatically identifies participants or tasks which have possible usability issues. By automatically mapping areas with a high FDI back onto the user interface, problem areas can be highlighted. These areas, tasks or user groups can then be investigated further by the expert analyst.

4.4.1 Proposed FDI process

The FDI process in the proposed method covers a number of steps in order to highlight usability issues; see Figure 4.5 and Algorithm 2 for an overview of the process.

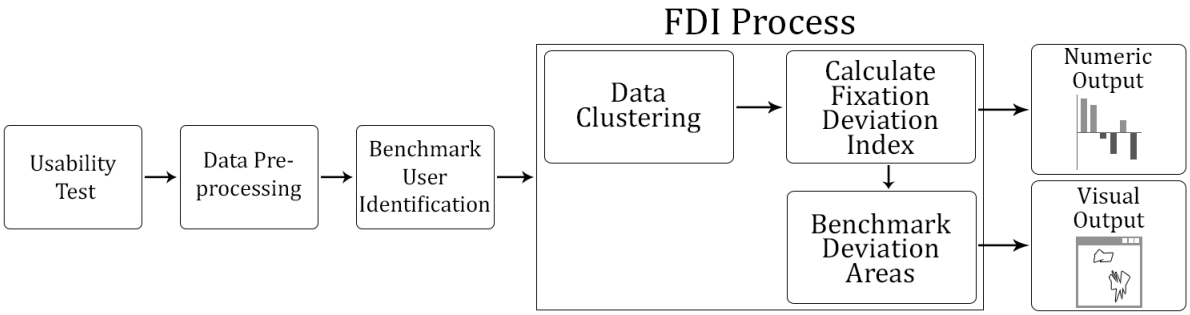


Figure 4.5: Fixation deviation index process diagram.

Algorithm 2 FDI process

```
for all tasks : t do
    for all participants : p do
        execute ← Data Clustering                          ▷ Algorithm 3
        execute ← Calculate FDI                            ▷ Equation 4.1, 4.3, 4.2 & 4.4
        execute ← Benchmark Deviation Areas                ▷ Algorithm 4
    end for
end for
```

The first step in the FDI part of the process is fixation *data clustering*. In order to determine how much the fixation data of the participants differ from the benchmark user, the benchmark user's fixations are used as centroids of a cluster. The fixations of each participant are then clustered with respect to the closest neighbouring benchmark user fixation. If the fixations of a cluster are very widely spread, the fixations of the participant are far away from the fixation of the benchmark user. This indicates that the participant was looking for the information and elements to complete the task in a different place on the user interface as compared to the benchmark user, indicating a possible usability issue.

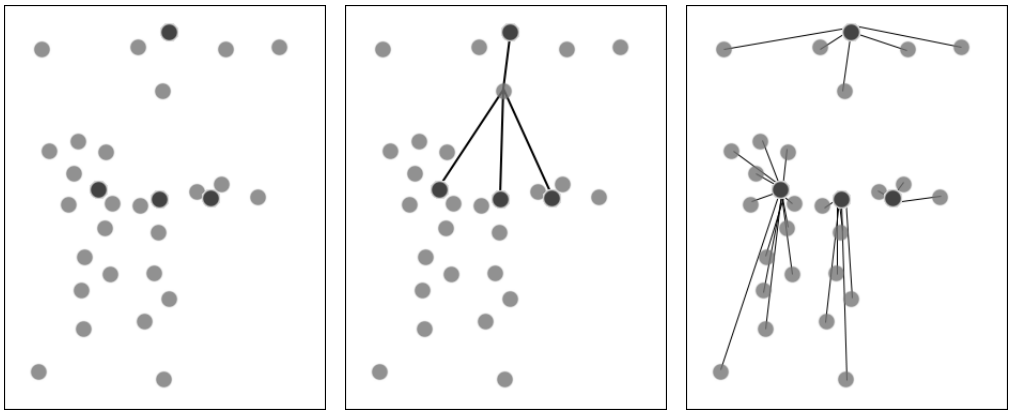
An index is calculated for each of the defined clusters; this represents the measure of how much a participant deviated from the benchmark user centroid in a specific area. To *calculate the* FDI for a task, an average deviation measure is calculated from all the cluster deviation indices.

The last step involves mapping the data back onto the user interfaces to highlight where the participant deviated, (*benchmark deviation area*). Clusters with high deviation indicates areas where the participants' interaction with the application varied a lot from the benchmark user and investigating visual representations of these areas provides significant information into usability problems of the application.

Each of these steps is applied to the data for each task and each participant. The phases are described in detail in the following subsections.

4.4.2 Data clustering

The areas on the user interface where the benchmark user fixated, represent relevant information to complete the task at hand. Determining the dispersion of the neighbouring participant fixations, from the benchmark user's fixations, gives insight into how close the participant was to the most efficient visual strategy. To establish if the participant's fixations are close to the benchmark user's fixation, the data is grouped or clustered according to their relative position. Using this approach, no information is needed on how many areas are important on the screen, or where the important areas are. Benchmark user's fixations are assumed to be the important areas. The benchmark user's fixates on the screen components that are required to be focussed on to complete the task, so these fixation points are used as the cluster centroids. The number of cluster centroids are therefore equal to the number of fixations of the benchmark user. Each participant's fixations are allocated to the cluster with the centroid closest to that fixation. All fixations are grouped into a cluster, even if they are not on the same component on the screen.



(a) Fixation points (b) Euclidean distance (c) Clusters

Figure 4.6: Illustration of FDI clustering process. Each participant fixation (lighter points) is added to the cluster of the benchmark user fixation point (darker points) closest to it.

Figure 4.6 illustrates the data clustering process. The darker points represent the centroids (benchmark user fixations) and the lighter points portray the fixation for a

Algorithm 3 FDI: Fixation clustering for a single participant

Input:
  fixations of the participant : $f_1, ..., f_i, ..., f_{n_f}$
  centroids (fixations of the benchmark user) : $c_1, ..., c_k, ..., c_K$
Output
  clusters of fixations : $cluster_1, ..., cluster_k, ..., cluster_K$
for all clusters : $cluster_k$ **do**
  $cluster_k \leftarrow \varnothing$
end for
for all fixations : f_i **do**
  $d_{min} \leftarrow \infty$
  $k_{min} \leftarrow 0$
  for all centroids : c_k **do**
    $d_k \leftarrow \mathrm{d}(f_i, c_k)$ ▷ Equation 4.1
    if $d_k < d_{min}$ **then**
      $d_{min} \leftarrow d_k$
      $k_{min} \leftarrow k$
    end if
  end for
  add f_i **to** $cluster_{k_{min}}$
end for

participant completing a single task, Figure 4.6(a). As depicted in Algorithm 3, to determine the closest centroid to a fixation, the Euclidean distance (Equation 4.1) is calculated between the participant fixation (f_i) and all centroids (c_k), shown in 4.6(b). The fixation f_i is then grouped into $cluster_{k_{min}}$, of the centroid closest to it. This is illustrated in Figure 4.6(c) for each of the fixations.

$$d(f_1, f_2) = \sqrt{(f_1.x - f_2.x)^2 + (f_1.y - f_2.y)^2} \tag{4.1}$$

4.4.3 Calculate fixation deviation index

To obtain a quantifiable measure of how much each participant deviated from the benchmark user while completing a task, the fixation deviation index (FDI) is defined. To calculate the FDI for a task requires a number of steps and calculations. A deviation value ($FDI_{cluster_k}$) is calculated for each cluster, by means of the average absolute deviation defined as:

$$FDI_{cluster_k} = \frac{\sum_{i=1}^{n_f} |d(f_i, c_k) - \bar{d_k}|}{n_f} \tag{4.2}$$

where

$$\bar{d_k} = \frac{\sum_{i=1}^{n_f} d(f_i, c_k)}{n_f} \tag{4.3}$$

$\bar{d_k}$ is the mean Euclidean distance of all fixations (n_f) from the centroid c_k, in the cluster $cluster_k$. The mean Euclidean distance of the cluster $\bar{d_k}$, is subtracted from the Euclidean distance of the centroid and fixation $d(f_i, c_k)$, to get an absolute difference. The sum of the absolute difference of every fixation is calculated and divided by the total number of fixations n_f to determine the $FDI_{cluster_k}$.

The FDI total, the sum of all the cluster deviation averages $FDI_{cluster_k}$, is then divided by the number of clusters K to get the FDI value for a participant completing a given task.

$$FDI = \frac{\sum_{k=1}^{K} FDI_{cluster_k}}{K} \tag{4.4}$$

Thus, the FDI is a statistical dispersion measure, the average absolute deviation, which reflects how compressed or scattered the fixations are for a specific participant completing a task. The FDI is a summary statistic that gives an overview of the data captured during task execution. The FDI results are real numbers, greater or equal to zero. If the data is exactly the same, the FDI would be zero; the FDI will increase the more the participants' fixations vary from the benchmark user's fixations. The FDI will

be small if the participant fixate close to where the benchmark user fixated. A small FDI is acceptable, because the participants would not necessarily fixate on the exact coordinates where the benchmark user fixated, but they could still fixate on the same component. The greater the FDI is, the further the participant fixated away from the relevant information on the screen.

The resulting metrics can be used to filter through large sets of data captured during an extensive usability test with a large number of participants. The data can be filtered according to a specific user group, who had difficulty completing a task. The data can also be filtered to see which tasks have high FDI values to identify tasks with possible usability issues for further investigation.

4.4.4 FDI results of the Pilot study

In this section, the first part of the process discussed is applied to real data in order to verify the approach. Data captured during the Pilot study was used in this section and the results are discussed.

A small FDI indicates that the participant focused in the same vicinity of the user interface as the benchmark user. Even if the fixations were not on the exact same position, they could still be on the same element of the interface. At least a slight deviation was thus expected from all the participants. Larger FDI values indicate greater variance; indicating that the user focussed on additional elements on the screen that were irrelevant to complete the task successfully. The FDI results are relative to the user interface screen and should be interpreted accordingly. There is no maximum deviation value, only a minimum optimal value of zero.

In Table 4.2 the FDI is listed per participant and for every task completed during the Pilot study. The time to complete the task and number of fixations (n_f) are also listed to give an overview of the other data captured. The FDI value for the benchmark user is zero, as expected.

Results show that Task 1 had the lowest average FDI value and very small variation between the participants' FDI values. The low FDI values indicated that the participants approached the task in a similar way as the benchmark user. Results for Task 2 showed the highest average FDI of all tasks as well as high variance between the results of the

	Task 1			Task 2			Task 3			
Participant	FDI	Time	n_f	FDI	Time	n_f	FDI	Time	n_f	FDI Average
1	**0.86**	26.88	51	**2.23**	32.60	91	**3.23**	56.64	94	**2.11**
2	**0.36**	19.27	41	**0.43**	13.38	28	**0.22**	11.78	17	**0.34**
3	**0.66**	21.54	37	**5.43**	118.90	262	**3.04**	52.67	118	**3.04**
4	**0.41**	32.70	32	**5.52**	161.32	250	**1.15**	52.32	40	**2.36**
Benchmark	**0**	13.33	16	**0**	17.23	20	**0**	13.52	9	0
Average	**0.574**	22.74	35.40	**3.404**	68.68	141.2	**1.909**	37.39	55.60	–

Table 4.2: FDI results for the Pilot study.

participants. The variance between the different participant FDI values indicated that some participants had more difficulty completing the task than others. This task should be investigated further to identify where exactly the usability issues lie. The results for Task 3 also showed high variance between the participants' results, but the task FDI was lower than the FDI average of Task 2. There could also be some usability issues to investigate in this task. Participant 2 was the participant with the lowest FDI value of all the tasks and also the only participant who used a computer prior to the usability test (Table 3.2). This could indicate that computer illiterate participants had trouble interpreting the user interface, which was also highlighted by the FDI value.

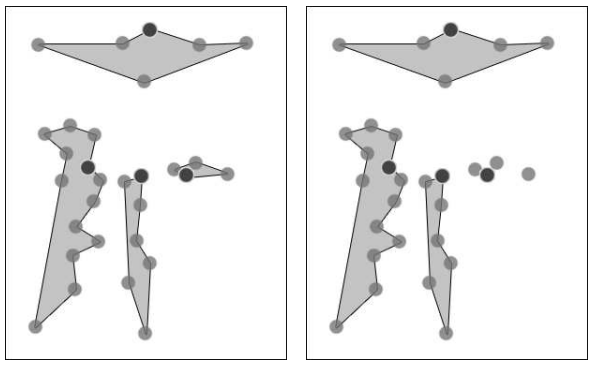
In the data, it should be noted that even though participant 2 had the lowest FDI value, this was not the participant with the lowest number of fixations. Considering the data of participants 3 and 4, the time spent completing the task was almost the same, but the FDI values differed quite significantly. Participant 3 also had a lower FDI value than participant 1, even though participant 1 had a lower number of fixations. This showed that the time spent on a task and the number of fixations, even though related, were not an indication of the amount of deviation.

The expert–based analysis study focused on the data captured during Task 2, the task which included the core functionality of the application. Task 2 also had the highest variation in FDI results and highest FDI average, indicating possible usability issues. Thus, the remainder of this analysis will also focus on the data from Task 2.

4.4.5 Benchmark deviation areas

The FDI measure highlights which participants deviated significantly from the benchmark users, but does not show where deviation occurred on the user interface. This part of the process makes use of benchmark deviation areas (BDA) to highlight areas on the user interface with possible usability issues.

To obtain additional information from the proposed method, clusters with high deviation are mapped back onto the original user interface. The areas with high FDI values are the areas where most deviation occurred, but can also indicate that the area was significant to the participants and they expected to find certain information there. By highlighting these areas, the expert can investigate the cause of the high deviation further. In Algorithm 4, the clusters with FDI values higher than the total FDI multiplied by ω, a specified weight ($FDI_{cluster_k} > \omega FDI$) are plotted back onto the user interface by means of polygons. Each fixation point of the cluster serves as a point in the polygon, indicating areas with noticeable deviation, as shown in Figure 4.7(a).



(a) All clusters (b) High deviation clusters

Figure 4.7: Illustration of drawing fixation clusters called benchmark deviation areas.

The illustration in Figure 4.7(a) shows how a cluster is represented by a polygon. Consider the smallest cluster in Figure 4.7(a), even though the participant did not focus on the exact same point as the benchmark user, both still fixated in approximately the same area or on the same user interface component. This behaviour results in a $FDI_{cluster_k}$ value that is less than the ωFDI for the task, thus this cluster will not be

Algorithm 4 FDI: Drawing benchmark deviation areas

1: **for all** clusters : $cluster_k$ **do**
2: **if** $FDI_{cluster_k} > \omega FDI$ **then**
3: Draw $cluster_k$ as polygon on UI
4: **end if**
5: **end for**

drawn back onto the user interface, as seen in Figure 4.7(b). The weight, ω, specified by the expert, can be used to change the threshold level of when a cluster should be visualised.

4.4.6 Related quantitative metrics

A method developed by Wooding [227] produced two metrics relating to fixation data. The first metric is the coverage percentage of the visual stimuli that had a number of fixations higher than a specified fixation count threshold. The second metric provides an indication of similarity between two fixation datasets. The fixation count difference between two datasets, at each point on the visual stimuli, is calculated. The resulting dataset is normalised between zero and one and the average fixation count is then calculated for the dataset.

The FDI is also a metric that considers the fixation data, but it depicts the dissimilarity of the fixation data of two datasets. The deviation considers the spread of the fixation data of the one dataset, in relation to another. Both the FDI and the metric produced by [227], shows similarity to some extent, but respectively one metric considers the fixation location while the other uses fixation count.

4.4.7 BDA results of the Pilot study

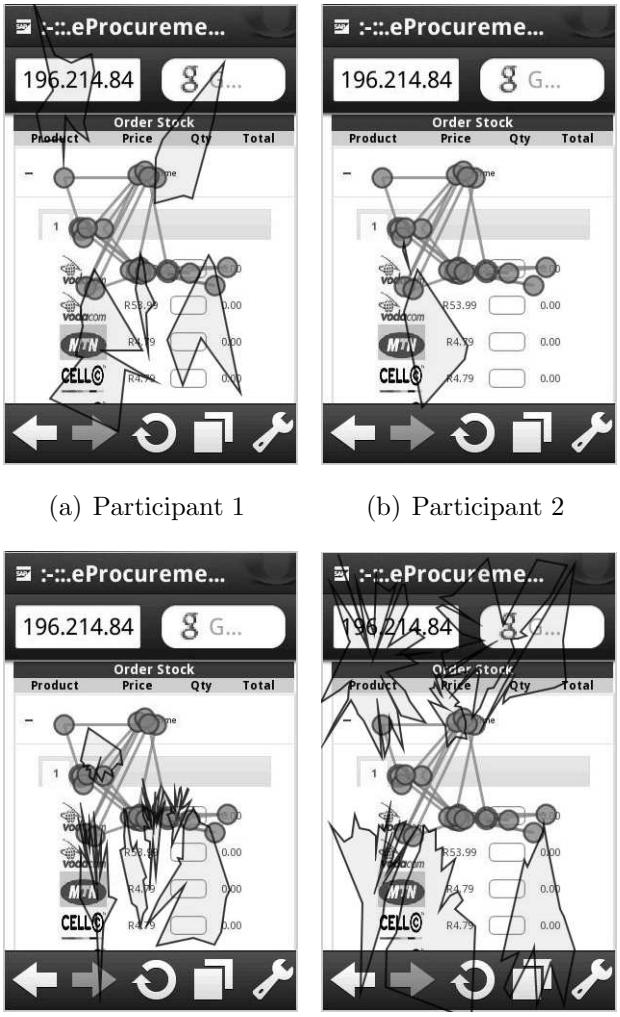
In this section the BDA part of the proposed process is applied to the Pilot study data, to determine if the proposed visualisations can provide insight into the usability issues of the user interface to an expert analyst.

The use of a benchmark user in this approach removed the need to map out areas

of interest on the user interfaces. The focal points of the benchmark user were already seen as significant areas. This saved the expert analyst time and effort by removing the need to map out the important areas on the interface. The benchmark deviation areas represented the clusters with high FDI values. Drawing the areas with high deviation back onto the user interface enabled the expert analyst to visually see where the participant fixated on areas which the benchmark user did not fixate on. The visualisation uses polygons to highlight the areas; each corner in the polygon represents a fixation point in the cluster. The edges are more prominent in the polygon than the inside area. The edges represent the fixation position and the polygon area indicates the general region on the user interface where the participant deviated. A cluster with smooth edges has fewer fixations than a cluster with jagged edges. The benchmark user's fixations were also visualised along with the BDAs on the user interface, see Figure 4.8. This was to accentuate how the deviation areas highlighted the fixation deviation.

Figure 4.8 shows the BDA for task 2 of the Pilot study, participants 1–4. Considering participant 2 (Figure 4.8(b)), even if the participant's average FDI for this task was low, there were areas where the participant deviated from the benchmark user, as indicated by the BDA. Comparing these results with Figure 4.2(b) it was clear that the deviated area was highlighted as expected. The remaining participants explored the user interface more widely before completing the task. Unlike participants 3 and 4 (Figure 4.8(c) and 4.8(d)), the edges of the clusters for participant 1 (Figure 4.8(a)) were not very jagged. This indicated that participant 1 scanned over the user interface, but not repetitively. On the other hand, participants 3 and 4, fixated on other parts of the user interface a significant number of times before completing the task. This could indicate uncertainty or searching for an element.

Interpreting the BDA results, a number of observations could be made from the visualisations. The product images drew the attention of all the participants and even though the product they had to add was the first one on the screen, all the participants fixated on the other images as well. Another significant part of the user interface was the prices; all participants fixated on the prices next to the images. The text boxes received little attention from the participants who seemed to focus on the sub–totals to the right of the text box. The header and footer of the emulator was used to load



(a) Participant 1 (b) Participant 2

(c) Participant 3 (d) Participant 4

Figure 4.8: BDA results for each participant completing Task 2.

the web application, but drew the attention of the participants more than expected. Participants 1 and 4 explored the table header, resulting in a BDA over the headers, whereas participants 2 and 3 did not deviate to the table headers much and no BDA was drawn in that area. The table names did not seem to contribute to the efficiency of completing the task, as participants 3 and 4 were the worst performers.

The benchmark user was specifically used to analyse comparative performance, rather than performance as assessed by an expert. In this case, the participant's demographic

data was also relevant and should be taken into consideration with the FDI results. The fact that participant 2 was the only participant who has some experience with computers in the past could contribute to the low FDI value. Participant 2 completed Task 2 similarly to the benchmark user by knowing how to interpret a text box.

4.4.8 Comparison of findings

Analysing clusters with high FDIs could result in discovering areas with usability problems. In order to support the use of the proposed approach, the findings were compared to the expert–based analysis conducted on the Rustica application (Section 3.5.5), using the same data. Thus, the expectation would be that if this proposed approach is valid, it should deliver the same results as the expert–based analysis, while being less resource intensive. The comparison was done while using the same user interface components as the expert analysis. Table 4.3 shows the findings from each analysis approach, separated into the different components discussed in the expert analysis.

There were a high number of correlations between the findings of the two methods. The high number of similarities indicates that the FDI method is a feasible, supplementary automated method for expert analyst to use. The two main advantages to use the FDI method are to get quantitative data and simplified visualisations. The quantitative results would assist in quickly identifying tasks with usability issues and user groups who had difficulty with a specific task. The simplified visualisations removed cluttered data from the interface, only highlighting where the participants deviated.

In the following section, additional indices and a visualisation technique are discussed, which provide information to expert analysts in the usability evaluation process.

UI	Expert Findings	BDA Findings
Product images	This was one of the first things the participants focused on, sometimes all the pictures were fixated on before the participants move to the prices.	Deviation occured one or more times on all the images. The images were the most investigated feature on the user interface as shown by the jagged edges of the BDAs over the images.
Product price	After fixating on the images, the participants backtracked up to the prices until the desired product was found.	Participants deviated significantly on the product prices as seen by the BDA edges on the prices. The edges are much smoother than over the images indicating less repetitive fixation.
Text boxes	Because of unfamiliarity, little attention was paid to the text boxes even though it was the key component in completing the task.	There were very few fixations on the text boxes, deviation clusters form over the text boxes, showing that the participants disregarded the text boxes and deviated to the surrounding components.
Product tabs	Tabs were possibly confused for input elements on the user interface.	As the benchmark user also fixated on the tabs, the deviation of that component is regarded as significant and is not highlighted by a BDA.
Table headings	Participants fixated on the table headings to determine the context of the other user interface components.	Two participants deviated towards the table headings. The very jagged edges show repetitive fixations, highlighting searching. The 'Product', 'Price' and 'Qty' heading was deviated to.
Emulator menu	The emulator was not part of the Rustica interface, yet participants searched the emulator assistance when they could not find the needed components.	The emulator menu only had a few sharp BDA corners, indicating that the participants did scan the menu, but rapidly returned to the interface.

Table 4.3: Comparison of findings of the proposed automated method FDI process and the expert findings on the Pilot study.

4.5 Saccade deviation index

During a fixation, cognitive processing occurs with respect to the element the user is focussing on. In contrast with a fixation, the brain selectively shuts off the visual processing during an eye movement. This is known as saccadic masking [166], where a person is effectively blind during a saccade. Even if there is not valuable information about the elements over which a saccade moves, the saccade movement still holds significant information. Data such as the scan path (consecutive saccades), saccade angle, saccade count and saccade length are relevant to user interface analysis. For the purpose of this study, the movement between elements, on the user interface (saccade position) and the saccade length are relevant. These two elements can be interpreted as information such as where the user expected to find an element, whether or not the user is deciding between two options, whether or not the user is searching for an element and how directed the eye movement was, to name a few.

The same approach was followed for the saccade data as with the fixation data – utilizing a benchmark user for automated analysis. The saccade data of the participants are compared to the benchmark user's data to identify where the participant deviated from the benchmark user's path. To quantify how much a participant deviated from the efficient benchmark user, two sets of metrics are defined: saccade deviation indices (SDI) and saccade length indices (SLI). These indices reflect deviation in metrics used by various usability studies [55]; the number of saccades (SDI) and the saccade length (SLI), respectively. These metrics can be used to filter through the data and identify user groups and tasks where there are possible inefficiencies in the design. To visualise the repetitive path deviation of a participant, the benchmark deviation vectors (BDVs) are introduced by this proposed method. The BDVs will highlight the path from one element to another that the participant followed repetitively, that differed from the benchmark user's path.

This automated analysis approach supports an expert in the usability analysis. The SDI together with the FDI can be used to identify where and by how much the participants deviated from the benchmark user in terms of fixation and saccade data. The following section describes each step of the SDI process in the proposed approach and how this supports the findings off an expert–based usability study.

4.5.1 Proposed SDI process

The SDI process which consists of four steps, is shown in Figure 4.9. Each step is applied to every task and every participant, as shown in Algorithm 5.

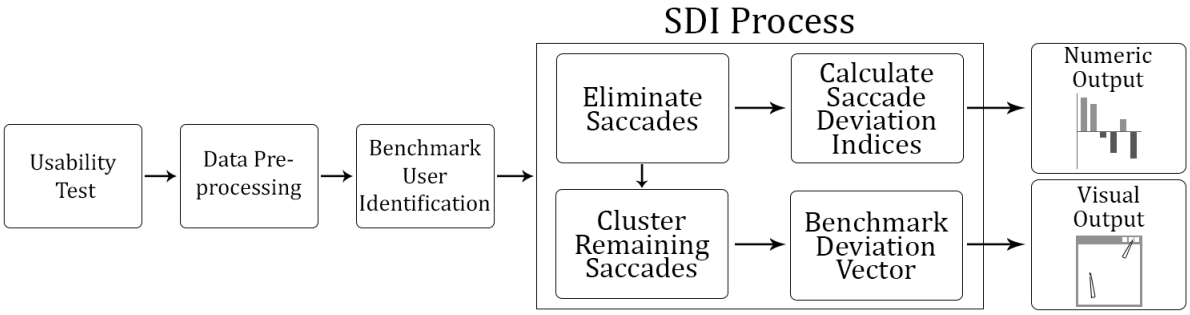


Figure 4.9: Saccade deviation index process diagram.

Algorithm 5 SDI process

```
for all tasks : t do
    for all participants : p do
        execute ← Eliminate saccades                                   ▷ Algorithm 6
        execute ← Calculate saccade deviation indices ▷ Equation 4.5, 4.6, 4.7 & 4.8
        execute ← Cluster remaining saccades                           ▷ Algorithm 7
        Draw Benchmark Deviation Vectors
    end for
end for
```

To *eliminate saccades* is the first step in the SDI process, this step removes the participant's saccades similar to the saccades of the benchmark user. The saccade elimination and remainder data is used to *Calculate the saccade deviation indices* (SDI and SLI). The next step is to *cluster the remaining saccades* into groups where the saccades are similar. These clusters are then used to draw the *benchmark deviation vectors* back onto the user interface. These vectors highlight the repetitive scan path deviation position and direction. A detailed description of each step follows in the subsections below.

4.5.2 Eliminate saccades

To allow the automated analysis process to remain as user interface independent as possible, there is no input specifying the correct path to follow or even the important elements on the screen. This information is extracted from the benchmark user's saccades and fixations. Presuming the path followed by the benchmark user is efficient, if the participants followed the same path they fixated on all the necessary components and in the right order. The information to extract is where the scan path of the participants differed from those of the benchmark user. For this reason, the first step is to eliminate the saccades of the participant which are similar to that of the benchmark user. Saccades are perceived as vectors in order to calculate the SDI and SLI values, cluster and visualise the data.

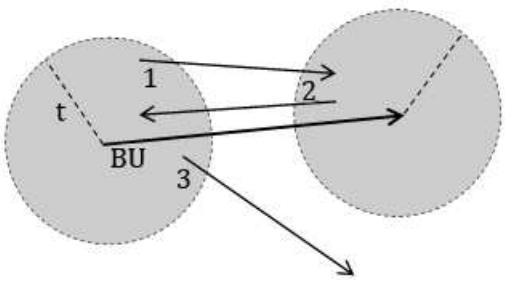


Figure 4.10: Illustration of benchmark user saccade elimination from participant saccades.

Saccades are considered similar if their start points are in close proximity to one another and their end points are also in the same vicinity. Not all eye movements will be in the exact same position, thus a threshold (t_e) is set to determine if the saccades are approximately the same. Figure 4.10 is an illustration of how saccade elimination works. The BU line represents a benchmark user's saccade and the remaining lines, *1–3*, represent the saccades of a participant. The BU saccade moves from one area to another on the user interface. Saccade *1* is eliminated, both within the threshold and in the same direction. Saccade *2* is within the same threshold, but moves in the opposite direction and is not eliminated. Lastly, saccade *3* has the same start point but the endpoint is not within the threshold, and consequently is not eliminated.

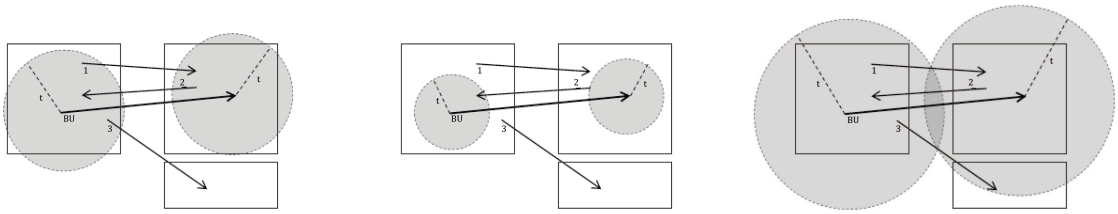
The eliminated saccades are not disregarded as seen in Algorithm 6. If the difference between the benchmark user's saccade and the participant's saccade is smaller than the threshold, then the saccade s_i is added to the $saccades_{eliminated}$ subset. If the saccades are not within a threshold of one another, the saccade s_i is added to the $saccades_{remainder}$ subset.

Algorithm 6 SDI: Eliminate saccades for a single participant completing a single task.

Input:
  threshold : t_e
  saccades of participants : $s_1, ..., s_i, ..., s_{n_s}$
  saccades of benchmark user : $b_1, ..., b_m, ..., b_M$
Output:
  $saccades_{eliminated}$: $e_1, ..., e_j, ..., e_J$
  $saccades_{remainder}$: $r_1, ..., r_p, ..., r_P$
$t \leftarrow threshold_{elimination}$
for all saccades of benchmark user: b_m **do**
  for all saccades of participant : s_i **do**
    if $d(b_m.start, s_i.start) < t_e$ **and** $d(b_m.end, s_i.end) < t_e$ **then**
      add s_i **to** $saccades_{eliminated}$
    else
      add s_i **to** $saccades_{remainder}$
    end if
  end for
end for

Saccades are similar if the difference between the start point of the benchmark user's saccade and participant's saccades and the end points of the saccades are within the defined threshold. The threshold is user interface dependent and a suitable value will depend on the size of the user interface and granularity of the elements. Refer to Figure 4.11(a), which shows a sufficient threshold around the BU start– and endpoints, covering a wide area of the two rectangles. The rectangles represent user interface components, such as buttons or text boxes. In the case where the threshold is too small, some

saccades moving between the same two components will not be eliminated, as seen in Figure 4.11(b). The opposite can happen if the threshold is too big. Figure 4.11(c) shows that saccades moving to other components on the screen can be incorrectly eliminated.



(a) Sufficient (b) Too small (c) Too big

Figure 4.11: Importance of the selection of a sufficient saccade elimination threshold.

The threshold should be smaller if the components on the screen are small. In a case where the screen size is relatively big, such as a desktop application, and the elements on the screen are small but not too close to one another, then the threshold can be slightly bigger. On a small screen with larger components, the threshold should be relatively larger. This study proposes a threshold value of 80% of the average component size on the screen. This ensures that the threshold is big enough to include most of the individual components on the screen and might only have a small overlap with other components, as users mostly focus on the center of an element and not its edges.

4.5.3 Calculate saccade deviation indices

Many different eye tracking studies make use of saccade length for quantitative analysis [36, 101, 118]. The same metrics are used in this proposed approach, but applied to the saccades eliminated and remainder as extracted in Algorithm 6.

The first proposed SDI metric is the saccade deviation index eliminated ($\text{SDI}_{\text{eliminated}}$), defined as a metric showing the similarity between the participant's scan path and the scan path of the benchmark user. The $\text{SDI}_{\text{eliminated}}$ is shown in Equation 4.5:

$$SDI_{eliminated} = \frac{J}{n_s} \tag{4.5}$$

where J is the number of saccades eliminated and n_s is the total number of saccades for that participant completing a specific task. The percentage of saccades eliminated shows how many of the participant's saccades move between the same components on the user interface as the benchmark user's saccades. If the $\text{SDI}_{\text{eliminated}}$ is high, the participant focussed on the relevant elements on the screen and followed a similar path as the benchmark user. For a low $\text{SDI}_{\text{eliminated}}$, it can be assumed the participant did not follow the same path as the benchmark user.

The next SDI metric is the saccade deviation index remainder ($\text{SDI}_{\text{remainder}}$), as shown in Equation 4.6 and indicates how much the scan path of the participant differed from the benchmark user:

$$SDI_{remainder} = P \tag{4.6}$$

where P is the number of saccades that was not eliminated. The $\text{SDI}_{\text{remainder}}$ provides insight into the performance of the participant while completing a task. If there was little deviation and the participant followed a similar path to the benchmark user, then a large number of saccades would be eliminated and the number of remaining saccades would be low, therefore the $\text{SDI}_{\text{eliminated}}$ would be high and the $\text{SDI}_{\text{remainder}}$ would be low. A high $\text{SDI}_{\text{eliminated}}$ does not always guarantee that the number of remaining saccades will be low. Both the $\text{SDI}_{\text{remainder}}$ and the $\text{SDI}_{\text{eliminated}}$ can be high, depicting that even though the participant did follow the same path as the benchmark user, there was still a lot of additional deviation.

The second set of saccade deviation metrics are the saccade length indices (SLI); the saccade length could infer searching efficiency and how directed and meaningful the eye movements were, indicating pre–planned movement [73, 118]. To calculate how much deviation occurred, only the saccades remaining after elimination are used. There are two proposed SLI metrics, the $\text{SLI}_{\text{total}}$ and the $\text{SLI}_{\text{average}}$. The $\text{SLI}_{\text{total}}$ is defined as the total length of the remaining saccades showing how much the scan path of the participant deviated from the benchmark user. This metric is calculated, as shown in Equation 4.7, by the Euclidean distance sum of all the $saccades_{remainder}$:

$$SLI_{total} = \sum_{p=1}^{r_P} d(r_p.start_{fixation}, r_p.end_{fixation}) \tag{4.7}$$

where r_P is the of number remaining saccades and d is the Euclidean distance (Equation 4.1) between the start ($r_p.start_{fixation}$) and end ($r_p.end_{fixation}$) fixations of the remaining saccade p.

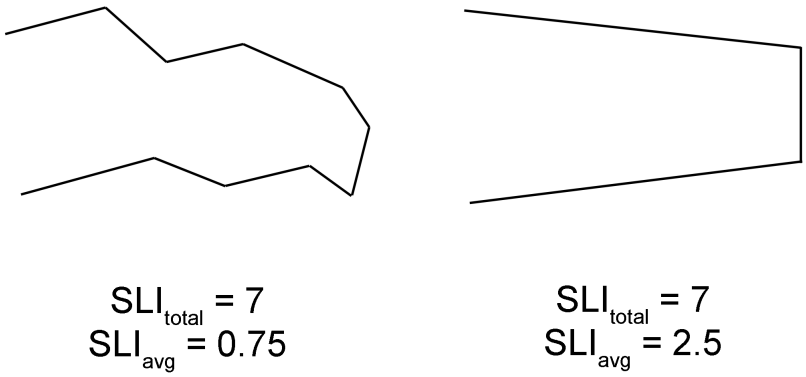
Next the $\text{SLI}_{\text{average}}$ is defined as the average length of the remaining saccades that depicts the type of eye movement of a participant that occurred during a specific task. This is calculated for the number of saccades remaining, as shown in Equation 4.8:

$$SLI_{average} = \frac{SLI_{total}}{r_P} \tag{4.8}$$

Even though the $\text{SLI}_{\text{average}}$ is derived from the $\text{SLI}_{\text{total}}$, it conveys different information that relates to eye movement [55, 156]. Figure 4.12 demonstrates how the $\text{SLI}_{\text{total}}$ and the $\text{SLI}_{\text{average}}$ differ. The $\text{SLI}_{\text{total}}$ shows how much deviation occurred and the $\text{SLI}_{\text{average}}$ shows the type of eye movement that occurred in the deviation. Both scan paths are the same length, but the averages differ. The first scan path (Figure 4.12(a)) consists of a number of small short saccades, resulting in a smaller $\text{SLI}_{\text{average}}$. The second scan path (Figure 4.12(b)) is more directed with fewer saccades; this is reflected by a larger $\text{SLI}_{\text{average}}$. The total and the average SLI values give separate information, but when interpreted together, additional information can be obtained, such as that there was a lot of deviation (high $\text{SLI}_{\text{total}}$) and searching occurred during the deviation (short $\text{SLI}_{\text{average}}$). When analysing the average saccade length, a larger average indicates that the saccades overall were deliberately aimed towards components on the screen and a smaller saccade average can indicate more searching on the screen.

4.5.4 SDI results of the Pilot study

The data captured from the Pilot study is used in the SDI process to automatically extract the SDI and SLI metrics.



(a) Small $SLI_{average}$ (b) Big $SLI_{average}$

Figure 4.12: Difference illustrated between the SLI_{total} and $SLI_{average}$.

Pilot study SDI results

Table 4.4 shows the results after eliminating the saccades of all the participants that were similar to the benchmark user. The best results for the $SDI_{eliminated}$ are if 100% of the saccades were eliminated. This would only happen if the participant followed the same path as the benchmark user, within the given threshold. The higher the $SDI_{eliminated}$ elimination percentage, the better because it indicates a lot of overlap with the benchmark user's scan path. The number of remaining saccades ($SDI_{remainder}$) should ideally be zero, because then the participant's scan path did not deviate from the benchmark user's scan path at all. The lower the $SDI_{remainder}$, the less deviation occurred. Results from Table 4.4 can be interpreted as follows:

Participant	Total	Eliminated	$SDI_{remainder}$	$SDI_{eliminated}$
1	90	41	49	45.46%
2	27	20	7	74.07%
3	261	188	73	72.03%
4	249	101	148	40.56%
Benchmark	19	19	0	100%

Table 4.4: SDI results for Pilot study for each participant completing Task 2.

Participant 1: The participant had more total saccades than the benchmark user, but far fewer than participant 3 and 4. The participant had a relatively low elimination percentage and intermediate saccade remainder. This reflected deviation during the task, but almost half of the saccades did overlap with those of the benchmark user.

Participant 2: The saccades of participant 2 aligned closely to the benchmark user's saccades. This is reflected in the low saccade remainder of only 7 saccades and a high percentage of 74.07% eliminated saccades. With a limited amount of data in the visualisation, it was still possible to visually compare the images in Figure 4.2. The benchmark user (Figure 4.2.e) and participant 2 (Figure 4.2.b) had similar scan paths, as reflected by the SDI results.

Participant 3: The SDI results showed a high elimination percentage of 72.03%. This indicated that the scan path data of participant 3 overlapped with that of the benchmark user significantly. The saccade elimination remainder was, however, also high (73 remaining saccades), indicating additional searching in other areas of the screen, even though most of the scanning occurred in the same area as that of the benchmark user.

Participant 4: With the lowest elimination percentage(40.46%), participant 4 had significant deviation from the benchmark user. The participant also had the highest number of remaining saccades, 148 saccades, even if the participant did not have the highest number of total saccades. This participant did not follow the same path as the benchmark user and focussed on many elements of the user interface that were unnecessary to complete the task.

Pilot study SLI results

The SLI makes use of the saccades remaining after elimination. The optimal SLI_{total} is zero, indicating no deviation from the benchmark user. A high SLI_{total} could suggest inefficient task completion, as the participant spent time scanning parts of the user interface not needed for task completion. Conversely, a low SLI_{total} relates to effective and efficient task completion. The $SLI_{average}$ indicates the type of eye movement. A low

$SLI_{average}$ value, indicating shorter eye movements, is related to searching on the user interface, while a higher $SLI_{average}$ shows more directed eye movements.

Table 4.5 shows the SLI data for Task 2 of the Pilot study. The following can be deduced from the SLI information for every participant.

Participant	SLI_{total}	$SLI_{average}$
1	5755.03	117.45
2	707.92	101.13
3	5304.78	72.67
4	13558.72	91.61
Benchmark	0	0

Table 4.5: SLI for Pilot study for each participant completing Task 2.

Participant 1: The participant had the highest $SLI_{average}$ and had very directed eye movements only focussing on key elements of the user interface, indicating little searching. Even though the participant deviated from the benchmark user, as indicated by the SLI_{total}, the deviated movement was directed.

Participant 2: With the lowest SLI_{total} there was very little deviation from the benchmark user. Participant 2 also had a relatively high $SLI_{average}$ showing that where the participant deviated, the eye movements were directed with limited searching.

Participant 3: The SLI_{total} of participant 3 was close to the SLI_{total} of participant 1, but participant 3 had a much lower $SLI_{average}$. This showed deviation from the benchmark user, but with short eye movements indicating more searching than participant 1.

Participant 4: This participant deviated from the benchmark user as indicated by the highest SLI_{total} of all the participants. This translated into inefficient task completion. The lower $SLI_{average}$ indicated less directed eye movements and more searching while completing the task.

From these results an expert analyst can investigate further why participant 4 struggled to complete the task and why participant 3 was searching excessively to complete the

task. In large datasets the SLI results could assist the experts in identifying high level usability issues on an interface.

4.5.5 Cluster remaining saccades

The resulting metrics assist in filtering through the data and identifying the tasks or user groups with issues concerning the use of the application. The next step involves visualising the deviation of the saccade data. When investigating the remaining saccades, it was found that the saccades formed clusters when superimposed onto the user interface. This indicates that there were elements on the user interfaces that captured the attention of the participants on the areas where deviation occurred and a repetitive path to that component could be seen. A cluster is defined as a group of similar objects; in this study three or more similar saccades within the specified threshold, t_g, are considered a cluster. Clusters are used rather than individual saccades to highlight repetitive scan paths the participant followed.

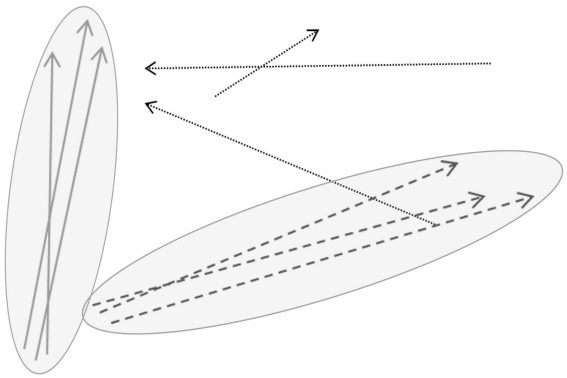


Figure 4.13: Similar saccades are grouped into clusters

The number of clusters in the saccade data are unknown after saccade elimination occurred and would differ for every participant and task, thus the number of clusters should be determined automatically. Figure 4.13 illustrates different saccades resulting from the previous phases in the proposed approach. From the illustration, it can be seen that there are two definite clusters with at least three saccades with closely related features. To cluster the data automatically, the saccades should be compared and grouped if similar, this is described in Algorithm 7.

Algorithm 7 SDI: Cluster remaining saccades for a single participant and task

Input:
  threshold : t_g
  list of saccades remaining : $list_r$
  minimum cluster size : c_{min}
Output:
  set of clusters : C
$C \leftarrow \varnothing$
$list_u \leftarrow list_r$
while $list_u$.size > 0 **do**
  $f \leftarrow$ first element in $list_u$
  add f **to** $cluster_{temp}$
  remove f **from** $list_u$
  $cluster \leftarrow \varnothing$
  add f **to** $cluster$
  while $cluster_{temp}$.size > 0 **do**
    $s \leftarrow$ first element of $cluster_{temp}$
    for r in $list_r$ **do**
      if r in $list_u$ **then**
        if $d(s.start, r.start) < t_g$ **and** $d(s.end, r.end) < t_g$ **then**
          add r **to** $cluster_{temp}$
          remove r **from** $list_u$
          add r **to** $cluster$
        end if
      end if
    end for
    remove s **from** $cluster_{temp}$
  end while
  if $cluster$.size $>= c_{min}$ **then**
    add $cluster$ **to** C
  end if
end while

The algorithm clusters the saccades within the given threshold t_g of one another. This threshold can be the same as was used for eliminating saccades, seeing that it is being applied to the same user interface. The algorithm has a cascading effect – each saccade that was added to a cluster, is also compared to all of the remaining saccades. This is to ensure that a saccade close to the cluster will be included, even though that saccade might not be within the threshold of the first saccade selected for comparison. Thus, every saccade (s) that has not been clustered is compared to all the other saccades ($list_u$) that have not been clustered. Each of the saccades allocated to the cluster of s, will then also be compared to the remaining saccades not clustered, until there are no saccades within the threshold of the current *cluster*. As input, a minimum size (c_{min}) is defined, only if the number of saccades within the *cluster* is more or equal to the c_{min} value, will the cluster be superimposed onto the user interface.

4.5.6 Benchmark deviation vectors

Drawing each of the lines in a cluster, as defined in the previous phase can cause user interface information to be hidden under the visualisation, due to too many lines cluttering the screen. The visualisation should highlight the path between objects where the participant varied from the benchmark user. To focus on the path followed and not on all the various lines in a cluster, an average saccade is calculated for every cluster and only the average saccade is superimposed back onto the user interface.

The saccades in a cluster all have similar start and end points. To determine the average saccade, the average position of the start coordinates of the saccade are calculated and the same is done for the end coordinates of the saccade. Using these two average coordinates, the $saccade_{avg}$ is defined. The $saccade_{avg}$, of the cluster in Figure 4.14, has a dotted line. To represent the average saccade position, magnitude and direction a triangle is drawn on the user interface. The triangle provides a clear visual representation of the $saccade_{avg}$ direction, length and position as opposed to just a line, representing the $saccade_{avg}$ vector, on the screen.

The base of the triangle (the shortest side of the triangle) indicates the starting position of the $saccade_{avg}$. The base can be set to 10% of the threshold t_g as it relates to the size of the screen. The apex (vertex where two equal sides of a triangle meet) is

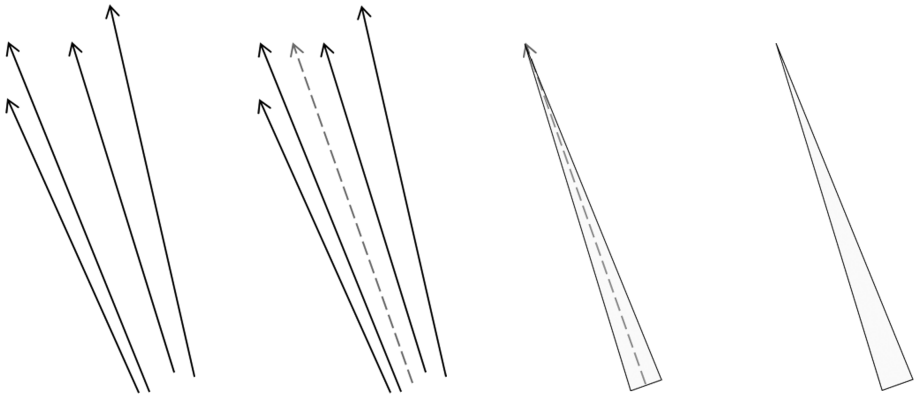


Figure 4.14: Illustration of how benchmark deviation vectors are created from cluster of saccades.

the end position for the $saccade_{avg}$. These highlighted average saccades are referred to as the benchmark deviation vectors (BDV). Algorithm 7 specifies that only the clusters with a number of saccades more or equal to the minimum cluster size will be drawn, the c_{min} value of this study is 3. The $saccade_{avg}$ is then calculated for all qualifying clusters and drawn back onto the user interface.

With this process, the expert analyst can focus on the saccade deviation indices (SDI) and saccade length indices (SLI) as well as on the user interfaces with the benchmark deviation vectors (BDV) visualisations. The following section shows the results of applying the proposed approach to the Pilot study data.

4.5.7 Related quantitative metrics

Jarodzka et al. [103] produced metrics to measure the similarity of scan paths. The method produces five different metrics that measure similarity in terms of shape, amplitude, position, direction and duration. The study calculates these metrics for all the saccades in a scan path that are aligned and then reports the averages.

The proposed approach produces four metrics from the saccade data. Two of the metrics relate to the number of saccades (eliminated and remaining) and the other two metrics depict the deviation in terms of the saccade amplitude (length). Both the pro-

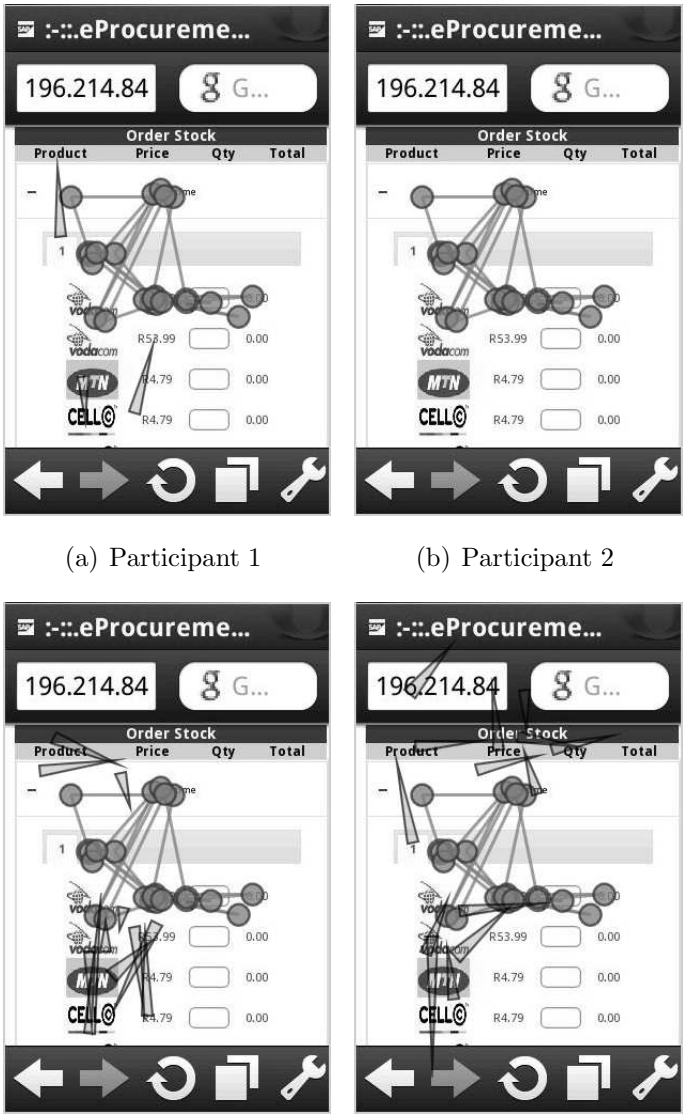
posed approach and the Jarodzka et al. [103] study interpret the saccades as vectors. The two studies also only consider the similarity between two datasets at a time, take sequence into account, and do not require areas of interest to be mapped out. The proposed approach also subtracts saccades (as vectors) from one another to find similar saccades, but the proposed approach does this within a set threshold to include saccades that are close to one another, even though they are not exactly aligned. Furthermore, the proposed approach only calculates the saccade lengths for saccades that differ from the benchmark user, not for all the saccades in the scan paths.

4.5.8 BDV results of the Pilot study

Saccade visualisations provide an accumulative overview, to easily see the scan path data of a participant. By drawing only the deviation data on the user interface, the visualisation is less cluttered and easier to analyse. The benchmark deviation vectors highlight repetitive scan paths between two objects on the user interface. Figure 4.15 shows the benchmark deviation vectors of participants with the fixation data of the benchmark user.

Inspecting the visualisation for each participant, the following observations can be made:

Participant 2 (Figure 4.15(b)) performed well and deviated very little, thus no deviating clusters formed from the resulting data, as there were no saccades moving from one element on the screen to another more than twice. This resulted in no BDV drawn back onto the user interface. The remaining participants had a number of similar scan paths. The scan path of all these participants scanned down the images and two participants even followed a scan path back up to the product, which had to be added. The scan path of all the participants, except for participant 2, backtracked up the product prices, towards the price that identified the relevant product they had to buy. The table headings were read from left to right by participant 3 and 4, and participants focussed on the numbered tabs before moving to the table headings. Participant 4 (Figure 4.15(d)) was the only participant who scanned the emulator header and footer repetitively even though the other participants also fixated on the emulator, seen from the BDA in Figure 4.8. In Figure 4.15(c), two short BDVs moving between the two top images (which



(a) Participant 1 (b) Participant 2

(c) Participant 3 (d) Participant 4

Figure 4.15: BDV results each participant completing Task 2.

are the same) could indicate that participant 3 was looking for differences between the two images. Participant 3 also scanned up and down the remaining images and prices repetitively, which could indicate searching or indecisiveness.

The visualisations are also generated automatically, and can assist experts to get insight into the usability evaluation results without spending too much time analysing

the replays.

4.5.9 Comparison of findings

The BDV visualisations give an informative overview of the main deviation in the path followed between these user interface components. Table 4.6 shows the comparison of the BDV and expert–based analysis results. The comparison is done to substantiate why it could be beneficial for an expert analyst to make use of the SDI method.

There are a number of similar observations that can be made using the two different methods. The advantage of using the SDI method is that the expert needs to spend much less time evaluating the eye tracking data. The BDV visualisation resulting from the SDI method provides a simplified visualisation of saccade data. These results can complement the expert–based study to speed up the analysis process.

UI	Expert Findings	BDV Findings
Product images	This was one of the first things the participants focused on. Sometimes all the pictures were fixated on before the participants move to the prices.	Participants repetitively scanned over the pictures on the screens as indicated by the multiple BDVs moving up and down the images.
Product price	After fixating on the images the participants backtracked up to the prices until the desired product was found.	There was a clear tendency for participants to move down the images before backtracking up to the prices. The BDVs moving down the images and up to the prices for participant 1, 3 and 4 supported this.
Text boxes	Because of unfamiliarity, little attention was paid to the text boxes even though it was the key component in completing the task.	Only one BDV moved directly from the product image towards the text boxes, as seen for participant 4. The users did not move to the text boxes and subtotals of each product line as expected, but rather down and up the images.
Product tabs	Tabs were possibly confused for input elements on the user interface.	Participant 1 and 4 repetitively fixated on the tabs before moving to the table headings for additional information as shown by the present BDVs.
Table headings	Participants fixated on the table headings to determine the context of the other user interface components.	Participant 3 and 4 read the table headers a number of times, as indicated by the BDVs moving from left to right.
Emulator menu	The emulator is not part of the Rustica interface, yet participants searched the emulator for assistance when they could not find the needed components.	Only participant 4 deviated to the emulator multiple times, as shown by the BDV moving to the emulator menu, while the other participants did focus on the emulator, but not repeatedly.

Table 4.6: Comparison of the findings of the proposed automated method SDI process to those of the expert–based analysis of the Pilot study.

4.6 Conclusion

Automated usability testing is considered to be one of the most time–efficient methods of usability evaluations. Data capturing is at an advanced stage of automation, while data analysis and critique are less developed in terms of automation. As an example, an eye tracker automatically captures the eye movements of a person while completing a usability test. The analysis of the eye tracking data should, however, then be conducted by an expert analyst. The process is extremely time–consuming and the expert needs extensive knowledge of the user interface to map out areas of interest for analysis.

The proposed approach in this chapter discussed a method to automate a part of the analysis approach. The premise of the approach was to select the most efficient participant in a usability test as the benchmark user. Because the benchmark user completed the task in the most efficient and effective way, it was assumed that the user followed a better visual path than the other participants. The eye tracking data of the remaining participants were compared to the benchmark user in order to determine how much each participant deviated from the benchmark user. The deviation was automatically translated into quantitative data. The method is applied to both fixation and saccade data. To get additional output from the automated method, the areas with high deviation were superimposed back onto the user interface.

There are a number of advantages of this proposed approach. The method saves time by automatically providing quantitative data, which can be used to filter through the data and quickly identify tasks or user groups where issues occurred. The quantitative output also provides an unbiased approach when analysing the results without partial views getting in the way. The relative data also makes the analysis user interface independent, so any size interface with any size components can be automatically analysed. The benchmark user already fixated on the important elements on the user interface and therefore the areas of interest do not need to be mapped out by an expert. The proposed approach is especially useful in a usability test in which a high volume of participants partake and/or a large number of tasks are completed. This allows for easy filtering and only focussing on the tasks with high deviation and high irregularity in the results.

In this chapter the proposed approach was discussed in detail and the results of the proposed method were compared to the results from an expert–based analysis to show

the feasibility for automatically analysing eye tracking data of the method. Another objective is met by visualising relevant resulting data onto the user interface, to highlight possible usability issues for the expert to consider.

In the following chapter, this approach is applied to a larger group of participants and a larger set of tasks to demonstrate the effectiveness of the method when applied to voluminous data.

Chapter 5

Validation

With eye trackers now even available for mobile devices, the possibility to conduct remote, large scale usability studies with the use of eye trackers has become a reality. Chapter 4 described how the proposed automated eye tracking analysis method was developed using the Pilot usability study consisting of five participants. This chapter applies the proposed approach to a larger dataset, analyses the results, and compares the findings to an independent expert analysis of the data.

5.1 Introduction

In this chapter the dataset collected from the Validation study is discussed first, to put the results to follow into context. The selection criteria and benchmark user selection are discussed to show why the specific participants were selected for each sub task. An initial investigation is described that makes use of some summarized results to filter through the data and identify problem tasks. The proposed approach is then applied to the identified tasks. Each step from the previous chapter is applied to the dataset and the results are discussed. Lastly, a comparison follows between the findings from the automated method and the findings from an independent expert analysis. This addresses the last objective of this study: to determine the feasibility of the study, by applying it to a bigger dataset and comparing the results.

5.2 Data

The usability study, referred to as the Validation study, captured the eye tracking data of 33 participants completing three given tasks, as explained in Section 3.6. This section provides more information on how the recorded data was interpreted.

5.2.1 Data separation into subtasks

Task 1 and Task 2 had the same objective: to place an order with the provided BiYP application. Task 3, on the other hand, required the participant to view an invoice of the order placed during Task 1 and 2. To give the gaze data context, the fixations and saccades are relative to a specific user interface, otherwise the data is just dots and lines on a random space. Since Task 1 and Task 2 had the same objective and used the same user interface screens, the eye tracking data from these tasks were grouped together and the data from Task 3 was grouped separately. These datasets were then divided into subtasks, for each individual user interface screen.

Table 5.1 lists the ten subtasks associated with the three tasks of the Validation study. Each subtask is given a name and is executed on a particular user interface screen (the figure number of the screen is given in the 3^{rd} column of the table). The 4^{th} column states the tasks that the subtask applies to and the final column gives a short description of what was required to complete the subtask on the given user interface.

5.2.2 Data pre–processing

The Tobii T120 records 120 images per second. Should the image not be sufficient to track the pupil because of external factors, it is recorded as a failure. From this data, an accuracy percentage could be calculated, presenting how many of the recording intervals succeeded in tracking the eye gaze. If the recording percentage of a participant was below 40%, the data was discarded for the remainder of the study. Table A.1, in Appendix A, provides the participant data for the Validation study, including the recording accuracy.

Using the Tobii Studio software, fixation data was exported to text files using the Tobii I–VT fixation classification algorithm. This contained all the fixation data captured

Subtask	Name	UI	Task(s)	Description
1	Menu1	3.4(a)	1 & 2	To place and order, the *Shop* option had to be selected from the main menu.
2	Menu2	3.4(a)	3	To view an invoice of the previous order placed, the *Shop* option had to be selected from the main menu.
3	Supplier	3.4(b)	1, 2 & 3	Only one supplier was available on the system, thus *Metro Hyper – Hillfox* had to be selected from the supplier list.
4	Categories1	3.4(c)	1 & 2	The category containing the required product had to be selected from the categories menu.
5	Categories2	3.4(c)	3	To view a previous order from the selected supplier, the *Orders* pivot (tab) had to be selected at the top of the categories page.
6	Products	3.4(d)	1 & 2	From the list of products, the specific product with the right quantity had to be selected.
7	Product	3.4(e)	1 & 2	When the correct product had been selected, the specified quantity had to be selected on the product page and the product added to the order.
8	Confirm	3.4(f)	1 & 2	A list of products showed the products and quantities ordered, the user could adjust the product quantities and/or confirm the order.
9	Orders	3.4(g)	3	On the orders page, the last order placed had to be selected to view the invoice.
10	Invoice	3.4(h)	3	The invoice of the order is displayed. From here the order could be cancelled or just closed.

Table 5.1: Tasks from the Validation study, divided into subtasks with a short description of each subtask.

for each task. The data was separated into datasets for each of the tasks completed, separated by the instructions before every task. To obtain the data for every user interface screen, the data was subdivided again into subtasks. The subtasks were separated by

event data captured in the BiYP application. This data for each subtask could then be considered individually and all the saccades were calculated from the fixation data.

The next section describes how the benchmark user was selected for the Validation study.

5.3 Benchmark user

The criteria for selecting a benchmark user are specific to the usability test depending on what is regarded as optimal, such as the least number of clicks or the shortest time spent on a task. The benchmark user is the user who is the best performer according to predefined criteria. In this study, a benchmark user was selected for each subtask to obtain results for each user interface screen. The following criteria were used:

1. *Completed overall task correctly*: A participant can only be considered as a benchmark user if all the objectives of the task were met as expected. If a task was unsuccessful, then all the subtasks of the given task were also considered unsuccessful.

2. *Accuracy* $\geq$ 90%: For a benchmark user, the percentage of eye tracking data correctly recorded should be 90% or higher. If the accuracy is too low, then the number of fixations recorded could be lower than the actual fixations required to perform the task.

3. *Least number of fixations*: The user who had optimal eye movements while completing a task should be selected as the benchmark user. Should a participant be searching for a component on the screen, then the number of fixations will increase compared to a participant who managed to locate the component efficiently.

4. *Shortest time on subtask*: Other than the number of fixations, efficiency can also be measured by means of time spent on a subtask. Should more than one participant comply with the above three criteria, then the participant with the shortest time on a subtask was selected as the benchmark user.

The complete dataset of the fixations and time of completion for each subtask can be found in Appendix A in Table A.1. Table 5.2 shows only the participants with an accuracy of 90% and higher. Some of the values for participant 1021 are crossed out, because this participant did not complete Task 1 and Task 2 correctly, therefore the results were not considered. All of the other users were considered according to fixation count and then on time; these values are listed in that order for each participant for each subtask. The fixation count and time (in milliseconds) for each participant, selected as the benchmark user for that subtask, is highlighted and in bold. Participant 1013 and 1014 had the same fixation count for Category T1, but participant 1013 was selected, because of the shorter subtask time. Participant 1033 was selected over participant 1014 in Category T3 for the exact same reason.

These highlighted participants were used as benchmark users for each of these subtasks throughout the remainder of this study.

Participant	Accuracy %	Main T1		Main T3		Supplier		Category T1		Category T3		Products		Product		Confirm		Orders		Invoice	
1009	99	13	35491	76	52716	13	7140	49	24262	17	6436	43	24102	46	32636	37	13944	12	7960	13	4478
1013	91	**11**	**18604**	38	32448	15	6503	**41**	**18626**	27	9114	92	43053	26	22464	**10**	**3492**	22	20899	6	2526
1014	97	12	18842	15	7153	**5**	**7907**	41	22432	5	4446	52	31725	31	19013	11	5115	12	11342	7	4502
1016	95	18	24510	23	20169	11	8447	88	39631	36	14008	96	50599	66	48512	20	7520	14	13117	17	7876
1020	94	12	10855	25	16206	6	6557	45	20563	38	14895	75	39141	59	35031	16	6592	33	14038	16	5503
1021	94	~~10~~	~~12842~~	21	16388	~~27~~	~~14644~~	~~33~~	~~18881~~	12	6414	~~55~~	~~24661~~	~~39~~	~~31983~~	~~3~~	~~2007~~	7	3033	**5**	**2931**
1023	90	18	42618	36	28605	11	8438	96	59749	54	27281	61	33919	40	36013	28	12939	8	5822	7	3739
1024	95	17	25683	58	42674	22	10674	99	43432	54	23152	138	60477	45	34940	14	3428	5	2386	15	6253
1027	95	17	21125	27	18366	33	15838	80	33631	94	36113	112	49953	50	46865	22	7981	14	11193	18	7418
1028	96	16	20578	72	56575	17	16061	70	47695	90	46663	94	55318	47	66658	17	5175	21	10903	12	5319
1030	98	27	22586	67	42374	21	9271	125	53955	59	27631	167	81801	95	47275	55	18125	35	15391	25	9918
1031	99	27	26309	60	46226	19	9032	97	40739	24	9599	132	57617	72	60020	27	10248	23	11879	27	9068
1032	97	30	25971	28	21163	8	6318	49	22860	108	58177	76	35766	43	49671	11	3510	**4**	**3508**	12	6958
1033	97	14	16668	**7**	**11401**	9	7335	44	22834	**5**	**2160**	**29**	**13307**	**14**	**13884**	22	7583	6	7413	12	6402

Table 5.2: The subset of participants who completed the task correctly and with an accuracy of 90% and higher. The benchmark users selected for every subtask according to the provided criteria are highlighted and bold.

5.4 Identification of subtasks with usability issues

Applying the proposed approach from this study assisted in filtering through large sets of eye tracking results collected from a usability study. The calculated metrics, such as the FDI, SDI and SLI, could initially be used to filter through the data at a very high level to identify the tasks to investigate further. Calculating the totals and averages of the resulting metrics for different tasks, subtasks, user interfaces and even user groups, could highlight different collections to focus on. An expert could then inspect the data indicating potential usability issues.

To investigate the data collected from the Validation study, the proposed approach was applied and the resulting FDI, SDI and SLI values were calculated for each subtask and each participant. The complete datasets are available in Appendix A. Thereafter, the totals and averages for each subtask were calculated in order to investigate which of the subtasks had potential usability issues. Figure 5.1 shows the normalized values for subtasks 1 to 10. The metrics have different value ranges; by normalising the data, to a range between zero and one to the observed maximum of each data set separately, they can be considered relative to one another. This was done to identify the tasks with high deviation depicted by the different metrics and the data in Figure 5.1 should be interpreted accordingly.

For the FDI value, a lower value presents less deviation from the benchmark user. The SLI_{total} value represents length of the remaining saccades, in which a lower value also shows less deviation from the benchmark user. The $\text{SDI}_{remainder}$ refers to the number of remaining saccades, also a metric of which the optimal approaches zero. On the other hand, the $\text{SDI}_{eliminated}$ metric presents the percentage of saccades that was eliminated, the higher the value the better, because there are more saccades of the participant similar to that of the benchmark user. For this reason, the average values of the $\text{SDI}_{eliminated}$ metric were not only normalised, but also inverted before being drawn on the line graph in Figure 5.1, indicated as $\text{SDI}^{-1}_{eliminated}$. Now all the lower values on the graph indicate a smaller deviation and the higher values indicate a higher deviation and a bigger chance of the occurrence of usability issues.

In Figure 5.1, subtask 5 clearly had very high deviation and was investigated in more detail. Subtask 2 had the second highest overall results, also indicating possible usability

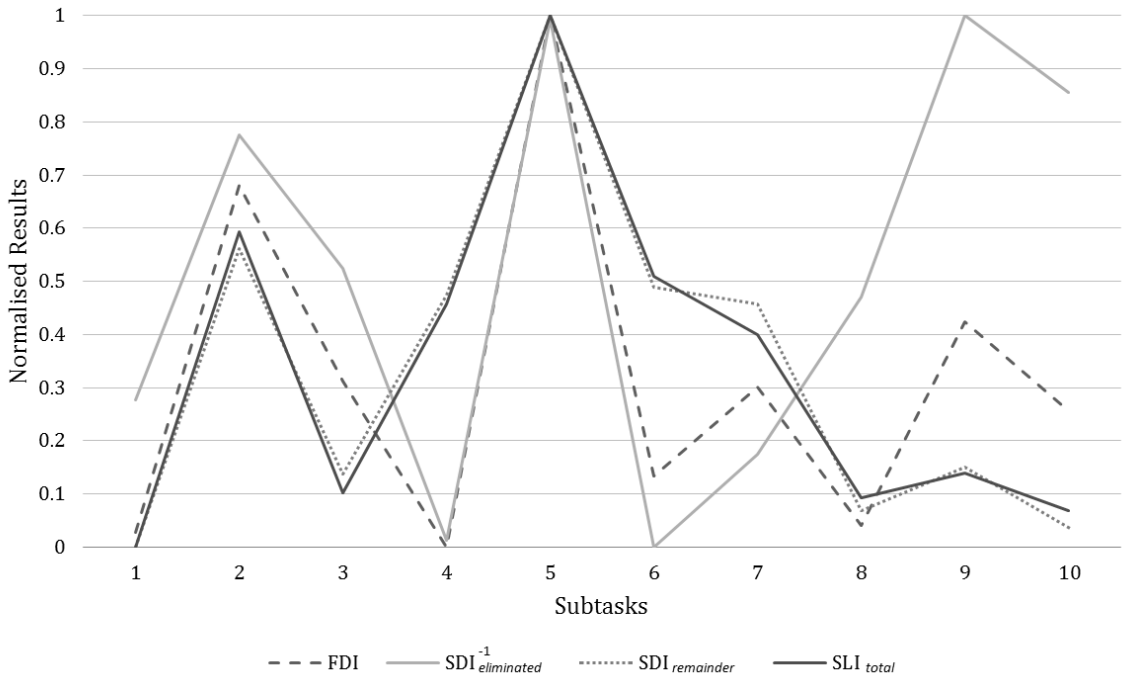


Figure 5.1: Average results from the proposed automated approached, normalised for relative comparison.

issues. Subtasks 1 and 8 had noticeably low results, indicating that users followed a similar tactic to that of the benchmark user to complete the given subtask. From the remaining subtasks, considering subtask 9 and 10 had the highest variance between the results, the $\text{SDI}^{-1}_{eliminated}$ value for both subtasks was exceedingly high. These tasks should also be investigated further to determine the cause of the high data variance. Interestingly, subtask 2, 5, 9 and 10 were all subtasks of Task 3 – viewing the invoice of an order placed with the given supplier.

Another set of values to consider was the $\text{SDI}_{remainder}$ and the $\text{SDI}_{eliminated}$ values in relation to one another. The $\text{SDI}_{eliminated}$ showed the average percentage of saccades that were eliminated by the benchmark user saccades. The $\text{SDI}_{remainder}$ metric is the average number of saccades remaining after elimination. The ideal is to have high elimination and low remainder, showing that the participants followed a similar path between components on the screen to that of the benchmark user. Subtracting the $\text{SDI}_{remainder}$ from the

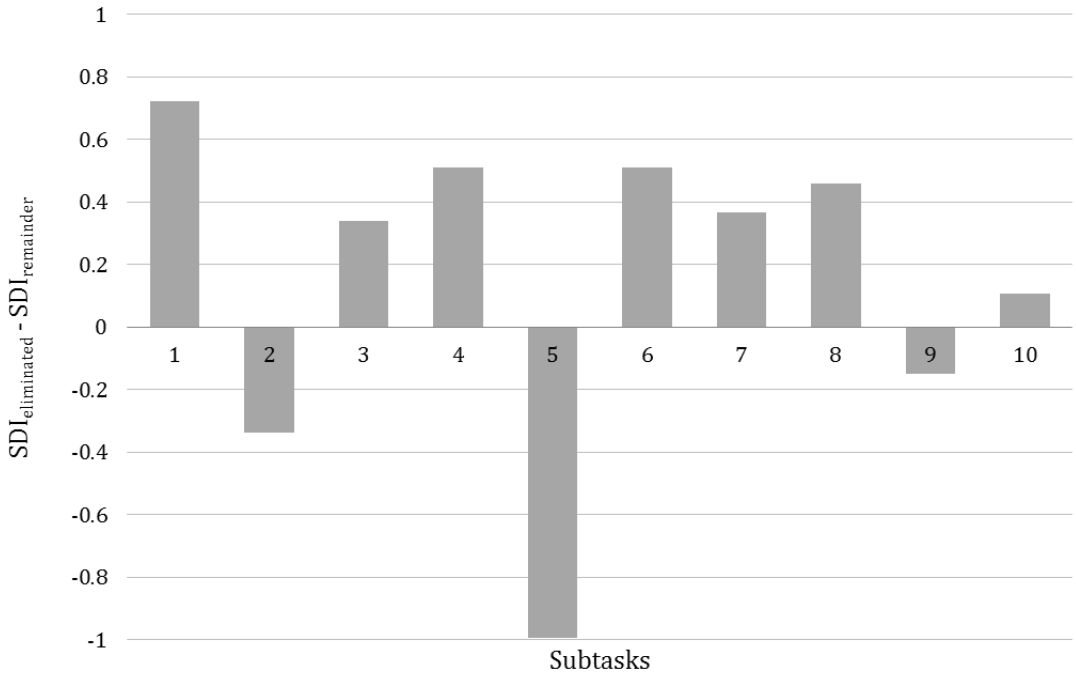


Figure 5.2: Difference between $SDI_{remainder}$ and $SDI_{eliminated}$.

$SDI_{eliminated}$ generated the graph in Figure 5.2. The subtasks with a high remainder and low elimination are highlighted by a negative difference. These were subtasks 2, 5 and 9, which were three of the four subtasks previously identified as tasks with possible usability issues.

From this initial inspection, it was decided to investigate subtasks 2 (Main T3), subtask 5 (Category T3) and subtask 9 (Orders) further.

5.5 Task specific metric inspection

In this section, the fixation deviation for each participant, completing the tasks identified in Section 5.4, was investigated to identify a number of participants who struggled to complete a given task. The visualizations of these participants could then be investigated in more detail to identify where the participants deviated. It should be noted that the participants with an accuracy percentage lower than 40% were omitted from the study.

5.5.1 FDI

Table 5.3 shows FDI values for participants on subtasks Main T2, Category T3 and Orders. The time, in milliseconds, for each subtask and each participant was also indicated in the table. The FDI values for the selected benchmark users were all 0, because no fixation deviation would occur if the benchmark user was compared to himself/herself. All FDI values should be considered relative to one another for each subtask. There is no maximum FDI value, only a minimum FDI value of zero. The information extracted from the FDI values are discussed for each of the selected subtasks in this section.

Main T3

Considering the Main T3 subtask, participant 1008, 1011 and 1012 deviated the most from the benchmark user. Even though participant 1008 had the highest FDI value, it was not the participant who took the longest to complete the task. Participant 1001 has the longest time on this task, but with an average FDI value, indicating that the participant fixated on similar areas to the benchmark user, even though this participant took longer to complete the task.

Participants 1003, 1009, 1028, 1030 and 1031 also had high FDI values in a similar range, all between 11 and 12. From the mentioned participants, participant 1003 took much longer to complete the task, but had a similar FDI value. The accuracy percentage of participant 1003 was much lower than the other mentioned participants, which could indicate that missing data was resulting in a lower–than–expected FDI value, or the participant fixated on the same areas as the other participant, but just took longer to complete the subtask.

Participants 1010, 1014 and 1025 had the lowest FDI values, showing that they fixated on similar points as the benchmark user. Participant 1014 demonstrated a noticeably lower subtask time and a very high accuracy percentage, which showed that the participant completed the subtask efficiently.

Participant	Accuracy	Main T3		Category T3		Orders	
		FDI	time	FDI	time	FDI	time
1001	62%	8.85	144.353	15.13	50.512	8.55	16.955
1003	74%	11.90	107.475	17.98	76.972	6.91	12.735
1004	66%	7.38	46.512	15.65	52.290	11.79	11.360
1005	61%	7.92	31.736	9.14	21.124	3.67	10.544
1007	73%	9.97	49.418	12.40	36.378	4.14	7.952
1008	55%	15.29	71.376	12.08	35.532	9.03	9.830
1009	99%	11.85	52.716	4.21	6.436	3.26	7.960
1010	72%	1.71	31.224	4.46	16.796	1.33	11.536
1011	59%	12.29	61.080	9.75	38.969	7.48	13.824
1012	41%	14.60	59.708	14.70	47.896	13.80	30.764
1013	91%	7.00	32.448	6.89	9.114	3.40	20.899
1014	97%	2.85	7.153	2.14	4.446	2.26	11.342
1016	95%	4.35	20.169	5.99	14.008	4.94	13.117
1017	69%	4.65	24.480	14.75	43.887	7.48	9.229
1018	47%	10.45	61.046	15.65	53.199	10.60	14.512
1020	94%	5.88	16.206	7.52	14.895	8.01	14.038
1021	94%	3.69	16.388	2.55	6.414	0.87	3.033
1022	68%	10.45	37.110	13.48	44.954	0.77	2.453
1023	90%	5.32	28.605	7.01	27.281	2.43	5.822
1024	95%	10.80	42.674	9.38	23.152	1.38	2.386
1025	68%	2.69	25.729	16.99	91.320	1.82	6.389
1026	81%	10.10	48.060	21.04	43.985	14.99	27.390
1027	95%	5.13	18.366	14.35	36.113	5.01	11.193
1028	96%	11.62	56.575	17.73	46.663	9.06	10.903
1029	87%	7.26	33.860	9.33	30.272	1.41	3.826
1030	98%	11.28	42.374	11.24	27.631	9.98	15.391
1031	99%	11.05	46.226	6.41	9.599	8.31	11.879
1032	97%	5.33	21.163	16.66	58.177	0.00	3.508
1033	97%	0.00	11.401	0.00	2.160	2.47	7.413
Average		7.99		10.85		5.70	

Table 5.3: FDI results and time for the Validation study for subtask Main T3, Category T3 and Orders. All participants completed all tasks successfully.

Category T3

The average FDI value for this subtask was the highest in the study, clearly indicating high deviation and a usability issue on this user interface. Participant 1026 had an extremely high FDI value, showing that the participant had a lot of difficulty completing the subtask. Participants 1003, 1025, 1028 and 1032 also had high FDI values in a range between 16 and 18. Considering participants 1026 and 1025, participant 1026 had the highest FDI value, but not the longest time spent on a subtask. Participant 1025 had the longest time on a subtask, but a low FDI value. A low FDI does not necessarily indicate low trouble in completing a task. If the data accuracy is low, it should be taken into consideration with the results.

Participants with lower FDI values, such as participants 1009, 1014 and 1021, also had high data accuracy percentages and low subtask completion times. Participant 1010 had a low FDI value, but the time spent on the subtask was higher and the data accuracy was lower, indicating that the missing data could have affected the FDI values.

Orders

The average FDI value of the Orders subtask was lower than the other two subtasks investigated. There was a high number of participants with low FDI values, indicating little deviation, thus fewer participants had trouble completing this subtask. Participants, such as participants 1004, 1012, 1018 and 1026 had high FDI values and lower data accuracy, showing that the FDI values could be even higher if more data was available.

Participants 1014, 1021, 1023, 1024, and 1033 had very low FDI values and also a data accuracy higher than 90%. Participant 1014 had very little deviation, but had a longer subtask completion time, showing the participant was less efficient in completing the subtask. Other participants who also had very low FDI values, but also lower data accuracy included participants 1010, 1022, 1025, and 1029. From these participants, only participant 1010 had a much longer time on the subtask, which could indicate that the data accuracy might have affected this FDI value.

5.5.2 SDI

The SDI metric consists of two values: the percentage of saccades eliminated ($SDI_{eliminated}$) and the number of saccades remaining after elimination ($SDI_{remainder}$). For this study elimination threshold (t_e) was set to 90 pixels, using the procedure described in Section 4.5.2. In Table 5.4, each participant and their data accuracy is shown, followed by the $SDI_{eliminated}$ and $SDI_{remainder}$ value for each of the selected subtasks. The benchmark user has 100% elimination and the higher the $SDI_{eliminated}$ for the remaining participants, the less they deviated. The $SDI_{remainder}$ value shows the number of saccades that were not eliminated, highlighting how much the participant deviated from the benchmark user. The benchmark user will have zero saccades remaining after elimination, thus a lower $SDI_{remainder}$ value for the participants is preferred. Therefore, a low $SDI_{eliminated}$ value and a high $SDI_{remainder}$ value are indicative of problems in the usability.

Main T3

In the Main T3 subtask, the number of participants with an extremely low $SDI_{eliminated}$ value was minimal. Only participant 1023 had a $SDI_{eliminated}$ value below 10%. In a case, such as participant 1014, where the participant had a low $SDI_{remainder}$ and low $SDI_{eliminated}$ value, it indicated that the participant followed an efficient scan path, although it differed from the scan path of the benchmark user.

A number of participants had a high remainder and an average elimination, such as participants 1003, 1011, 1012 and 1026. These participants followed a different scan path or explored the interface more than the benchmark user.

Participant 1010 was a good example of a high $SDI_{eliminated}$ and low $SDI_{remainder}$ value, indicating high elimination and only a few saccades remaining. Participant 1008 had a relatively high $SDI_{eliminated}$ value, but the $SDI_{remainder}$ value was also high. Taking this and the low data accuracy into consideration could indicate that eye movements not recorded during the test, might affect the $SDI_{remainder}$ value, or the participant repetitively searched for the correct information by following the same scan path as the benchmark user. Participants with relatively higher $SDI_{eliminated}$ and lower $SDI_{remainder}$ values included participant 1025 and 1027.

Participant	Accuracy	Main T3		Category T3		Orders	
		SDI_e	SDI_r	SDI_e	SDI_r	SDI_e	SDI_r
1001	62%	23.73%	45	13.38%	123	28.21%	28
1003	74%	24.35%	87	21.38%	114	25%	12
1004	66%	27.27%	24	18.1%	95	11.76%	30
1005	61%	34%	33	14.89%	40	25%	6
1007	73%	29.69%	45	19.77%	69	5.26%	18
1008	55%	38.38%	61	14.1%	67	19.23%	21
1009	99%	28%	54	18.75%	13	18.18%	9
1010	72%	46.15%	7	4.55%	21	0%	7
1011	59%	34.38%	63	15%	51	17.86%	23
1012	41%	23.47%	75	1.1%	90	17.46%	52
1013	91%	24.32%	28	7.69%	24	9.52%	19
1014	97%	14.29%	12	0%	4	9.09%	10
1016	95%	18.18%	18	8.57%	32	15.38%	11
1017	69%	27.59%	21	17.31%	86	24%	19
1018	47%	25.45%	41	11.46%	85	22.58%	24
1020	94%	25%	18	5.41%	35	0%	32
1021	94%	25%	15	27.27%	8	33.33%	4
1022	68%	19.61%	41	14.89%	80	0%	3
1023	90%	8.57%	32	15.09%	45	14.29%	6
1024	95%	24.56%	43	7.55%	49	25%	3
1025	68%	42.11%	11	18.52%	88	22.22%	7
1026	81%	18.67%	61	23.77%	93	5.45%	52
1027	95%	30.77%	18	24.73%	70	0%	13
1028	96%	26.76%	52	15.73%	75	0%	20
1029	87%	28.57%	30	8.62%	53	0%	6
1030	98%	28.79%	47	3.45%	56	5.88%	32
1031	99%	18.64%	48	8.7%	21	18.18%	18
1032	97%	25.93%	20	14.95%	91	100%	0
1033	97%	100%	0	100%	0	0%	5
Average		25.57%	38.1	13.26%	60.1	12.93%	18

Table 5.4: $\text{SDI}_{\text{eliminated}}$ (SDI_e) and $\text{SDI}_{\text{remainder}}$ (SDI_r) results and time for the Validation study for subtask Main T3, Category T3 and Orders. All participants completed all tasks successfully.

Category T3

This subtask had much higher $SDI_{remainder}$ values and much lower $SDI_{eliminated}$ values, indicating high deviation. Participants 1001, 1003, 1004 and 1026 had an extremely high $SDI_{remainder}$ value, showing additional scanning over the user interface. From these participants, participant 1026 did have a high $SDI_{eliminated}$ value as well, showing some similar scan paths to the benchmark user, but also extra exploration.

Participant 1014 once again followed a different path from that of the benchmark user, but still finished the subtask efficiently, indicated by an $SDI_{remainder}$ of only 4 and an $SDI_{eliminated}$ value of 0%. Participant 1021 also had a low $SDI_{remainder}$ value and a relatively high $SDI_{eliminated}$ value showing very little deviation. Participant 1012 had an extremely low $SDI_{eliminated}$ value and a high $SDI_{remainder}$ value, but also a very low data accuracy percentage; the missing data could have an effect on the results and should be investigated further.

Orders

In the $SDI_{eliminated}$ results of the Orders subtask, the number of participants who had 0% elimination was extremely high, showing that a different scan path was followed to complete this task. Some of the participants who had no elimination also had low $SDI_{remainder}$ values, for example participants 1010, 1016, 1022, 1023, 1027, 1029 and 1033. This showed that there was another effective way to complete the task efficiently other than the path followed by the benchmark user.

Considering participants 1001, 1004, 1020 and 1030, all had $SDI_{remainder}$ values in a close range to one another, but the $SDI_{eliminated}$ values differed significantly. These participants all had similar amounts of deviation, but some followed the same path as the benchmark user as well. Some of these participants even completed the task without any scan paths matching that of the benchmark user.

For the participants who followed a similar scan path as the benchmark user, the $SDI_{eliminated}$ values were relatively high, for example participants 1001, 1003, 1005, 1024 and 1025 were all between 22% and 25%. These participants also had very low $SDI_{remainder}$ values, except for participant 1001 who explored more than the other participants.

5.5.3 SLI

After saccade elimination, the remaining saccades are used to calculate the SLI values. From the remaining saccades, the total length, SLI_{total}, is calculated to show how much the participant deviated. The $SLI_{average}$ provides more information on how long the average saccade was. Longer saccades show more directed eye movement and shorter saccades can indicate more searching, confusion, indecision and even more reading. For the SLI_{total} the optimal value is zero, as can be seen, in Table 5.5, for the benchmark user totals which are 0. The $SLI_{average}$ for the benchmark users cannot be calculated, because there are no remaining saccades to divide by. Table 5.5 shows each participant, the data accuracy and the SLI_{total} and $SLI_{average}$ values for each of the selected subtasks.

Main T3

As indicated by Holland et al. [86] longer saccades occur when users have trouble finding an element in a list, which could account for the higher average $SLI_{average}$ of the main menu searches performed in this subtask, see Table 5.5. Participant 1010 had the desired deviation data – a very small amount of deviation because of the low SLI_{total} and directed eye movements presented by the high $SLI_{average}$ value. Comparing this data to the data of participant 1025, with a similar SLI_{total} value, but a much lower $SLI_{average}$ value, indicated less directed eye movements and possibly more searching.

Participants with higher SLI_{total} values included 1003, 1008 and 1031. Only participant 1031 had a relatively high $SLI_{average}$ value. A number of participants with $SLI_{average}$ values, almost half the length of that of participant 1010, included 1001, 1004, 1007, 1011 and 1012. The SLI_{total} of these participants ranged between 2000 and 6000, which was significantly different. All the participants had short eye movements, some participants just scanned, searched or read more than others.

Category T3

The $SLI_{average}$ for this subtask was the lowest, indicating many short saccades in the deviation sections. Participants 1001, 1003, 1004 and 1026 had extremely high SLI_{total} values, highlighting high deviation saccades in this subtask. In this dataset, there were

Participant	Accuracy	Main T3		Category T3		Orders	
		SLI_t	SLI_a	SLI_t	SLI_a	SLI_t	SLI_a
1001	62%	3475.21	77.23	10110.5	82.2	2219.17	79.26
1003	74%	9088.92	104.47	10367	90.94	1010.69	84.22
1004	66%	2074.05	86.42	10906.93	114.81	3758.15	125.27
1005	61%	3612.76	109.48	3342.56	83.56	631.6	105.27
1007	73%	3982.59	88.5	6071.37	87.99	1892.25	105.12
1008	55%	6336.02	103.87	6704.6	100.07	1559.41	74.26
1009	99%	5767.04	106.8	1032.53	79.43	1038.4	115.38
1010	72%	1213.42	173.35	2626.28	125.06	1378.47	196.92
1011	59%	5179.88	82.22	4218.75	82.72	2057.76	89.47
1012	41%	5766.29	76.88	9772.36	108.58	2943.86	56.61
1013	91%	4106.00	146.64	3228.36	134.52	1758.19	92.54
1014	97%	1358.13	113.18	382.96	95.74	1548.25	154.83
1016	95%	2058.37	114.35	2893.45	90.42	975.17	88.65
1017	69%	1792.41	85.35	7464.19	86.79	2404.84	126.57
1018	47%	5032.27	122.74	8629.93	101.53	2660.85	110.87
1020	94%	2508.57	139.36	2991.06	85.46	3028.95	94.65
1021	94%	1532.25	102.15	629.27	78.66	316.14	79.04
1022	68%	4384.13	106.93	7586.39	94.83	431.88	143.96
1023	90%	4079.62	127.49	4629.74	102.88	655.12	109.19
1024	95%	4829.24	112.31	4881.8	99.63	461.92	153.97
1025	68%	1222.15	111.1	8240.03	93.64	312.3	44.61
1026	81%	6072.98	99.56	9799.9	105.38	5136.12	98.77
1027	95%	1809.85	100.55	6752.18	96.46	1068.61	82.2
1028	96%	5794.78	111.44	6719	89.59	2555.17	127.76
1029	87%	3103.07	103.44	4659.45	87.91	705.97	117.66
1030	98%	4634.73	98.61	6311.88	112.71	4141.96	129.44
1031	99%	6555.43	136.57	2401.07	114.34	1881.37	104.52
1032	97%	2143.13	107.16	9107.41	100.08	0	0
1033	97%	0	0	0	0	511.35	102.27
Average		3935.68	104.16	5884.26	95.47	1760.41	103.99

Table 5.5: $\text{SLI}_{\text{total}}$ (SLI_t) and $\text{SLI}_{\text{average}}$ (SLI_a) results and time of the Validation study for subtask Main T3, Category T3 and Orders. All participants completed all tasks successfully.

also very low SLI_{total} values, as with participants 1014 and 1021. There was very little deviation in these cases, but the $SLI_{average}$ for these were low with participant 1021 having the lowest $SLI_{average}$ value.

Participants with low $SLI_{average}$ value, included 1001, 1011 and 1021. Participant 1001 had a very high SLI_{total} value and a very low $SLI_{average}$ value. This showed that the participant deviated a lot, and in the deviation the eye movements were very short, indicating possible excessive searching or scanning.

Orders

Overall this subtask had the lowest SLI_{total} values, with relatively high $SLI_{average}$ values. This indicated little deviation and directed and meaningful eye movements in the deviation areas. Only the SLI_{total} of participants 1026 and 1030 was excessively high, where a lot of deviation occurred. A long list of participants, 1005, 1021 – 1025, 1029 and 1033 had a SLI_{total} under 800. The $SLI_{average}$ of these participants had a much wider range, indicating different eye movements for the same amount of deviation.

Participant 1025 had the lowest SLI_{total} value and also the lowest $SLI_{average}$ value. This could be effected by the lower data accuracy for this participant and should be investigated in further detail. Participant 1010, as with the Main T3 subtask, had the highest $SLI_{average}$ value, indicating long and purposeful eye movements.

5.5.4 Participants with high deviation to investigate further

A number of participants were highlighted in the sections above. For some of the indicated participants the FDI and SDI data will be visualised and super imposed over the relevant user interfaces. Table 5.6 shows the participants selected for each subtask.

Subtasks	Participants				
Main T3	1003	1008	1012	1026	1030
Category T3	1001	1003	1012	1026	1032
Orders	1004	1012	1020	1026	1030

Table 5.6: Participants selected for BDA and BDV visualisation.

5.6 Task specific visualisation inspection

There are two visualisations generated by this proposed method: the benchmark deviation areas (BDA) showing the fixation points, and the benchmark deviation vectors (BDV), presenting repetitive saccadic data. The BDA uses a weight, ω of 2.0, to determine which clusters to draw. For the BDV, the grouping threshold (t_g) is 70 pixels and the minimum number of saccades in a cluster (c_{min}) to be drawn is 3.

To complete Task 3 of the usability study, the participants were asked to view the latest order that they had placed. Three subtasks were selected in Section 5.4 with the highest deviation. The Main T3 subtask (subtask 2) required the participant to select the Shop option to navigate to the supplier from whom they ordered the stock. To complete the Category T3 (subtask 5) subtask , the participants had to navigate to the Orders tab (pivot) to view a list of orders placed with the selected supplier. The last subtask investigated, was the Orders subtask (subtask 9), where the participants had to select the latest new order, which was at the top of the list to view the invoice of the order.

(a) Subtask 2 (b) Subtask 5 (c) Subtask 9

Figure 5.3: Eye tracking data of the benchmark users for the selected tasks.

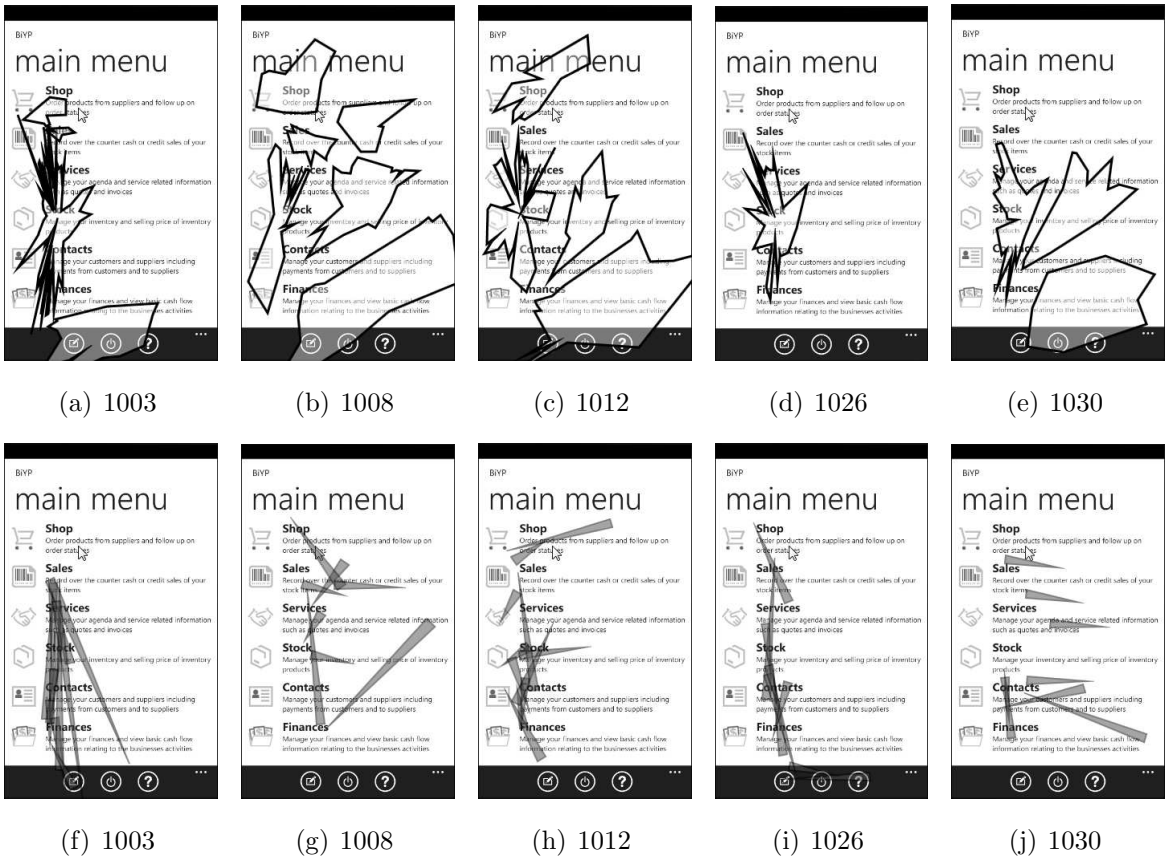
Figure 5.3 illustrates the eye tracking data of the benchmark users for subtasks se-

lected for investigation. In Figure 5.3(a), the benchmark user (participant 1033) followed a path up to the Shop menu item and scanned over the description. For subtask 5, the benchmark user, also participant 1033, fixated on the tab at the top for subtask 5, then checked that the total of the current order was R0 before going to the required Orders tab, as can be seen in Figure 5.3(b). For subtask 9, the benchmark user (participant 1032) fixated on the new order, fixated down the list to note that all the items had the same description, before fixating on the date of the first order, which was the item that had to be selected.

The benchmark deviation areas (BDA) are visual representations of areas on the user interface where the fixation data of the participant had a high deviation from the benchmark user fixation points. The clusters for data are drawn as a polygon on the user interface, each corner of the polygon representing a fixation point of the participant. Benchmark deviation vectors (BDV) represent a number of saccades repetitively moving in the same direction between or on components on the user interface. These vectors are represented by arrows to show the direction of the movement. Not all saccades are visualised, but only the ones that are repetitive and not eliminated by the benchmark user scan paths. The BDA and BDV visualisations will be considered and discussed together for each of the selected participants and each of the selected subtasks. Figures 5.4, 5.5 and 5.6 show the data for subtask Main T3, Category T3 and Orders respectively.

BDA and BDV for Main T3

According to Figure 5.4 all the participants fixated on all the titles of the menu items. Very few participants paid attention to the heading of the page, but more participants focussed on the bottom menu of the user interface. Participants 1003 (Figure 5.4(a) and 5.4(f)) and 1026 (Figure 5.4(d) and 5.4(i)), fixated on the menu titles a number of times, but did not read the descriptions in detail. Their saccades over the headings were much longer and more repetitive saccades than those of the other participants. The repetition could highlight that the participant was trying to make a decision between these elements. Participant 1008 (Figure 5.4(b) and 5.4(g)), 1012 (Figure 5.4(c) and 5.4(h)) and 1030 (Figure 5.4(e) and 5.4(j)) had more distributed fixations over the menu items, as can be seen by the smoother edges of the clusters, indicating more exploration.



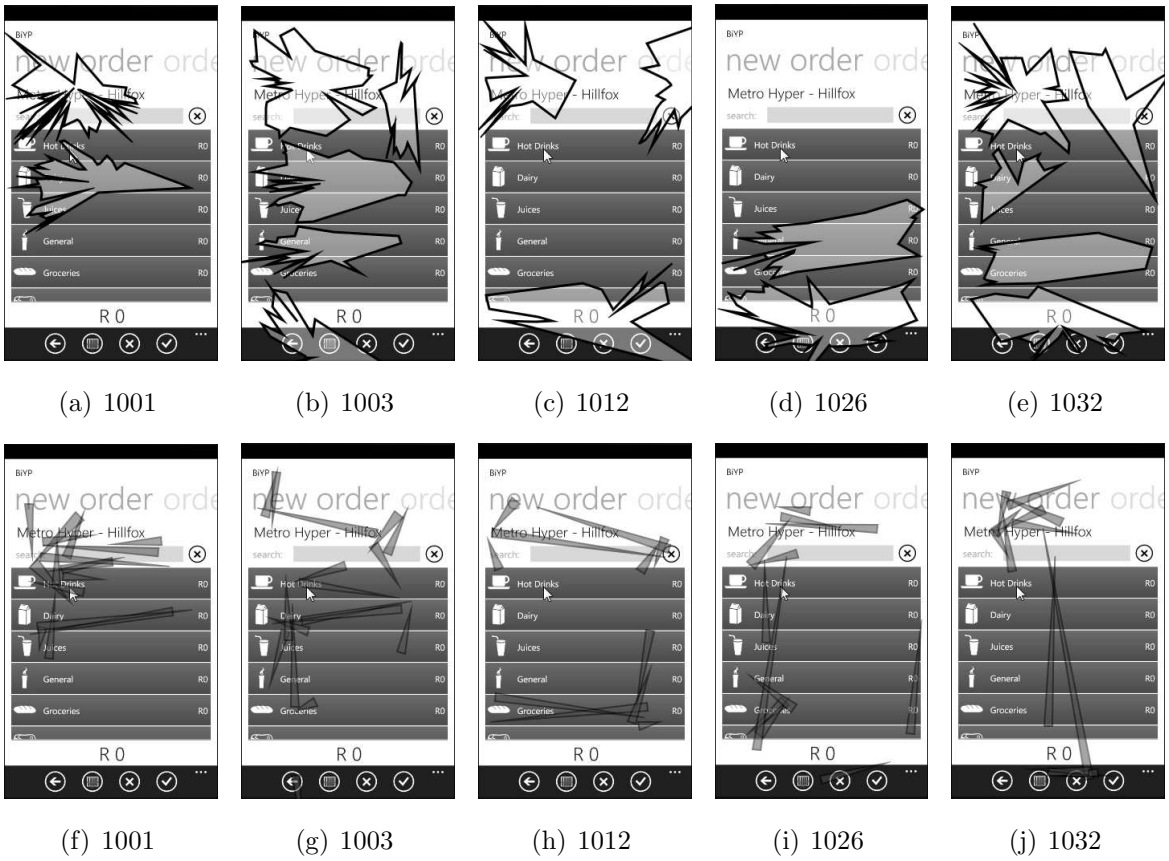
(a) 1003 (b) 1008 (c) 1012 (d) 1026 (e) 1030

(f) 1003 (g) 1008 (h) 1012 (i) 1026 (j) 1030

Figure 5.4: BDA [5.4(a) – 5.4(e)] and BDV [5.4(f) – 5.4(j)] of subtask Main T3 for selected participants.

These participants read the menu descriptions, as seen from the shorter saccade vectors moving from left to right over the descriptions.

Different approaches were taken by those to find the necessary information to efficiently complete the given task. Neither of these participants completed the task efficiently, as can be seen by the high deviation values. Even the bottom menu was explored to determine where to find the previous orders. Thus, the users could not effectively extract the necessary information from the user interface either by scanning the headings or exploring the descriptions of the menu items.



(a) 1001 (b) 1003 (c) 1012 (d) 1026 (e) 1032

(f) 1001 (g) 1003 (h) 1012 (i) 1026 (j) 1032

Figure 5.5: BDA [5.5(a) – 5.5(e)] and BDV [5.5(f) – 5.5(j)] of subtask Category T3 for selected participants.

BDA and BDV for Category T3

Considering all BDA visualisations in Figure 5.5, most deviation occurred in the header as well as the bottom menu. Many short saccades can be seen, from the BDVs in the header, highlighting how much the users searched in this area. All the participants showed repetitive scanning patterns over the header, with a noticeable backtracking saccade moving from right to left. Participant 1001 (Figure 5.5(a) and 5.5(f)) had low deviation in the bottom of the interface, whereas participant 1026 (Figure 5.5(d) and 5.5(i)) showed little deviation in the header. Even though participant 1026 did explore the header, the fixations were not wide spread and close to the benchmark user fixation. Some attention was paid to the category names, as seen by the number of fixations

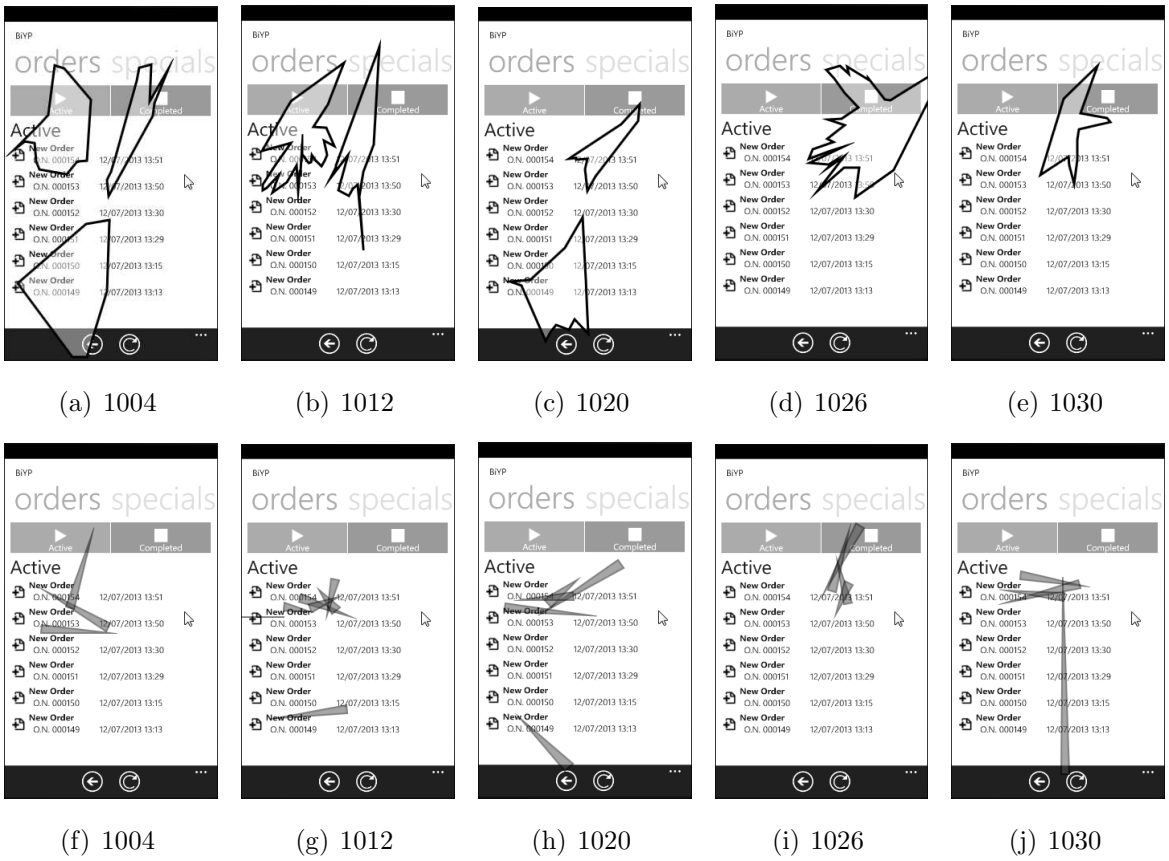
on the names by participants 1003 (Figure 5.5(b) and 5.5(g)) and 1026. Both of these participants as well as participant 1032 (Figure 5.5(e) and 5.5(j)) backtracked to the top menu from the bottom a number of times, indicating that they searched to the bottom and directly went back to the heading to find the Orders tab. The Orders tab is at the top right corner of the user interface. The jagged edges around the wholesaler's name showed a high number of fixations, as the participants found the name significant.

These participants had high deviation indices, indicating that they had trouble finding the Orders tab, which was necessary to complete the subtask. The amount of exploration on the interface can be seen from the visualisations. The users did expect to find the Orders tab in the header of the page, but the searching that occurred showed that it was not obvious. Even though the benchmark user did fixate on the header, the amount of exploration around those fixations resulted in high deviation in the area.

BDA and BDV for Orders

The BDAs for the Orders task in Figure 5.6 had smoother edges, showing less fixation points in each of the clusters, than the Main T3 and Category T3 BDAs. All of the participants, except for participant 1026 (Figure 5.6(i)) had saccades moving between the order number and the order timestamps, these long directed saccades showed meaningful eye movements as the participants were making sure that the top order was the latest one they placed. From the BDA (Figure 5.6(a) – 5.6(e)), smooth deviation edges can be seen for all the participants to the top menu, indicating a few fixations in the menu, but the BDVs indiate that most of the fixations in these clusters occurred between the order number and date. This can be seen from the horizontal BDV vectors (Figure 5.6(f), 5.6(g), 5.6(h) and 5.6(j)) with selected BDVs moving to or from the header of the of the page (see Figure 5.6(f), 5.6(h) and 5.6(i)).

From these visualisations it can be seen that the participants scanned the menu, but paid more attention to the list of orders to select the correct one. Less exploration occurred throughout this subtask as the participants could more easily have located the list item to select. The deviation indices showed little deviation, but very few saccades were eliminated. On further investigation it can be seen that the participants fixated on the relevant and expected areas on the interface, but the scan path of the benchmark user



(a) 1004 (b) 1012 (c) 1020 (d) 1026 (e) 1030

(f) 1004 (g) 1012 (h) 1020 (i) 1026 (j) 1030

Figure 5.6: BDA [5.6(a) – 5.6(e)] and BDV [5.6(f) – 5.6(j)] of subtask Orders for selected participants.

differed from most of the participants, which accounts for the low number of saccades eliminated for this subtask.

5.7 Data relevance

This section discusses the metrics resulting from the proposed approach with respect to time – a metric frequently used to determine the performance while completing a task. The relationship between each of the metrics and time are investigated to determine if the metrics provide similar or different information regarding time.

The number of fixations and the time spent on a task are some of the metrics that

would normally be considered during expert–based eye tracking usability evaluation. Even though time and number of fixations can be used to determine the performance, the resulting metrics from this approach provide additional information relative to the user interface.

The average time spent on a subtask and the number of fixations captured during each subtask are normalised to a range between zero and one and plotted on a line graph in Figure 5.7. This data was extracted from the raw data exported by means of Tobii Studio. On the same graph, the normalised averages of the FDI and $\text{SDI}^{-1}_{\text{eliminated}}$ metrics are also charted.

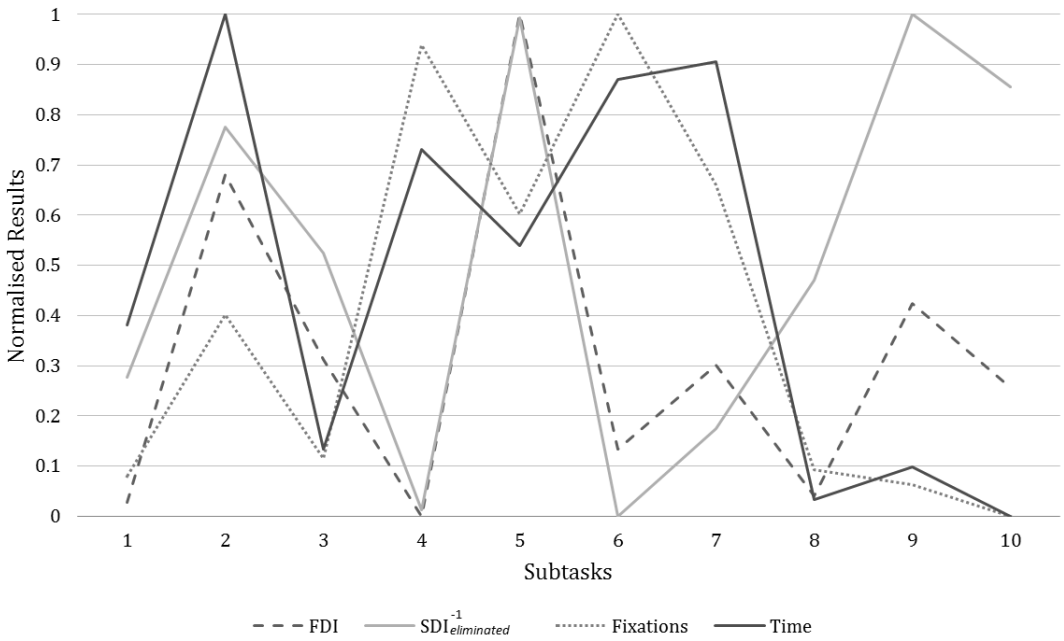
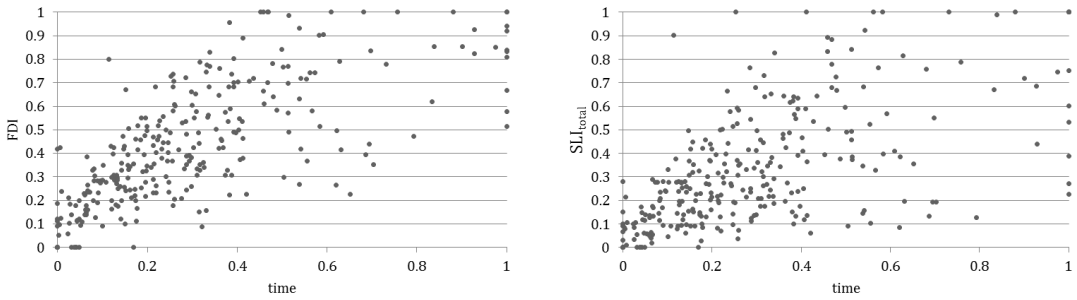


Figure 5.7: Comparison between raw data and processed FDI and SDI data.

This data shows that, if only the number of fixations and the time spent on a task were considered to decide which subtasks to investigate, the expert analyst would have investigated subtasks 2, 4, 6 and 7. The expert study in Section 3.6.5 highlights Task 3 (subtask 2, 5, 9 and 10) as the task with the most concerning usability issues. These are the same subtasks where the most deviation or variation occurred, as highlighted by the FDI and $\text{SDI}^{-1}_{\text{eliminated}}$ values exported by the automated method.

For each of the metrics from the proposed approach the Spearman's correlation coefficient was calculated in relation to time. For each subtask of the usability study, the metrics and time were normalised before the Spearman's correlation coefficient was calculated. Figure 5.8(a) and 5.8(b) show the normalised FDI and SLI_{total} metrics against the normalised time, for each subtask and each participant. Table 5.7 shows Spearman's correlation for each of the indices from the proposed approach.



(a) FDI/time correlation

(b) SLI_{total}/time correlation

Figure 5.8: The correlation between normalised FDI and SLI_{total} metrics and time normalised for every task and every participant.

Index	FDI	$SDI^{-1}_{eliminated}$	$SDI_{remainder}$	SLI_{total}	$SLI_{average}$
Spearman's correlation	0.75	0.21	0.77	0.63	0.02

Table 5.7: The Spearman's correlation coefficient of each index from the proposed method in relation with time.

The first performance measure is the FDI, which showed a strong correlation of 0.75 with relation to time. The strong correlation indicated that the FDI, just as time, was an indication of the performance of the participant. Considering Figure 5.8(a), the FDI value was not directly proportional to the time, there was some variation. The FDI value provided additional information on the behaviour of the users. Participants could have a high FDI value, but completed the task in a relatively short time, showing that the users were effective in completing the task, even though they deviated from the benchmark user. This could highlight more than one efficient way in which the task could be completed, depending on the user group.

There is a very weak correlation between the inverted $SDI^{-1}_{eliminated}$ and time, of only 0.21. The $SDI_{eliminated}$ value depicted the percentage of paths that the participant followed, which were similar to that of the benchmark user. Unlike the FDI, the $SDI_{eliminated}$ is not a measure of how much the participant deviated, but it is a measure of similarity. The $SDI_{remainder}$ depicted how much the scan paths of the participant differed from the benchmark user. This measure also has a strong correlation of 0.77 against time, which can also be used as a performance measure.

The last set of metrics is the saccade length. The SLI_{total} is the sum of the length of all the remaining saccades. This is another indication of how much the participants deviated in terms of distance, another performance measure as indicated by the strong correlation of 0.63 relative to time. The correlation is slightly lower than the FDI and $SDI_{remainder}$, because of the values being more disperse, see Figure 5.8(b). The $SLI_{average}$ has a very weak correlation, of 0.02, in relation to time, indicating that the $SLI_{average}$ is not a measure of performance. The $SLI_{average}$ is used to determine the type of eye movements that occurred, not how much the participant deviated.

In eye tracking usability studies, both the fixations and saccades are used as performance metrics. Even though they are both performance metrics, like time, different information can be extracted from these metrics. The correlations between some of the indices produced by the proposed method shows that they can be used as performance metrics, but they provide additional information concerning the amount of deviation. Despite the fact that the $SDI^{-1}_{eliminated}$ and $SLI_{average}$ does not have a strong correlation with time, these metrics provide valuable insight into the type of eye movement and the similarity of the scan path followed with relation to the benchmark user.

5.8 Expert analysis comparison

This section compares the findings from the proposed approach and that of an expert analyst. As discussed in Section 3.6.5, an expert analyst investigated the usability of the BiYP application, by watching replays of the eye tracking data captured during the usability study. The major usability issues identified by the expert are compared to the usability issues identified in the proposed approach when it was applied to the Validation

study.

The expert manually investigated the time it took to complete the tasks and identified Task 3 as a task that could be completed relatively quickly, but most participants took excessively long to complete the task. The same observation was made for the subtasks of Task 3 in the initial inspection of the proposed approach metrics, see Section 5.4.

The expert also identified three main usability issues while analysing the recorded eye tracking data. The one issue occurred because the usability test was not performed on a touch screen, as the system was designed to be used. This issue will not be taken into consideration. The remaining two usability issues relate to Task 3 and are compared to the BDA and BDV findings of the proposed approach as can be seen in Table 5.8.

Similar observations can be made from the proposed approach in the investigated tasks as from the expert review. The proposed approach also provides detail into the type of eye movement, such as searching, scanning and reading. It takes a fraction of the time to investigate a single image, produced by the proposed approach, compared to watching a replay of the eye tracking data several times. The metrics provided by the proposed approach also supply the necessary data to select participants to investigate further, compared to the participants who were selected randomly by the expert analyst to investigate. This supports the automated proposed approach as a feasible method for analysts to utilise and investigate eye tracking data.

UI	Expert Findings	Proposed Approach Findings
Main Menu	In order to view previous orders, the users were expected to select the 'Shop' item, which was counter–intuitive. The description of the menu item did not provide the necessary information to the users that the orders could be viewed by navigating to the 'Shop' service. The participants scanned up and down the main menu to find the correct item. One of the participants selected a help icon from the bottom menu, other participants just looked at the menu for assistance.	This usability issue relates directly to subtask Main T3, the subtask with the second highest deviation. Some participants repetitively scanned the headings of the menu items looking for the item to select, as the 'shop' description was not sufficient. Even though participants 1008, 1012 and 1030 read the menu items as can be seen by the BDVs, the high deviation indicated that the descriptions were not sufficient to complete the task efficiently. The bottom menu was considered by the participants to provide help after the titles and descriptions of the menu items were not sufficient. Different scan patterns could be seen between users who read the descriptions and those who just scanned the headings.
Orders Header	The participants had to navigate to the page by selecting 'orders' at the top, right of the screen. All of the participants eventually navigated to this page either on purpose or by accident. The 'order' title was very light and the word was cut–off. This made it very difficult for the participants to determine how they should navigate to the order history.	Subtask Category T3, the subtask with the highest deviation also reflected this usability issue. The participants deviated to the wholesaler name repetitively and considered the categories a few times, but most of their attention was on the top and bottom of the interface. Very little attention was paid to the 'order' title, thus the heading did not draw the attention of the users. There was excessive searching in the header of the interface, shown by the number BDVs that each highlight a repetitive path followed.

Table 5.8: Comparison of findings of the proposed automated method and the expert findings on the Validation study.

5.9 Conclusion

This chapter showed how the proposed approach can be applied, especially to larger datasets. By applying this method to the large dataset, usability issues could be identified from the eye tracking data, by automatically generating metrics and visualisations. The data comparison is numeric, which removes biased analysis, allowing analysts to identify tasks or subtasks where the participants show high deviation. The metrics also provide insight into the type of eye movement while completing a task, automatically providing rich numeric data for the analyst. This chapter highlighted the power of filtering through the data and focusing on problematic tasks, in order to save time in the analysis process. The visualisations provide user interface related information by visualising the fixation and scan path data of the selected tasks onto the relevant interface.

The correlation between time and the FDI, $SDI_{remainder}$ and the SLI_{total}, emphasises the use of the metrics produced by the proposed approach to be used not only to analyse the performance while completing a task, but also provides additional comparative information. The comparison of the main identified usability issues of the expert analyst investigation correspond to the findings produced by applying the proposed approach, which took a fraction of the time.

Chapter 6

Conclusion

The problem addressed by this study is the tremendous amount of time an expert needs to spend analysing eye tracking data and the amount of knowledge needed to identify usability issues. This chapter contains a summary of the main results as well as suggestions for some future direction regarding the current research.

6.1 Summary of findings

This research study was conducted with the aim to minimise the time spent and system knowledge required by an expert analyst to analyse eye tracking usability data – but still gain insight into the usability of the applications.

Knowledge was gained in the field of eye tracking and usability testing to form a good foundation and understanding of these two fields and how they can be used together to provide insight into usability testing. The research investigation then became more focused on the automation of eye tracking data analysis, to understand what has been done and to build on the work of other researchers. Various ways to visualise eye tracking data and visualisation design aspects were investigated to apply the knowledge acquired in this study.

Two usability studies were conducted in a lab–based environment capturing eye tracking data and an expert analysed the data. This was resource intensive and time–consuming work, but a number of usability issues were identified. During the devel-

opment of the proposed automated method, the expert findings served as a basis with which to compare the results of the Pilot study. This was done to ensure that the metrics and visualisations provided similar information. For the Validation study, the expert findings were used to investigate the results of the proposed automated methods when applied to a larger dataset.

The first objective was to develop a method to derive comparative metrics from eye tracking data. To achieve this, benchmark user eye tracking data was used as a baseline against which the eye tracking data of the other participants were compared. Resulting quantitative, comparative metrics indicated to what extent the participants' eye movement differed from the visual strategy of the benchmark user. These average metrics were used to filter through the eye tracking data to identify tasks where a lot of deviation occurred. Further investigation into the metrics provided information in the deviation and eye movements of specific users for a task. The same Pilot study task with usability issues, as identified by the expert analyst, was highlighted by the proposed automated method.

Selecting a benchmark user was the second objective. The benchmark user could either be a user proficient in the application at hand or one of the participants who showed the best performance throughout the usability study. The criteria according to which the benchmark user is selected should be predefined and is dependent on the software being tested. Multiple benchmark users can be used; one for each task or even a benchmark to represent each user group. For the two usability studies applied in this study, a benchmark user was selected per task and subtask, based on eye tracking and usability performance metrics.

The third objective was to visualise where on the user interface usability issues occurred. The fixation metric provided insight into the areas on the user interface where the deviation occurred and the saccade metrics indicated repetitive deviation paths followed between components. The visualisations were generated for tasks with high deviation indices. Only the fixation and saccade data where high deviation occurred was superimposed back onto the original user interface. Using these visualisations, similar usability issues were recognised as were identified by the expert analyst.

The last objective was to apply the proposed approach to a larger validation usability

study and investigate the feasibility. The results from an independent expert–based study was compared to the results from the Validation study. Similar tasks with usability issues were identified by the proposed method and the expert analyst. Furthermore, similar usability issues were deduced from the visualisations produced by the proposed method, as was identified by the expert analyst.

6.2 Conclusions

The following conclusions can be deduced from the findings:

- The eye tracking data of a benchmark user can be used as a baseline against which to compare eye tracking data of other users.
- In usability testing, a benchmark user who completed a task effectively and efficiently, can be utilised to automate the process of analysing eye tracking data.
- Deviation indices produced by the proposed approach highlight tasks with usability issues. It can be used by an expert to filter through large datasets and identify specific tasks as well as users that require further investigations, reducing the time that is required for an analyst to go through the entire eye tracking dataset.
- The resulting benchmark deviation indices and visualisations provide analysts with information that can be used to identify usability issues of an application in a reduced time frame.
- It is assumed that the benchmark user focuses on all the necessary components on the user interface to complete the task. This makes the proposed method user interface independent, minimising the knowledge the expert analyst needs with regard to the tested application.

6.3 Future work

The following section discusses a number of ways to extend the current work, as well as some opportunities to apply the work in other scenarios were identified.

6.3.1 Alternative benchmark users

The participant selected as the benchmark user, for comparison, has a direct effect on the results. Research has shown the difference between visual strategies of men and women [189]. Therefore, selecting an appropriate benchmark user (according to gender, age, or other demographic information) is crucial. An interesting study would be to note the effect of selecting a benchmark user for every demographic group in the study.

Studies have made use of benchmarks mapped out by expert analysts [98, 115, 178]. By adopting this strategy, it is possible to investigate the effect on the deviation indices, if an application expert manually maps out the expected visual strategy, which is then used as the benchmark user data.

In this study, most of the participants followed a different path from that of the benchmark user while completing subtask 9. For future work a feed backward system can be considered, similar to the strategy used by the WebRemUSINE tool [151]. A participant can be added as a benchmark user, who had similar visual strategies than the majority of other users and completed the task effectively and efficiently. For such a system multiple benchmark users can be considered based on the visual strategies followed by the users.

6.3.2 Extensions to mobile and dynamic user interface

The current method is designed for static screens as the eye movements are relative to the components on the user interface. Future work could investigate how to use a benchmark user to produce deviation indices on a dynamic user interface. One of the options is to create a database of fixations, the components focused on, and the time the fixation occurred, as was done with the WebEyeMapper tool [170]. The database will allow comparison of eye movement deviation relative to the components on the screen, overcoming the dynamic movements of the user interface.

The main reason why the mobile applications utilised in this study were evaluated through the use of an emulator, was because of the limitation of the available eye tracking devices for mobile phones. Screen capture of the mobile user interface was not available and would have been recorded by means of a video camera. The mobile device had to

be set up in a stationary position relative to the eye tracker, which did not allow for natural mobile usage. With the latest technology [60], this can be overcome by small infra–red eye trackers that can be attached to mobile devices. Another solution is to record the eye tracking data relative to the user interface components as was done with the WebEyeMapper tool.

6.3.3 Investigating the effect of parameters

The proposed method requires two threshold values for the SDI process and two additional values for the BDA and BDV visualisations. The first threshold is the saccade elimination threshold (t_e). If the start– and endpoint of a participant saccade is within the threshold of the benchmark saccade, it is assumed that the saccade moves between similar components and the participant saccade is eliminated. The effect of different t_e values are illustrated in Figure 4.11. The second threshold is the saccade clustering threshold (t_g). If the start– and endpoint of the saccades are within the threshold of each other, then the saccades are grouped together. This is to highlight repetitive paths that occur between components. These thresholds should be relative to the average component size on the user interface to ensure that the saccadic movements between components are captured and represented. Investigations into different t_e thresholds should be considered for each benchmark saccade, depending on size of the components at the endpoints of the saccade. Sadasivan et al. [181] visualised general paths followed between fixation clusters – a similar method can be considered for this clustering method, avoiding the use of the t_g threshold.

The last two parameters are related to the visualisations. The benchmark deviation areas utilise a weight (ω) that is multiplied with the average task FDI value. Consequently, a BDV cluster will be drawn if the average cluster FDI is above ωFDI. The weight affects the number of BDVs that will be drawn on the user interface. If the ω is too small, then too many clusters will be drawn and the areas with excessively high deviation will not be highlighted. The c_{min} is the last parameter, specifying the minimum number of saccades that should be in a cluster before a benchmark deviation vector is drawn for that cluster. The BDVs should represent repetitive paths followed between the components on the user interface. Fewer BDVs will be drawn if the c_{min} parameter

is very high. An appropriate size is dependent on the requirements of the study.

New or existing clustering and parameter optimisation algorithms could be considered to automatically determine the optimal thresholds, as input for the proposed approach.

6.3.4 Limitations

Some limitations have been identified during the development of the proposed method, like evaluating dynamic screens and the possibility that the benchmark user eye tracking data does not represent the majority of the visual strategies. Currently, the application generates clusters for all the benchmark fixations, before calculating the FDI values, even if more than one fixation is on a component. To overcome this, the fixations of the benchmark user can be clustered and the centroids of clusters can be used as the benchmark user fixations for the remainder of the FDI process (see Algorithm 2). Adjusting the clustering method of Santella and DeCarlo [187], to keep only the centroids of the cluster instead of resulting in clusters of fixations could be an appropriate clustering method.

There can also be some improvement on the visualisations produced by the proposed approach. The visualisations distinctly show the areas with high deviation and repetitive paths followed between components. In the Validation study the two visualisations were considered together during the analysis, but on individual representations. Combining the BDA and BDV visualisations into one design with an appropriate visualisation could be considered.

Another task that requires some additional work from the analyst, is to divide the data into segments that are specific to a user interface screen. This allows eye tracking comparison relative to users focussing on the same visual stimuli. The segmentation can be done by automatically or manually logging when navigation to different screens occur. The prototyping tool developed for this study can import these log files. To overcome the need to add logging functionality to an application or to manually separate the data, automated segmentation should be considered. Various eye tracking data segmentations were implemented by Holland et al. [86], an adoption of these segmentation algorithms can be considered to automate the data pre–processing of the proposed approach.

6.3.5 Application of other fields

The proposed approach should not be limited to analysing user interfaces but can be extended to other fields where eye tracking data is compared. This could be applied in expert–novice based eye tracking studies, to compare how much a novice differs from an expert. The areas where the novices deviated a lot from the benchmark user can also be superimposed on the visual stimuli.

These are just some ideas on how to build on the current work. The proposed method should not be limited to applications in the eye tracking field, but could be applied to data captured from other peripherals that capture human behaviour.

6.3.6 Develop open source tool

In order to enable this proposed method to be used by other researchers, the prototype tool, shown in Section A.1, should be developed into a complete solution and made available to researchers. With the finalisation of the tool, the visualisations can be extended to provide a additional views, such as a combined view of the BDA and BDV visualisations. The source code can also be made available to enable researchers to expand or adapt the current tool.

Bibliography

[1] B. Abbey. Instructional and Cognitive Impacts of Web–Based Education. pages 1–270. Idea Group Global Inc, Hershey, first edition, 1999.

[2] F. Adebesin and P. Kotzé. The Design of Application–Specific Heuristics for the Usability Evaluation of the Digital Doorway. *South African Computer Journal*, 48:9–30, 2012.

[3] D. L. Akers. *Backtracking Events as Indicators of Software Usability Problems*. Thesis, Stanford University, 2009.

[4] M. Albanesi, R. Gatti, M. Porta, and A. Ravarelli. Towards Semi–Automatic Usability Analysis Through Eye Tracking. In *Proceedings of the 12th International Conference on Computer Systems and Technologies*, pages 135–141, Vienna, June 2011. ACM.

[5] American National Standards for Information Technology. Common Industry Format for Usability Test Reports. Technical report, American National Standards Institute Inc., New York, 2001.

[6] T. S. Andre. *Determining the Effectiveness of the Usability Problem Inspector: A Theory–Based Model and Tool for Finding Usability Problems*. Thesis, Virginia Polytechnic Institute and State University, 2000.

[7] T. S. Andre, R. H. Hartson, and R. C. Williges. Determining the Effectiveness of the Usability Problem Inspector: A Theory–Based Model and Tool for Finding Usability Problems. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 45(3):455–482, 2003.

[8] T. Arh and B. J. Blažič. A Case Study of Usability Testing – the SUMI Evaluation Approach of the EducaNext Portal. *WSEAS Transactions on Information Science & Applications*, 5(2):175–181, 2008.

[9] F. T. W. Au, S. Baker, I. Warren, and G. Dobbie. Automated Usability Testing Framework. In *Proceedings of the Ninth Conference on Australasian User Interface*, volume 76, pages 55–64, Wollongong, Jan. 2008. Australian Computer Society Inc.

[10] J. S. Babcock, M. Lipps, and J. B. Pelz. How People Look at Pictures Before, During, and After Scene Capture: Buswell Revisited. In B. E. Rogowitz and T. N. Pappas, editors, *Proceedings of the Society of Photo–optical Instrumentation Engineers*, pages 34–47, San Jose, 2002. SPIE Inc.

[11] C. M. Barnum. Usability Testing Essentials: Ready, Set...Test! pages 1–408. Morgan Kaufmann, Burlington, first edition, 2010.

[12] B. Battleson, A. Booth, and J. Weintrop. Usability Testing of an Academic Library Website: a Case Study. *The Journal of Academic Librarianship*, 27(3):188–198, May 2001.

[13] R. Bednarik. *Methods to Analyze Visual Attention Strategies: Applications in the Studies of Programming*. Thesis, University of Joensuu, 2007.

[14] N. Bevan. Measuring usability as quality of use. *Software Quality Journal*, 4(2):115–130, June 1995.

[15] A. Bleicher. Eye–Tracking Software Goes Mobile. *IEEE Spectrum*, pages 9–10, May 2013.

[16] P. Blignaut, T. Beelders, J. Plessis, D. Wium, and R. Brown. Demystifying the Black Box: From Raw Data to Applications. In *Proceedings of the Conference on Eye Tracking South Africa*, pages 1–18, Cape Town, 2013. ACM.

[17] J. Bojko. Informative or Misleading? Heatmaps Deconstructed. In J. Jacko, editor, *Proceedings of the 13th International Conference on Human–Computer Interaction.*

Part I: New Trends, volume 5610, pages 30–39, San Diego, 2009. Springer Berlin Heidelberg.

[18] Z. Boraston and S. J. Blakemore. The Application of Eye–Tracking Technology in the Study of Autism. *The Journal of Physiology*, 581(3):893–898, June 2007.

[19] G. Brône, B. Oben, and T. Goedemé. Towards a More Effective Method for Analyzing Mobile Eye–Tracking Data: Integrating Gaze Data with Object Recognition Algorithms. In *Proceedings of the 1st International Workshop on Pervasive Eye Tracking and Mobile Eye–based Interaction*, pages 53–56, Beijing, 2011. ACM.

[20] G. Buscher, E. Cutrell, and M. Morris. What do You See when You're Surfing? Using Eye Tracking to Predict Salient Regions of Web Pages. In *Proceedings of the SIGCHI Conference on Human Computer Interaction*, pages 21–30, Boston, 2009. ACM.

[21] G. Buswell. *How People Look at Pictures: a Study of the Psychology and Perception in Art*. University of Chicago Press, Oxford, 1935.

[22] K. A. Butler. Usability Engineering Turns 10. *Interactions*, 3(January):58–75, Jan. 1996.

[23] M. D. Byrne, J. R. Anderson, S. Douglass, and M. Matessa. Eye tracking the visual search of click–down menus. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*, pages 402–409, Pittsburgh, May 1999. ACM.

[24] S. Card and T. Moran. User technology – from pointing to pondering. In *Proceedings of the ACM Conference on The history of personal workstations*, pages 183–198, New York, 1986. ACM.

[25] S. Card, T. Moran, and A. Newell. The Model Human Processor: An Engineering Model of Human Performance. In K. R. Boff, L. Kaufman, and J. P. Thomas, editors, *Handbook of perception and human performance*, volume 2, pages 41–45. Wiley Publishing, second edition, 1986.

[26] R. Carpenter. *Movement of the Eyes*. Pion Ltd, second edition, 1988.

[27] S. Cashmore. Adding Muscle to Mobile Apps. *ITWeb Brainstorm Magazine*, Apr. 2012.

[28] W. Chang, P. A. Shen, K. Ponnam, H. Barbosa, M. Chen, and S. Bermudez. WAYLA: Novel Gaming Experience Through Unique Gaze Interaction. In *ACM SIGGRAPH 2013 Emerging Technologies*, page 4503, Anaheim, July 2013. ACM.

[29] N. Charness, E. M. Reingold, M. Pomplun, and D. M. Stampe. The Perceptual Aspect of Skilled Performance in Chess: Evidence from Eye Movements. *Memory & Cognition*, 29(8):1146–1152, Dec. 2001.

[30] W. G. Chase and H. A. Simon. Perception in Chess. *Cognitive Psychology*, 4(1):55–81, 1973.

[31] P. Chynal, J. M. Szymaski, and J. Sobecki. Using Eyetracking in a Mobile Applications Usability Testing. *Intelligent Information and Database Systems*, 7198:178–186, 2012.

[32] T. Comber and J. R. Maltby. User Operations as Language Elements: Measuring Usability and User Competence through Redundancy. In *Proceedings of the 4th Annual Conference of the ACM Special Interest Group on Computer–Human Interaction*, pages 57–62, Dunedin, 2003. ACM.

[33] L. Cooke. Improving Usability Through Eye Tracking Research. In *International Professional Communication Conference Proceedings*, pages 195–198, Minneapolis, 2004. IEEE.

[34] L. Cooke. Is Eye Tracking the Next Step in Usability Testing? In *International Professional Communication Conference*, pages 236–242, Saratoga Springs, Oct. 2006. IEEE.

[35] C. Courage and K. Baxter. *Understanding Your Users: A Practical Guide to User Requirements Methods, Tools, and Techniques*. Morgan Kaufmann Publications, San Francisco, first edition, 2005.

[36] L. Cowen, L. J. Ball, and J. Delin. *An eye movement analysis of webpage usability.* Dissertation, Lancaster University, 2001.

[37] H. D. Crane and C. M. Steele. Generation–V dual–Purkinje–image eye tracker. *Applied Optics*, 24(4):527–537, 1985.

[38] J. A. de Bruin, K. M. Malan, and J. H. P. Eloff. Saccade Deviation Indicators for Automated Eye Tracking Analysis. In *Proceedings of Eye Tracking South Africa*, volume 1, pages 47–54, Cape Town, Aug. 2013. ACM.

[39] J. A. de Bruin, K. M. Malan, J. H. P. Eloff, and M. P. Zielinski. The Use of a Benchmark Fixation Deviation Index to Automate Usability Testing. In P. S. P. Gamito and P. J. Rosa, editors, *I See Me, You See Me: Inferring Cognitive and Emotional Processes from Gazing Behavior*, chapter six, pages 104–124. Cambridge Scholars Publishing, Lisboa, first edition, 2014.

[40] A. D. de Groot. Thought and Choice in Chess. In *Psychological Studies*, number 2 in Psychological Studies, pages 1–463. Walter de Gruyter, illustrate edition, 1978.

[41] E. B. Delabarre. A Method of Recording Eye–movements. *The American Journal of Psychology*, 9(4):572–574, 1898.

[42] L. Dell'Osso and R. Daroff. Eye Movement Characteristics and Recording Techniques. In J. S. Glaser, editor, *Neuro–ophthalmology*, chapter nine, pages 327–344. Lippincott Williams & Wilkins, Philadelphia, third edition, 1990.

[43] A. Dix, J. Finlay, G. D. Abowd, and R. Beale. *Human–Computer Interaction.* Pearson Education Limited, Essex, third edition, 2003.

[44] H. Drewes. *Eye Gaze Tracking for Human Computer Interaction.* Thesis, Ludwig Maximilians Universität, 2010.

[45] S. Duan, A. Kementsietsidis, K. Srinivas, and O. Udrea. Apples and Oranges: A Comparison of RDF Benchmarks and Real RDF Datasets. In *Proceedings of the SIGMOD International Conference on Management of Data*, pages 145–156, Athens, 2011. ACM.

[46] S. Dubey and A. Rana. Analytical Comparison of Usability Measurement Methods. *International Journal of Computer Applications*, 39(15):11–18, 2012.

[47] A. T. Duchowski. A Breadth–first Survey of Eye Tracking Applications. *Behavior Research Methods, Instruments, & Computers*, 34(4):455–470, Nov. 2002.

[48] A. T. Duchowski. *Eye Tracking Methodology: Theory and Practice*. Springer–Verlag, Secaucus, second edition, 2007.

[49] A. T. Duchowski, N. Cournia, and H. Murphy. Gaze–Contingent Displays: Review and Current Trends Model–Based Graphical Displays. *CyberPsychology and Behavior*, 7(6):621–635, 2004.

[50] A. T. Duchowski, J. Driver, S. Jolaoso, W. Tan, B. N. Ramey, and A. Robbins. Scanpath Comparison Revisited. In *Proceedings of the Symposium on Eye–Tracking Research & Applications*, volume 1, pages 219–226, Austin, 2010. ACM.

[51] J. Dumas and J. Redish. Introducing Usability Testing. In *A Practical Guide to Usability Testing*, chapter two, pages 1–404. Intellect Ltd, Exeter, first edition, 1999.

[52] J. S. Dumas and J. E. Fox. Usability Testing: Current Practice and Future Directions. In A. Sears and J. Jacko, editors, *The Human–Computer Interaction Handbook*, chapter 57, pages 1129–1149. Taylor and Francis, second edition, 2008.

[53] P. Eachus. The Use of Eye Tracking Technology in the Evaluation of e–Learning: A Feasibility Study. In *Proceedings of the 6th International Conference on Education in a Changing Environment*, number 1947, pages 239–247, Salford, 1998. Informing Science Press.

[54] M. R. Ebling and B. E. John. On the Contributions of Different Empirical Data in Usability Testing. In *Proceedings of the Conference on Designing Interactive Systems Processes, Practices, Methods, and Techniques*, pages 289–296, New York, 2000. ACM.

[55] C. Ehmke and S. Wilson. Identifying Web Usability Problems from Eye–Tracking Data. In *Proceedings of the 21st British HCI Group Annual Conference on People and Computers: HCI*, pages 119–128, Swinton, Sept. 2007. ACM.

[56] S. Ellis and R. Candrea. Windows to the Soul? What Eye Movements Tell Us About Software Usability. In *Proceedings of the 7th Annual Conference of the Usability Professionals Association*, pages 151–178, Washington DC, 1998. UPA Press.

[57] J. Eloff. Business in Your Pocket. *SAP Research Magazine*, (1):1–4, 2011.

[58] P. Fabo and R. Durikovic. Automated Usability Measurement of Arbitrary Desktop Application with Eyetracking. In *16th International Conference on Information Visualisation*, pages 625–629, Montpellier, July 2012. IEEE.

[59] L. Faulkner. Beyond the Five–user Assumption: Benefits of Increased Sample Sizes in Usability Testing. *Behavior Research Methods, Instruments, and Computers*, 35(3):379–383, Aug. 2003.

[60] O. Ferhat, F. Vilariño, and F. J. Sánchez. A Cheap Portable Eye–tracker Solution for Common Setups. *Journal of Eye Movement Research*, 7(3):1–10, 2014.

[61] X. Ferré, N. Juristo, U. P. D. Madrid, H. Windl, S. Ag, and L. Constantine. Usability Basics for Software Developers. *IEEE Software*, 18(1):22–29, 2001.

[62] P. M. Fitts, R. E. Jones, and J. L. Milton. Eye Movements of Aircraft Pilots During Instrument–landing Approaches. *Aeronautical Engineering Review*, 9(2):24–29, 1950.

[63] R. Fitzpatrick. Strategies for Evaluating Software Usability. *Computer Engineering Commons*, 353(1):1–10, 1998.

[64] E. Frø kjær, M. Hertzum, and K. Hornbæ k. Measuring Usability: are Effectiveness, Efficiency, and Satisfaction Really Correlated? In *Proceedings of the SIGCHI Conference on Compter Human Interaction*, volume 2, pages 345–352, Amsterdam, 2000. ACM.

[65] A. Gegenfurtner, E. Lehtinen, and R. Säljö. Expertise Differences in the Comprehension of Visualizations: a Meta–Analysis of Eye–Tracking Research in Professional Domains. *Educational Psychology Review*, 23(4):523–552, July 2011.

[66] H. Gelderblom. Project Rustica Evaluation: Analysis of Eye Tracking Data. Technical report, UNISA, Pretoria, South Africa, http://osprey.unisa.ac.za/TechnicalReports/h3.pdf, 2011.

[67] H. Gelderblom, J. A. de Bruin, and A. Singh. Three Methods for Evaluating Mobile Phone Applications Aimed at Users in a Developing Environment: a Comparative Case Study. In *Proceedings of the 3rd International Conference on Mobile Communication for Development*, volume 3, pages 321–334, New Delhi, Feb. 2012.

[68] J. Gips and P. Olivieri. EagleEyes: An Eye Control System for Persons with Disabilities. In *The Eleventh International Conference on Technology and Persons with Disabilities*, pages 1–15, Los Angeles, 1996.

[69] J. H. Goldberg and J. Helfman. Visual Scanpath Representation. In *Proceedings of the Symposium on Eye Tracking Research & Applications*, pages 203–210, Austin, 2010. ACM.

[70] J. H. Goldberg and J. I. Helfman. Scanpath Clustering and Aggregation. In *Proceedings of the Symposium on Eye Tracking Research & Applications*, pages 227–234, New York, 2010. ACM.

[71] J. H. Goldberg and X. P. Kotval. Eye Movement–based Evaluation of the Computer Interface. *Advances in Occupational Ergonomics and Safety*, 2:529–532, 1998.

[72] J. H. Goldberg and X. P. Kotval. Computer Interface Evaluation using Eye Movements: Methods and Constructs. *International Journal of Industrial Ergonomics*, 24(6):631–645, Oct. 1999.

[73] J. H. Goldberg, M. J. Stimson, and M. Lewenstein. Eye Tracking in Web Search Tasks: Design Implications. In *Proceedings of the Symposium on Eye Tracking Research & Applications*, pages 51–58, New Orleans, 2002. ACM.

[74] D. Hansen and Q. Ji. 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, 2010.

[75] H. Hartson, T. Andre, and R. Williges. Criteria for Evaluating Usability Evaluation Methods. *International Journal of Human–Computer Interaction*, 13(4):373–410, 2001.

[76] H. R. Hartson. Human–computer interaction: Interdisciplinary roots and trends. *Journal of Systems and Software*, 43(2):103–118, 1998.

[77] H. R. Hartson, J. C. Castillo, J. Kelso, and W. C. Neale. Remote Evaluation: The Network as an Extension of the Usability Laboratory. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 228–235, Vancouver, 1996. ACM.

[78] M. Hayhoe. Vision Using Routines: A Functional Account of Vision. *Visual Cognition*, 7(1–3):43–64, 2000.

[79] M. Hayhoe and D. Ballard. Eye Movements in Natural Behavior. *Trends in Cognitive Sciences*, 9(4):188–194, Apr. 2005.

[80] R. Haynes. Communications Infrastructure Fuelling African Revolution, http://www.itwebinformatica.co.za/index.php?option=com_content&view=article&id=4075:communications-infrastructure-fuelling-african-revolution&catid=84:communications-2012&Itemid=123, 2012.

[81] J. Heminghous and A. T. Duchowski. iComp: A Tool for Scanpath Visualization and Comparison. In *Symposium on Applied Perception in Graphics and Visualization*, volume 1, page 152, Boston, 2006. ACM.

[82] J. M. Henderson and A. Hollingworth. Eye Movements During Scene Viewing: An Overview. In G. Underwood, editor, *Eye Guidance in Reading and Scene Perception*, chapter twelve, pages 269–293. Elsevier, 1998.

[83] J. M. Henderson and A. Hollingworth. Eye Movements and Visual Memory: Detecting Changes to Saccade Targets in Scenes. *Perception & Psychophysics*, 65(1):58–71, 2003.

[84] G. Hervet, K. Guérard, S. Tremblay, and M. S. Chtourou. Is Banner Blindness Genuine? Eye Tracking Internet Text Advertising. *Journal of Applied Cognitive Psychology*, 25:708–716, 2011.

[85] D. Hilbert and D. F. Redmiles. Extracting usability information from user interface events. *ACM Computing Surveys*, 32(4):384–421, Dec. 2000.

[86] C. Holland, O. Komogortsev, and D. Tamir. Identifying Usability Issues via Algorithmic Detection of Excessive Visual Search. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 2943–2952, Austin, 2012. ACM.

[87] T. Hollingsed and D. G. Novick. Usability Inspection Methods After 15 Years of Research and Practice. In *Proceedings of the 25th Annual International Conference on Design of Communication*, pages 249–255, New York, New York, USA, 2007. ACM.

[88] K. Holmqvist, M. Nyström, and R. Andersson. *Eye Tracking: A Comprehensive Guide to Methods and Measures*. Oxford University Press, 1 edition, 2010.

[89] K. Holmqvist, M. Nyström, and F. Mulvey. Eye Tracker Data Quality: What It is and How to Measure It. In *Proceedings of the Symposium on Eye Tracking Research and Applications*, volume 1, pages 45–52, Santa Barbara, 2012. ACM.

[90] A. Holzinger. Usability Engineering Methods for Software Developers. *Communications of the ACM*, 48(1):71–74, 2005.

[91] A. Hornof, A. Cavender, and R. Hoselton. Eyedraw: A System for Drawing Pictures with Eye Movements. In *Proceedings of the 6th International Conference on Computers and Accessibility*, pages 86–93, Atlanta, 2004. ACM.

[92] H. Hua, P. Krishnaswamy, and J. P. Rolland. Video–based Eyetracking Methods and Algorithms in Head–mounted Displays. *Optics Express*, 14(10):4328–4350, May 2006.

[93] E. B. Huey. *The Psychology and Pedagogy of Reading: With a Review of the History of Reading and Writing and of Methods, Text, and Hygiene of Reading.* The Macmillan Company, New York, 1908.

[94] B. R. A. Hurley, A. Ouzts, J. Fischer, and T. Gomes. Effects of Private and Public Label Packaging on Consumer Purchase Patterns. *Packaging Technology and Science: An International Journal*, 23(January):399–412, 2013.

[95] W. Hwang and G. Salvendy. Number of People Required for Usability Evaluation. *Communications of the ACM*, 53(5):130–133, May 2010.

[96] S. T. Iqbal and B. P. Bailey. Using Eye Gaze Patterns to Identify User Tasks. In *The Grace Hopper Celebration of Women in Computing – Making History*, pages 1–6, Chicago, 2004. ACM.

[97] D. Irwin. Fixation Location and Fixation Duration as Indices of Cognitive Processing. In J. M. Henderson and F. Ferreira, editors, *Interface of Language, Vision and Action: Eye Movements and the Visual World*, number 217, chapter three, pages 105–144. Psychology Press, 2004.

[98] M. Ivory and M. Hearst. Comparing Performance and Usability Evaluation: New Methods for Automated Usability Assessment. Technical report, Namahn, www.namahn.com/resources/documents/note-eyetracking.pdf, 1999.

[99] M. Ivory and M. A. Hearst. The State of the Art in Automating Usability Evaluation of User Interfaces. *ACM Computing Surveys*, 33(4):470–516, Dec. 2001.

[100] R. J. K. Jacob. What You Look at is What You Get: Eye Movement–based Interaction Techniques. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 11–18, Seattle, 1990. ACM.

[101] R. J. K. Jacob and K. S. Karn. Eye Tracking in Human–Computer Interaction and Usability Research: Ready to Deliver the Promises. In J. Hyönä, R. Radach, and H. Deubel, editors, *In the Mind's Eye: Cognitive and Applied Aspects of Eye Movement Research*, chapter four, pages 573–605. Elsevier, Amsterdam, 2003.

[102] N. E. Jacobsen. *Usability Evaluation Methods: The Reliability and Usage of Cognitive Walkthrough and Usability Test*. Thesis, University of Copenhagen Denmark, 1999.

[103] H. Jarodzka, K. Holmqvist, and M. Nyström. A vector-based, multidimensional scanpath similarity measure. In *Proceedings of the 2010 Symposium on Eye-Tracking Research & Applications - ETRA '10*, volume 1, page 211, New York, New York, USA, 2010. ACM Press.

[104] M. W. M. Jaspers. A Comparison of Usability Methods for Testing Interactive Health Technologies: Methodological Aspects and Empirical Evidence. *International Journal of Medical Informatics*, 78:340–353, May 2009.

[105] J. Jeng. Usability Assessment of Academic Digital Libraries: Effectiveness, Efficiency, Satisfaction, and Learnability. *International Journal of Libraries and Information Services*, 55(2–3):96–121, 2005.

[106] Q. Ji, Z. Zhu, and P. Lan. Real–Time Nonintrusive Monitoring and Prediction of Driver Fatigue. *IEEE Transactions on Vehicular Technology*, 53(4):1052–1068, July 2004.

[107] S. A. Johansen, M. Tall, J. S. Agustin, and H. Skovsgaard. The Eye Tribe: Going Mobile... and Getting There, https://theeyetribe.com/android-mobile, 2013.

[108] S. Josephson and M. E. Holmes. Attention to Repeated Images on the World–Wide Web: Another Look at Scanpath Theory. *Behavior ResearchMethods, Instruments, & Computers*, 34(4):539–548, Nov. 2002.

[109] S. Josephson and M. E. Holmes. Visual Attention to Repeated Internet: Testing the Scanpath Theory on the World Wide Web. In *Proceedings of the Symposium on Eye Tracking Research & Applications*, pages 43–49, New Orleans, 2002. ACM.

[110] M. A. Just and P. A. Carpenter. Eye Fixations and Cognitive Processes. *Cognitive Psychology*, 8:441–480, Oct. 1976.

[111] B. K. Kahn, D. M. Strong, and R. Y. Wang. Information Quality Benchmarks: Product and Service Performance. *Communications of the ACM*, 45(4):184–192, 2002.

[112] P. Kasarskis, J. Stehwien, J. Hickox, A. Aretz, and C. Wickens. Comparison of Expert and Novice Scan Behaviors During VFR Flight. In *Proceedings of the 11th International Symposium on Aviation Psychology*, pages 1–6, Columbus, 2001.

[113] D. J. Kasik and H. G. George. Toward Automatic Generation of Novice User Test Scripts. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 244–251, Vancouver, 1996. ACM.

[114] D. E. Kieras. A Guide to GOMS Task Analysis. Technical report, IBM, Michigan, 1994.

[115] D. E. Kieras, S. D. Wood, K. Abotel, and A. Hornof. GLEAN: A Computer–Based Tool for Rapid GOMS Model Usability Evaluation of User Interface Designs. In *Proceedings of the 8th Annual Symposium on User Interface and Software Technology*, pages 91–100, Pittsburgh, 1995. ACM.

[116] B. Kirwan, A. Evans, L. Donohoe, A. Kilner, T. Lamoureux, T. Atkinson, and H. MacKendrick. Human Factors in the ATM System Design Life Cycle. In *FAA/Eurocontrol ATM R&D*, pages 1–21, Paris, 1997.

[117] O. Komogortsev and C. Holland. Aiding Usability Evaluation via Detection of Excessive Visual Search. In *Extended Abstracts on Human Factors in Computing Systems*, pages 1825–1830, Vancouver, 2011. ACM.

[118] O. Komogortsev, C. Mueller, D. Tamir, and L. Feldman. An Effort Based Model of Software Usability. In *International Conference on Software Engineering Theory and Practice*, pages 1–9, Orlando, 2009.

[119] O. V. Komogortsev, S. Jayarathna, D. H. Koh, and S. M. Gowda. Qualitative and Quantitative Scoring and Evaluation of the Eye Movement Classification Algorithms. In *Proceedings of the Symposium on Eye Tracking Research & Applications*, pages 65–68, Austin, 2010. ACM.

[120] R. B. Kulkarni and S. K. Dixit. Empirical and Automated Analysis of Web Applications. *International Journal of Computer Applications*, 38(9):1–8, 2012.

[121] M. Kumar, A. Paepcke, and T. Winograd. EyePoint: Practical Pointing and Selection Using Gaze and Keyboard. In *Conference on Computer–Human Interaction*, pages 1–10, San Jose, 2007. ACM.

[122] M. F. Land and M. Hayhoe. In What Ways Do Eye Movements Contribute to Everyday Activities? *Vision research*, 41:3559–3565, Jan. 2001.

[123] M. F. Land and D. N. Lee. Where We Look When We Steer. *Nature Journal*, 369(6483):742–744, 1994.

[124] M. F. Land, N. Mennie, and J. Rusted. The Roles of Vision and Eye Movements in the Control of Activities of Daily Living. *Perception*, 28(11):1311–1328, 1999.

[125] S. Lauesen. *User Interface Design: A Software Engineering Perspective*. Pearson/Addison–Wesley, Essex, first edition, 2005.

[126] B. Law and M. Atkins. Eye Gaze Patterns Differentiate Novice and Experts in a Virtual Laparoscopic Surgery Training Environment. In *Proceedings of the Symposium on Eye Tracking Research & Applications*, volume 1, pages 41–48, San Antonio, 2004. ACM.

[127] J. R. Leigh and D. S. Zee. *The Neurology of Eye Movements*. Oxford University Press, Michigan, 2006.

[128] J. L. Levine. An Eye–controlled Computer. Technical report, IBM Research Division, TJ Watson Research Center, New York, 1981.

[129] D. Li, J. Babcock, and D. J. Parkhurst. openEyes: a Low–cost Head–mounted Eye–tracking Solution. In *Proceedings of the Symposium on Eye Tracking Research & Applications*, pages 95–100, San Diego, 2006. ACM.

[130] X. Li, A. Çöltekin, and M. Kraak. Visual Exploration of Eye Movement Data Using the Space–Time–Cube. *Geographic Information Science*, 6292:295–309, 2010.

[131] S. Liversedge and J. Findlay. Saccadic Eye Movements and Cognition. *Trends in Cognitive Sciences*, 4(1):6–14, Jan. 2000.

[132] G. Loftus and N. Mackworth. Cognitive Determinants of Fixation Location During Picture Viewing. *Journal of Experimental Psychology: Human Perception and Performance*, 4(4):565–572, 1978.

[133] P. Majaranta, A. Aula, and K. Räihä. Effects of Feedback on Eye Typing with a Short Dwell Time. In *Proceedings of the Symposium on Eye Tracking Research & Applications*, pages 139–146, San Antonio, 2004. ACM.

[134] M. Manhartsberger and N. Zellhofer. Eye Tracking in Usability Research: What Users Really See. In A. Holzinger and K. H. Weidmann, editors, *Empowering Software Quality: How Can Usability Engineering Reach These Goals?*, volume 198, pages 141–152. OCG Publication, Vienna, 2005.

[135] F. Mars and J. Navarro. Where We Look When We Drive with or without Active Steering Wheel Control. *PLoS ONE*, 7(8):1–6, 2012.

[136] D. J. Mayhew. The Usability Engineering Lifecycle. In *Extended Abstracts on Human Factors in Computer Systems*, pages 147–148, Pittsburgh, May 1999. ACM.

[137] S. Milekic. The More You Look the More You Get: Intention–based Interface using Gaze–tracking. In D. Bearman and J. Trant, editors, *Proceedings of Museums and the Web*, pages 1–27, Charlotte, Mar. 2003. Archives & Museum Informatics.

[138] E. Miluzzo, T. Wang, and A. T. Campbell. EyePhone: Activating Mobile Phones with Your Eyes. In *Proceedings of the Second ACM SIGCOMM Workshop on*

Networking, Systems, and Applications on Mobile Handhelds, pages 15–20, New Delhi, 2010. ACM.

[139] D. Miniotas, O. Špakov, I. Tugoy, and I. S. Mackenzie. Speech–Augmented Eye Gaze Interaction with Small Closely Spaced Targets. In *Proceedings of Eye Tracking Research and Applications*, pages 27–29, San Diego, Mar. 2006. ACM.

[140] C. Morimoto and M. Mimica. Eye Gaze Tracking techniques for Interactive Applications. *Computer Vision and Image Understanding*, 98(1):4–24, 2005.

[141] Namahn. Using Eye Tracking for Usability Testing Eye–tracking Technology. Technical report, Namahn – Human–centered Design Agency, Brussels, 2001.

[142] J. Nielsen. Finding Usability Problems Through Heuristic Evaluation. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 373–380, Monterey, 1992. ACM.

[143] J. Nielsen. *Usability Engineering*. Morgan Kaufmann, San Francisco, 1 edition, 1993.

[144] J. Nielsen. Usability Inspection Methods. In *Conference Companion on Human Factors in Computing Systems*, pages 413–414, Boston, 1994. ACM.

[145] J. Nielsen. Usability 101: Introduction to Usability. *Nielsen Norman Group*, Jan. 2012.

[146] J. Nielsen and K. Pernice. *Eyetracking Web Usability*. New Riders, 2010.

[147] J. Nielsen and V. Phillips. Estimating the Relative Usability of Two Interfaces: Heuristic, Formal, and Empirical Methods Compared. In *Proceedings of the Conference on Human Factors in Computing Systems*, pages 214–221, Amsterdam, 1993. ACM.

[148] K. L. Norman and R. La. Levels of Automation and User Participation in Usability Testing. *Interacting with Computers*, 18(2):246–264, 2006.

[149] A. North. Accuracy and Precision of Electro–oculographic Recording. *Investigative Ophthalmology & Visual Science*, 4(3):343–348, 1965.

[150] M. Obrist, R. Bernhaupt, E. Beck, and M. Tscheligi. Focusing on Elderly: An iTV Usability Evaluation Study with Eye–tracking. In P. César, K. Chorianopoulos, and J. F. Jensen, editors, *European Conference on Interactive TV*, volume 4471, pages 66–75, Amsterdam, May 2007. Springer–Verlag.

[151] L. Paganelli and F. Paternò. Intelligent Analysis of User Interactions with Web Applications. In *Proceedings of the International Conference on Intelligent User Interfaces*, pages 111–118, San Francisco, Jan. 2002. ACM.

[152] F. Papenmeier and M. Huff. DynAOI: A Tool for Matching Eye–movement Data with Dynamic Areas of Interest in Animations and Movies. *Behavior research methods*, 42(1):179–187, Feb. 2010.

[153] S. Pekkala. *Usability Evaluation of Design Solutions for Tablet Magazines*. Thesis, Aalto University, 2012.

[154] J. B. Pelz, R. L. Canosa, D. Kucharczyk, J. S. Babcock, A. Silver, and D. Konno. Portable Eyetracking: A Study of Natural Eye Movements. In B. E. Rogowitz and T. N. Pappas, editors, *Human Vision and Electronic Imaging*, pages 566–582, San Jose, Jan. 2000. SPIE Inc.

[155] P. G. Polson, C. Lewis, J. Rieman, and C. Wharton. Cognitive Walkthroughs: A Method for Theory–based Evaluation of User Interfaces. *International Journal of Man–Machine Studies*, 36(5):741–773, May 1992.

[156] A. Poole and L. J. Ball. Eye Tracking in Human–Computer Interaction and Usability Research: Current Status and Future Prospects. In C. Ghaoui, editor, *Encyclopedia of Human Computer Interaction*, volume 10, pages 211–219. Idea Group Inc., 2005.

[157] J. Preece. Sociability and Usability in Online Communities: Determining and Measuring Success. *Behaviour & Information Technology Journal*, 20(5):347–356, Jan. 2001.

[158] J. Preece, Y. Rogers, and H. Sharp. Interaction Design: Beyond Human–Computer Interaction. In *Interaction Design: Beyond Human–Computer Interaction*, pages 1–519. John Wiley & Sons, second edition, 2007.

[159] W. J. Price. A Benchmark Tutorial. *IEEE Micro*, 9(5):28–43, 1989.

[160] C. M. Privitera and L. W. Stark. Algorithms for Defining Visual Regions–of–Interest: Comparison with Eye Fixations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(9):970–982, 2000.

[161] D. Purve, G. J. Augustine, D. Fitzpatrick, L. C. Katz, A. S. LaMantia, J. O. McNamara, and S. M. Williams. Types of Eye Movements and Their Functions. In *Neuroscience*, chapter twenty, page 457. Sinauer Associates, Sunderland, second edition, 2001.

[162] K. Räihä, A. Aula, P. Majaranta, H. Rantala, and K. Koivunen. Static Visualization of Temporal Eye–tracking Data. In M. Costabile and F. Paternò, editors, *IFIP Conference on Human–Computer Interaction*, volume 3585, pages 946–949, Lisbon, 2005. Springer–Verlag.

[163] G. Rakoczi and M. Pohl. Visualisation and Analysis of Multiuser Gaze Data: Eye Tracking Usability Studies in the Special Context of E–learning. In *12th International Conference on Advanced Learning Technologies*, pages 738–739, Rome, July 2012. IEEE.

[164] R. Ramloll and C. Trepagnier. Gaze Data Visualization Tools: Opportunities and Challenges. In *Proceedings of the 8th International Conference on Information Visualisation*, pages 173–180, London, July 2004. IEEE.

[165] K. Rayner. Eye Movements in Reading and Information Processing. *Psychological Bulletin*, 85(3):618–660, May 1978.

[166] K. Rayner. Eye movements in Reading and Information Processing: 20 Years of Research. *Psychological Bulletin*, 124(3):372–422, 1998.

[167] K. Rayner, B. Miller, and C. M. Rotello. Eye Movements When Looking at Print Advertisements: The Goal of the Viewer Matters. *Applied Cognitive Psychology*, 22:697–707, 2008.

[168] K. Rayner, C. M. Rotello, A. J. Stewart, J. Keir, and S. A. Duffy. Integrating Text and Pictorial Information: Eye Movements when Looking at Print Advertisements. *Journal of Experimental Psychology: Applied*, 7(3):219–226, 2001.

[169] R. Razeghi. *Usability of Eye Tracking as a User Research Technique in Geo–information Processing and Dissemination.* Dissertation, University of Twente, 2010.

[170] R. W. Reeder, P. Pirolli, and S. Card. WebEyeMapper and WebLogger: Tools for Analyzing Eye Tracking Data Collected in Web–use Studies. In *Extended Abstracts on Human Factors in Computing Systems*, pages 19–20, Seattle, 2001. ACM.

[171] E. D. Reichle, A. Pollatsek, D. L. Fisher, and K. Rayner. Toward a Model of Eye Movement Control in Reading. *Psychological Review*, 105(1):125–157, Jan. 1998.

[172] E. M. Reingold, N. Charness, M. Pomplun, and D. M. Stampe. Visual Span in Expert Chess Players: Evidence From Eye Movements. *Journal of American Psycological Society*, 12(1):48–55, 2001.

[173] J. A. Renshaw and N. Webb. Eye Tracking in Practice. In *British HCI Group Annual Conference on People and Computers: HCI*, pages 239–241, University of Lancaster, Sept. 2007. British Computer Society.

[174] S. Riihiaho. *Experiences with Usability Evaluation Methods.* Thesis, Helsinki University of Technology, 2000.

[175] K. Rodden, H. Hutchinson, and X. Fu. Measuring the User Experience on a Large Scale: User–Centered Metrics for Web Applications. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 2395–2398, Atlanta, 2010. ACM.

[176] K. Rogers, editor. *The Eye: The Physiology of Human Perception*. Rosen Education Service, New York, first edition, 2010.

[177] R. G. Ross, A. Olincy, J. G. Harris, B. Sullivan, and A. Radant. Smooth Pursuit Eye Movements in Schizophrenia and Attentional Dysfunction: Adults with Schizophrenia, ADHD, and a Normal Comparison Group. *Biological psychiatry*, 48(3):197–203, Aug. 2000.

[178] J. Rubin and D. Chisnell. *Handbook of Usability Testing: How to Plan, Design, and Conduct Effective Tests*. John Wiley & Sons Ltd, Indianapolis, second edition, 2008.

[179] B. M. C. Russell. Hotspots and Hyperlinks: Using Eye–tracking to Supplement Usability Testing. *Usability News*, 7(2):1–11, 2005.

[180] W. Ryan. *Limbustrack: Stable Eye–tracking in Imperfect Light Conditions*. Thesis, Clemson University, 2007.

[181] S. Sadasivan, J. S. Greenstein, A. K. Gramopadhye, and A. T. Duchowski. Use of Eye Movements as Feedforward Training for a Synthetic Aircraft Inspection Task. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 141–149, Portland, 2005. ACM.

[182] G. Salvendy. *The Human–Computer Interaction Handbook*. Human Factors and Ergonomics. Taylor & Francis, New York, second edition, Sept. 2007.

[183] D. D. Salvucci. An Interactive Model–based Environment for Eye–movement Protocol Analysis and Visualization. *Proceedings of the Eye Tracking Research and Applications*, pages 57–63, 2000.

[184] D. D. Salvucci and J. Anderson. Automated Eye–Movement Protocol Analysis. *Human–Computer Interaction*, 16:39–86, 2001.

[185] D. D. Salvucci and J. Goldberg. Identifying Fixations and Saccades in Eye–tracking Protocols. In *Proceedings of the Symposium on Eye Tracking Research & Applications*, pages 71–78, Palm Beach Gardens, 2000. ACM.

[186] A. Santella and D. DeCarlo. Abstracted Painterly Renderings Using Eye–Tracking Data. In *Proceedings of the 2nd International Symposium on Non–photorealistic Animation and Rendering*, pages 75–82, Annecy, 2000. ACM.

[187] A. Santella and D. DeCarlo. Robust Clustering of Eye Movement Recordings for Quantification of Visual Interest. In *Proceedings of the Symposium on Eye Tracking Research & Applications*, pages 27–34, San Antonio, 2004. ACM.

[188] J. Sauro and E. Kindlund. A Method to Standardize Usability Metrics into a Single Score. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 401–409, Portland, 2005. ACM.

[189] M. Schiessl, S. Duda, A. Thölke, and R. Fischer. Eye Tracking and its Application in Usability and Media Research. *MMI Interaktiv – Eye Tracking*, 1(6):41–50, 2003.

[190] J. Scholtz. Common Industry Format for Usability Test Reports. In *Extended Abstracts on Human Factors in Computing Systems*, pages 301–301, The Hague, 2000. ACM.

[191] J. Scholtz, S. Laskowski, and L. Downey. Developing Usability Tools and Techniques for Designing and Testing Web Sites. In *Proceedings of the 4th Conference on Human Factors & the Web*, pages 1–10, Basking Ridge, 1998. AT&T Labs.

[192] M. Sendín, J. J. Rodríguez, and C. Cuadrat. Validating a Method for Quantitative Mobile Usability Testing Based on Desktop Eyetracking. In V. M. Penichet, A. Peñalver, and J. A. Gallud, editors, *New Trends in Interaction, Virtual Reality and Modeling*, pages 1–17. Springer–Verlag, London, 2013.

[193] SensoMotoric Instruments. BeGaze User Manual. Technical Report 2.4, SMI, number 2.4, Feb. 2010.

[194] B. Shackel. Pilot Study in Electro–oculography. *The British Journal of Ophthalmology*, 44(2):89–113, 1960.

[195] P. Shilane, P. Min, M. Kazhdan, and T. Funkhouser. The Princeton Shape Benchmark. In *Shape Modeling Applications*, pages 167–178, Genova, 2004. IEEE.

[196] J. Sibert and M. Gokturk. The Reading Assistant: Eye Gaze Triggered Auditory Prompting for Reading Remediation. In *Proceedings of the 13th Annual ACM Symposium on User Interface Software and Technology*, pages 101–107, San Diego, 2000. ACM.

[197] E. Siegenthaler, P. Wurtz, and R. Groner. Improving the Usability of e–book Readers. *Journal of Usability Studies*, 6(1):25–38, 2010.

[198] H. Skovsgaard. *Noise Challenges in Monomodal Gaze Interaction*. Thesis, IT University of Copenhagen, 2011.

[199] P. A. Smith. Towards a Practical Measure of Hypertext Usability. *Interacting with Computers*, 8(4):365–381, Dec. 1996.

[200] R. S. Snell and M. A. Lemp. *Clinical Anatomy of the Eye*. Wiley–Blackwell, New York, second edition, 1997.

[201] M. L. Spezio, R. Adolphs, R. S. E. Hurley, and J. Piven. Abnormal Use of Facial Information in High–Functioning Autism. *Journal of Autism and Developmental Disorders*, 37(5):929–939, May 2007.

[202] J. Spool, T. Scanlon, C. Snyder, W. Schroeder, and T. DeAngelo. *Web Site Usability: A Designer's Guide*. Morgan Kaufmann, first edition, 1998.

[203] S. Ssemugabi. *Usability Evaluation of Web–based e–learning Applications: A Study of two Evaluation Methods*. Dissertation, University of South Africa, 2006.

[204] A. Straube and U. Büttner. *Neuro–ophthalmology: Neuronal Control of Eye Movements*, volume 40. Krager, 2007.

[205] T. Tiedtke, C. Märtin, and N. Gerth. AWUSA – A Tool for Automated Website Usability Analysis. In P. Forbrig, editor, *PreProceedings of the 9th International Workshop on the Design, Specification and Verification of Interactive Systems*, pages 1–15, Rostock, June 2002. University of Rostock.

[206] Tobii Technology AB. Tobii Eye Tracking: An Introduction to Eye Tracking and Tobii Eye Trackers. Technical report, Tobii Technology AB, Jan. 2010.

[207] Tobii Technology AB. Using Eye Tracking to Test Mobile Devices. Technical report, Tobii Technology AB, Feb. 2010.

[208] Tobii Technology AB. Tobii Studio 2.2 User Manual. Technical report, Tobii Technology AB, 2012.

[209] M. J. Tovée. *An Introduction to the Visual System*, volume 74. Cambridge University Press, illustrate edition, 1996.

[210] N. Tractinsky and M. Hassenzahl. Arguing for Aesthetics in Human–Computer Interaction. *i–com*, 4(3):66–68, 2005.

[211] D. Travis. *ISO 9241 for Beginners*. Userfocus Ltd., London, ninth edition, 2013.

[212] H. Y. Tsang, M. Tory, and C. Swindells. eSeeTrack – Visualizing Sequential Fixation Patterns. *IEEE Transactions on Visualization and Computer Graphics*, 16(6):953–962, 2010.

[213] P. H. Tseng, I. G. M. Cameron, G. Pari, J. N. Reynolds, D. P. Munoz, and L. Itti. High–throughput Classification of Clinical Populations from Natural Viewing Eye Movements. *Journal of Neurology*, 260:275–284, Jan. 2013.

[214] T. Tullis and W. Albert. *Measuring the User Experience: Collecting, Analyzing, and Presenting Usability Metrics*. Morgan Kaufmann, first edition, 2008.

[215] M. A. Vernon. Methods of Recording Eye Movements. *The British Journal of Ophthalmology*, 12(3):113–130, 1928.

[216] O. Špakov and D. Miniotas. Visualization of Eye Gaze Data using Heat Maps. *Electronics and Electrical Engineering*, 2(74):55–58, 2007.

[217] N. Wade and B. Tatler. Did Javal Measure Eye Movements During Reading? *Journal of Eye Movement Research*, 2(5):1–7, 2009.

[218] N. J. Wade. Pioneers of Eye Movement Research. *i–Perception*, 1:33–68, 2010.

[219] N. J. Wade and B. Tatler. *The Moving Tablet of the Eye: The Origins of Modern Eye Movement Research*. Oxford University Press, first edition, 2005.

[220] N. J. Wade, B. W. Tatler, and D. Heller. Dodge–ing the Issue: Dodge, Javal, Hering, and the Measurement of Saccades in Eye–movement Research. *Perception*, 32(7):793–804, 2003.

[221] M. Wattenberg and F. B. Viégas. The Word Tree, an Interactive Visual Concordance. *IEEE Transactions on Visualization and Computer Graphics*, 14(6):1221–1228, 2008.

[222] J. M. West, A. R. Haake, E. P. Rozanski, and K. S. Karn. eyePatterns: Software for Identifying Patterns and Similarities Across Fixation Sequences. In *Proceedings of the Symposium on Eye Tracking Research & Applications*, pages 149–154, San Diego, 2006. ACM.

[223] R. West and K. Lehman. Automated Summative Usability Studies: An Empirical Evaluation. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 631–639, Montréal, 2006. ACM.

[224] M. E. Wiklund, editor. *Medical Device and Equipment Design: Usability Engineering and Ergonomics*. CRC Press, Boca Raton, first edition, 1995.

[225] A. Wilbik and J. Kacprzyk. On a Benchmark Related Assessment of the Performance of Mutual (Investment) Funds. In *IEEE International Conference on Fuzzy Systems FUZZ*, pages 2934–2939, Taipei, June 2011. Systems Research Institute, Polish Academy of Sciences, Warsaw, Poland, IEEE.

[226] J. O. Wobbrock, J. Rubinstein, M. Sawyer, and A. T. Duchowski. Not Typing but Writing: Eye–based Text Entry Using Letter–like Gestures. In *The 3rd Conference on Communication by Gaze Interaction*, pages 4–6, Leicester, Sept. 2007.

[227] D. S. Wooding. Fixation Maps: Quantifying Eye-movement Traces. In *Proceedings of the 2002 symposium on Eye tracking Research & Applications*, pages 31–36, New Orleans, 2002. ACM.

[228] L. R. Young and D. Sheena. Survey of eye movement recording methods. *Behavior Research Methods & Instrumentation*, 7(5):397–429, 1975.

[229] S. Zhai, C. Morimoto, and S. Ihde. Manual and Gaze Input Cascaded (MAGIC) Pointing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 246–253, Pittsburgh, 1999. ACM.

Appendix A

Validation Study Output

A.1 Process prototype tool

In order to test the proposed automated approach, the functionality had to be implemented. For the purpose of this study, the Automated Eye Tracking Analysis (Process) Prototype was developed in C# (see Figure A.1). This prototype provides the functionality to load the raw eye tracking and event data and apply all the necessary data pre–processing. From there the prototype can be used to automatically select the benchmark user for each task to be used in the FDI and SDI processes. Each step of the process can be applied sequentially and the relevant data output is displayed at the bottom.

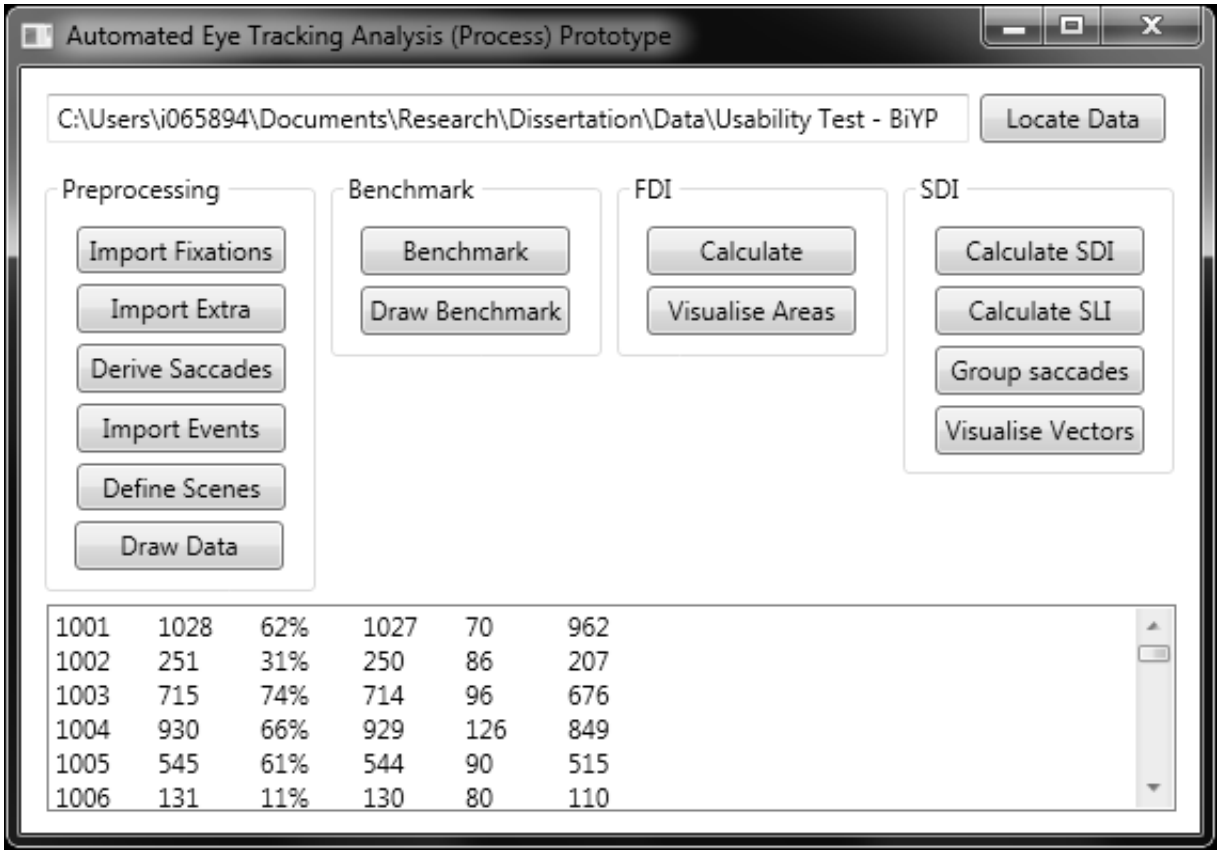


Figure A.1: Screen–shot of the process prototype tool to automate eye tracking analysis

A.2 Benchmark user table

Part	Accu	Main T1		Main T3		Supplier	Category T1		Category T3		Products		Product		Confirm		Orders		Invoice	
1001	62%	20	33914	60	144353	32	12479	219	83908	143	50512	184	77874	149	92957	83	37149	40	16955	32	12126
1002	31%	26	24872	46	560930	10	11573	26	54123	56	30961	11	60246	0	84120	0	5284	27	17015	5	9582
1003	74%	32	34317	116	107475	74	39445	78	35795	146	76972	87	46069	105	72355	16	4549	17	12735	5	3339
1004	66%	25	29357	34	46512	28	18408	169	88976	117	52290	280	183630	120	99026	25	11014	35	11360	16	6830
1005	61%	16	18750	51	31736	26	14340	129	43870	48	21124	102	40158	108	81386	20	7252	9	10544	6	2406
1006	11%	6	25472	17	31550	2	8810	37	55822	0	46552	18	53684	26	42738	4	4088	0	16230	0	5982
1007	73%	20	36799	65	49418	25	16344	177	87286	87	36378	60	37558	80	40896	25	10140	20	7952	26	14018
1008	55%	33	31555	100	71376	24	9654	124	50978	79	35532	240	105068	76	47656	23	12342	27	9830	11	4392
1009	99%	13	35491	76	52716	13	7140	49	24262	17	6436	43	24102	46	32636	37	13944	12	7960	13	4478
1010	72%	17	19054	14	31224	19	11640	60	27944	23	16796	50	24000	48	28592	16	6188	8	11536	9	3572
1011	59%	22	27750	97	61080	77	28892	185	60297	61	38969	148	65083	112	66835	21	7701	29	13824	14	7010
1012	41%	31	36176	99	59708	43	14860	162	61520	92	47896	102	51856	68	61148	33	10652	64	30764	14	5136
1013	91%	11	18604	38	32448	15	6503	41	18626	27	9114	92	43053	26	22464	10	3492	22	20899	6	2526
1014	97%	12	18842	15	7153	5	7907	41	22432	5	4446	52	31725	31	19013	11	5115	12	11342	7	4502
1015	31%	5	14919	27	42254	9	9151	21	36418	37	39846	23	29222	14	44631	6	5029	7	18820	0	7066
1016	95%	18	24510	23	20169	11	8447	88	39631	36	14008	96	50599	66	48512	20	7520	14	13117	17	7876
1017	69%	35	26372	30	24480	30	13869	65	31720	105	43887	88	38297	32	35783	18	6416	26	9229	37	14964
1018	47%	17	19501	56	61046	37	19144	143	36408	97	53199	133	39136	191	63459	44	6009	32	14512	14	5176
1019	3%	20	36441	0	34715	2	25073	10	46843	0	47698	13	47238	2	83549	20	14715	0	10778	0	6041
1020	94%	12	10855	25	16206	6	6557	45	20563	38	14895	75	39141	59	35031	16	6592	33	14038	16	5503
1021	94%	10	12842	21	16388	27	14644	33	18881	12	6414	55	24661	39	31983	3	2007	7	3033	5	2931
1022	68%	26	26241	52	37110	26	15236	115	52928	95	44954	116	47282	84	67680	17	6891	4	2453	11	5459
1023	90%	18	42618	36	28605	11	8438	96	59749	54	27281	61	33919	40	36013	28	12939	8	5822	7	3739
1024	95%	17	25683	58	42674	22	10674	99	43432	54	23152	138	60477	45	34940	14	3428	5	2386	15	6253
1025	68%	42	29705	20	25729	51	23356	120	54853	109	91320	99	46999	115	73741	24	7148	10	6389	14	7612
1026	81%	32	56579	76	48060	29	13994	139	49289	123	43985	135	65877	196	78399	11	4353	56	27390	47	16336
1027	95%	17	21125	27	18366	33	15838	80	33631	94	36113	112	49953	50	46865	22	7981	14	11193	18	7418
1028	96%	16	20578	72	56575	17	16061	70	47695	90	46663	94	55318	47	66658	17	5175	21	10903	12	5319
1029	87%	14	17982	43	33860	17	10393	77	32585	59	30272	67	36138	55	46443	16	6401	7	3826	11	4074
1030	98%	27	22586	67	42374	21	9271	125	53955	59	27631	167	81801	95	47275	55	18125	35	15391	25	9918
1031	99%	27	26309	60	46226	19	9032	97	40739	24	9599	132	57617	72	60020	27	10248	23	11879	27	9068
1032	97%	30	25971	28	21163	8	6318	49	22860	108	58177	76	35766	43	49671	11	3510	4	3508	12	6958
1033	97%	14	16668	7	11401	9	7335	44	22834	5	2160	29	13307	14	13884	22	7583	6	7413	12	6402

Table A.1: Complete benchmark selection dataset of the Validation study: participant, accuracy, number of fixations and time on each subtask.

A.3 FDI results

Participant	*Samples*	*Main T1*	*Main T3*	*Supplier*	*Category T1*	*Category T3*	*Products*	*Product*	*Confirm*	*Orders*	*Invoice*
1001	62%	1.32	8.85	6.47	3.66	15.13	3.87	10.07	5.55	8.55	8.94
1002	31%	2.58	7.17	3.1	0.57	11.98	0.65	0	0	7.33	1.6
1003	74%	3.4	11.9	9.86	1.45	17.98	2.85	4.82	1.38	6.91	1.72
1004	66%	2.42	7.38	5.35	3.29	15.65	7.21	6.28	2.17	11.79	3.76
1005	61%	1.09	7.92	3.54	2.55	9.14	3.64	5.78	2.54	3.67	1.21
1006	11%	1.15	3.26	0	0.49	0	0.42	1.84	0	0	0
1007	73%	2.57	9.97	4.97	3.36	12.4	1.43	4.03	2.23	4.14	6.62
1008	55%	4.42	15.29	3.35	2.41	12.08	6.71	5.97	2.3	9.03	3.51
1009	99%	1.19	11.85	2.48	0.79	4.21	1.06	2.84	4.55	3.26	2.73
1010	72%	1.83	1.71	2.88	1.16	4.46	1.16	2.69	2.54	1.33	1.35
1011	59%	1.87	12.29	11.75	3.58	9.75	3.96	6.08	2.02	7.48	1.67
1012	41%	3.16	14.6	6.82	3.95	14.7	3.4	4.48	2.62	13.8	5.91
1013	91%	0	7	5	0	6.89	3.32	1.11	0	3.4	2.56
1014	97%	0.88	2.85	0	0.43	2.14	2	1.71	1.12	2.26	1.08
1015	31%	0.57	5.31	1.54	0.2	7.67	0.47	0.67	0.07	3.61	0
1016	95%	1.14	4.35	2.08	1.51	5.99	2.9	4.56	2.33	4.94	5.24
1017	69%	3.67	4.65	6.34	1	14.75	3.02	2.19	2.51	7.48	9.16
1018	47%	1.43	10.45	8.67	2.87	15.65	4.84	11.02	4.72	10.6	5.32
1019	3%	2.34	0	0.29	0.01	0	0.52	0.04	2.58	0	0
1020	94%	1.85	5.88	1.47	0.73	7.52	1.73	3.39	1.07	8.01	4.91
1021	94%	0.5	3.69	5.72	0.38	2.55	1.9	2.83	0	0.87	0
1022	68%	2.49	10.45	4.88	2.31	13.48	3.7	5.07	1.53	0.77	2.81
1023	90%	1.94	5.32	1.63	2.03	7.01	1.44	2.52	1.82	2.43	2.94
1024	95%	1.6	10.8	3.19	1.98	9.38	3.81	2.87	1.05	1.38	2.23
1025	68%	3.93	2.69	11.59	1.94	16.99	2.35	4.31	2.79	1.82	3.28
1026	81%	2.95	10.1	4.9	2.84	21.04	4.17	12.23	1.39	14.99	10.71
1027	95%	2.29	5.13	4.56	1.14	14.35	2.56	3.76	2.1	5.01	4.52
1028	96%	2.08	11.62	4.33	1.5	17.73	2.91	3.23	1.37	9.06	3.64
1029	87%	1.31	7.26	4.66	1.37	9.33	1.81	2.73	1.82	1.41	3.19
1030	98%	3.02	11.28	3.34	3.03	11.24	5.08	8.56	5.9	9.98	6.78
1031	99%	3.02	11.05	3.34	1.93	6.41	4.39	5.11	3.28	8.31	8.38
1032	97%	3.3	5.33	1.43	0.88	16.66	2.22	2.77	0.68	0	3.65
1033	97%	1.26	0	1.64	0.9	0	0	0	1.96	2.47	3.78

Table A.2: Complete FDI dataset of the Validation study. *FDI* value for every subtask.

A.4 SDI and SLI results

Part	Accu	Main T1		Main T3		Supplier		Category T1		Category T3		Products		Product		Confirm		Orders		Invoice	
1001	62	4	78.95%	46	22.03%	27	12.9%	85	61.01%	129	9.15%	38	79.23%	79	46.62%	53	35.37%	29	25.64%	24	22.58%
1002	31	13	48%	32	28.89%	5	44.44%	8	68%	49	10.91%	8	20%	0	0%	0	0%	20	23.08%	3	25%
1003	74	16	48.39%	87	24.35%	67	8.22%	30	61.04%	125	13.79%	41	52.33%	31	70.19%	9	40%	12	25%	4	0%
1004	66	14	41.67%	25	24.24%	21	22.22%	57	66.07%	99	14.66%	117	58.06%	41	65.55%	14	41.67%	30	11.76%	13	13.33%
1005	61	4	73.33%	36	28%	15	40%	47	63.28%	42	10.64%	33	67.33%	34	68.22%	11	42.11%	6	25%	4	20%
1006	11	5	0%	14	12.5%	1	0%	6	83.33%	0	0%	8	52.94%	16	36%	2	33.33%	0	0%	0	0%
1007	73	13	31.58%	47	26.56%	17	29.17%	74	57.95%	72	16.28%	16	72.88%	48	39.24%	16	33.33%	18	5.26%	22	12%
1008	55	17	46.88%	69	30.3%	20	13.04%	47	61.79%	70	10.26%	66	72.38%	32	57.33%	11	50%	23	11.54%	7	30%
1009	99	5	58.33%	56	25.33%	8	33.33%	13	72.92%	15	6.25%	8	80.95%	12	73.33%	24	33.33%	9	18.18%	8	33.33%
1010	72	4	75%	7	46.15%	8	55.56%	17	71.19%	21	4.55%	12	75.51%	25	46.81%	9	40%	7	0%	5	37.5%
1011	59	11	47.62%	72	25%	56	26.32%	51	72.28%	51	15%	46	68.71%	54	51.35%	10	50%	24	14.29%	12	7.69%
1012	41	16	46.67%	76	22.45%	30	28.57%	47	70.81%	90	1.1%	52	48.51%	34	49.25%	20	37.5%	54	14.29%	12	7.69%
1013	91	0	100%	29	21.62%	11	21.43%	0	100%	24	7.69%	31	65.93%	12	52%	0	100%	20	4.76%	4	20%
1014	97	2	81.82%	12	14.29%	0	100%	5	87.5%	4	0%	17	66.67%	11	63.33%	6	40%	10	9.09%	5	16.67%
1015	31	0	100%	16	38.46%	5	37.5%	6	70%	29	19.44%	7	68.18%	3	76.92%	1	80%	5	16.67%	0	0%
1016	95	6	64.71%	19	13.64%	5	50%	39	55.17%	33	5.71%	30	68.42%	23	64.62%	9	52.63%	13	0%	13	18.75%
1017	69	16	52.94%	21	27.59%	20	31.03%	19	70.31%	90	13.46%	27	68.97%	10	67.74%	12	29.41%	23	8%	26	27.78%
1018	47	10	37.5%	44	20%	29	19.44%	71	50%	87	9.38%	51	61.36%	112	41.05%	37	13.95%	26	16.13%	11	15.38%
1019	3	15	21.05%	0	0%	1	0%	6	33.33%	0	0%	4	66.67%	0	100%	17	10.53%	0	0%	0	0%
1020	94	7	36.36%	19	20.83%	2	60%	13	70.45%	35	5.41%	18	75.68%	33	43.1%	8	46.67%	32	0%	13	13.33%
1021	94	4	55.56%	15	25%	13	50%	8	75%	9	18.18%	14	74.07%	15	60.53%	2	0%	5	16.67%	0	100%
1022	68	10	60%	44	13.73%	15	40%	40	64.91%	82	12.77%	33	71.3%	37	55.42%	10	37.5%	3	0%	6	40%
1023	90	8	52.94%	32	8.57%	3	70%	37	61.05%	47	11.32%	16	73.33%	16	58.97%	13	51.85%	7	0%	6	0%
1024	95	10	37.5%	47	17.54%	17	19.05%	29	70.41%	51	3.77%	56	59.12%	13	70.45%	7	46.15%	3	25%	9	35.71%
1025	68	33	19.51%	12	36.84%	32	36%	38	68.07%	92	14.81%	24	75.51%	37	67.54%	11	52.17%	7	22.22%	12	7.69%
1026	81	13	58.06%	63	16%	19	32.14%	59	57.25%	97	20.49%	36	73.13%	105	46.15%	6	40%	52	5.45%	35	23.91%
1027	95	10	37.5%	18	30.77%	13	59.38%	21	73.42%	73	21.51%	44	60.36%	22	55.1%	10	52.38%	13	0%	15	11.76%
1028	96	8	46.67%	53	25.35%	7	56.25%	17	75.36%	78	12.36%	24	74.19%	14	69.57%	8	50%	20	0%	11	0%
1029	87	5	61.54%	34	19.05%	7	56.25%	23	69.74%	55	5.17%	18	72.73%	23	57.41%	7	53.33%	6	0%	9	10%
1030	98	18	30.77%	50	24.24%	10	50%	41	66.94%	56	3.45%	60	63.86%	39	58.51%	34	37.04%	33	2.94%	23	4.17%
1031	99	8	69.23%	51	13.56%	10	44.44%	28	70.83%	22	4.35%	54	58.78%	30	57.75%	16	38.46%	19	13.64%	19	26.92%
1032	97	13	55.17%	22	18.52%	5	28.57%	13	72.92%	93	13.08%	19	74.67%	12	71.43%	4	60%	0	100%	7	36.36%
1033	97	6	53.85%	0	100%	3	62.5%	11	74.42%	0	100%	0	100%	0	100%	13	38.1%	5	0%	9	18.18%

Table A.3: Complete SDI dataset of the Validation study. $SDI_{remainder}$ and $SDI_{eliminated}$ value for every subtask

Part	Accu	Main T1	Main T3	Supplier	Category T1	Category T3	Products	Product	Confirm	Orders	Invoice
1001	62	374.7	93.67	3518.95	76.5	1738.18	64.38	6522.7	76.74	10714.55	83.06	2521.36	66.35	6805.79	86.15	3992.12	75.32	2343.27	80.8	2024.21	84.34
1002	31	1125.63	86.59	3615.24	112.98	548.93	109.79	1059.88	132.49	5230.1	106.74	1502.65	187.83	0	0	0	0	2362.9	118.14	343.35	114.45
1003	74	2026.24	126.64	9088.92	104.47	6443.15	96.17	2502.78	83.43	11471.11	91.77	4503.89	109.85	2047.07	66.03	1108.64	123.18	1010.69	84.22	492.04	123.01
1004	66	1031.93	73.71	2122.14	84.89	2741.33	130.54	5055.11	88.69	11607.88	117.25	16858.26	144.09	3494.25	85.23	2367.23	169.09	3758.15	125.27	2351.06	180.85
1005	61	603.36	150.84	3824.75	106.24	1750.96	116.73	4401.57	93.65	3676.87	87.54	2561.1	77.61	1987.65	58.46	1230.52	111.87	631.6	105.27	369.18	92.29
1006	11	630.13	126.03	1817.63	129.83	38.48	38.48	395.25	65.88	0	0	614.7	76.84	924.96	57.81	375.4	187.7	0	0	0	0
1007	73	1348.32	103.72	4150.26	88.3	2335.32	137.37	7095.97	95.89	6580.66	91.4	1612.34	100.77	2730.77	56.89	1588.98	99.31	1892.25	105.12	2468.46	112.2
1008	55	1627.39	95.73	7069.28	102.45	2135.25	106.76	4765.45	101.39	7147.96	102.11	5818.54	88.16	2735.22	85.48	1343.13	122.1	1825.29	79.36	867.19	123.88
1009	99	593.57	118.71	5932.12	105.93	545.05	68.13	1374.78	105.75	1319.64	87.98	662.07	82.76	1099.35	91.61	3296.67	137.36	1038.4	115.38	1354.98	169.37
1010	72	346.53	86.63	1213.42	173.35	583.78	72.97	1859.34	109.37	2626.28	125.06	1067.09	88.92	2162.13	86.49	1506.38	167.38	1378.47	196.92	765.41	153.08
1011	59	1126.52	102.41	5773.43	80.19	4880.85	87.16	5401.11	105.9	4218.75	82.72	3505.68	76.21	5974.31	110.64	792.7	79.27	2098.79	87.45	893.92	74.49
1012	41	2156.02	134.75	5830.24	76.71	1970.47	65.68	3899.9	82.98	9772.36	108.58	4635.22	89.14	1612.62	47.43	2051.51	102.58	3083.52	57.1	1611.56	134.3
1013	91	0	0	4166.3	143.67	1370.37	124.58	0	0	3228.36	134.52	3835.86	123.74	716.24	59.69	0	0	1817.09	90.85	390.44	97.61
1014	97	117.72	58.86	1358.13	113.18	0	0	539.51	107.9	382.96	95.74	1552.89	91.35	732.05	66.55	676.26	112.71	1548.25	154.83	924.55	184.91
1015	31	0	0	2003.77	125.24	498.65	99.73	542.38	90.4	2722.86	93.89	347.49	49.64	207.46	69.15	62.8	62.8	744.79	148.96	0	0
1016	95	565.35	94.22	2178.39	114.65	335.43	67.09	3924.84	100.64	2919.53	88.47	2171.24	72.37	1845.1	80.22	1273.83	141.54	1192.52	91.73	1434.9	110.38
1017	69	1205.52	75.35	1792.41	85.35	2290.94	114.55	1223.86	64.41	7873.2	87.48	2608.91	96.63	560.22	56.02	1118.5	93.21	2972.47	129.24	2647.29	101.82
1018	47	1237.18	123.72	5358.42	121.78	3534.83	121.89	9499.31	133.79	8873.91	102	7040.86	138.06	15517.57	138.55	3596.43	97.2	2757.83	106.07	1543.17	140.29
1019	3	1801.56	120.1	0	0	60.73	60.73	719.6	119.93	0	0	312.03	78.01	0	0	1936.43	113.91	0	0	0	0
1020	94	1155.36	165.05	2621.11	137.95	63.12	31.56	1007.46	77.5	2991.06	85.46	1518.21	84.34	2969.81	89.99	1098.23	137.28	3028.95	94.65	1844.06	141.85
1021	94	482.55	120.64	1532.25	102.15	1223.79	94.14	740.31	92.54	713.33	79.26	918.5	65.61	1245.78	83.05	268.75	134.37	508.91	101.78	0	0
1022	68	891.36	89.14	4536.21	103.1	1570.33	104.69	3580.73	89.52	7758.34	94.61	3606.59	109.29	3017.42	81.55	1114.13	111.41	431.88	143.96	821.44	136.91
1023	90	792.63	99.08	4079.62	127.49	116.55	38.85	3762.2	101.68	4813.96	102.42	1317.45	82.34	1118.49	69.91	1283.15	98.7	739.18	105.6	871.2	145.2
1024	95	1422.03	142.2	5301.05	112.79	1784.97	105	3269.25	112.73	5139.01	100.76	5572.79	99.51	1510.76	116.21	680.45	97.21	461.92	153.97	783.34	87.04
1025	68	4115.27	124.71	1273.69	106.14	2412.09	75.38	3671.09	96.61	8734.54	94.94	2699.46	112.48	2999.06	81.06	1981.24	180.11	312.3	44.61	1795.22	149.6
1026	81	1115.42	85.8	6167.17	97.89	2083.11	109.64	6123.01	103.78	10269.6	105.87	2965.36	82.37	12215.62	116.34	699.25	116.54	5136.12	98.77	3677.26	105.06
1027	95	1225.44	122.54	1809.85	100.55	711.78	54.75	1895.91	90.28	7240.31	99.18	3210.04	72.96	2054.34	93.38	1293.04	129.3	1068.61	82.2	1765.84	117.72
1028	96	893.77	111.72	5837.08	110.13	872.11	124.59	1285.41	75.61	6897.78	88.43	2378.37	99.1	1320.51	94.32	1112.99	139.12	2555.17	127.76	1448.45	131.68
1029	87	664.64	132.93	3360.04	98.82	586.34	83.76	2213.2	96.23	5181.93	94.22	1986.06	110.34	1530.92	66.56	946.61	135.23	705.97	117.66	1016.81	112.98
1030	98	1520.05	84.45	4774.8	95.5	594.32	59.43	4653.58	113.5	6311.88	112.71	7869.29	131.15	3712	95.18	3565.66	104.87	4279.27	129.67	2505.24	108.92
1031	99	772.59	96.57	6951.91	136.31	732.49	73.25	2490.39	88.94	2712.88	123.31	5870.95	108.72	2450.55	81.69	2652.72	165.79	1963.65	103.35	2664.81	140.25
1032	97	1486.14	114.32	2512.82	114.22	188.89	37.78	1195.27	91.94	9464.9	101.77	1569.72	82.62	939.15	78.26	595.78	148.94	0	0	798.05	114.01
1033	97	745.7	124.28	0	0	343.29	114.43	1283.78	116.71	0	0	0	0	0	0	1406.17	108.17	511.35	102.27	914.26	101.58

Table A.4: Complete SLI dataset of the Validation study. SLI_{total} and $SLI_{average}$ value for every subtask.

Appendix B

Validation Usability Study

This section holds additional information on the BiYP usability study. First, the user questionnaire is listed and then the questionnaire results of all the participants are provided. Lastly, the full expert review report of the eye tracking data captured during the BiYP usability study is given.

B.1 User questionnaire

Each participant completed the following questionnaire for the BiYP usability study:

1. Please enter the unique number as provided by the facilitator:

   _____

2. Please select your age group?

   ☐ Younger than 15
   ☐ 15–25
   ☐ 26–35
   ☐ 36–45
   ☐ 46–55
   ☐ 56–65
   ☐ Older than 65

3. What is your gender?

☐ Male ☐ Female

4. What is the highest degree or level of school you have completed?

   ☐ No schooling completed
   ☐ High school until grade 10
   ☐ Matric
   ☐ Bachelors degree / Certificate
   ☐ Honours degree
   ☐ Masters degree
   ☐ Doctorate degree

5. Please specify your occupation:

   _____

6. In a typical day, how often do you use your mobile phone?

   ☐ Every hour
   ☐ Every two hours
   ☐ Three times a day
   ☐ Once a day
   ☐ Less frequently than the above

7. Do you have a smart phone?

   ☐ Yes ☐ No

8. Which of the following online shopping sites do you use, or have you used in the past?

   ☐ Kalahari ☐ Take a lot
   ☐ Want it all ☐ Pick 'n Pay

- ☐ Woolworths
- ☐ Expansys
- ☐ Zando
- ☐ Bid or Buy
- ☐ Other (Please Specify): _____

9. Select what you use your mobile phone for, other than making voice calls?

- ☐ Text messaging (SMS)
- ☐ Reading e–mail
- ☐ Searching for specific information
- ☐ Viewing content on social networks
- ☐ Weather forecasts
- ☐ Maps, GPS
- ☐ Social networks
- ☐ News
- ☐ Listening to music
- ☐ Chatting
- ☐ Watching video
- ☐ Listening to audio podcasts
- ☐ Solo video games
- ☐ Multi–player video games
- ☐ Reading books
- ☐ Other (Please Specify): _____

10. Have you ever worked with the Business in Your Pocket / GaRO application?

- ☐ Never seen the application before
- ☐ Saw a demo of the application
- ☐ Used the application prototype on a mobile device
- ☐ Part of the development team for the application

B.2 User questionnaire results

The following section holds the results of the questionnaire for the Validation study.

Figure B.1: What is your age group?

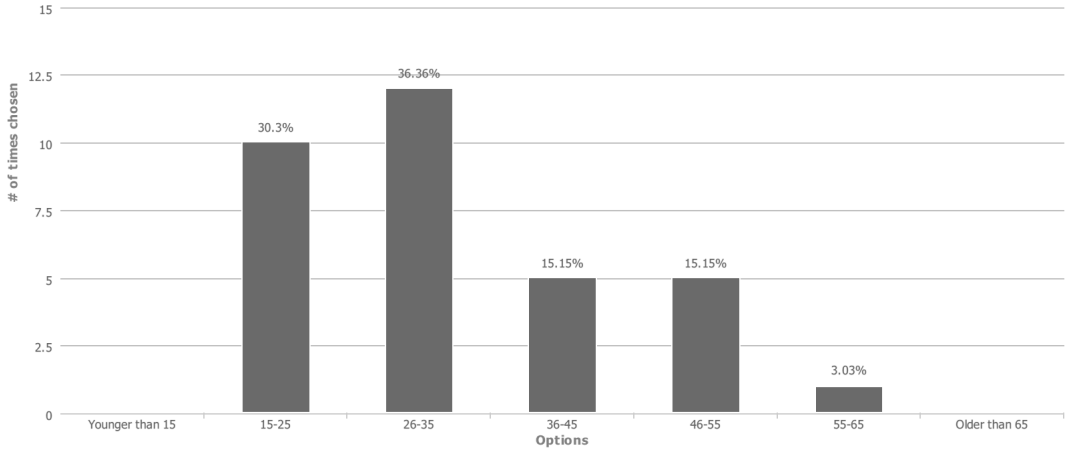


Figure B.2: What is your gender?

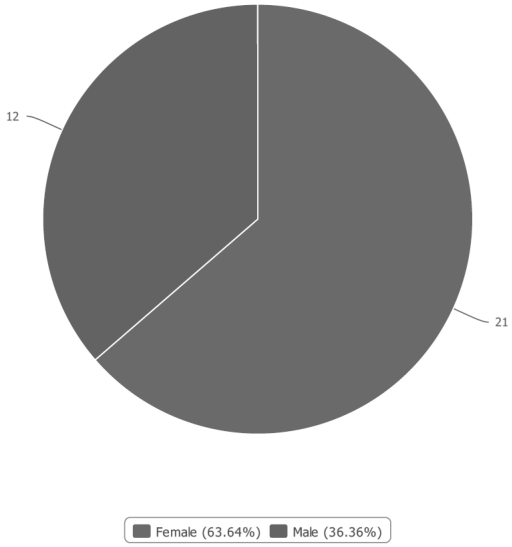


Figure B.3: What is the highest degree or level of schooling that you have completed?

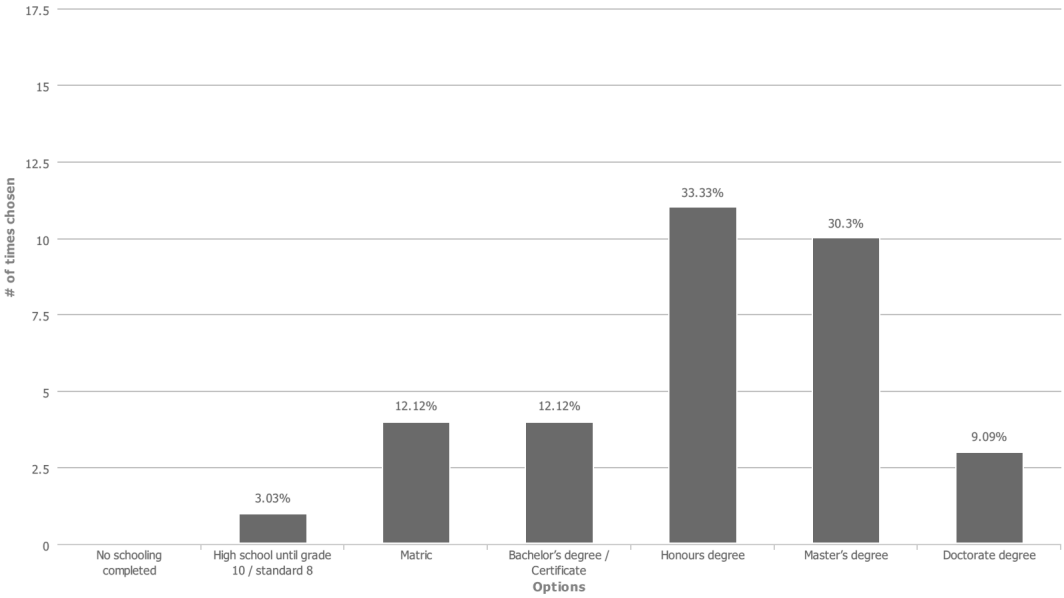


Figure B.4: In a typical day, how often do you use your mobile phone?

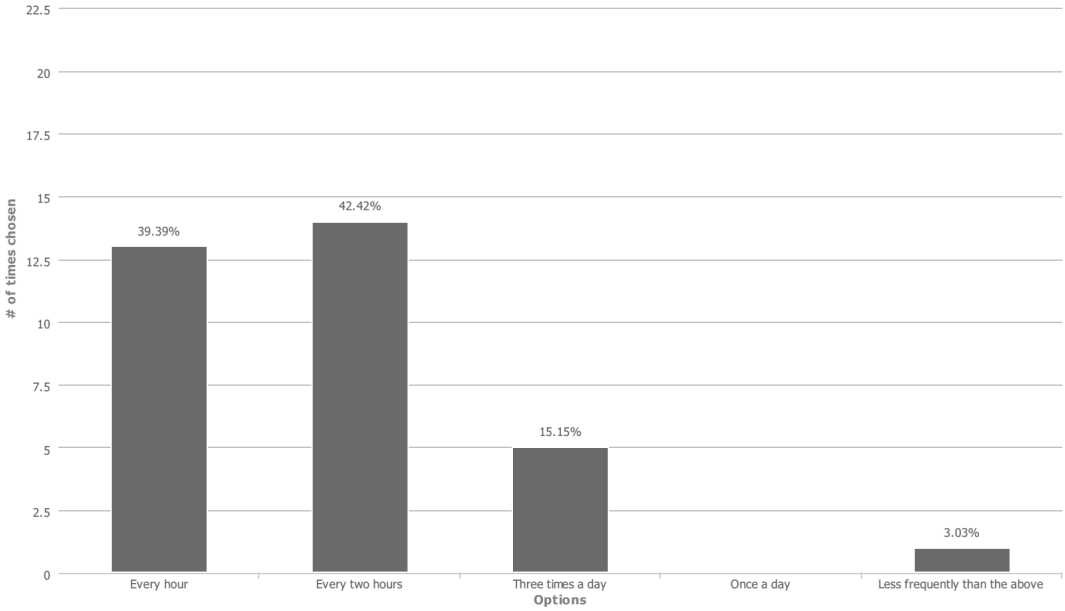


Figure B.5: Do you have a smart phone?

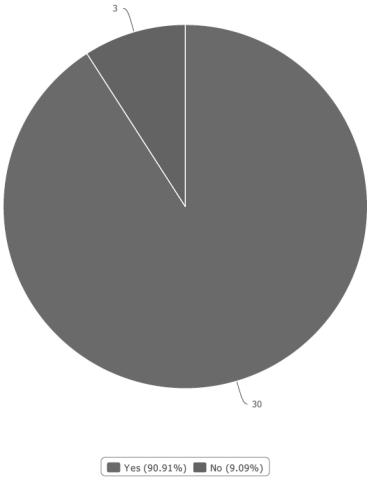


Figure B.6: Which of the following on–line shopping sites do you use, or have you used in the past?

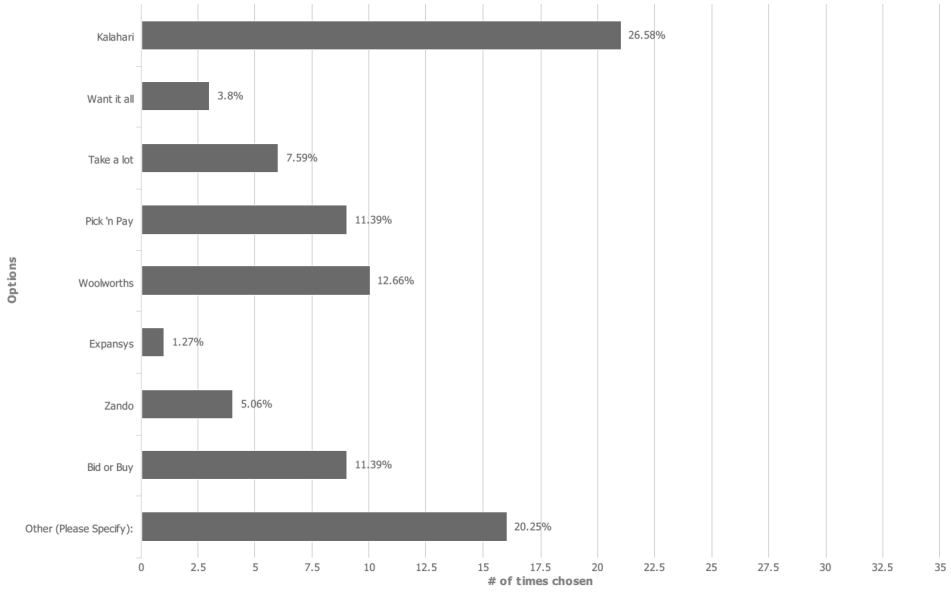


Figure B.7: Select what you use your mobile phone for, other than making voice calls?

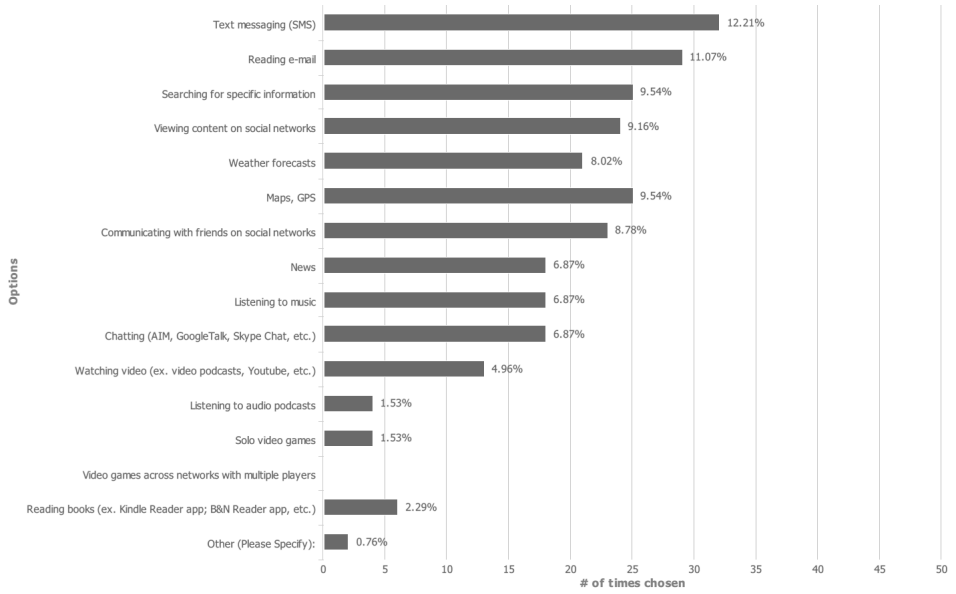
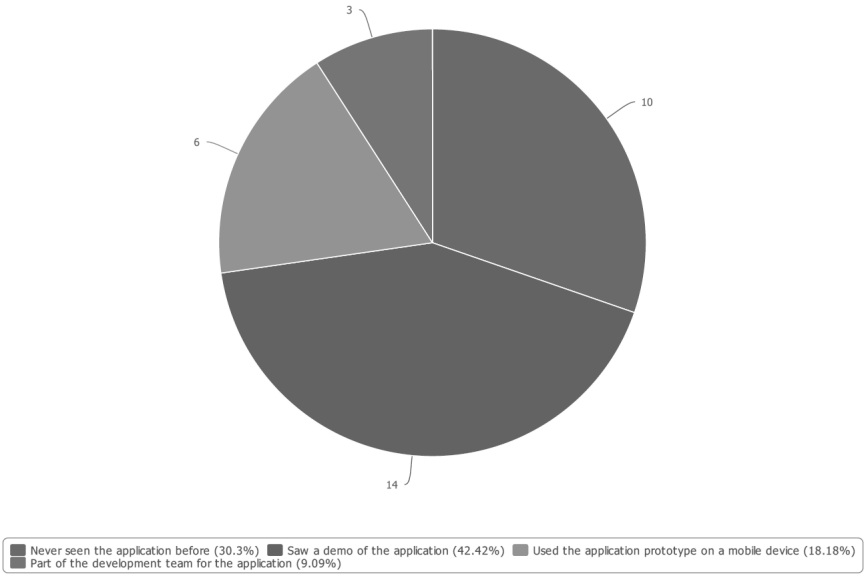


Figure B.8: Have you ever worked with the Business in Your Pocket / GaRO application?



B.3 Expert review

Analysis of eye tracking videos for usability evaluation of the mobile procurement application, BiYP. Performed by Prof Helene Gelderblom, Department of Informatics, University of Pretoria on 10 March 2014.

B.3.1 Time taken to complete tasks:

Task 1 and Task 2 required similar actions from the users (placing an order for a given number of 2 different products). Task 3 required them, from the main menu, to view their orders.

From the task completion times it is clear that there was an improvement from Task 1 to Task 2, with Task 2 being completed in half the time it took to complete Task 1. This is no surprise because once they have determined how to place an order any further orders will be easy.

Task 3 was a seemingly simple task that required a selection (Shop) from the menu and then clicking on the 'Order' heading at the top of the next screen. The task completion times of participants 1021 and 1014 demonstrated that this could be completed very quickly (22 and 24 seconds respectively). However, the average completion time is over one minute, with all the other participants taking more than one minute and one participant taking more than two minutes.

B.3.2 Qualitative analysis of eye tracking videos

Procedure:

A sample of ten eye tracking recordings were selected for the expert evaluation. While watching the videos the evaluator noted down observations that relate to the usability of the application. Each video was watched repeatedly until no new observations could be added to the observation notes.

Participant	Time on Task 1	Time on Task 2	Time on Task 3
1014	66	40	24
1021	61	67	22
1024	126	60	64
1026	207	60	125
1027	138	55	64
1028	159	71	109
1029	104	67	65
1030	171	82	83
1031	140	79	71
1032	96	57	88

Table B.0: Expert review of the time spent on a task, in seconds, for the Validation study.

Results:

All participants identified Shop as the correct main menu item for Tasks 1 and 2. In Task 1, five of the participants spent some time deciding which of the Ricoffy icons to select, but they were quicker in selecting the desired tea bag icon.

A few of the participants did not immediately understand how to scroll the number list with the mouse. This problem is probably linked to the fact that this is an emulation of a touch screen and on the touch screen this would probably be more intuitive. One participant tried to use the keyboard to enter a number in that field.

When the items and their numbers were selected there were some hesitation from some participants before clicking on the 'tick' icon, but not enough to regard this as a usability problem.

Analysis of the observations provided explanation for the problems experienced in Task 3. Eight of the ten participants took some time to decide which option on the main menu to select, scanning up and down through the menu. Three participants first selected an incorrect option (Sales or Services) and had to return to the main menu to try again.

One participant clicked on the '?' icon when he did not know which main menu item to select to see his orders. This was not at all helpful, but he eventually selected Shop.

Eventually all ten selected the Shop item.

Once on the next screen the task required them to click on the word 'Order' that appears at the top left of the screen. On this screen the word is cut off. Also, it appears

in a dim font that may make it seem inactive. There is no clear indication that it represents an active area on the screen. All participants did eventually select it, some of whom "stumbled" upon it almost by accident.

To summarise:

The following are severe usability problems:

- There is no clear path to viewing orders already placed. The fact that this falls under the 'Shop' main menu item is counter–intuitive. The description next to the Shop icon should at least mention that orders can be viewed there, if it is not desirable to include a new designated main menu item for this kind of query.
- Once on the correct screen to access the orders, it is again very difficult to determine from the visible interface how to access the order history.
- When ordering new items, the mechanism for entering the number of items to order was not sufficiently successful on the emulator. Testing on the touch screen is recommended to determine if this problem also occurs on the actual screen.

Other aspects that could be improved:

- For some products there are more than one icon. This requires the user to make a decision w.r.t. the correct one to choose and slows down the overall interaction.
- Two participants selected the X icon when they should have selected the 'tick' icon. This did not lead to any severe problems in the interaction and they could correct their actions easily.