Detecting Confusion Using Eye-Tracking and Machine Learning Techniques

Yupeng Lan

June 8, 2017

Abstract

Eye Tracking is widely applied in studying how people interact with the interface of software. Researchers use eye trackers to find confusion during users' interactions with the interface. They search for flaws in the user interface design and make users less confused in the development stage. This project explores the possibility of using machine learning methods to find the patterns of confusion in eye movements and mouse activity using real time interaction data. An experiment is designed to collect eye tracking data from participants. The positions of gaze, fixation, and cursor are used to generate feature data. Two versions of feature data are generated: the Euclidean distances of gaze, fixation, and cursor position and the standard deviation of gaze, fixation, and cursor position in a five-second windows. Then the models built from two feature sets are compared. 60% of two feature sets are training sets, and the rest of 40% are validation sets. Models produce insignificant result on both test sets. A K-Nearest Neighbor model classifies the first feature set with the highest classification rate of 60% on both class instances with kappa statistics of 0.14. The KStar model best classifies the second feature set with 53.5117% of classification accuracy on both class instances with kappa statistics of 0.09. Individual categories of feature data are evaluated to find the correlations with confusion using logistic regression. The cursor feature data in both feature sets are found to be strongly correlated with confusion.

Contents

List of Figures

List of Tables

1 Introduction

Confusion is one of the frustrating parts of user experience when performing some tasks in complex software systems. Researches were focusing on designing user interface making users less confused in the development stage. However, even well designed user interface can still make users confused. Some research projects have attempted to reduce confusion on the fly during the interaction between the user and the software system. Bosch et al. [1] proposed a method to resolve confusion in an Intelligent Tutoring System. Based on the work of Bosch et al. [1], Lallé et al. [20] has conducted a study to show how to predict confusion when the user is interacting with a visualization-based interface, which is a proof of concept for automating confusion detection. They have produced promising result using pupil size as the most significant feature to predict confusion. In this project, a similar approach is used to prove the viability of automating confusion detection.

Confusion can happen in many different kinds of scenarios. When performing a task using some software, the user needs to plan a set of concrete steps to interact with the user interface. Users may get confused if they do not know the correct steps to complete the task, or see different outcome of the actions from their expectations. In this project, one type of user confusion is investigated: users cannot locate the functional component of the user interface corresponding to their plan. When the user cannot find a corresponding option in the interface, one of the most direct reaction is to perform visual search on the likely areas of the target in the interface. In this case, it is worth investigating users' gaze patterns since these patterns may predict confusion. There are many types of gaze patterns. For example, users may display a gaze pattern called back-tracing. When users search for items in the interface, they gaze from top to bottom and then gaze back up to search for missed items[2]. Users may also stare at a particular area of the interface, which is a pattern called fixation[15]. Such pattern may generate variations in the concentration of gaze points, which is represented as the standard deviation of gaze, fixation, and cursor position. The standard deviation of gaze and fixation may also suggest a change in users' interest[15]. In addition, mouse activities can also suggest confusion[22]. In this project, these two differences are used to build models for confusion in order to investigate the possibility of using these two patterns as confusion predicting features.

In this project, I hope to answer the question: Can eye movement reveal confusion? If so, what eye movement features are the best indicator of confusion? The hypothesis of this project is that these two patterns are possible confusion predictors: the concentration of gaze and fixation points on areas of the interface and the relative positions between the cursor and gaze points. In this project, The data analysis methods similar to the one used Lallé et al. [20]s work were applied to prove the hypothesis. Lallé et al. [20] applied the random forest algorithm to build a confusion prediction model. Their result showed that combining a set of metrics could accurately predict confusion. This project used the following algorithms to build confusion prediction models: KStar and IBK. The performance of these models are cross compared. Logistic Regression is used to test the statistical significance of both hypotheses.

An experiment is designed to collect data to build confusion prediction models. The basic approach is to let subjects complete eight common tasks in Excel and use a GazePoint GP3 eye tracker to record their eye movements. Subjects were asked to verbally report confusion when getting confused. The 5 provides more details about the process.

The body is organized as follows: Section 2 gives an overview of eye tracking technology and its application. Section 3 discusses the related projects that provides inspiration, methodology, and theoretical and practical foundations. Section 4 proposes a method to answer the research question. Section 5 discusses details on the experiment design and process. Section 6 describes the procedure of formatting raw experiment data and generating feature data. Section 7 presents a summary and an analysis on the experiment data, feature data, and classification result. An overall summary of the result is in the section 8. Section 9 discusses limitations in the experiment data and data processing, and this section also presents an reflection on the mistakes made in this project. Section 9 will also talk about works needed to be done to improve the result of this project. Section A provides raw data, eye tracker information, etc. that are related to this project. The A section will also include some algorithms used in this project.

2 Introduction to Eye Tracking

2.1 What is eye tracking?

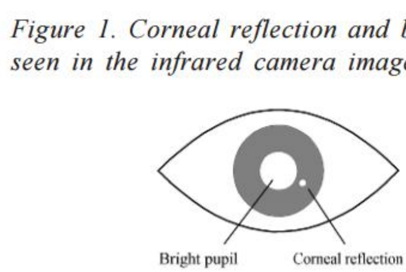


Figure 1: corneal reflection from emitted infrared light[7]

In the book *Encyclopedia of Human Computer Interaction* by Ghaoui [7], Poole and Ball [23] describe eye tracking as the following: eye tracking is a research technique for the researcher to understand where and when one is seeing and how one shift the gaze from one place to another. Eye tracking can help HCI

(Human-Computer Interaction) researchers to understand the factors affecting the usability of user interfaces and to study visual and display-based information processing. In this way, eye movement recordings help objectively evaluating and informing the design of user interfaces. People can also use eye tracking as an indirect mean of interacting with computers without keyboards or mouse, which can be especially beneficial for the disabled.

2.2 Introduction to Eye Trackers

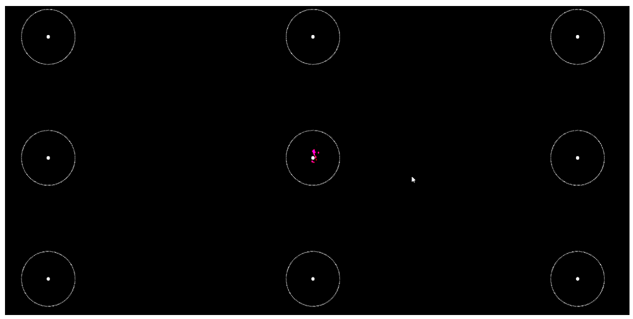


Figure 2: A typical 9-point calibration grid

Most commercial grade eye trackers on the market measure point-of-regard through the corneal-reflection/pupil-center method[9]. These kinds of trackers are usually installed under the screen of a standard computer. The trackers use some image processors (typically a software) to locate eyes and find the features used for eye tracking. The eye tracker emits an infrared light source embedded in the device to illuminate the target eyes and uses a camera to track the location of eyes consistently. Infrared light is used because it does not dazzle the user. A large proportion of the emitted infrared light is reflected back from users' retinas so that the pupil appear as a bright, well-defined disc (known as the corneal reflection as shown in 2.1). Once the center of the pupil and the location of the corneal reflection is being consistently tracked, then the image processor measures vector between them and uses trigonometry to calculate the point-of-regard. Eye movements can be disassociated from head movements even though solely using corneal reflection is enough to approximate the point-of-regard[5][18]. Video-based eye trackers need to be calibrated to adapt different biometrics of each persons eye movements. In the calibration process, a dot displays on the screen, and the system records the pupil-center/corneal-reflection relationship corresponding to a specific x,y coordinate on the screen if users' eyes fixate on that dot long enough to pass the timing threshold. Users repeatedly calibrate the eye tracker over a 5 points, 9 points, or 13 points grid (shown in figure 2.2) until desired accuracy is reached[8].

2.3 Why is eye tracking interesting to HCI researchers?

Why people are so interested in using eye tracking to conduct usability research? Just and Carpenter [19] suggested that what a person is looking at implied his or her cognitive processes. Poole and Ball [23] pointed out that eye movement recordings can dynamically trace a persons attention being directed in relation to a visual display through Just and Carpenter [19]'s hypothesis. In practice, when the HCI researcher needs to infer useful information from eye-movement recordings, they need to define areas of interest over certain parts of a display or interface, and then they analyze the eye movements that fall within such areas[23]. After that, the result from eye tracking data can help evaluating the meaningfulness, visibility, and placement of specific interface elements objectively, and then the findings can be used to improve the design of the interface[10]. For example, if some participants are asked to search for a button, a longer-than-expected gaze on the button before eventual selection would indicate that that button lacks meaningfulness and probably needs to be redesigned[23].

2.4 Commonly Used Eye Tracking Metrics

Eye-Movement Metric	What it Measures	Reference
Number of fixations overall	More overall fixations indicate less efficient search (perhaps due to sub-optimal layout of the interface).	Goldberg and Kotval (1999)
Fixations per area of interest	More fixations on a particular area indicate that it is more noticeable, or more important, to the viewer than other areas.	Poole et al. (2004)
Fixations per area of interest and adjusted for text length	If areas of interest are comprised of text only, then the mean number of fixations per area of interest can be divided by the mean number of words in the text. This is a useful way to separate out a higher fixation count, simply because there are more words to read, from a higher fixation count because an item is actually more difficult to recognize.	Poole et al. (2004)
Fixation duration	A longer fixation duration indicates difficulty in extracting information, or it means that the object is more engaging in some way.	Just and Carpenter (1976)
Gaze (also referred to as dwell, fixation cluster, and fixation cycle)	Gaze is usually the sum of all fixation durations within a prescribed area. It is best used to compare attention distributed between targets. It also can be used as a measure of anticipation in situation awareness, if longer gazes fall on an area of interest before a possible event occurring.	Mello-Thoms et al. (2004); Hauland (2003)
Fixation spatial density	Fixations concentrated in a small area indicate focused and efficient searching. Evenly spread fixations reflect widespread and inefficient search.	Cowen et al. (2002)
Repeat fixations (also called post-target fixations)	Higher numbers of fixations off target after the target has been fixated indicate that it lacks meaningfulness or visibility.	Goldberg and Kotval (1999)
Time to first fixation on target	Faster times to first fixation on an object or area mean that it has better attention-getting properties.	Byrne et al. (1999)
Percentage of participants fixating on an area of interest	If a low proportion of participants is fixating on an area that is important to the task, it may need to be highlighted or moved.	Albert (2002)
On target (all target fixations)	Fixations on target divided by total number of fixations. A lower ratio indicates lower search efficiency.	Goldberg and Kotval (1999)

Figure 3: Fixation-derived metrics and their interpretation in the usability studies[23]

In *Eye Tracking: A comprehensive guide to methods and measures* by Holmqvist et al. [15], fixation is the state when the eye stays still over a period of time, e.g. stopping at a word during reading. It lasts anywhere from some tens of milliseconds up to several seconds. Fixation can have quite different interpretations in various scenarios. According to Jacob and Karn [17] and Just and Carpenter [19], in a task like browsing web pages, higher fixation frequency on a particular area can be suggests greater interest in the target (e.g., a photograph in a news report), or it can be indicate that the target is hard to be understood by the user. However, in a search task, the interpretation of fixation is reversed: a higher number of single fixations, or clusters of fixations, are the signals of the difficulties or uncertainty in interpreting a target item[17]. The duration of a fixation also is linked to the processing time applied to the object being fixated[19]. It is widely accepted that external representations associated with long fixations are not as meaningful to the user as those associated with short fixations[8].

According to Hauland [13] and Mello-Thoms et al. [21], Gaze is the aggregated fixation durations within a certain area. Gaze is used to compare attention distribution between targets. Gaze can be also used as a measure of anticipation in situation awareness, if longer gazes fall on an area of interest before a possible event occurring[23].

The explanations of other metrics are shown in figure 2.4, figure 2.4, and figure 2.4.

Eye-Movement Metric	What it Measures	Reference
Number of saccades	More saccades indicate more searching.	Goldberg and Kotval (1999)
Saccade amplitude	Larger saccades indicate more meaningful cues, as attention is drawn from a distance.	Goldberg et al. (2002)
Regressive saccades (i.e., regressions)	Regressions indicate the presence of less meaningful cues.	Sibert et al. (2000)
Saccades revealing marked directional shifts	Any saccade larger than 90 degrees from the saccade that preceded it shows a rapid change in direction. This could mean that the user's goals have changed or the interface layout does not match the user's expectations.	Cowen et al. (2002)

Figure 4: Saccade derived metrics and their interpretations[23]

Eye-Movement Metric	What it Measures	Reference
Scanpath duration	A longer-lasting scanpath indicates less efficient scanning.	Goldberg and Kotval (1999)
Scanpath length	A longer scanpath indicates less efficient searching (perhaps due to a suboptimal layout).	Goldberg et al. (2002)
Spatial density	Smaller spatial density indicates more direct search.	Goldberg and Kotval (1999)
Transition matrix	The transition matrix reveals search order in terms of transitions from one area to another. Scanpaths with an identical spatial density and convex hull area can have completely different transition values—one is efficient and direct, while the other goes back and forth between areas, indicating uncertainty.	Goldberg and Kotval (1999); Hendricson (1989)
Scanpath regularity	Once cyclic scanning behavior is defined, and then deviation from a normal scanpath can indicate search problems due to lack of user training or bad interface layout.	Goldberg and Kotval (1999)
Spatial coverage calculated with convex hull area	Scanpath length plus convex hull area define scanning in a localized or larger area.	Goldberg and Kotval (1999)
Scanpath direction	This can determine a participant's search strategy with menus, lists, and other interface elements (e.g., top-down vs. bottom-up scanpaths). *Sweep* denotes a scanpath progressing in the same direction.	Altonen et al. (1998)
Saccade/ fixation ratio	This compares time spent searching (saccades) to time spent processing (fixating). A higher ratio indicates more processing or less searching.	Goldberg and Kotval (1999)

Figure 5: Scanpath derived metrics and their interpretation[23]

3 Related Work

HCI Researches are putting efforts to make more proactive and user-friendly user interface. Prendinger et al. [24] made an Attentive User Interface that guess the intention of disabled users based on their gaze. Hyrskykari [16] leveraged eye tracker to make a gaze aware intelligent dictionary that provides in-time translation for users reading foreign text. All of these project infers the mental process of users from their eye movements in order to make correct decisions to provide help. Similarly, my project has the similar goal: inferring confusion from the eye movements and mouse usages so that software can decide to provide help to users when predicting confusion.

a) Gaze Features (149)
Overall Gaze Features (9)
Fixation rate
Mean & Std. deviation of fixation durations
Mean & Std. deviation of saccade length
Mean & Std. deviation of relative saccade angles
Mean & Std. deviation of absolute saccade angles
AOI Gaze Features for each AOI (140)
Fixation rate in AOI
Longest fixation in AOI, Time to first & last fixation in AOI
Proportion of time, Proportion of fixations in AOI
Number & Prop. of transitions from this AOI to every AOI
b) Pupil Features (6) and Head Distance Features (6)
Mean, Std. deviation, Max., Min. of pupil width/head distance
Pupil width/head distance at the *first* and *last fixation* in the data window
c) Mouse Event Features (Overall and for each AOI) (32)
Left click rate, Double click rate
Time to first left click, Time to first double click

Table 1: Sets of feature considered for classification.

Figure 6: features used in Lalle et al.'s paper

Lallé et al. [20] provides the evidence that predicting confusion in real time is possible. Thus, in my project, I would like to add another piece of evidence to show that real-time confusion prediction is possible. Major parts of the experiment design and methodology are inspired by their work. In terms of selecting test platform,Lallé et al. [20] developed a software called InfoVis as the test platform to collect eye tracking data. They claimed that complex decisions can be modeled as preferential choices. In contrast, complex decisions are more close to plan a set of operations on the user interface and map the operations to the corresponding locations. In terms of collecting confusion events, they asked participants to self-report confusion by clicking a button on the interface of InfoVis and confirmed their reports after the experiment. Similarly, I decide to let participants self-report confusion verbally. Holmqvist et al. [15] suggested that verbal communication may affect the result of eye tracking. Thus, a less intrusive method of confusion report is considered, which is to let participants click left mouse key three times at the beginning of confusion event and click right mouse key three times to end the confusion report. However, due to the limitation of the experiment control software coming along with the eye tracker, mouse clicks cannot be reliably recorded and resulted in lost data. The same assumption of confusion report is used in my project: assuming participants are not confused until they report to be confused. In terms of feature selection, they decided to use a collection of features showed in 3. They found that pupil size was the most promising feature to predict confusion. I also considered to include pupil size in the feature set, but given the unreliable accuracy of eye tracking data and no proper calibration process is incorporated in the experiment, I have to only include gaze, fixation, and cursor position in the feature set. A similar data sampling method is used in my project. They randomly chose some pivot points in time as the start of confusion just before the start of real confusion event in their ”short confusion window” method. I borrowed this idea of sampling a confusion interval into a five-second window to compress eye tracking data into one feature data point. We both used full windowed data sampling in the second feature data set. They also evaluated different combinations of feature set and concluded that a larger feature set can produce more accurate result than using a smaller feature set. A small set of features are used in my project given the limited time to complete, but the first feature set has a large data size for each feature.

Type	Duration (ms)	Amplitude	Velocity
Fixation	200–300	–	–
Saccade	30–80	4–20°	30–500°/s
Glissade	10–40	0.5–2°	20–140°/s
Smooth pursuit	–	–	10–30°/s
Microsaccade	10–30	10–40′	15–50°/s
Tremor	–	$< 1'$	20′/s (peak)
Drift	200–1000	1–60′	6–25′/s

Figure 7: common eye movements

The book *Eye Tracking: A comprehensive guide to methods and measures* by Holmqvist et al. [15] has provided guidance on the experiment designs in my project. In term of adjusting the lab environment, this book suggests that the lab should have lighting source that emits no infrared light, e.g. fluorescent lighting and neon lights. The lab room should have no window that allows direct sunlight. Thus, the experiment room of the lab used in this project is not directly exposed to sunlight, and the experiment time is usually after the sunset. The book also advices to have moderate lighting level because high lighting environment will restrict the size changes in the subject's pupil, and dark room will make pupil size too large. In this project, brightness level in the lab is adjusted to moderate level. The book points out that sound can easily affect participants' visual behavior and suggests to use a sound proof room. In principle, any vibration can affect the accuracy of eye tracking. In addition, the book suggests not to allow participants to operate on keyboard or mouse. However, given the purpose of the project, participants must use keyboard and mouse to interact with the computer and verbally communicate with the experimenter to facilitate progress of the experiment. Thus, I have to ignore these two suggestions of blocking sound and vibration.

Byrne et al. [2] studies how users interact with the drop-down menu and found that users primarily look from top to bottom, and they may skip a few items. This type of interaction is very common in my project since participants need to interact with drop down menus in Excel frequently. If subjects cannot find their target, they will do a back-tracing search, which intensifies the concentration of gaze points in that area. Thus, if back-tracing happens frequently, then the gaze points should be more concentrated in the area of the menu where the participant looks for the target function.

Hyrskykari [16] proposed a reading assistant software that assists non-native speakers reading texts in foreign languages through eye-tracking. The system measured how long the reader was looking for a particular word, in another word, the duration of eye fixation, to determine whether the user needs the definition of a particular word. In other words, the result of the study indicates the correlation between fixation and uncertainty. In some cases, confusion may happen when participants are not sure of what to do to complete the next step in the assigned task. Then the uncertainty level of participants may increase and result in longer fixations. Thus, it is reasonable to include fixation as one of the features in my project.

Pentel [22] used interaction data from users to predict confusion. They designed a special computer game and collected mouse click data from participants. They concluded that mouse clicks could be used to predict confusion when users were playing their specially designed games. Thus, the mouse click is considered as a feature to predict confusion in this project.

4 Approach

The goal of this project is to find geometric patterns that indicates confusion. In particular the spread of gaze points, fixations, and cursor positions are evaluated.

This project contains two stages: experiment and data analysis. In the experiment stage, I choose Microsoft Excel as the test platform. In the experiment instructions, each tasks contains sequential steps that correspond to particular menu options or buttons of the User Interface. The participants' task is to follow the instructions and find those menu options and buttons. Then they interact with those items on the User Interface to complete the task. I took the lap time of each step for each task. If subject cannot find the items of the instructed step after multiple attempts, then they need to self-report confusion verbally. Then I will mark the time frame of the step that the participant is working on as confusion. I collect the eye-tracking data, time frame data, and screen recording to prepare for data analysis. I will provide more details in the

Section 5.

In the data analysis stage, I wrote a data processing program using Python to turn unformatted raw data (including time intervals of each step and raw eye-tracking data) into processed training data. Then I feed the training data to WEKA [11] to run machine learning algorithms. I will describe the details of data format, the algorithms of data processors, the algorithm of feature generator, and the structure of training data. The source code and data of this project is available on `https://stevelan1995@bitbucket.org/stevelan1995/senior_thesis.git.`.

5 The Eye Tracking Experiment

5.1 Experiment Design

I designed eight Excel tasks that simulates the scenario of analyzing the data from a social study. The simulated social study investigates the population, marriage status, and church attendance of a small town.

I chose Microsoft Excel as the test platform because it has a complex user interface, and large user base. Notice that the first 7 subjects performed the experiment on Office 2010 and other subjects did the experiment on the newer version of Excel from Office 365 due to a system upgrade. A lot of users, especially less experienced users, will get confused while using Excel. If the gaze patterns or other eye-related features are found to indicate confusion, then the result is generalizable than the result found on the test platform used in Lalle et al.'s project. They used a self-made Data Visualization software called ValueChart, which contains an interactive UI for visualizing preferences. The software is specifically designed for their research purposes. Even though the usability of the software was investigated in multiple related research, the user base is negligible compared with Microsoft Excel. In addition, their test platform focused on the confusion caused by not only the complexity of the User Interface but also the complexity of making preferential choices.

The primary purpose of these eight tasks is to simulate the procedure of mapping planned interaction to the location of the menu options or buttons and interact with them in correct order. The following scenario is a confusion event: when users plan a sequence of actions to interact with the User Interface to actuate some intentions, they cannot find the correct location of the corresponding menu options or buttons or the sequence of interactions is not correct after making several failed attempts. Thus, the instruction of each task has two parts: the goal of the task and the steps to complete the goal. For example, in the first task, I asked participants to calculate the total frequency of all categories of data. In the context of the simulated social study, the total frequency represents the total participants of the social study. This task simulates the case that the researcher wants to know how many participants participated the study so that the researcher can calculate the percentage for each class of subjects later on. Thus, the intention of this simulated researcher (the experiment subject) is to get the sum of the population in each data class. In order to actuate this intention, the subject needs to form a plan to interact with the User Interface: find the summation function in Excel, then select all data categories, and finally perform the calculation. Then the subject must map this plan to the specific location of menu options corresponding to each step of the plan. For example, one of the possible thought process that the subject may go through the following mental process in order to complete task 1 is the following:

Step 1: Where to put the result of summation?

The subject reads the instruction, which tells the subject to look for the cell next to the cell named ″Total″

The corresponding reaction of the subject is to visually search for the cell named ″Total″ and then move the gaze right next that cell, which would find the correct position. The cursor will also follow the gaze and locate on the target cell once the subject has located the cell.

Step 2: Where is the insert function button located?

Step 3: Where is the summation function located?

Step 4: Where does the data range start and end?

I divided tasks into two groups by difficulties: easy and hard. The difficulty level is manifested by the number of steps required to complete the goal and the concreteness of instructed step. In order to let subjects fully exhibit the potential patterns in visual search, I deliberately avoided giving instructions on completing the task through keyboard short cuts. All tasks have some likelihood to make participants confused.

Before letting the participants do the actual tasks, I also give them a similar spreadsheet containing completed data and charts so that they can get familiar with the User Interface of Microsoft Excel, regardless of their familiarity and skill level. Another purpose of giving them this exercise material is to let them practice the confusion reporting protocol.

The very first version of the experiment, participated by 2 subjects, designs did not require the participants to verbally report confusion. Instead, I asked them to use the left key and right key of mice to report confusion, since the experiment control software of the eye-tracker can record mouse clicks. This protocol prevents eye movements of participants from being influenced by talking. However, the experiment control software is not always stable. It crashed several times while I was doing the experiment and lose all the recorded data. Even when the software work properly, it may not record mouse clicks. Thus, I have to let subjects verbally report confusion since this is a more reliable way.

5.2 Experiment Setup

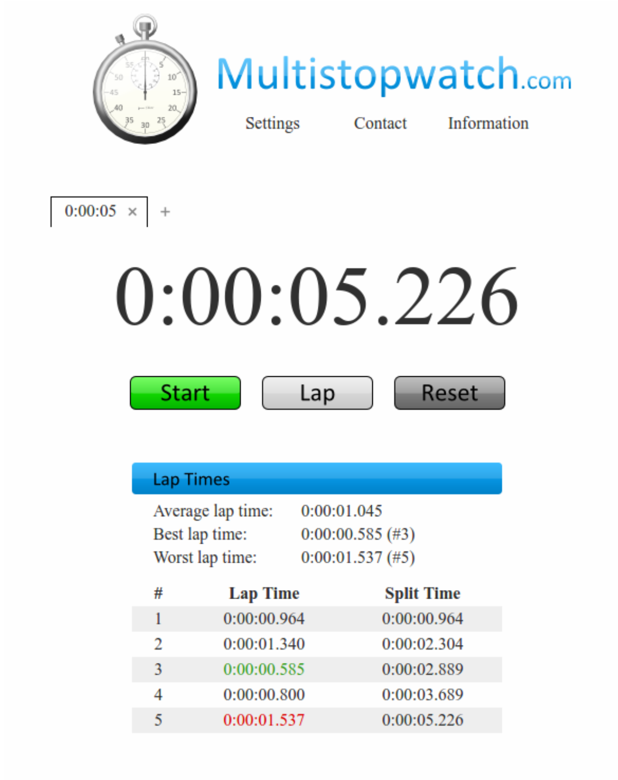


Figure 8: Stopwatch used in this project

The lab environment is consists of two parts: the control area and the experiment room. The experiment room is on the left side of the control area. Both the control area and the experiment room have a screen that connects to a computer, which is upgraded in the experiment done to the last three subjects. Two sets of mice and keyboards are used, which one is for the experimenter, and the other is for the subject. The experiment room has a light switch to control the brightness in the room. In order to minimize the effect of lighting on the eye-tracker, I set the brightness of the room to the lowest level. I used a GazePoint GP3 eye tracker to collect eye data from participants. The eye tracker uses IR reflection from participants pupils to locate eye movements. The sampling rate of this eye-tracker is 60 Hz. It has 0.5 1 degree of visual angle accuracy.

The experiment control software comes along with the eye tracker. The software has a data collector and a monitor program. The data collector manages experiment data and the status of the eye-tracking device. I use this data collector to initialize recording and export data. The monitor program displays the real-time recordings captured by the eye tracker. The monitor program also performs calibration on subjects' head distance and the accuracy of eye-tracking.

Figure 9: Lab Environment, Experimenter sits outside

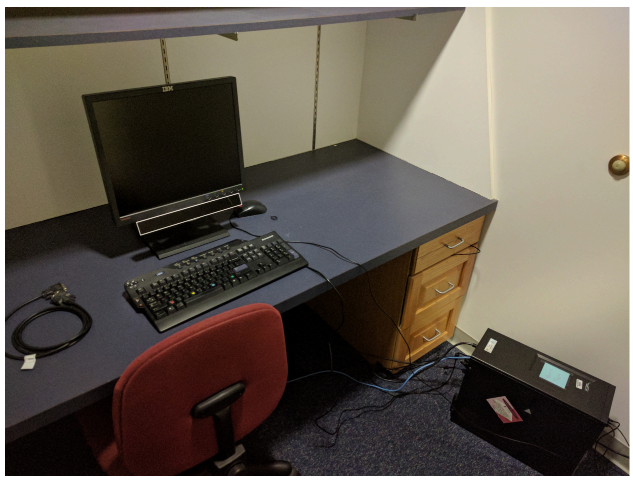


Figure 10: Lab Environment, subject sits outside

Since the experimenter and the subject share the same computer, both of them need to control the computer asynchronously. I installed two sets of mice and keyboard for such purpose. I prepare the experiment materials and control the experiment controlling program. After I start recording, I hand over the control to subjects so they can do the tasks without interference.

The version of Excel used on first 7 subjects was 2010. Due to a system upgrade, the version has been updated to the latest version (Office 365). Some changes in the User Interface happened after the upgrade. I made the experiment instruction according to the older version so the software upgrade brought some minor discordance to the instructions. Some subjects got confused because the steps did not completely match the UI components in the newer version.

The lab computer has an Intel i7 2400 CPU and 4 GB of memory before the upgrade. The RAM capacity increased to 8 GB after the upgrade. An upgrade was performed because the experiment control software

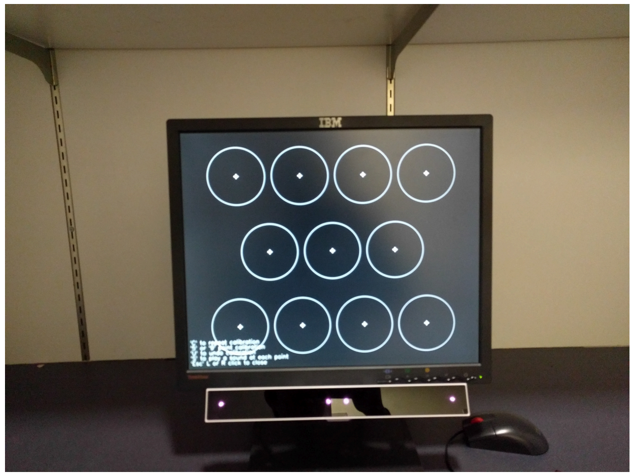


Figure 11: Eye tracker and calibration screen

requires huge amount of resources to operate properly. The software is very likely to crash and lose data when the memory runs out.

I used a web-based lap timer to record step intervals.

More details about the experiment setup is available in the Appendix.

5.3 Pre-Experiment Procedure

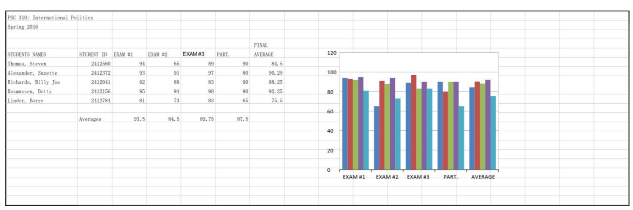


Figure 12: Warm up exercise given to participants

I hand out the background questionnaire, the consent form, and the pre-experiment instruction (see A) when a participant comes to the lab. The consent form introduces the basic content of the experiment and asks whether the participant agrees to be the experiment subject. I tell subjects that they are video-taped. The background questionnaire asks for the background of the participant, their age, their Excel skill level, their computer skill level, and their visual ability (whether they have visual impairment, e.g. nearsightedness). The pre-experiment instruction provides details of doing calibration and completing the pre-experiment exercise. The full content of the consent form, background questionnaire, and the pre-experiment instructions are available in the Appendix.

The first stage of calibration process adjust the head distance and pose from the screen and the eye tracker. I require each participant to calibrate their biometrics for the eye-tracker. The subject sits on a chair and adjust the height of the chair to find a comfortable position. I ask participants to find the position they are comfortable with so that they do not constantly move and adjust their body position during the experiment. Some participants may get too relaxed and reduce their body height, which will break the tracking of the eye tracker. To prevent that from happening, I adjust the angle of the eye tracker so that participants' eyes are positioned at the bottom half of the screen in the monitor program. Then I ask participants to adjust their head distance from the eye tracker. A dot on the monitor program will move left if the participant is too close to the screen. The dot will move to the right if the participant is too far away. Whenever they change their head distance of height of the chair, I adjust the angle of the eye tracker to make sure the camera is directly facing the participants. When the dot is positioned roughly in the middle, then the head distance and the angle of the eye tracker is correctly calibrated.

The second stage of calibration evaluates the accuracy of eye tracing by matching the device-inferred gaze with points on the screen. The participant performs a nine-point calibration. At the beginning of the experiment, a shrinking white circle with a red center appears on the top left of the screen. The participant need to fixate the sight on that point until it disappears. Then the point will reappear on the top center. Then the point will keep moving to the right until it reaches the down right corner of the screen. To see a complete procedure of eye tracking calibration, the appendix has screen shots showing the positions of the nine dots. After the calibration, I evaluate the accuracy of gaze tracing. In the evaluation mode, the monitor program displays nine circles of equal sizes with nine crosses as the centers of these circles. Ideally, the eye tracker will exactly show where the participant is looking. When the participant looks at another location on the screen, the gaze path will be displayed smoothly. If stable tracking cannot be established, gaze points will appear randomly. I point the mouse cursor to the center of the top left circle. The participant looks at where I point to. Then I move the mouse cursor from left to right, top to left until the participant has seen all the centers of the circle. If the gaze point almost matches the centers, then I will end the calibration session and move on the exercise stage. If not, I will repeat the calibration until the accuracy becomes acceptable.

After the calibration process, I give each participant a sample spreadsheet to get familiar with the User Interface of Excel. Each participant has three minutes to practice. While the participant is trying out functions of Excel, I explain the confusion report process and functions involved in the real experiment. I verbally give instructions on how they should interact with the User Interface. To let them practice data calculation, I ask them to click on the cells where data are calculated using various functions. Then they will see the formula that produce the result of calculation. To let participants get familiar with chart building, I let them enable the data selection options in the chart, and they can see the cells included in the chart. For cell formatting and data formatting, I use mouse cursor to point out the locations of the buttons related to these two tasks. After this step, I let them practice the confusion report protocol. While practicing the confusion report protocol, I tell them that minimal hint will be given if they are confused. When they get confused, I only say 'yes' or 'no' to suggest whether they are heading towards the right direction.

5.4 Formal Experiment

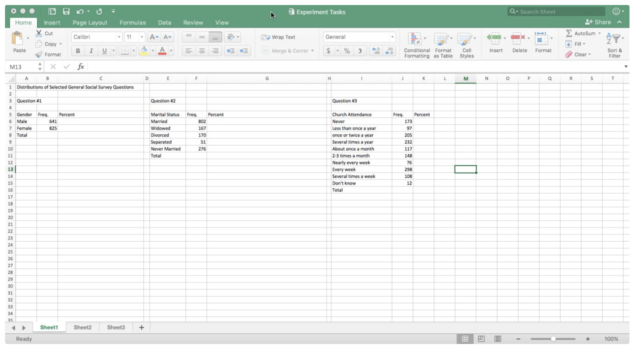


Figure 13: Experiment Task given to participants

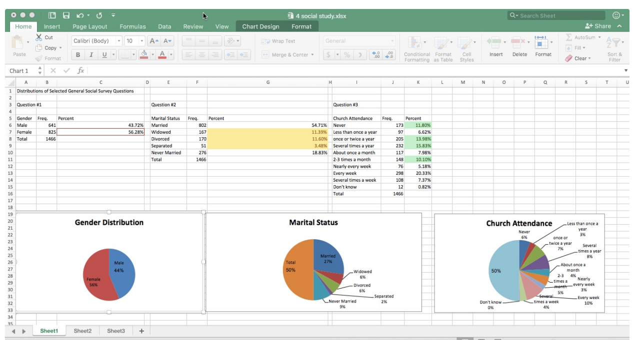


Figure 14: Finished Experiment Task

After the warm up, the participant starts to do experiment tasks along with the instruction document. I divided eight tasks to two groups according to their difficulty levels. I assign the tasks in sequential order to participants. Before each task, they have sufficient time to read the task description and understand what to do. I wait until they report to be prepared. I explain the content of the task if they have any question. When the participant gets ready, I start timing their steps of each task and initialize eye tracking and screen recording simultaneously. Then I immediately hand over the control to the participant. When the participant starts to do the task, I keep watching the screen recordings. I always make sure that participants do not to leave the scope of the eye tracker. If they do, I remind them to adjust their head position and body height to move back into the scope of the eye tracker. The first time interval is immediately recorded by clicking the lap button on the timer when the participant clicks a button or menu tab. I wait until the first step of that task is completed. Then I immediately insert a very short time interval as the separator between the recorded time interval and the next time interval (time elapse is approximately 0.3 second). After that I repeat the process to record the next time interval until the participant complete the task. If the participant reports confusion, I will write down the start time of the confusion event, and write down the end time of confusion event when the participant reports to be not confused. When the participants get confused, they tend to ask questions about how to proceed correctly. I only give minimum amount of hint to avoid the effect on eye movement[15]. If participants get confused and ask me how to proceed, I only give minimal hints if they ask questions during confusion interval. If they are heading to the right direction, I say ”yes”. If they are not doing the task the right way, I say ”no”. Participants can keep trying until either finishing the task correctly or giving up. After finishing all the tasks, participants get debriefed and receive $8 of cash rewards.

6 Data Processing

I need to process two kinds of raw data before making the training set: eye tracking data from the eye tracker and raw time frame data from a timer. The eye tracking data contains many metrics related to eye movements and biometrics of the subjects' eyes. Since I am only interested in gaze, fixation, cursor position and time, I extracted these columns from the CSV files containing the data of each participant. The raw time interval data consists of the index of lap time, the length of lap time, and the end of lap time. When I copy the time intervals from the timer, all data go into one column, and I need to label these intervals with their confusion class label (True or False). I need to format these time intervals into four separate columns: elapse, start time, end time, and confusion. After formatting all the data, I label them with set number and task number. The set number represent the subjects, and task number represent the order of the tasks.

The first training set is directly generated from source data by taking needed features: fixation (FPOGX, FPOGY), gaze (BPOGX, BPOGY), and cursor positions (CX, CY). Then I use the top left corner of the screen as reference point to calculate the Euclidean distance of each x and y coordinate data using $euclidian = \sqrt{x^2 + y^2}$. Then I use reference data to manually label the feature data as confusion or non-confusion.

To generate the second feature set, I wrote two Python scripts to format the time data and eye tracking data. The script reference data formatter will format the raw time data and combine all the formatted data to one file. The combined data set is called reference data because it is one of the input used by the training data generator to generate training data. The purpose of reference data is telling the training data generator the start and end time of extracting preprocessed eye tracking data and label the generated data with the value of confusion label. The script source data extractor selects the columns I need and combine data from all subjects. If some data sets contain missing data, the source data extractor will use linear interpolation method to fill those empty data.

The training set generator will take formatted reference data and source data to generate training set. The script reads the start time and end time in the reference data. The time intervals in the reference data are sliced by a 5-second time windows used in Lalle et al.'s work. Then the time intervals are used to extract source data from the source data set. After that, functions in the feature generator are used to generate feature data. I wanted to measure the spread of the gaze points, fixation points, and cursor positions. Thus, I applied standard deviation to the data in the time window. Then I used the sum of square of the standard deviation data of x and y coordinate data to combine the standard deviation of the three selected feature categories. When this process is completed, the training data is ready to be used by WEKA [11].

7 Result And Analysis

Ten participants participated the study. Data from 8 subjects were used, and the rest were discarded because non confusion data and time interval data were recorded. Among the 80 data sets (8 tasks from 10 participants), I selected 14 data sets that contain confusion to balance the non-confusion and confusion classes.

The first feature set contains 218306 data points. The the second feature set generated from the training data generator contains 744 training instances with 414 non-confusion class instances and 333 confusion class instances for the second feature set. Notice that the difference in the data size is caused by compressing the data in a 5-second window.

Descriptive statistics (Quantitative data):

Statistic	Task 1	Task 2	Task 3	Task 4	Task 5	Task 6	Task 7	Task 8
Nbr. of observations	8	8	8	8	8	8	8	8
Minimum	25.711	51.865	27.123	184.427	23.345	47.183	24.711	29.472
Maximum	246.478	209.300	161.624	534.142	207.180	319.637	99.639	63.432
Range	220.767	157.435	134.501	349.715	183.835	272.454	74.928	33.960
1st Quartile	62.142	92.320	34.157	270.655	33.346	93.650	37.638	32.861
Median	66.043	147.520	58.724	342.690	55.004	106.357	44.045	42.123
3rd Quartile	98.284	161.271	90.649	409.310	85.895	147.446	63.862	51.323
Mean	98.028	132.549	71.742	343.405	75.647	133.219	52.424	43.815
Variance (n-1)	5953.730	2621.214	2361.668	13455.026	3869.643	6841.291	610.411	168.032
Standard deviation (n)	72.177	47.891	45.458	108.504	58.189	77.370	23.111	12.126
Standard deviation (n-1)	77.160	51.198	48.597	115.996	62.206	82.712	24.706	12.963

Figure 15: Task Completion Time Summary

14 data sets containing confusion are selected for evaluation. The task completion time is shown in figure 7. The task completion time for each task is in A.4. Task 4 is the most confusing time for participants since the difficulty level is designed to be high. Task 6 is the second most confusing task.

An analysis is done on the distribution of gaze points and fixations. Most of the gaze and fixation points are distributed between 0 to 0.3. This represents that participants are mostly looking at the top left to middle

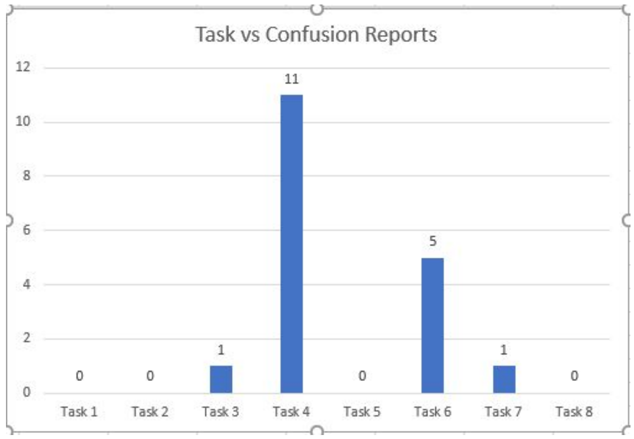


Figure 16: Confusion reports of each task

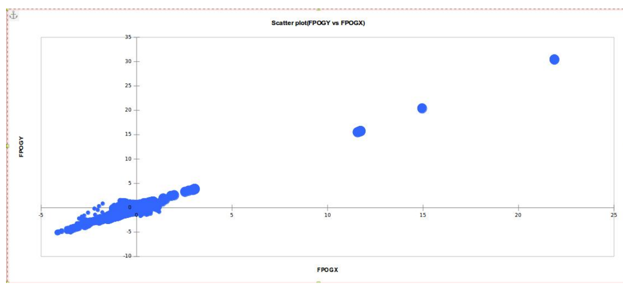


Figure 17: Fixation X and Y coordinate distribution

left part of the screen. Since the experiment design is to let participants perform tasks on the left side of the screen, then this observation matches the fact of experiment design. Data quality is quite low, since according to the GazePoint API manual, the validate coordinate data of gaze should only appear between 0 to 1, but in figure 7 there are large amount of data points that lie beyond that range. A possible explanation of this phenomenon is that experiment tasks involves using keyboard to enter data so participants may look down to make sure hitting on the right key. Participants also tend to change their body distance from time to time, and the eye tracker may lose the consistent tracking on their eyes. Another important fact is that participants will blink their eyes in 2 seconds to 5 seconds. Blinking also makes eye tracker losing the tracking.

After generating all the feature data, 60% of both feature set are used to train models and 40% of the feature data are used for testing. Cohen's Kappa[4] is used along with classification accuracy is used to evaluate both models. Cohen's kappa coefficient is a statistic which measures inter-rater agreement for qualitative (categorical) items[4]. The kappa statistics of the model build on the first feature set is 0.14, and the kappa statistics of the model build on the second feature set is 0.07. The model using the first feature set is better than the second model in terms of classification accuracy and kappa statistics. However, the

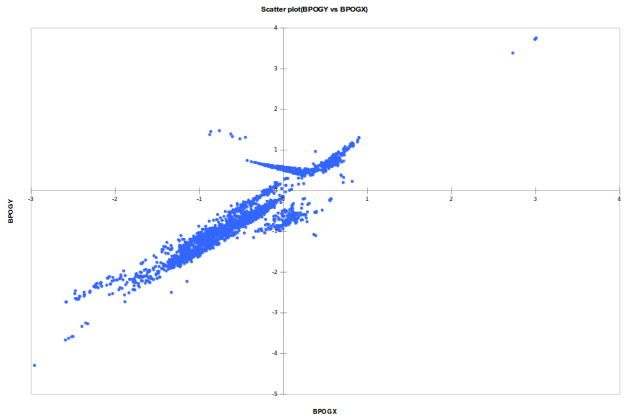
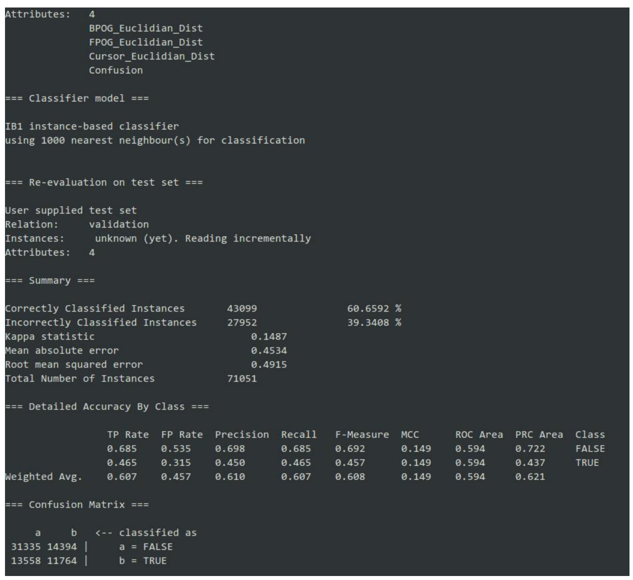


Figure 18: Gaze X and Y coordinate distribution



```
Attributes:   4
              BPOG_Euclidian_Dist
              FPOG_Euclidian_Dist
              Cursor_Euclidian_Dist
              Confusion

=== Classifier model ===

IB1 instance-based classifier
using 1000 nearest neighbour(s) for classification


=== Re-evaluation on test set ===

User supplied test set
Relation:     validation
Instances:     unknown (yet). Reading incrementally
Attributes:   4

=== Summary ===

Correctly Classified Instances       43099               60.6592 %
Incorrectly Classified Instances     27952               39.3408 %
Kappa statistic                          0.1487
Mean absolute error                      0.4534
Root mean squared error                  0.4915
Total Number of Instances            71051

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.685    0.535    0.698      0.685    0.692      0.149    0.594     0.722     FALSE
                 0.465    0.315    0.450      0.465    0.457      0.149    0.594     0.437     TRUE
Weighted Avg.    0.607    0.457    0.610      0.607    0.608      0.149    0.594     0.621

=== Confusion Matrix ===

     a     b   <-- classified as
 31335 14394 |     a = FALSE
 13558 11764 |     b = TRUE
```

Figure 19: K Nearest Neighbor Algorithm classification result on the first feature set

kappa statistics must be larger than 0.5 to be significant. Thus, both models show no significance. Logistics regression is used on both data set. The cursor feature in both feature sets are found to be very statistical significant (p ¡ 0.0001) because only the cursor feature data rejects the null hypothesis of Chi Square.

8 Conclusion

No significant result has been found in the time span of this project. Eye tracking data were collected and analyzed. A training data set is generated to train models to recognize confusion. The best performing model is built by K* algorithm, which reached 54% of classification accuracy and kappa statistics of 0.09. Since kappa statistics is too low to be meaningful, I conclude that the model has no sufficient accuracy

```
=== Summary ===

Correctly Classified Instances         409               54.7523 %
Incorrectly Classified Instances       338               45.2477 %
Kappa statistic                          0.0789
Mean absolute error                      0.4792
Root mean squared error                  0.5392
Relative absolute error                 96.9793 %
Root relative squared error            108.4738 %
Total Number of Instances              747

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.616    0.538    0.588      0.616    0.601      0.079    0.536     0.571     False
                 0.462    0.384    0.492      0.462    0.477      0.079    0.536     0.478     True
Weighted Avg.    0.548    0.469    0.545      0.548    0.546      0.079    0.536     0.530

=== Confusion Matrix ===

   a   b   <-- classified as
 255 159 |   a = False
 179 154 |   b = True
```

Figure 20: KStar Algorithm classification result on the second feature set

Term		Estimate	Std Error	ChiSquare	Prob>ChiSq
Intercept		0.0791906	0.013755	33.15	<.0001*
FPOG_Euclidean_Dist		-0.0476478	0.0076074	39.23	<.0001*
Cursor_Euclidean_Dist		0.3938179	0.0213947	338.83	<.0001*

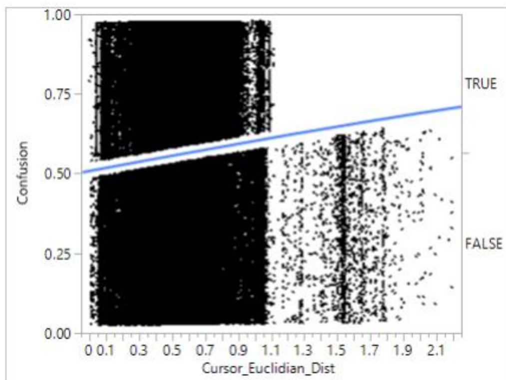


Figure 21: Logistic Regression result on the cursor feature in Feature Set 1

to predict confusion. Given the limitations in this project, future work should focus on designing better experimental procedures, recruiting more subjects, improving data processing, and make new training features.

9 Limitation, Reflection and Future Work

There are several major limitations in this project. The first major limitation is having a unnecessarily complicated experiment. I realized this issue when I found that only data that contain confusion are useful. I could have design at most four tasks with confusing steps so that I can collect both confusion data and non-confusion data while taking less time and having more balanced data set. I also think letting participant verbally report confusion negatively affects the eye tracking data. The classification method is also problematic. Since eye tracking data is produced in relation of time, I should use time series classification method, or develop proper geometric features to summarize them. I only used standard deviation in this

Term		Estimate	Std Error	ChiSquare	Prob>ChiSq
Intercept		0.27960788	0.0089414	977.88	<.0001*
std_fp		0.00944844	0.0056372	2.81	0.0937
std_cursor		0.45865963	0.0402538	129.83	<.0001*

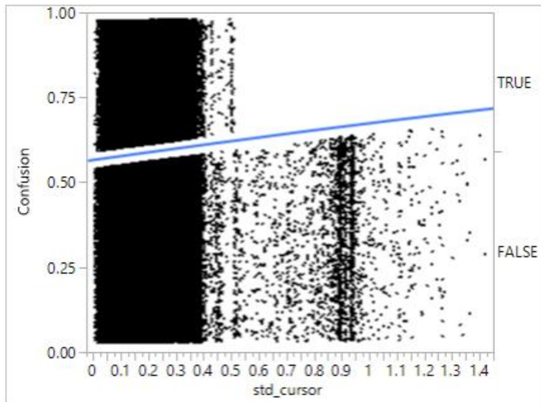


Figure 22: Logistic Regression result on the cursor feature in Feature Set 1

project, which is overly simplistic. The way of how I combine the data from x axis and y axis is arbitrary. I should use the Euclidean distance from the average point of the time window to represent them.

Since no real meaningful result is found in this project, I need to work on additional features, update the experiment method, and improve the data processing. Several aspects of this project can be improved. The first aspect of improvement is to deepen the understanding of the relation between eye movement and confusion by researching on more related literature. The second aspect is to improve the experiment design. The experiment design in this project may be over complicated so I need simplify it properly so that I can record more confusion events without too much interference. Since the eye tracking data contain a lot of empty values, I need to improve the hardware and find a better way to handle missing values. In term of developing new features, I need to focus on either summarizing the data using the geometric patterns of gaze or using time series classification methods to classify eye tracking data. I also need to recruit more subjects to increase the data size.

Appendices

A Appendix

A.1 Raw Data and Feature Data

A	B	C	D	E	F	G	H	I	J	K	L	M	N
MEDIA_ID	MEDIA_NAME	CNT	TIME(2017/02/20 13:51:45.313)	TIMETICK(f=3312871)	FPOGX	FPOGY	FPOGS	FPOGD	FPOGID	FPOGV	BPOGX	BPOGY	BPOGV
0	NewMedia	0	0	13037028874	0.1769	-0.04929	0	0	1	1	0.12283	-0.12823	1
0	NewMedia	1	0.01636	13037083083	0.17285	-0.05972	0	0.01635	1	1	0.15264	-0.11186	1
0	NewMedia	2	0.03296	13037138028	0.17111	-0.06671	0	0.03295	1	1	0.16063	-0.10863	1
0	NewMedia	3	0.04932	13037192080	0.1698	-0.07195	0	0.04931	1	1	0.16063	-0.10863	1
0	NewMedia	4	0.0658	13037246675	0.16672	-0.07753	0	0.06579	1	1	0.14205	-0.12222	1
0	NewMedia	5	0.08215	13037300859	0.17286	-0.06863	0	0.08215	1	1	0.22815	0.01152	1
0	NewMedia	6	0.09851	13037355172	0.17789	-0.06134	0	0.09851	1	1	0.22815	0.01152	1
0	NewMedia	7	0.11499	13037409824	0.18242	-0.05762	0	0.11499	1	1	0.23224	-0.01666	1
0	NewMedia	8	0.13135	13037464013	0.17968	-0.06194	0	0.13134	1	1	0.14683	-0.11376	1
0	NewMedia	9	0.14795	13037518908	0.17733	-0.06564	0	0.14795	1	1	0.14683	-0.11376	1
0	NewMedia	10	0.16431	13037573240	0.19682	-0.03953	0	0.1643	1	1	0.46971	0.32603	1
0	NewMedia	11	0.18091	13037628088	0.20769	-0.00286	0	0.1809	1	1	0.37071	0.54713	1
0	NewMedia	12	0.19714	13037682042	0.20769	-0.00286	0	0.19714	1	1	0.37071	0.54713	0
0	NewMedia	13	0.21545	13037742657	0.20769	-0.00286	0	0.21545	1	1	0.37071	0.54713	0
0	NewMedia	14	0.22998	13037790829	0.20769	-0.00286	0	0.22998	1	1	0.37071	0.54713	0
0	NewMedia	15	0.24646	13037845468	0.21345	0.00604	0	0.22998	1	0	0.37071	0.54713	0
0	NewMedia	16	0.2627	13037899227	0.21345	0.00604	0	0.22998	1	0	0.37071	0.54713	0
0	NewMedia	17	0.2793	13037953916	0.21345	0.00604	0	0.22998	1	0	0.37071	0.54713	0
0	NewMedia	18	0.29834	13038017031	0	0	0	0.22998	1	0	0.37071	0.54713	0
0	NewMedia	19	0.31226	13038063222	0	0	0	0.22998	1	0	0.37071	0.54713	0
0	NewMedia	20	0.32874	13038117768	0	0	0	0.22998	1	0	0.37071	0.54713	0
0	NewMedia	21	0.34509	13038171973	0	0	0	0.22998	1	0	0.37071	0.54713	0
0	NewMedia	22	0.36133	13038225951	0	0	0	0.22998	1	0	0.37071	0.54713	0
0	NewMedia	23	0.37781	13038280621	0	0	0	0.22998	1	0	0.37071	0.54713	0
0	NewMedia	24	0.39453	13038335859	0	0	0	0.22998	1	0	0.37071	0.54713	0
0	NewMedia	25	0.41077	13038389586	-1.35559	-1.6586	0	0.22998	1	0	-2.71119	-3.31719	1
0	NewMedia	26	0.42712	13038444008	-1.37194	-1.54535	0	0.22998	1	0	-1.40464	-1.31885	1
0	NewMedia	27	0.44373	13038498681	-1.83902	-1.98534	0	0.22998	1	0	-1.40122	-1.31996	1
0	NewMedia	28	0.45996	13038552688	-1.61137	-1.64513	0	0.22998	1	0	-1.38372	-1.30493	1
0	NewMedia	29	0.47632	13038607018	-1.41863	-1.36935	0	0.22998	1	0	-1.03316	-0.81779	1
0	NewMedia	30	0.49463	13038667598	-1.15002	-0.98017	0	0.22998	1	0	-1.03316	-0.81779	1
0	NewMedia	31	0.50928	13038715881	-1.27221	-1.14999	0	0.22998	1	0	-1.3944	-1.31981	1
0	NewMedia	32	0.52576	13038770629	-1.29957	-1.17021	0	0.22998	1	0	-1.3543	-1.21066	1
0	NewMedia	33	0.54211	13038824947	-1.36767	-1.24704	0	0.22998	1	0	-1.3543	-1.21066	1
0	NewMedia	34	0.55847	13038879106	-1.33312	-1.24295	0	0.22998	1	0	-1.29857	-1.23886	1
0	NewMedia	35	0.57507	13038933984	-1.43086	-1.36317	0	0.22998	1	0	-1.62635	-1.60361	1
0	NewMedia	36	0.59277	13038992517	-1.51709	-1.48203	0	0.22998	1	0	-1.62635	-1.60361	1
0	NewMedia	37	0.60791	13039042648	-1.49284	-1.44724	0	0.22998	1	0	-1.46859	-1.41246	1
0	NewMedia	38	0.62427	13039096869	-1.49598	-1.44035	0	0.22998	1	0	-1.50227	-1.42657	1
0	NewMedia	39	0.64075	13039151503	-1.49657	-1.43725	0.59277	0.04797	2	1	-1.49832	-1.42795	1
0	NewMedia	40	0.6571	13039205843	-1.50614	-1.45622	0.59277	0.06433	2	1	-1.54443	-1.53207	1

Figure 23: An example of data generated from eye tracker

O	P	Q	R	S	T	U	V	W	X	Y	Z	AA	AB	AC	AD	AE
CX	CY	CS	USER	LPCX	LPCY	LPD	LPS	LPV	RPCX	RPCY	RPD	RPS	RPV	BKID	BKDUR	BKPMIN
1.54453	0.09766	0		0.31901	0.77171	17.72471	1	1	0.65846	0.74988	16.40849	1.35758	1	0	0	22
1.54453	0.09766	0		0.31739	0.7723	17.09442	1	1	0.65576	0.75169	17.28342	1.35758	1	0	0	22
1.54453	0.09766	0		0.31567	0.77288	17.39791	1	1	0.65349	0.75148	16.21966	1.35758	1	0	0	22
1.54453	0.09766	0		0.31479	0.77267	17.70088	1	1	0.65374	0.75213	16.14207	1.35758	1	0	0	22
1.54453	0.09766	0		0.31155	0.7728	16.89957	1	1	0.64928	0.7527	17.19653	1.35758	1	0	0	22
1.54453	0.09766	0		0.30874	0.7735	17.20368	1	1	0.64654	0.75223	16.47931	1.35758	1	0	0	22
1.54453	0.09766	0		0.30818	0.77261	16.8234	1	1	0.64623	0.75201	15.58576	1.35758	1	0	0	22
1.54453	0.09766	0		0.30287	0.77411	17.28258	1	1	0.64013	0.75164	16.16109	1.35758	1	0	0	22
1.54453	0.09766	0		0.29956	0.77351	16.95	1	1	0.63723	0.7506	15.93799	1.35758	1	0	0	22
1.54453	0.09766	0		0.29921	0.77356	17.49132	1	1	0.63704	0.7514	16.11197	1.35758	1	0	0	22
1.54453	0.09766	0		0.29505	0.77531	14.59347	1	1	0.6312	0.75079	15.19672	1.35758	1	0	0	22
1.54297	0.09863	0		0.29505	0.77531	14.59347	1	0	0.62715	0.75185	6.91994	1.35758	1	0	0	22
1.54297	0.09863	0		0.29505	0.77531	14.59347	1	0	0.62715	0.75185	6.91994	1.3495	0	0	0	22
1.51953	0.10645	0		0.29505	0.77531	14.59347	1	0	0.62715	0.75185	6.91994	1.34142	0	0	0	22
1.51953	0.10645	0		0.29505	0.77531	14.59347	1	0	0.62715	0.75185	6.91994	1.33334	0	430	0	22
1.51953	0.10645	0		0.29505	0.77531	14.59347	1	0	0.62715	0.75185	6.91994	1.33334	0	430	0	22
1.34063	0.14746	0		0.29505	0.77531	14.59347	1	0	0.62715	0.75185	6.91994	1.33334	0	430	0	22
1.24219	0.17383	0		0.29505	0.77531	14.59347	1	0	0.62715	0.75185	6.91994	1.33334	0	430	0	22
1.24219	0.17383	0		0.29505	0.77531	14.59347	1	0	0.62715	0.75185	6.91994	1.33334	0	430	0	22
0.99453	0.21191	0		0.29505	0.77531	14.59347	1	0	0.62715	0.75185	6.91994	1.33334	0	430	0	22
0.99453	0.21191	0		0.29505	0.77531	14.59347	1	0	0.62715	0.75185	6.91994	1.33334	0	430	0	22
0.99453	0.21191	0		0.29505	0.77531	14.59347	1	0	0.62715	0.75185	6.91994	1.33334	0	430	0	22
0.63594	0.20801	0		0.29505	0.77531	14.59347	1	0	0.62715	0.75185	6.91994	1.33334	0	430	0	22
0.63594	0.20801	0		0.29505	0.77531	14.59347	1	0	0.62715	0.75185	6.91994	1.33334	0	430	0	22
0.63594	0.20801	0		0.29505	0.77531	14.59347	1	0	0.62715	0.75185	6.91994	1.33334	0	430	0	22
0.6	0.20703	0		0.56484	0.7135	16.64287	1	1	0.62715	0.75185	6.91994	1.33334	0	0	0.18074	23
0.6	0.20703	0		0.22428	0.74464	17.38039	1	1	0.56063	0.71298	16.75602	1.33334	1	0	0	23
0.54844	0.19727	0		0.22282	0.74232	14.3316	1	1	0.5584	0.71014	13.89876	1.3265	1	0	0	23
0.49609	0.19238	0		0.21529	0.74381	15.27745	1	1	0.55153	0.71022	15.55097	1.31967	1	0	0	23
0.49609	0.19238	0		0.21178	0.74578	16.72069	1	1	0.54843	0.71077	17.09052	1.31283	1	0	0	23
0.47266	0.18652	0		0.2108	0.74403	14.46699	1	1	0.54634	0.70895	15.17267	1.31283	1	0	0	23
0.47266	0.18652	0		0.2052	0.74633	16.42965	1	1	0.54099	0.70883	15.29706	1.31283	1	0	0	23
0.47266	0.18652	0		0.20258	0.747	16.35861	1	1	0.53869	0.70888	16.09649	1.31283	1	0	0	23
0.47188	0.18359	0		0.20174	0.74576	15.24491	1	1	0.53731	0.70835	16.42067	1.31283	1	0	0	23
0.47422	0.18164	0		0.19862	0.74761	18.13155	1	1	0.53408	0.70727	16.45733	1.31283	1	0	0	23
0.47422	0.18164	0		0.19705	0.74693	17.29812	1	1	0.53226	0.70598	16.05844	1.31283	1	0	0	23
0.47969	0.17578	0		0.19656	0.74681	17.68846	1	1	0.53123	0.70568	17.10298	1.31283	1	0	0	23
0.47969	0.17578	0		0.19483	0.74693	17.65758	1	1	0.52962	0.70437	16.69093	1.31283	1	0	0	23
0.47969	0.17578	0		0.19441	0.74699	18.33339	1	1	0.52929	0.70416	16.67178	1.31283	1	0	0	23
0.47969	0.17578	0		0.19414	0.74735	18.78346	1	1	0.52852	0.70356	17.04187	1.30599	1	0	0	23
0.47969	0.17578	0		0.19393	0.74753	18.61394	1	1	0.52821	0.70153	16.86661	1.29915	1	0	0	23
0.50156	0.12988	0		0.19449	0.74652	17.88545	1	1	0.52841	0.70166	16.84819	1.29232	1	0	0	23
0.50234	0.12695	0		0.19449	0.7467	17.95339	1	1	0.52871	0.70103	17.32963	1.29232	1	0	0	23
0.50234	0.12695	0		0.19604	0.74638	17.86966	1	1	0.52966	0.69937	17.08057	1.29232	1	0	0	23
0.50234	0.12695	0		0.19731	0.74657	18.24882	1	1	0.53087	0.69813	16.37467	1.29232	1	0	0	23
0.50313	0.12402	0		0.19773	0.74638	17.95782	1	1	0.53191	0.69861	17.63626	1.3004	1	0	0	23
0.50313	0.12402	0		0.20101	0.74588	18.09243	1	1	0.5341	0.69607	16.65859	1.30848	1	0	0	23

Figure 24: An example of data generated from eye tracker cont'd

A	B	C	D
BPOG_Euclidian_Dist	FPOG_Euclidian_Dist	Cursor_Euclidian_Dist	Confusion
0.895361037	0.965415905	1.54963085	FALSE
1.026350253	0.995827106	1.54963085	FALSE
1.028659459	1.006761657	1.54963085	FALSE
1.192801572	1.08259959	1.545307291	FALSE
0.806456468	0.937819478	1.545307291	FALSE
0.807886476	0.892702215	1.545307291	FALSE
0.744480474	0.786234317	1.282669628	FALSE
0.891794869	0.836750247	1.282669628	FALSE
0.890371975	0.854153661	1.282669628	FALSE
0.779015678	0.833917989	1.081896267	FALSE
0.792846868	0.825207626	1.081896267	FALSE
0.791264779	0.819276664	0.931455457	FALSE
0.757928369	0.810291918	0.855781666	FALSE
0.771141306	0.805220811	0.855781666	FALSE
0.768103087	0.800977962	0.828030445	FALSE
0.719253801	0.792552641	0.828030445	FALSE
0.748582645	0.788527559	0.828030445	FALSE
0.747071546	0.785064082	0.772672422	FALSE
0.701178869	0.778486073	0.714737612	FALSE
0.665133781	0.770141904	0.714737612	FALSE
0.663636111	0.74174486	0.714737612	FALSE
0.53302112	0.634625761	0.625274327	FALSE
0.578523919	0.614975875	0.625274327	FALSE
0.577029237	0.562856616	0.625274327	FALSE
0.58411802	0.572481969	0.625274327	FALSE
0.631526139	0.590974268	0.625274327	FALSE
0.631526139	0.615357245	0.625274327	FALSE
0.677576172	0.646111199	0.625274327	FALSE
0.870616202	0.715079513	0.625274327	FALSE
0.833810065	0.372668423	0.625274327	FALSE
0.833810065	0.372668423	0.625274327	FALSE
0.833810065	0.372668423	0.625274327	FALSE
0.833810065	0.331635322	0.625274327	FALSE
0.833810065	0.290932066	0.625274327	FALSE
0.833810065	0.250719355	0.625274327	FALSE
0.833810065	0.211277471	0.625274327	FALSE
0.833810065	0.173134028	0.625274327	FALSE
0.833810065	0.137374892	0.625274327	FALSE
0.833810065	0.106431093	0.625274327	FALSE
0.833810065	0.08568863	0.625274327	FALSE
0.833810065	0.083167014	0.625274327	FALSE

Figure 25: First version of feature set generated from source data

A	B	C	D	E	F	G
Set	time_stamp	Time	BPOG_Euclidian_Dist	FPOG_Euclidian_Dist	Cursor_Euclidian_Dist	Confusion
3.3	00:00.0	0	0.895361037	0.965415905	1.54963085	FALSE
3.3	00:00.0	0.01636	1.026350253	0.995827106	1.54963085	FALSE
3.3	00:00.0	0.03296	1.028659459	1.006761657	1.54963085	FALSE
3.3	00:00.1	0.04944	1.192801572	1.08259959	1.545307291	FALSE
3.3	00:00.1	0.0658	0.806456468	0.937819478	1.545307291	FALSE
3.3	00:00.1	0.08215	0.807886476	0.892702215	1.545307291	FALSE
3.3	00:00.1	0.09851	0.744480474	0.786234317	1.282669628	FALSE
3.3	00:00.1	0.11646	0.891794869	0.836750247	1.282669628	FALSE
3.3	00:00.1	0.13135	0.890371975	0.854153661	1.282669628	FALSE
3.3	00:00.2	0.14795	0.779015678	0.833917989	1.081896267	FALSE
3.3	00:00.2	0.16467	0.792846868	0.825207626	1.081896267	FALSE
3.3	00:00.2	0.18079	0.791264779	0.819276664	0.931455457	FALSE
3.3	00:00.2	0.19714	0.757928369	0.810291918	0.855781666	FALSE
3.3	00:00.2	0.21362	0.771141306	0.805220811	0.855781666	FALSE
3.3	00:00.2	0.22998	0.768103087	0.800977962	0.828030445	FALSE
3.3	00:00.3	0.24646	0.719253801	0.792552641	0.828030445	FALSE
3.3	00:00.3	0.26306	0.748582645	0.788527559	0.828030445	FALSE
3.3	00:00.3	0.2793	0.747071546	0.785064082	0.772672422	FALSE
3.3	00:00.3	0.29919	0.701178869	0.778486073	0.714737612	FALSE
3.3	00:00.3	0.31226	0.665133781	0.770141904	0.714737612	FALSE
3.3	00:00.3	0.32861	0.663636111	0.74174486	0.714737612	FALSE
3.3	00:00.3	0.34497	0.53302112	0.634625761	0.625274327	FALSE
3.3	00:00.4	0.36145	0.578523919	0.614975875	0.625274327	FALSE
3.3	00:00.4	0.37793	0.577029237	0.562856616	0.625274327	FALSE
3.3	00:00.4	0.39441	0.58411802	0.572481969	0.625274327	FALSE
3.3	00:00.4	0.41077	0.631526139	0.590974268	0.625274327	FALSE
3.3	00:00.4	0.42712	0.631526139	0.615357245	0.625274327	FALSE
3.3	00:00.4	0.44373	0.677576172	0.646111199	0.625274327	FALSE
3.3	00:00.5	0.46008	0.870616202	0.715079513	0.625274327	FALSE
3.3	00:00.5	0.47644	0.833810065	0.372668423	0.625274327	FALSE
3.3	00:00.5	0.49463	0.833810065	0.372668423	0.625274327	FALSE
3.3	00:00.5	0.50928	0.833810065	0.372668423	0.625274327	FALSE
3.3	00:00.5	0.52576	0.833810065	0.331635322	0.625274327	FALSE
3.3	00:00.5	0.54211	0.833810065	0.290932066	0.625274327	FALSE
3.3	00:00.6	0.55872	0.833810065	0.250719355	0.625274327	FALSE
3.3	00:00.6	0.57495	0.833810065	0.211277471	0.625274327	FALSE
3.3	00:00.6	0.59143	0.833810065	0.173134028	0.625274327	FALSE
3.3	00:00.6	0.60791	0.833810065	0.137374892	0.625274327	FALSE
3.3	00:00.6	0.62427	0.833810065	0.106431093	0.625274327	FALSE
3.3	00:00.6	0.64099	0.833810065	0.08568863	0.625274327	FALSE
3.3	00:00.7	0.6571	0.833810065	0.083167014	0.625274327	FALSE

Figure 26: The second version of feature set

A.2 Algorithms Used In This Project

Random forests or random decision forests [14] algorithm is an ensemble learning method for classification, regression and other tasks, that operate by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees. Random decision forests correct over-fitting behavior for its random decision trees to their training set [12].

K* algorithm uses entropy to define the distance metric by calculating the mean of the complexity of transforming an instance into another. The algorithm takes into account the probability of instance transformation in a random walk away manner. The K* algorithm classifies instances summing the probabilities from the new instance to all of the members of a category and to the rest of the categories in order to finally select the class with the highest probability [3].

AdaBoost M1 algorithm belongs to a class of Boosting algorithm, which is a general and provably effective algorithm of producing a very accurate prediction rule through a combination of rough and moderately inaccurate rules[25]. The algorithm takes as input a training set $(x_1, y_1), ..., (x_m, y_m)$ where each x_i belongs to some domain or instance space X, and each label y_i is in some label set y_i [6].

A.3 Handouts Given to Participants

INFORMED CONSENT FORM

My name is Yupeng Lan, and I am a student at Union College in Schenectady, NY. I am inviting you to participate in a research study. Involvement in the study is voluntary, so you may choose to participate or not. A description of the study is written below.

I am interested in learning about identifying user confusion when users are using complex software like Excel through applying machine learning techniques. You will be asked to complete a set of tasks using Excel. This will take approximately an hour. It is important to inform you that <u>this study does not test your skills on using Excel, and you don't need to feel bad if you do not perform well</u> . You just need to relax and complete tasks as the way you normally do. There are no foreseeable risks to taking part in this study. Completing the entire task is not mandatory, and if you no longer wish to continue, you have the right to withdraw from the study, without penalty, at any time.

Your responses will be anonymous, such that it would be impossible to link your name with any of your responses. Notice that you will be video taped, and screen shots of the video may appear in the publication, but your face will not be identifiable.

Even though all aspects of the study may not be explained to you beforehand (e.g., the method of the study), during the debriefing session you will be given additional information about the study and have the opportunity to ask questions.

By signing below, you indicate that you **understand the information above**, and that you **<u>wish to participate in this research study</u>**.

Figure 27: Consent Form

1. Please indicate your computer skill level
    a) Novice
    b) Moderate
    c) Experienced user
2. Please indicate your Excel skill level
    a) Novice
    b) Moderate
    c) Experienced
3. How old are you
    a) 17-22
    b) 23-30
    c) 31-40
    d) 41-50
    e) 51-59
    f) >=60
4. Please indicate your gender
    a) Male
    b) Female
5. Please indicate your background
    a) Student
    b) Faculty
    c) Staff
    d) Visitor
    e) Other
6. Do you have visual impairment
    a) Yes
    b) No

Figure 28: background survey

Pre-Experiment Instruction

Thank you for interest to participate my study. Here are some steps you need to do before starting the experiment.

First, the experiment involves an eye-tracker, which records both of your eye movements and your face. **If you do not agree to be video taped, you can leave the experiment now, or leave any time during the experiment if you do not feel comfortable to be video taped. You can collect your reward even if you quit the experiment at any time.** If you agree to be video taped, we may continue.

Before the experiment, I need to calibrate the eye tracker for you. This process may repeat several times until an acceptable accuracy is reached. This step may take longer especially if you wear glasses or contact lenses. Your glasses or contact lenses may compromise the accuracy of the device so I may ask you to adjust your head distance and positions to achieve better accuracy.

The first step of calibration is to adjust your head distance and angle of the device. Constantly adjust your head position until the moving dot on the control panel stays roughly in the middle. If you are within the optimal distance, the dot will turn green. If you are too far away, the dot will shift to the left, and you need to move closer. If you are too close, the dot will shift right, and you need to move farther from the screen. I will remind you during the experiment if you leave the optimal range of the device.

The second step is to calibrate your eyes' positions. When the calibration starts, a shrinking white circle will appear on the top left of the screen. You need to focus your sight on the circle, and the circle will shrink into a red dot. Then the dot will move right to the top center of the screen, and so on. Follow your sight with the dot, and don't move your eyes until the circle moves. Please do not anticipate the position of the next circle. If you wear

Figure 29: Pre-Experiment Instruction

glasses, you might need to adjust the angle of your glasses to prevent infrared reflection.

Once the calibration is finished, try to maintain the position. Also, try not to blink too frequently. When the calibration finishes, a black screen with circles and crosses will appear. Focus your sight on the crosses, and a moving green line will try to follow your line of sight. Please verbally indicate where you are looking at. When the green line covers where you are looking at, move to other crosses. Again, please tell me where you are looking at whenever you move your sight to another location. I may ask you to restart calibration if random jumps of tracing path appear, or the tracing path is too far off from your actual point of view. When the accuracy is satisfactory, we may move on to the actual experiment.

Figure 30: Pre-Experiment Instruction cont’d

Please carefully read the task instruction below.

<u>Task 1.</u>

Please calculate the total frequencies besides the entry “Total”.

Click the cell right next to the cell “Total” **->** **Insert Function** **->** **Sum** **->** Select all cells under cell “Frequency” **->** Press enter

Please tell me whenever you have finished the task.

<u>Task 2.</u>

Calculate percentage for all three studies.
Percentage = frequency of a data entry / total frequency

Enter “=” under cell “Percent” **->** **Select data cell** **->** **Enter “/”** **->** **Select Total** **->** Press enter

Please tell me whenever you have finished the task.

<u>Task 3</u>

Now all percent are in decimals, convert them to the percent form by using the formatting feature in Excel.
Select Cells need to be formatted **->** **Home Tab** **->** **Format Cells** **->** **Percentage** **->**Enter

Figure 31: Experiment Instruction

Please tell me whenever you have finished the task.

Task 4

Make three pie charts to visualize percentage for all three studies.
Insert Tab -> Chart -> Pie Chart -> Select Data -> Select Horizontal Axis (All the data field under Gender, Marital Status, and Church attendance, excluding Total) -> **Select Data Field** (Freq.) -> **Select Legend** (Study topics of three social studies)

Please tell me whenever you have finished the task.

Task 5

If done correctly, the result of Task 4 should show three pie charts without percentage of the data entries. You want to show the percentage of each data entry on the chart for better visualization. Find the button that display percentage under the **Design Tab**.

You want to emphasize certain entries in the data fields of three social studies. Conditionally format cells that satisfy the following conditions

Task6

In the first study, add a red frame around the percentage cell that has a higher percentage.

a) Home -> **Conditional Formatting** -> **Greater than** -> Select the cell with lower percent -> **Red Border** -> OK

Task 7

Figure 32: Experiment Instruction cont'd

In the second study, highlight cell under percent column whose value is between **0** and **15%** with **Yellow Fill** with **Dark Yellow Text**

Follow the similar step showed in instruction 1

Task 8

In the third study, highlight cells under percent column with green fill and dark green text if the value is between **10%** and **20%**

Follow the similar step introduced above.

Save your result in the hard disk.

End of Experiment task

Figure 33: Experiment Instruction cont'd

Debriefing

This study is a part of my research to build an Adaptive User Interface that recognizes users' intentions, adapts to their habits, and identifies whether they need help. This research project is divided into three general research questions: how does the user interface know users need help, on what topic do users need help, and how can the user interface help users on that topic. To effectively answer these three questions, I need to find an effective way to recognize users intentions and mental states. However, accomplishing this task is very difficult and need a proper metrics to make inferences on users intentions and mental states. Based on previous studies, I believe that eye movement patterns can be a good indicator of users' intentions and mental states. Thus, I collected your eye movements and use the data to build models. The study I am currently doing tries to identify whether you are confused during tasks by tracking how your eyes moved. Once all data are collected, I will train a model and find out how successfully it predicts user confusion. If the model predicts user confusion accurately, then I can conclude that eye movement pattern is a good indicator of user confusion, and the model I trained can be applied to determine whether users need help. Otherwise, I will conclude that eye-tracking may not be a good way to determine whether users need help, and I need to find another way.

Figure 34: debrief

A.4 Experiment Task Durations

Set	start	end
3.3	00:25.0	00:43.0
3.4	01:23.0	03:28.0
3.6	00:02.9	00:54.0
3.7	00:29.7	01:06.0
4.4	01:28.0	04:30.0
4.4	05:37.0	06:19.0
4.4	05:37.0	06:19.0
4.6	00:27.5	01:13.0
5.6	00:41.0	01:20.0
6.4	01:55.8	03:17.0
6.4	05:13.8	05:37.0
6.4	06:17.7	07:10.9
7.4	04:49.0	05:43.0
7.6	01:23.0	05:17.0
8.4	02:30.0	02:23.0
8.6	00:45.0	02:29.0
9.4	02:48.0	06:49.0
10.4	01:24.0	01:40.0

Figure 35: Confusion reported task and the durations

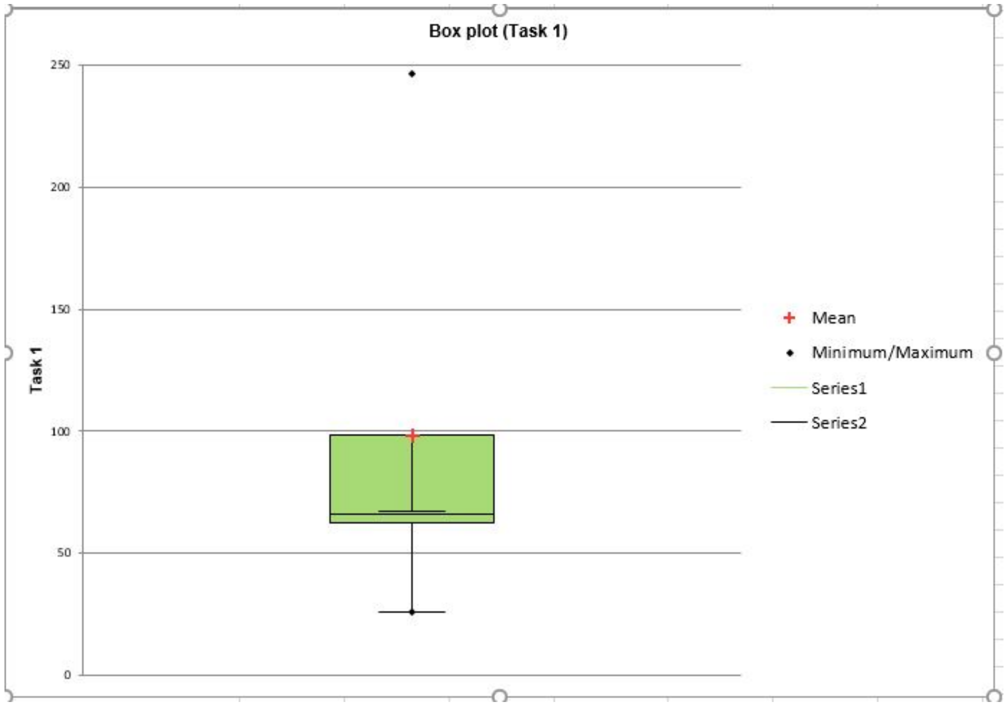


Figure 36: Task Completion Time of Task 1

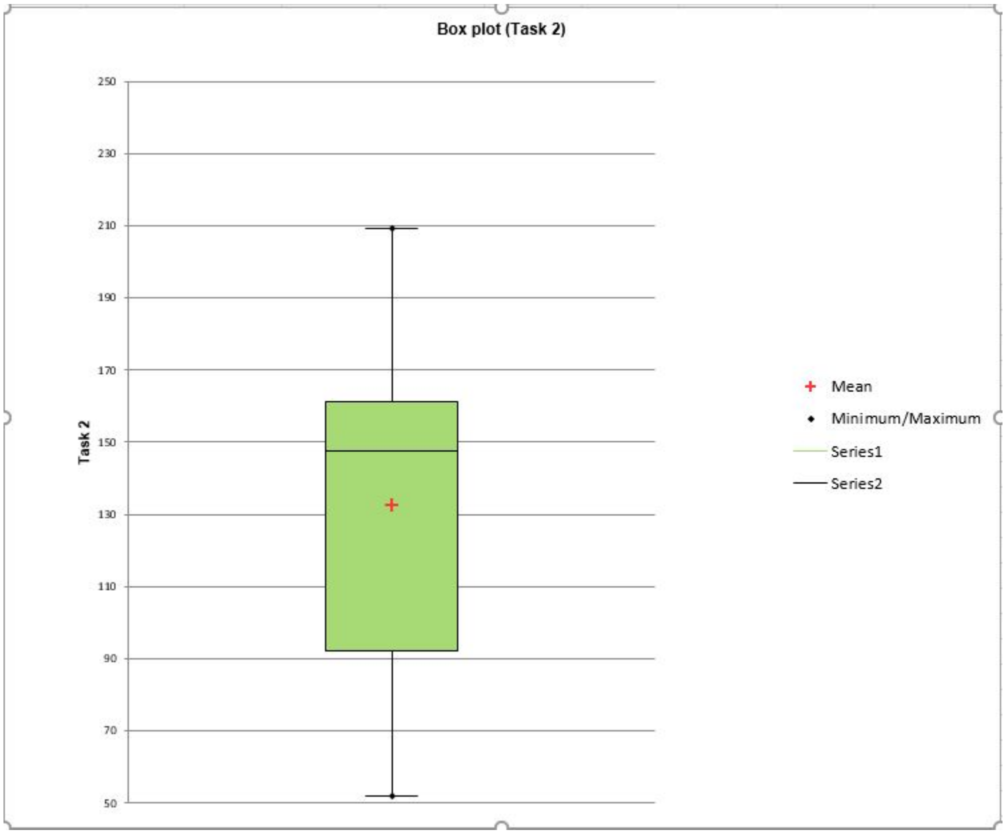


Figure 37: Task Completion Time of Task 2

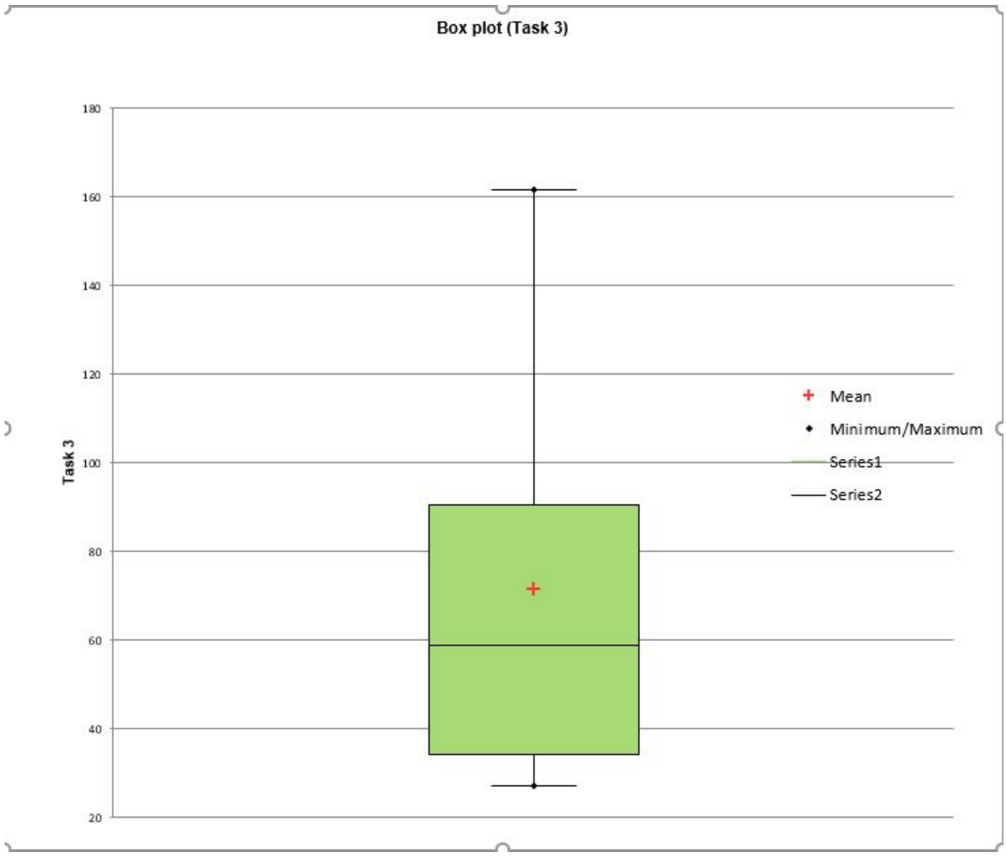


Figure 38: Task Completion Time of Task 3

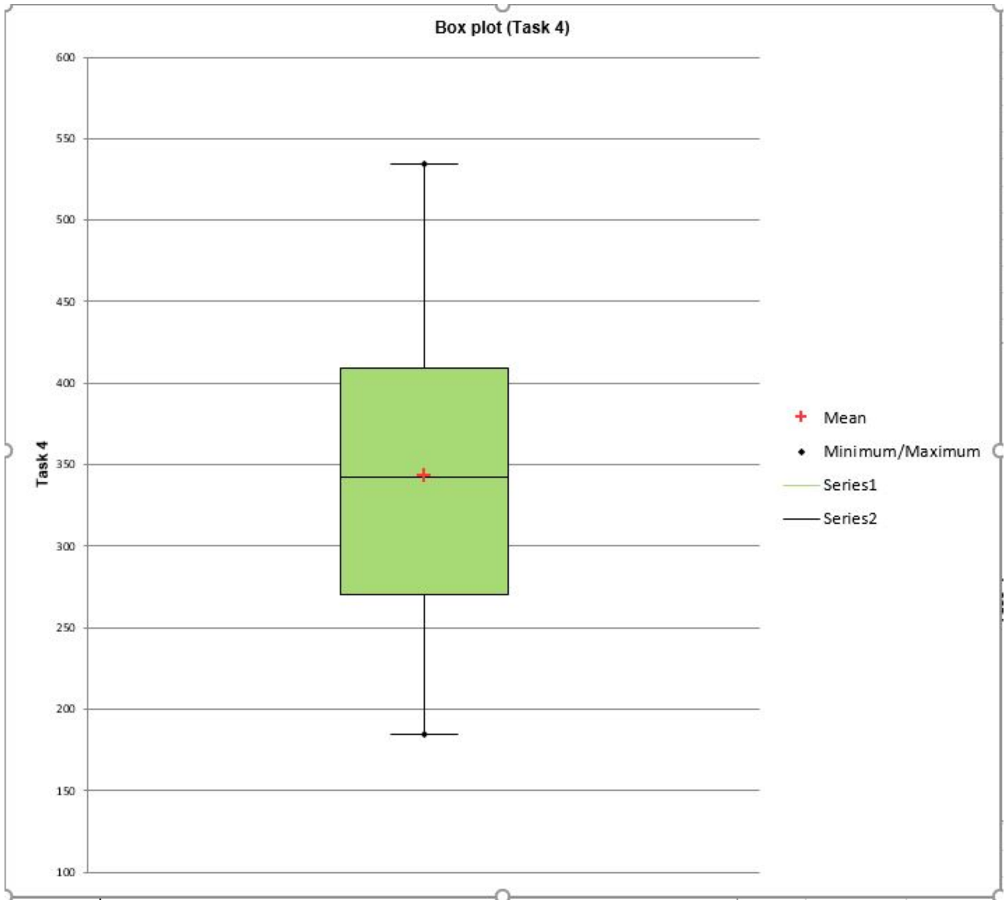


Figure 39: Task Completion Time of Task 4

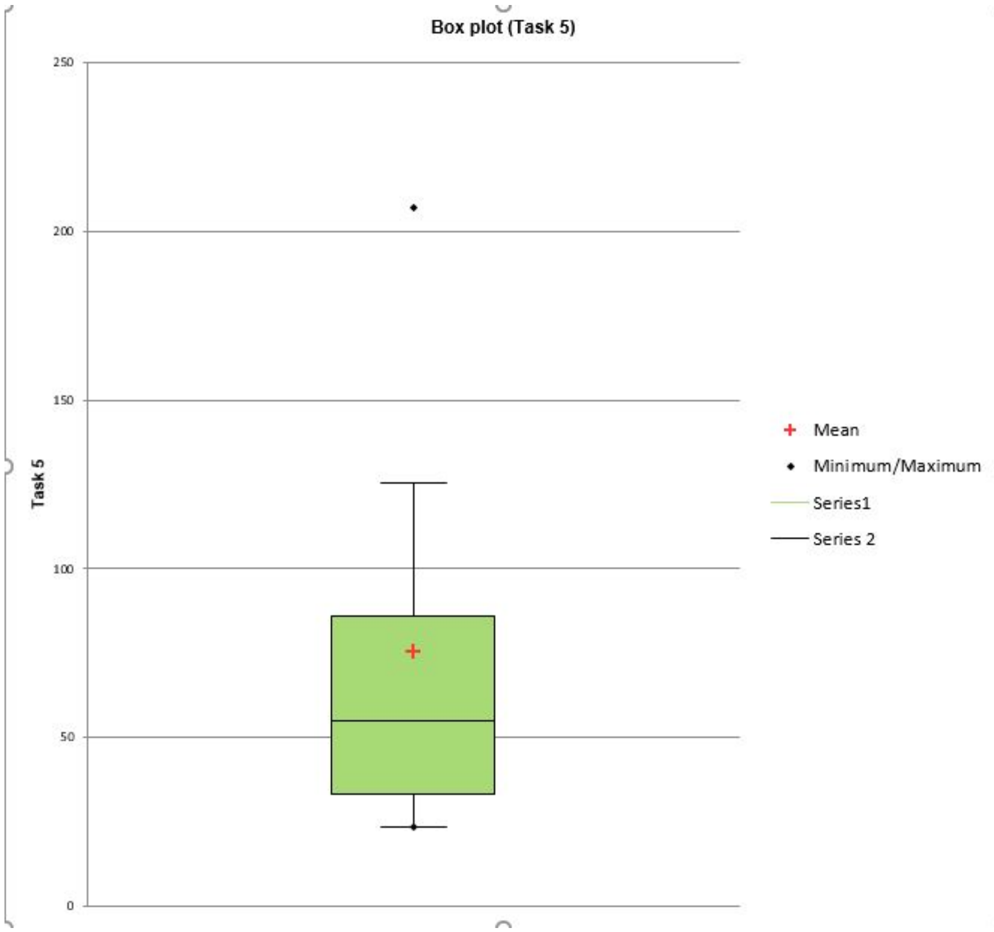


Figure 40: Task Completion Time of Task 5

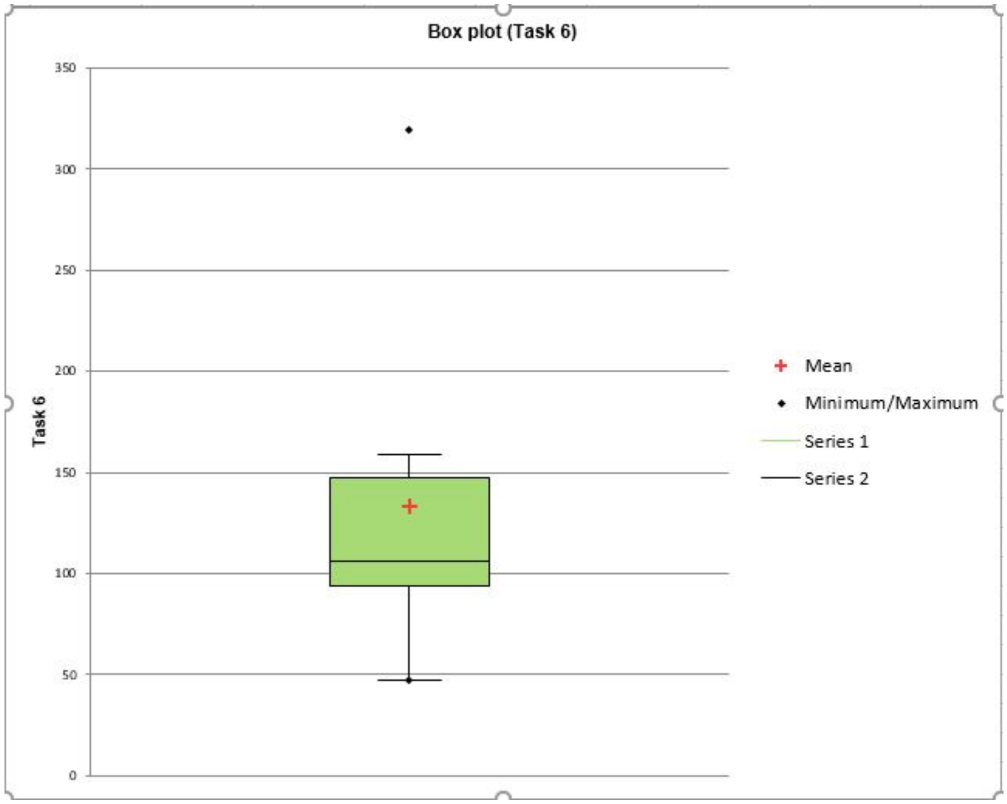


Figure 41: Task Completion Time of Task 6

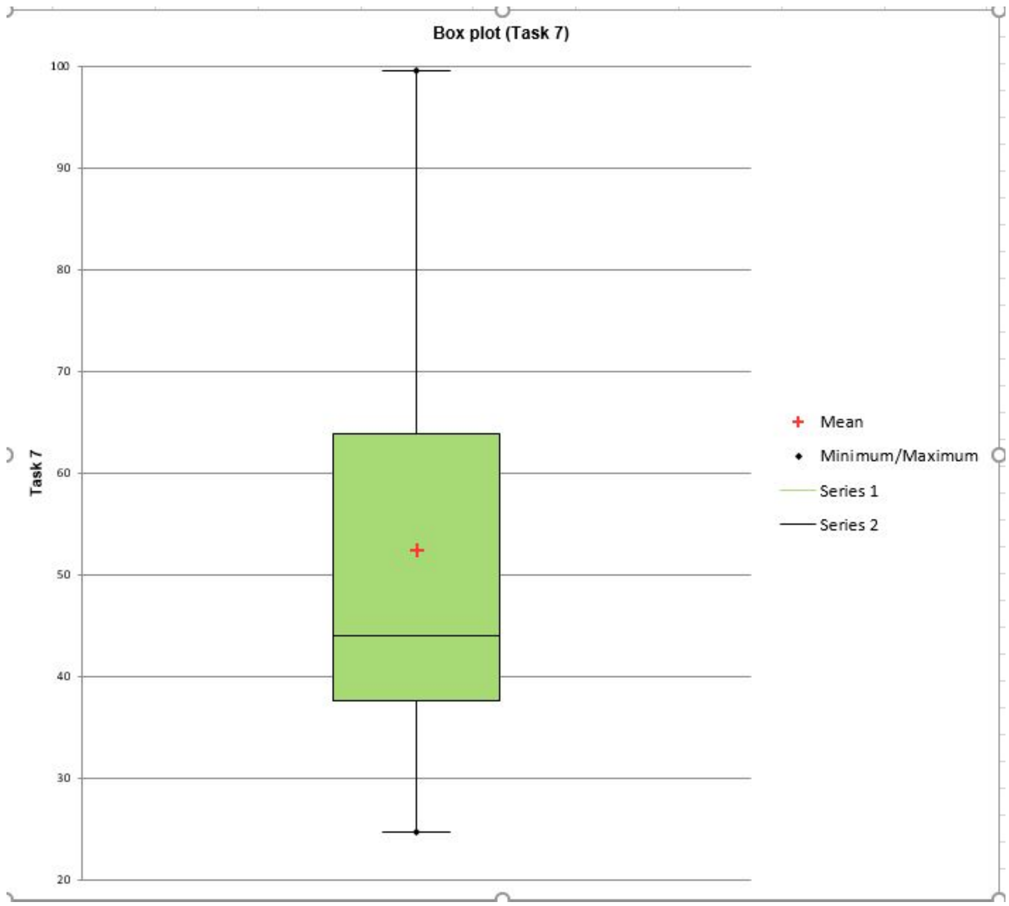


Figure 42: Task Completion Time of Task 7

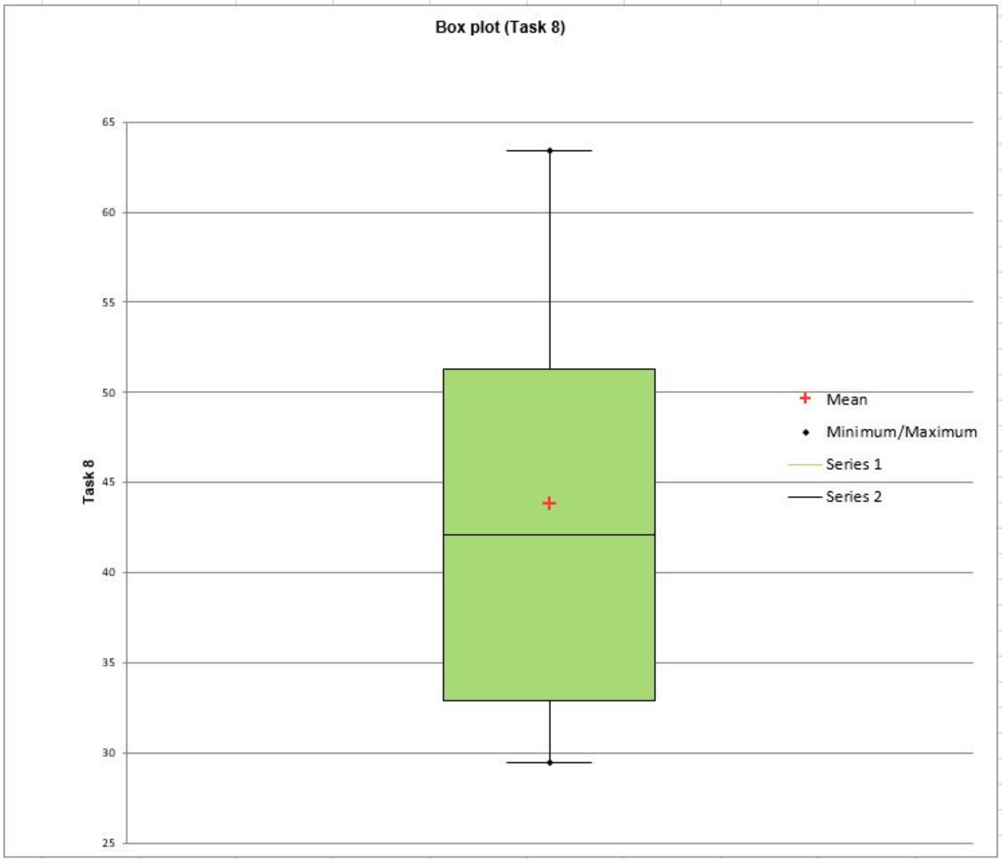


Figure 43: Task Completion Time of Task 8

References

[1] Nigel Bosch et al. “Automatic Detection of Learning-Centered Affective States in the Wild”. In: *Proceedings of the 20th International Conference on Intelligent User Interfaces*. IUI ’15. New York, NY, USA: ACM, 2015, pp. 379–388. ISBN: 978-1-4503-3306-1. DOI: 10.1145/2678025.2701397. URL: http://doi.acm.org/10.1145/2678025.2701397.

[2] Michael D. Byrne et al. “Eye Tracking the Visual Search of Click-down Menus”. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI ’99. New York, NY, USA: ACM, 1999, pp. 402–409. ISBN: 0-201-48559-1. DOI: 10.1145/302979.303118. URL: http://doi.acm.org/10.1145/302979.303118.

[3] John G. Cleary and Leonard E. Trigg. “K*: An Instance-based Learner Using an Entropic Distance Measure”. In: *12th International Conference on Machine Learning*. 1995, pp. 108–114.

[4] J. Cohen. “Weighted kappa: Nominal scale agreement with provision for scaled disagreement or partial credit”. In: *Psychological Bulletin* 70 (1968), pp. 213–220.

[5] Andrew T. Duchowski. *Eye Tracking Methodology: Theory and Practice*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2007. ISBN: 1846286085.

[6] Yoav Freund and Robert E Schapire. “A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting”. In: *Journal of Computer and System Sciences* 55.1 (1997), pp. 119 –139. ISSN: 0022-0000. DOI: http://dx.doi.org/10.1006/jcss.1997.1504. URL: http://www.sciencedirect.com/science/article/pii/S002200009791504X.

[7] Claude Ghaoui. *Encyclopedia of Human Computer Interaction*. Hershey, PA: Information Science Reference - Imprint of: IGI Publishing, 2005. ISBN: 1591405629, 9781591405627.

[8] Joseph H Goldberg and Xerxes P Kotval. “Computer interface evaluation using eye movements: methods and constructs”. In: *International Journal of Industrial Ergonomics* 24.6 (1999), pp. 631 –645. ISSN: 0169-8141. DOI: https://doi.org/10.1016/S0169-8141(98)00068-7. URL: http://www.sciencedirect.com/science/article/pii/S0169814198000687.

[9] Joseph H Goldberg and Anna M Wichansky. “Eye tracking in usability evaluation: A practitioners guide”. In: (2002).

[10] Joseph H. Goldberg et al. “Eye Tracking in Web Search Tasks: Design Implications”. In: *Proceedings of the 2002 Symposium on Eye Tracking Research & Applications*. ETRA ’02. New Orleans, Louisiana: ACM, 2002, pp. 51–58. ISBN: 1-58113-467-3. DOI: 10.1145/507072.507082. URL: http://doi.acm.org/10.1145/507072.507082.

[11] Mark Hall et al. "The WEKA Data Mining Software: An Update". In: *SIGKDD Explor. Newsl.* 11.1 (2009), pp. 10–18. ISSN: 1931-0145. DOI: 10.1145/1656274.1656278. URL: http://doi.acm.org/10.1145/1656274.1656278.

[12] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning*. Springer Series in Statistics. New York, NY, USA: Springer New York Inc., 2001.

[13] Gunnar Hauland. "Measuring Individual and Team Situation Awareness During Planning Tasks in Training of En Route Air Traffic Control". In: *The International Journal of Aviation Psychology* 18.3 (2008), pp. 290–304. DOI: 10.1080/10508410802168333. eprint: http://dx.doi.org/10.1080/10508410802168333. URL: http://dx.doi.org/10.1080/10508410802168333.

[14] Tin Kam Ho. "Random Decision Forests". In: *Proceedings of the Third International Conference on Document Analysis and Recognition (Volume 1) - Volume 1*. ICDAR '95. Washington, DC, USA: IEEE Computer Society, 1995, pp. 278–. ISBN: 0-8186-7128-9. URL: http://dl.acm.org/citation.cfm?id=844379.844681.

[15] K. Holmqvist et al. *Eye Tracking: A comprehensive guide to methods and measures*. OUP Oxford, 2011. ISBN: 9780191625428. URL: https://books.google.com/books?id=5rIDPV1EoLUC.

[16] Aulikki Hyrskykari. "Eyes in Attentive Interfaces: Experiences from Creating iDict, a Gaze-Aware Reading Aid". In: *Hyrskykari, Aulikki* (2006).

[17] R. J. K. Jacob and K. S. Karn. "Eye Tracking in Human-Computer Interaction and Usability Research: Ready to Deliver the Promises". In: *The Mind's eye: Cognitive and Applied Aspects of Eye Movement Research* (2003). Ed. by J. Hyona, R. Radach, and H. Deubel, pp. 573–603. URL: http://www.sciencedirect.com.lt.ltag.bibl.liu.se/science.

[18] Robert J. K. Jacob and Keith S. Karn. "Eye Tracking in Human-Computer Interaction and Usability Research: Ready to Deliver the Promises". In: *Mind* 2.3 (2003), p. 4.

[19] Marcel Adam Just and Patricia A Carpenter. "Eye fixations and cognitive processes". In: *Cognitive Psychology* 8.4 (1976), 441480. DOI: 10.1016/0010-0285(76)90015-3.

[20] Sébastien Lallé et al. "Prediction of Users' Learning Curves for Adaptation While Using an Information Visualization". In: *Proceedings of the 20th International Conference on Intelligent User Interfaces*. IUI '15. New York, NY, USA: ACM, 2015, pp. 357–368. ISBN: 978-1-4503-3306-1. DOI: 10.1145/2678025.2701376. URL: http://doi.acm.org/10.1145/2678025.2701376.

[21] Claudia Mello-Thoms, Calvin F Nodine, and Harold L Kundel. “What Attracts the Eye to the Location of Missed and Reported Breast Cancers?” In: *Proceedings of the 2002 Symposium on Eye Tracking Research & Applications*. ETRA ’02. New Orleans, Louisiana: ACM, 2002, pp. 111–117. ISBN: 1-58113-467-3. DOI: 10.1145/507072.507095. URL: http://doi.acm.org/10.1145/507072.507095.

[22] A. Pentel. *Patterns of Confusion: Using Mouse Logs to Predict Users ...* URL: http://ceur-ws.org/Vol-1388/PALE2015-paper5.pdf.

[23] Alex Poole and Linden J Ball. “Eye tracking in HCI and usability research”. In: vol. 1. Idea Group Reference Hershey, PA, 2006. Chap. Eye tracking in HCI and usability research, pp. 211–219.

[24] Helmut Prendinger et al. “Attentive interfaces for users with disabilities: eye gaze for intention and uncertainty estimation”. In: *Univers. Access Inf. Soc.* 8 (4 2009), pp. 339–354. ISSN: 1615-5289. DOI: 10.1007/s10209-009-0144-5. URL: http://portal.acm.org/citation.cfm?id=1667488.1667489.

[25] Robert E. Schapire. “A Brief Introduction to Boosting”. In: *Proceedings of the 16th International Joint Conference on Artificial Intelligence - Volume 2*. IJCAI’99. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1999, pp. 1401–1406. URL: http://dl.acm.org/citation.cfm?id=1624312.1624417.