

**E-Prime and Tobii Eye-Tracker Configuration:
A Methodological Manual**

**Mandy Maguire, Ph.D.
Grant Magnon
& Anna Fitzhugh
Developmental Neurolinguistics Laboratory
University of Texas at Dallas**

TABLE OF CONTENTS

- I. Introduction
 - a. Study Overview and Objectives
- II. Methodological Considerations
 - a. Port Configuration
 - b. Timing in E-Prime
 - c. Defining ROIs
 - d. Tracking Eye Gaze
 - e. Sending Triggers
- III. Study-Specific Information
 - a. Stimulus Design and Specifications
 - b. Working with Event Files

INTRODUCTION

This manual is intended to provide a guide for anyone interested in linking together the Tobii 750 eyetracker and a SynAmps II Neuroscan system. The funding for this project was provided by an R03 Grant awarded to the PI, Mandy Maguire. As specified in the original grant, the goal was to create an integrated system that (1) linked eye tracking and EEG in real-time and (2) could allow for stimulus presentation contingent on eye gaze, such that looking to one specific point on the screen (or away from the screen) would dictate what would come up next. This was particularly important for us as a way to show “attention getting” stimuli for infants and young children completing visual EEG studies. This manual summarizes important aspects of our study, with specifics about the software and hardware needed to the interlocked system, written with the hope that it will provide a brief but useful introduction for others who may wish to adopt a similar methodological design.

Study Overview and Objectives

The Eye-Tracker Dog/Animal Study is a modification of a previous object categorization task, revised so that eye-tracking could be performed simultaneously with collection of EEG data. Subjects viewed 200 slides with images of animals—either “dog” or “non-dog”—and then indicate which type of slide they were shown on the response pad. Specifically aspects of the eye gaze that were important to us included (1) if visual attention was lost and (2) when visual fixation was on the head of the animal vs the body. The first point was to allow for attention getters to be played to reorient the child’s visual attention towards the screen. The second point was to identify the temporal relationship between gaze to a defining feature used in categorization (the head) and the ERP responses thought to index categorization (P300).

The primary aim of the study is methodological: to see whether online eye-tracking and EEG collection could be performed simultaneously by two computers, using two different software platforms, with an acceptable level of timing precision. Additionally, the latency of the P300 ERP in adults and children can be compared to the amount of time it took them to direct their eyes to the different animals’ heads—the part of the stimulus that is crucial for object categorization—to see whether differences in the onset of the P300 in adults and children may be correlated to differences in eye-movement.

Stimulus presentation occurs in E-Prime, which has been configured to communicate with Tobii Studio through the use of drivers and protocols installed from the “E-Prime Extensions for Tobii” CD-ROM. On each animal picture, certain regions of interest (ROIs) have been drawn around the animal’s head and body, and code has been written so that E-Prime will send different triggers to SCAN for when subjects are looking at the head vs. the body of the animal.

The use of triggering directly from E-Prime to SCAN is important for timing purposes: having a timestamp of when (a) the picture first appeared and (b) when the subject first moved their eyes toward the animal’s head will allow us to calculate, on the .CNT files, variations in the latency of the P300 from each of these markers. Because the two systems are separate, however, certain offline corrections for timing were needed to account for delays in stimulus presentation and detection of eye-gaze information.

Configuring E-Prime to define key ROIs, track eye-gaze data, and send triggers involved modification of E-Basic scripts in the form of InLine objects that were then inserted into the study timeline.

METHODOLOGICAL CONSIDERATIONS

Port Configurations

Linkage of the E-Prime stimulus presentation software and Neuroscan, for the purpose of sending time-stamped triggers, was accomplished by connecting the stimulus presentation computer (here referred to as “STIM” computer) to the Neuroscan amps unit via the “Stim-to-Scan” cable provided by Neuroscan. The “Stim” end was plugged into an available parallel port on the STIM computer, and the SCAN end fed into the back of the Synamps2 amplifier unit. Detailed information on the pinout of the Stim-to-Scan cable can be found in Neuroscan’s documentation. This configuration allows triggers to be sent from the presentation computer to the amplifiers, where they are sampled along with the EEG data and incorporated into the EEG continuous (.CNT) file at the time of collection.

Output of the triggers was done in E-Prime by means of the “Writeport” command, which allows users to specify an available port over which to send trigger pulses. For information on the syntax used by E-Prime, see the E-Prime documentation pertaining to the “Writeport” command. Note that the computer port being used may need to be specified in hexadecimal notation. These commands were then added to the experiment in E-Prime as blocks of In-Line code inserted at various points in the experiment timeline, as will be seen in subsequent sections of this document.

Timing in E-Prime

When trying to precisely time studies, there are a number of influences that can cause errors in timing. For example, unexpected operating system processes can interrupt the program running the experiment and cause an unnoticeable delay. There’s also the amount of time it takes the computer to find the file that it’s looking for in memory (i.e. a picture or a slide of a word that’s stored on the hard drive). Lastly, there is a built-in delay inherent in the way computers update their monitors and draw new images to the screen: an image can only be sent to the monitor, or altered, at certain intervals called “refresh cycles.” For more information, see E-Prime User’s Guide, Chapter 3.

The eye-tracker monitor’s refresh rate is 59Hz (or ~ 1 screen refresh every 16.66ms), meaning that images can only be sent at discrete intervals of 16.66ms. Take the following example: If the time it takes the computer to access an image in memory is 20ms, it will then have to wait around for the next refresh cycle before it can send that image to the monitor. If the first refresh came at 16.66ms, the next will come at 33.33ms. Thus, instead of the delay being 20ms, it’s actually 33.33ms.

E-Prime offers its own built-in time audit system (see User’s Guide Chapter 3) that can keep track of most of this stuff for us. Our focus involves making corrections to the Neuroscan event files to account for these delays so that we can get to ~1ms timing precision.

Specific Timing Delays:

- 1) **Processing Time/Refresh Rate for Slide Triggers.** As described in the introduction, there is a lag time between when an image is supposed to appear, and how long it takes the computer to retrieve it from memory and then wait around for the next refresh cycle. Adding code to the E-Prime scripts will let us calculate the differential between when the slide trigger was sent to Neuroscan (which is simultaneous with when that trigger appears on the .CNT file) and when the image was actually drawn on the screen. The trigger is *ahead* of when the image was actually shown on the monitor, and so its timestamp on the .CNT file has to be adjusted accordingly.

You can find this “lag” time in the E-Data file for each subject, for each trial. In the E-Data file there will be a column named “Stimulus3.OnsetDelay”—that tells you the number of milliseconds that elapsed between when the trigger was sent to Neuroscan and when the image actually showed up. You will want to add this time to the timestamp on the .CNT for the slide triggers. (For an explanation of how we coded for this, see Appendix A). Notice that nearly all lag times will be integer multiples of the screen refresh rate. Some may be particularly long—those are typically when the computer’s operating system temporarily took over to perform some housekeeping task. Nonetheless, by adding the elapsed millisecond time value to the .CNT timestamp, we can correct for such delays.

- 2) **Processing Time Delays for Eye-Position Triggers.** There is a processing delay in Tobii of ~3ms from when the eye-tracker registers eye-contact on the screen to when it can send a corresponding trigger to Neuroscan. Hence the eye-position triggers will show up on Neuroscan slightly *behind* when they really occurred. Adjusting the triggers on the .CNT file accordingly can correct for this delay. For how we determined this lag time, see Appendix B.
- 3) **Responses.** Because we are using simulated mouse clicks for responses, we sacrifice any ability to achieve ~1ms precision in the reaction time (RT). Thus the RT times are currently not being adjusted for delays.

Defining Regions of Interest in E-Prime

This was perhaps our biggest challenge in configuring the eye-tracking system to interface with E-Prime. The following are the steps we took to define ROIs that corresponded to the relevant parts of the animal stimuli and configure E-Prime to check to see when a subject’s gaze fell within that ROI.

The method we settled on to “draw” ROIs that would contain either the animals’ heads or bodies was to use empty textboxes that we then sized appropriately to cover each animal’s head or body. In some cases, because of the shape of the animal and the fact that E-Prime’s textboxes are square-shaped by default, we needed to cover the animal’s body with more than one textbox.

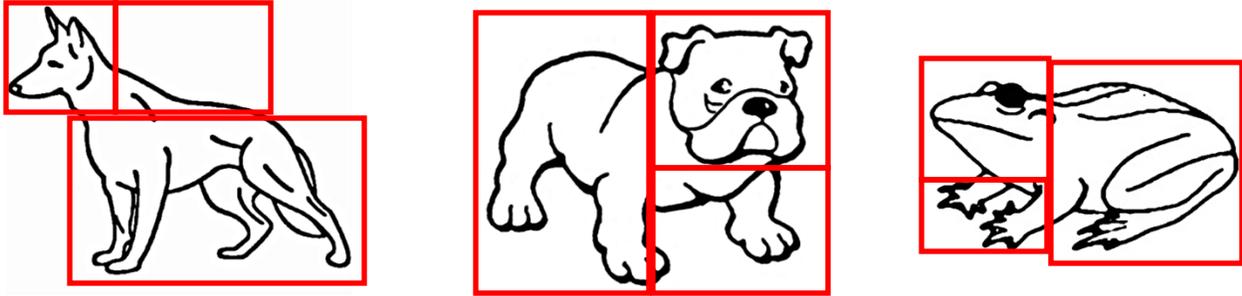


Figure 1. Examples of study stimuli coded for eye gaze ROIs.

To do this efficiently, we created a Slide Object which would display a different animal on each trial and would essentially be our main stimulus. On this one slide, we then created several different “sub-objects” which would appear on every trial with the same name, but whose properties would vary from trial to trial.

The first of these sub-objects was an image sub-object labeled “background,” which is what actually displays the drawing of the animal. It is set to fill the screen while leaving the animals in their original 720x720 cropped dimensions, without stretching. The animal shown by the background object varies, trial-by-trial, depending on which animal is due to be shown next (this order is specified in the “Trial Block List” spreadsheet) On the Property Pages for the background object, we set the file pathway to call up the name of the animal that matches what can be found in the Trial Block List, using E-Prime’s ability to create and then call up the different levels of a variable at run-time.

The second sub-object is an empty textbox named “head.” Its dimensions are defined by a variable such that, for each animal, a different set of x and y coordinates are used to shape the box so that it matches where that animal’s head will appear on the screen. This way, instead of needing 200 separate textboxes with different dimensions for each animal, we only need one textbox whose dimensions are defined by a variable that changes with each animal.

**Note: To determine the dimensions each animal would need in the first place, we drew a sample textbox over the head of each animal, recorded the dimensions, and then saved them as attributes along with that animal in the Trial Block List. The same was done for the body(s) ROIs as well.*

The remaining three sub-objects on the Stimulus slide are also empty textboxes labeled “body1,” “body2” and “body3.” Again, we defined the x and y coordinates of the boxes according to variables that would change with each animal, so that we don’t need separate ones for each animal. Because of their shapes, a few animals needed as many as three separate boxes to cover the body, hence why we have “body1,” “body2” and “body3.” However, for many animals these extra boxes weren’t needed; in these cases, the x and y coordinates of the boxes are left blank and thus the boxes simply don’t appear on those trials.

In this way we were able to accommodate all of our 200 stimuli (dog and animal pictures) on a single E-Prime trial timeline, with only 1 Stimulus Object and 5 sub-objects that use changing

variables to display all the different animals and to draw uniquely-shaped ROIs that match where each animal's head and body are.

Tracking Eye-Gaze Data

After establishing our setup, coding E-Prime to recognize when a subject's eyes were looking at either a head or body was relatively easy. E-Prime has a built-in function called "HitTest," which will access Tobii to get the x and y coordinates on the screen of where the subject is looking at any point in time. The HitTest then returns a string value that is equal to the name of any sub-object that overlaps the area on the screen where the eye-gaze is focused (note: because it interacts with sub-objects, the HitTest only works on Slide Objects, which are the only "Objects" in E-Prime capable of having sub-objects).

So, for example, E-Prime finds the current x and y coordinates of the subject's gaze; if these x and y coordinates fall inside a part of the screen that is currently occupied by a sub-object (for instance, our "head" textbox), the HitTest function returns a value of "head" (since "head" is the string for the name of our sub-object). The same occurs for "body1," "body2," and "body3" if the subject's eye-gaze falls within those areas of the screen.

Because the x and y coordinates of the head/body sub-objects change appropriately with each animal slide, we know that, if HitTest = "head" for any trial, it means the subject has directed their focus to the head. Some animals' heads may be on the left side of the screen, others on the right side, etc; however, with the HitTest and our moving sub-objects, none of that matters. If HitTest = "head" then we know they're looking at the head, without having to calculate for ourselves the x and y coordinates, compare them to the textbox, see if they match, etc. This is the crux of how E-Prime is able to communicate with Tobii and link eye-gaze coordinates to what is actually being shown on the screen.

Sending the Appropriate Triggers

With the HitTest doing most of our work for us, sending triggers that indicate whether the subject is looking at a head or body is relatively simple. We simply use an InLine code that says "If HitTest = "head," send a "100" trigger through the port to SCAN. If HitTest = "body1," "body2" or "body3" (they all refer to the body of the animal, so it doesn't matter which), then send a "200" trigger to SCAN.

STUDY-SPECIFIC INFORMATION

Stimuli: Animals & Orders

The stimuli in the E-Prime version of DA consist of 160 "non-dog" animal pictures (80 mirrored pairs) and 40 dog pictures (20 mirrored pairs) for a total of 200 slides, just as in the original. As with the original, these are line drawings of animals on a blank background slide, with all slides being the same dimensions (but with some size discrepancy among the animals, stemming from their different shapes).

Sizing

To provide consistency in terms of the amount of display area taken up by each image, all of the slides on which the animal images were drawn were cropped to pixel dimensions of 720 x 720

(occupying a square in the middle of the display roughly equivalent to the display height). This was chosen as the preferred method for sizing for several reasons: (1) Because the animals were different shapes, there was no way to assure consistent sizing by measuring the borders from the animal to the edges of the screen and striving to keep these a consistent width from animal to animal. (2) E-Prime’s sizing options did not allow for sizing the animals to a pixel dimension or % of the display area without selecting the “stretch” option, which, because the animals were not all square, ended up distorting their dimensions. We found it easiest to crop the animal slides to a consistent size and then import this size into E-Prime, which had the effect of ensuring each animal’s slide occupied a consistent % of the display screen (although within each slide, the actual size of the line-drawing varies slightly from animal to animal).

Sequencing

The order in which animals were presented in E-Prime was kept identical to the original Dog/Animal sequence files (including the substitutions—see below). Orders were created by taking the original Neuroscan sequence files and stripping away Stim2 codes until all that was left was a list of the images in the order they were presented. This list was then used as a template onto which each image’s E-Prime attributes (coding, X and Y coordinates, etc) were added (again including the substitutions made above). For Orders 1-3 (left orders) the “CorrectAnswer” column was populated with “1” for dogs and “2” for animals”; for Orders 4-6 (right orders), “1” was given to animal slides and “2” was given to the dog slides.

The six orders (with animal names, attributes, etc) were then pasted into the List Object of the E-Prime .es file and then saved separately as orders 1-6, respectively.

Working with Event Files

About the Event Files

The event files incorporating eye-gaze data will look different from most of the event files generated by stimulus-response studies. This is because the number of lines per trial is not consistent, but will depend on the eye-gaze pattern of the subject in response to each image. For instance, in our study, depending on how many times the subject may have looked back and forth from the animal’s body to its head, we may see multiple “100s” and “200s” either alternating or back-to-back in blocks.

The way to single out individual trials will be to look for the trigger codes that mark the onset of a new trial with the display of the image, and the code that indicates the subject’s response. All the lines in between will relate to that trial.

Code	Trigger
55	Dog image
88	Non-dog image
100	Eye gaze directed to head
200	Eye gaze directed to body
1	Correct response

2	Incorrect response
---	--------------------

Table 1. Eye-tracker Dog/Animal Trigger Codes

Time Adjustments

Because of inherent delays in stimulus presentation and stimulus triggering, time corrections are needed on the event file, depending on the desired temporal precision. A desired time precision of ~33ms should not require any correction; however, for <10ms precision, the following corrections need to be made to the timestamps that appear alongside trigger codes in the event file:

- 1) **Adjusting for presentation delays in presenting visual stimuli.** The timestamps for all image slides will need to be adjusted by adding a certain number of milliseconds to the time value shown in the elapsed time column. This is because, in actuality, the triggers for the pictures were sent to the SCAN computer ahead of when the STIM computer was actually able to draw them on the screen for the subject to view.

To determine how much time should be added, you need to find each subject's E-Data file containing E-Prime's internal data-collection and time-audit records for that experimental session. There will be a column labeled "Stimulus3.OnsetDelay": the value in this column (in milliseconds) is a measure of the amount of time that elapsed between when the trigger was sent to SCAN and when the actual image was shown to the subject. Take this value for each trial and add it to the time already contained in the event file for each image. Be careful to match each one trial-for-trial. This can currently be done manually, albeit with some effort. It may also be possible to import the E-Data files to Excel and write a macro that can automatically perform this adjustment trial-by-trial on the event file.

- 2) **Adjusting for Tobii's hardware delay in the detection of eye-gaze location.** There is a slight delay in the time between when Tobii first recognizes that a subject's eye-gaze has encountered part of an image (in our study, either the head or the body) and when the corresponding trigger actually reaches the SCAN computer. Therefore, as with the slide images, we will need to modify the time listed in the event file. In this case, we will be subtracting an amount of time, since the triggers are delayed from when the subject's eyes actually encountered the head/body. Based on some calculations described in Appendix B, the best estimate of this delay is ~3ms. Thus, 3ms should be subtracted from all eye-related trigger pulses.
- 3) **Response Times.** Currently, because we are using a response device that functions as a mouse to register responses, we don't have a precise way to account for delays in response times (see "Timing in E-Prime" above for an explanation). Thus, no modifications have been made at this time.

Appendix A

Triggers representing whether the current slide is a dog (55) or animal (88) are sent in a few lines of code that immediately precede the “SlideObject” (the image of a dog or animal) on E-Prime’s timeline. We wrote this code to also set-down a “marker” of what the computer time was when that trigger was sent. Specifically, we told E-Prime to set the “TargetTime” of the SlideObject to whatever the clock value happened to be at the instant the trigger was sent to Neuroscan. This “TargetTime” is a built-in system variable that E-Prime will keep track of for all objects, and then E-Prime will use this variable to calculate something called the OnsetDelay. The OnsetDelay is the elapsed time between “TargetTime” and when the given image actually appeared on the screen—*accounting for processing and refresh cycle delays*. Hence, the OnsetDelay gives us the time between when the trigger was sent to Neuroscan, and when the image actually appeared on the screen. Code is reproduced below:

```
Display.WaitForVerticalBlank
Writeport &HDEF0, c.GetAttrib("TriggerCode")
SetNextTargetOnsetTime (Clock.Read)
Sleep 10
Writeport &HDEF0, 0
```

Appendix B

We set about trying to measure Tobii’s internal lag-time in the following way. We ran a test file in E-Prime where the entire screen surface was an ROI. We told E-Prime to simulate a keystroke response the instant it first received feedback from the eye-tracker that someone was looking at the ROI. (*This was done while having someone sit in front of Tobii and stare continuously at the screen*). Then we asked E-Prime to calculate the “reaction time” between when the image/ROI appeared on the screen and when it received the simulated keystroke response, and the lag time was ~3 ms on each of 100 trials (meaning that, 3ms after the eye-tracker could begin checking for eye-contact in the ROI, E-Prime received feedback from the eye-tracker).

This also correlates with what we find on the Neuroscan .CNT files. The time delay between a trigger indicating onset of the ROI (t_1) and a trigger indicated that eye-contact was made (t_2) was consistently ~36/37ms. This is precisely the delay we would expect: it consists of two refresh-cycle delays for drawing the image (~33/34ms) + Tobii’s 3ms delay. NOTE: E-Prime factors in the refresh cycle delay in its reaction time calculation—hence why it showed the reaction time to be = 3ms. On Neuroscan we have to adjust for these delays (see above), so 36/37ms – 33/34ms = ~3ms, which gives us Tobii’s lag time.

References

Scan 4.5 (2003). Neuroscan User’s Guide. El Paso, TX: Compudemics Neuroscan.

Schneider, W., Eschman, A., & Zuccolotto, A. (2002). E-Prime User's Guide. Pittsburgh: Psychology Software Tools, Inc.

Schneider, W., Eschman, A., & Zuccolotto, A. (2002). E-Prime Extensions for Tobii. Pittsburgh: Psychology Software Tools, Inc.

Tobii Studio (2010). Tobii Studio 2.2 User's Manual. Stockholm, Sweden: Tobii Technology AB.