Linguistic Approaches to Bilingualism

The Art of Wrangling: Best Practices for Reporting Web-based Eye-tracking Data in Language Research

--Manuscript Draft--

Manuscript Number:	
Full Title:	The Art of Wrangling: Best Practices for Reporting Web-based Eye-tracking Data in Language Research
Short Title:	The art of wrangling
Article Type:	Special Issue article
First Author:	Adam A. Bramlett
Other Authors:	Adam A. Bramlett
Corresponding Author:	Seth Wiener, Ph.D. Carnegie Mellon University Pittsburgh, PA UNITED STATES
Funding Information:	
Section/Category:	Basic Science Section
Keywords:	data quality; online research; open science; eye-tracking; psycholinguistics
Manuscript Classifications:	10.050: Typical adult L2 acquisition; 20.500: Cognition
Abstract:	Web-based eye-tracking is more accessible than ever. Researchers can now carry out visual world paradigm studies remotely and access never before tested populations via the internet without the need for an expensive eye-tracker. Web-based eye-tracking, however, requires careful experimental design and extensive data wrangling skills. In this paper, we provide a guide for building reproducible, open science eye-tracking studies using the online experiment builder Gorilla. We provide step-by-step instructions to building a typical linguistics eye-tracking study, and walk the reader through a series of data wrangling steps needed to prepare the data for visualization and analysis using the open-source software environment, R. Importantly, we highlight the key decisions researchers need to make and report in order to reproduce an analysis. We demonstrate our approach by carrying out a single change replication of an in-person eye-tracking study, Porretta et al. (2020). We conclude with best practices and recommendations for researchers carrying out web-based eye-tracking studies.
Author Comments:	This is a submission for Dr. Cunnings, Dr. Pontikas, and Dr. Pliatsikas' special issue on "Advanced Quantitative Methods in Bi-/Multilingualism." There was no option in the dropdown menu to include it there.
Suggested Reviewers:	
Opposed Reviewers:	

Cover letter

Carnegie Mellon University

Seth Wiener
Associate Professor
Director of Graduate Studies
Department of Modern Languages
Carnegie Mellon University
5000 Forbes Ave.
Pittsburgh, PA 15213
(412) 268-5669
sethw1@cmu.edu

30 November 2023

Re: Cover letter for submission to *Linguistic Approaches to Bilingualism*

Dear Dr. Cunnings, Dr. Pontikas, and Dr. Pliatsikas:

This letter accompanies our manuscript, "**The Art of Wrangling: Best practices for reporting web-based eye-tracking data in language research**," which we wish to be considered for publication in the special issue of *Linguistic Approaches to Bilingualism* on *Advanced Quantitative Methods in Bi-/Multilingualism* as a full-length research article. We believe that this contribution fulfills two major needs in the field. First, we provide a step-by-step guide to carrying out fully replicable eye-tracking studies using the increasingly popular web-based experiment building platform, Gorilla. Second, we outline a series of best practices that researchers should follow (and hopefully report) in their own web-based eye-tracking study. We demonstrate both of these contributions by way of replicating Porretta, V., Buchanan, L. & Järvikivi, J. (2020). When processing costs impact predictive processing: The case of foreign-accented speech and accent experience. *Attention, Perception & Psychophysics, 82*, 1558–1565. The highlights from our manuscript, which we believe will appeal to readers of *Linguistic Approaches to Bilingualism,* include:

- Step-by-step guide to building visual world paradigm studies on Gorilla
- Detailed R code demonstrating eye-tracking data wrangling steps
- Recommended best web-based eye-tracking practices for the field
- Successful replication of Porretta et al.'s lab-based eye-tracking study

The manuscript has not been published and is not under consideration elsewhere. Every effort has been made to follow the format requirements for the journal. We have also complied with APA ethical standards in the treatment of our human subjects and the data. All data, R code, and stimuli are available on our Open Science link, which we have anonymized for peer reviewers and included in our manuscript. We also include links to our open-source Gorilla experiment and anonymized links to Shiny apps, which allow the reader and reviewer the opportunity to explore our data in educational, interactive ways.

There are many experts in the field who can serve as knowledgeable reviewers on the topic of eye-tracking. Here are three reviewers whose work we cite and are unaware of our submission, and can therefore serve as objective readers:

- Myrte Vos**,** UiT the Arctic University of Norway, myrte.vos@uit.no
- Aine Ito, National University of Singapore, aine.ito@nus.edu.sg
- Pia Knoeferle, Humboldt University, Berlin, pia.knoeferle@hu-berlin.de

We hope you will find this work to be an important contribution to the ongoing shift to web-based psycholinguistic research and the efforts to increase open science and reproducibility in our field. On behalf of my co-author, Adam Bramlett, we thank you for your consideration of our manuscript.

Sincerely,

Seth Wiener, Ph.D.

The Art of Wrangling: Best Practices for Reporting Web-based Eye-tracking Data in Language Research

University: XXX

Authors: XXX

XXX

Abstract

Web-based eye-tracking is more accessible than ever. Researchers can now carry out visual world paradigm studies remotely and access never before tested populations via the internet without the need for an expensive eye-tracker. Web-based eye-tracking, however, requires careful experimental design and extensive data wrangling skills. In this paper, we provide a guide for building reproducible, open science eye-tracking studies using the online experiment builder Gorilla. We provide step-by-step instructions to building a typical linguistics eye-tracking study, and walk the reader through a series of data wrangling steps needed to prepare the data for visualization and analysis using the open-source software environment, R. Importantly, we highlight the key decisions researchers need to make and report in order to reproduce an analysis. We demonstrate our approach by carrying out a single change replication of an in-person eye-tracking study, Porretta et al. (2020). We conclude with best practices and recommendations for researchers carrying out web-based eye-tracking studies.

***Keywords**:* data quality, online research, open science, eye-tracking, psycholinguistics

1. Introduction

1.1 Data Wrangling is Data Analysis

Data analysis is not only statistical analysis. Data analysis also includes data clean-up, transformation in and between data sets, visualization, and statistical analysis (Wickham & Grolemund, 2017). Yet, quantitative multilingual research often reports vague practices or fails to report any decisions made outside of statistical models, in part, because pre-processing software has already made the decisions for the researchers (Prystauka et al., 2023). These decisions, however, have pervasive implications across data analyses that affect replicability and reliability (Coretta et al., 2023). This is especially true for methods that capture real-time language processing, such as eye-tracking. Whereas open research practices, including shared data and code (Bolibaugh et al., 2021), serve as a positive first step, the field still has a long way to go.

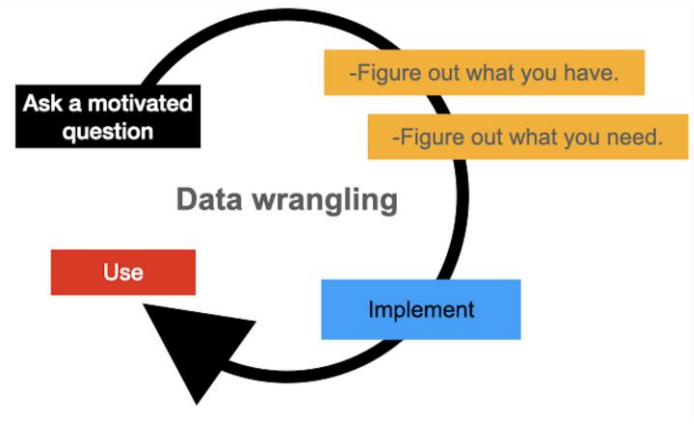

Figure 1. *The data wrangling cycle: an iterative process including all steps, which reduce, reorder, extend, tidy, transform, and/or combine your data.*

Here, we focus on data wrangling (see Figure 1); the iterative process of cleaning raw data straight from an experiment and transforming it into a usable structure (i.e., tidy data or data ready for visualization and statistical analysis) within a typical visual world paradigm

eye-tracking study. Web-based eye-tracking has become more accessible and reliable than ever, capturing many effects found in in-person experiments for a fraction of the cost (e.g., Degen et al., 2021; Prystauka et al., 2023; Semmelmann & Weigelt, 2017; Vos et al., 2022). Access to this method, however, comes at the cost of time-consuming, multipart data wrangling (e.g., Prystauka et al., 2023; Vos et al., 2022). Data from web-based experiments are currently even more complex than their in-person counterparts given the lack of subscriber-based pre-processing software, which means that the choices made during data wrangling are a new opportunity for radical open science practices. We present ***The Art of Wrangling*** as a best practice guide for web-based eye-tracking using the lingua franca of data analytics, R (Mizumoto & Plonsky, 2015; R Core Team, 2022).

1.2 Designing and Building a Typical Eye-tracking Experiment

Eye-movements provide a fine grain measure of various levels of language processing (e.g. Allopenna et al., 1998; Cooper, 1974; Tanenhaus et al., 1995). The now classic visual world paradigm (VWP) involves displaying visual stimuli including a target, competitor(s), and distractor(s) with a variety of possible layouts and formats, from pictures to words.

As seen in Figure 2, a set of images are displayed on a screen time-locked to a point in an audio stimulus (e.g., beaker). The participant then either needs to select the correct answer based on the audio that they perceived or simply listen and look as the sound stimulus plays (e.g., passive listening). VWP experiments vary widely in what linguistic process is being investigated (e.g., referent prediction, sentence processing, word recognition, phonetic cue integration). However, all VWP experiments carefully control three core constructs (i.e., time, audio stimuli, and visual stimuli) in order to bring meaning to a fourth core construct: eye-fixations. For the remainder of this paper, these "core four" constructs will be used to guide the reader’s understanding of how variation in eye-movement behavior can be captured,

organized, and analyzed. We use color consistently throughout this paper for reference to the core four constructs: blue (time), green (audio stimuli), black (visual stimuli), red (eye-fixations).

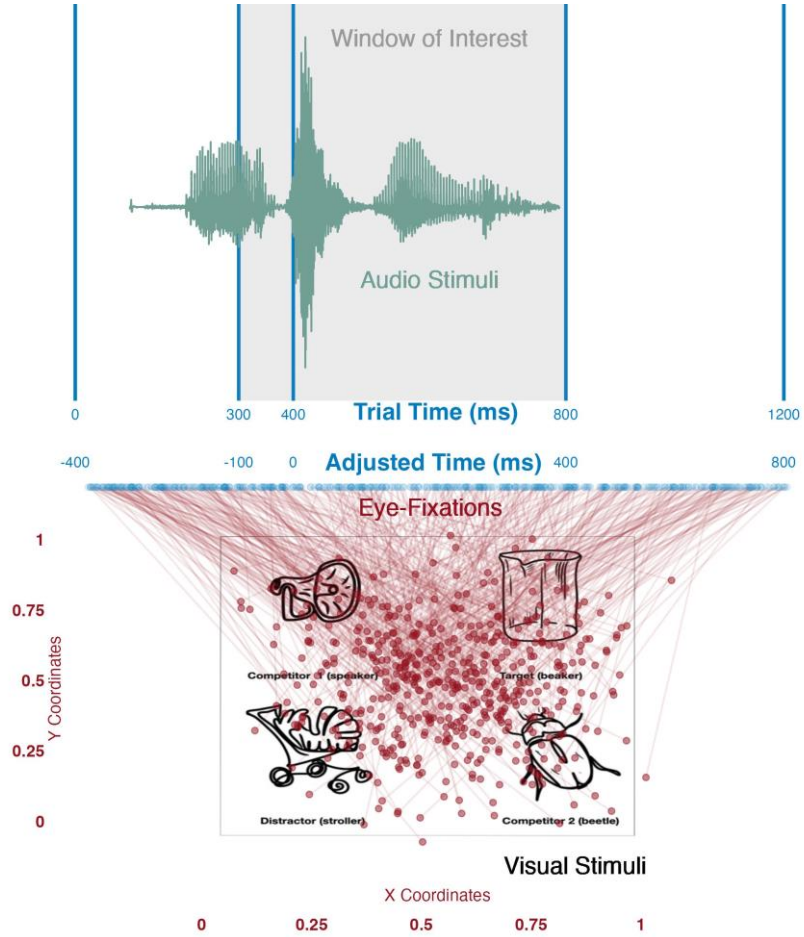

Figure 2. *Illustration of the core four constructs of the VWP. Eye fixations, represented by red dots, and respective times (blue dots).*

1.3 The Core Four of a 2x2 VWP Experiment

Time. Eye-tracking is especially valuable because it provides the time-course of processing. Time can be measured from the beginning of the trial to the end of the trial ('Trial Time' in Figure 2). There are two adjustments, however, that are typically made ('Adjusted Time' in Figure 2). First, it typically takes a listener about 200ms to plan an eye-movement (Matin et al., 1993). Eye-movements within the first 200ms are therefore discarded and researchers typically account for this 200ms delay by adjusting the analysis. Second, within

each trial there exists a window of interest (grey area in the top of Figure 2), which contains the crucial information necessary to identify the target. For example, time in which any carrier phrase is presented is typically ignored and time after the start of the target word is examined.

Audio Stimuli. This is the auditory input a participant receives on each trial. The audio stimuli can be a word, a sentence, or even a non-speech noise. The audio informs the participant about the visual stimuli, often indicating which on-screen visual stimulus is the target or topic of the sentence. The audio stimuli must be carefully locked to time. For example, the end of the green audio stimuli in Figure 3 is time-locked to end at 800ms (trial time).

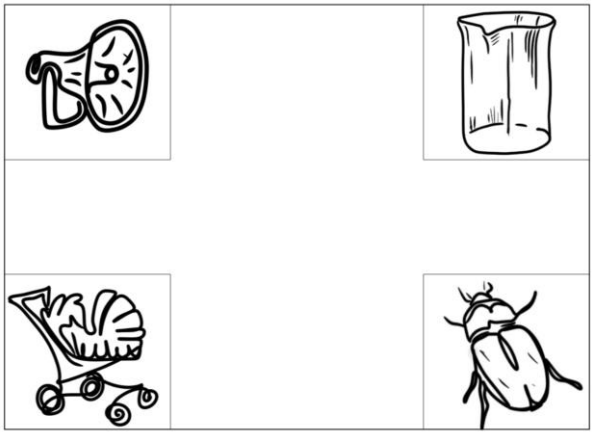

Figure 3. *Example visual stimuli including a target 'beaker', onset competitor 'beetle', rhyme competitor 'speaker', and distractor 'stroller' (stimuli inspired by Allopenna et al. (1998)).*

Visual Stimuli. Figure 3 is an example of the visual information shown to a participant on each trial. Visual stimuli can be presented with a preview time or simultaneously with the audio stimuli (Apfelbaum et al., 2021). Ultimately, the specific timing used in a study depends on the research question. Visual stimuli are minimally made up of two types: targets and competitors. In the case of four visual stimuli, an additional two visual stimuli include either a second competitor and a single distractor (or two distractors, if a second competitor is not built into the design). Visual stimuli are always counterbalanced

across the four quadrants so as to reduce the chances of bias in eye-movements in a particular direction. Note that quadrants are absolute positions on the computer screen (e.g., upper right ('beaker' in Figure 3), upper left, bottom left, bottom right).

Eye-Fixations. Eye-fixations are time-stamped x- and y- coordinates on the screen that are recorded throughout a trial. In other words, where a participant is looking at a particular time. In Figure 2, red dots are specific x- and y- coordinates and red lines tie those fixations to specific times (blue dots). The rate of recording is a function of the measurements recorded per second (e.g., measuring 1000 times in one second = 1000Hz). Eye-fixations get categorized into absolute positions on the screen (quadrants) and then mapped to visual stimuli. Where a participant is looking over time is informed by the audio stimuli.

2. Building an Eye-tracking Experiment in Gorilla

Beginning with Task Builder[1], each trial should start with a fixation cross for roughly 200ms. Next, a simple forced-choice task can serve as the foundation of the experiment with four visual stimuli as the choices (see Experimental ET Tasks: simple forced choice at Gorilla link). Audio input comes from the *web audio zone* that plays at the beginning of a specific *screen*, whereas *response button image* is the *zone* for visual stimuli. The audio and visual stimuli must be time-locked. When building the experiment, it is essential to focus on the timing of the trials, the types of data you want out of the trial[2], and when the webcam should track eye-fixations.

---

[1] This section was written with both Gorilla Task Builder 1 and Task Builder 2 in mind (e.g., zones and objects are the same thing). However, the terminology used follows Task Builder 1 as Task Builder 2 does not yet have eye-tracking functionality.

[2] Feedback is often used in multilingual studies, and would simply require an additional *screen* indicating the correct target, such as a circle around the beaker or written corrective feedback.

In Gorilla, *tasks* are made up of *displays*, which can be thought of as trials. The *display* is a useful unit because it allows the researcher to make recursive functionality (e.g., run the same trial with different content *n* times). Within each *display*, *screens* are played consecutively. That is, to control the overall relationship between the timing of audio stimuli and visual stimuli (i.e., preview, simultaneous), *web audio* should be placed either in the *screen* before or after the visual stimuli (examples for each type of timing is provided in the Gorilla link). As seen in Figure 4, the exact location of your *web audio* depends on where you want it time-locked to the visual stimuli in terms of *screens*.

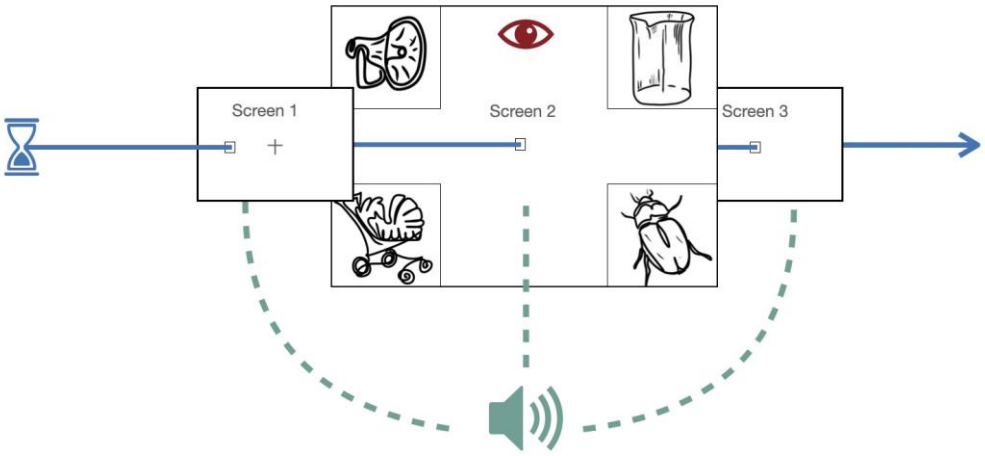

Figure 4. *Example Gorilla display with three screens.*

While the overall timing of a trial is controlled by the placement of *screens*, the transition between *screens* and fine-grained timing within *screens* is manipulated through *zones* within each *screen*. *Zones* include functionality like (*Response Buttons*, *web audio*, and *eye-tracker 2*). There are two general types of *zones* which are important in the context of eye-tracking (i.e., *content-response* and *control*). *Content-response zones* allow the user to put in audio and visual stimuli in specific locations. Importantly, not all of the *Content-response zones* enable screen progression. For example, the *web audio zone* can end a *screen* on completion;

however, you may need the *screen* to progress at a fixed time. In these cases, *control zones* allow a *screen* to progress at specific time intervals between *screens*.

2.1 Gorilla Settings

In each *zone*, *configuration settings* allow the user to enable features for both the experiment and data output. For example, *screens* with *web audio* can either progress automatically or continue until another *zone* forces progression. Similarly, *response button images* for the visual stimuli can end a *screen* on-click or the *screen* can end through control zones *time limit section/screen*, with exact timing in *configuration settings*. In terms of data output, each *response-button image* should be named (e.g., "image_1") so that you can differentiate clicks in the data analysis.

Eye-tracker 2 is found in the *advanced zone*; it has two explicit modes: *calibration* and *recording*. Either a five-point or nine-point calibration can be used, with any level set for calibration fail points or repeat calibrations. Nine-point calibration provides a better standard but takes longer and may fail more often. While not fully necessary because of the manner in which webgazer.js functions (Chen et al., 2001), it is recommended that the researcher calibrate participants at the beginning of the experiment and throughout the experiment (Prystauka et al., 2023). We recommend always reporting calibration metrics.

The *recording screen* is used when the visual stimuli begins. All screens you wish to record eye-fixations for should have an *eye-tracker zone*. The choice of *Basic eye-tracking data* and *detailed eye-tracking data* is in *configuration settings*. *Basic data* does not provide time course data, it only provides percentages of looks. For this reason, you will need to select *detailed data*. Additionally, in cases where you want to record multiple *screens*, you should select *continue recording* in *Eye-tracker 2 configurations settings*. Importantly, webcams have variable frame rates that depend on the lighting, participant movement, and the participant's device,

which can range between 20Hz and 60Hz (Vos et al., 2022). The typical raw eye-fixation samples captured per second is 15, 30, 60, and 120 (standard webcam frame rates) but likely much lower. Additionally, the lighting environment of the participant has a strong effect on the number of fixations recorded. For example, darker rooms will lead to the camera capturing less eye-fixation per second (lower fps). This means that some trials will capture more eye-fixations than other trials (Prystauka et al., 2023). Additionally, the timing of eye-fixations can vary within a trial with non-equal measurements between captured eye fixations. This means that the eye-fixations being captured start to drop throughout the trial. This variability in frame rate can be somewhat attenuated by doing in-person eye-tracking with web-gazer but is nonetheless somewhat unavoidable (e.g., Papoutsaki et al., 2016).

2.2 Gorilla Data and Tidy Data

Raw data from a VWP experiment downloaded from Gorilla has two basic parts: task data by *node* and an *uploads folder*. Task data will include all selections and timings of those selections (e.g., reaction time, condition, trial order). The additional *uploads folder* will contain trial-by-trial eye-fixation data that is paired with within-trial trial-time. This format is not limited to eye-tracking data and is true for any Gorilla experiment that would need continuous trial-specific data (e.g., voice recordings, mouse-tracking) beyond simple behavioral responses (e.g., reaction time, accuracy). A simplified example of this is shown in Figure 5.

Task data

ID	Trial	Audio file	Response	Correct
		Audio Stimuli	Visuali stimuli selected	
1	1	beaker.wav	image_1	1
1	2	stroller.wav	image_3	1
2	1	beatle.wav	image_4	0
2	2	speaker.wav	image_1	1

Eye-tracking data (1 participant x a single trial)

ID	Trial	Time	Screen location x	Screen location y
		Trial Time	Eye-fixations	
1	1	0.01	57.6	-23.6
1	1	0.029	59.2	-22.2
1	1	0.038	106.7	98.6
1	1	0.082	113	85.2

Figure 5. *Task data (left) and trial-specific eye-tracking data (right)*

The number of task data sheets has a direct relationship with the number of experimental (and/or questionnaire) *nodes* in the *experiment*. If counterbalancing audio stimuli by condition so that participants do not hear the same audio from two speakers, then each of these will also have their own *node* separated by *spreadsheets*. That is, if you were to have four counterbalanced stimuli *spreadsheets* in your experiment design then you will have four raw task data sheets, one from each of these *nodes* (see, "Experimental ET Tasks: All," for an example). However, the number of data sheets in the *uploads folder* will depend on how many participants and trials you have. For example, if you have 60 participants who finish 48 trials, you will have a total of 2,880 eye-tracking data sheets (60x48) that reference the behavior trials in your four task data sheets. This is the size of the data we use in the data wrangling section.

In sum, the raw data that Gorilla provides is maximally informative to enable a variety of analyses to be done. The challenge, however, and the focus of the remainder of this paper is how one tidies the data so that each column refers to a single variable (e.g., audio stimuli) and each row is exactly one observation (e.g., “beaker.wav”). In order to better demonstrate this

process, we next walk the reader through a replication study involving predictive sentence processing.

3. Replication of Porretta et al. (2020)

2.1 Background and Motivation

We carried out a single change (web-based data collection) replication study of Porretta et al. (2020)'s in-person VWP experiment. Porretta et al. (2020) showed that a foreign accent impedes predictive processing but does not preclude it, and as experience with that accent increases, the predictive processing behavior approximates that seen for non-accented speech. Porretta et al. (2020) was chosen for replication for two principled reasons following the recommendations of Marsden et al. (2018): 1) The majority of materials were made available by the researchers, which minimizes heterogeneity. 2) The recency, novelty, and theoretical impact of the initial study warrant replication for the sake of validation and generalizability.

Porretta et al. (2020) involved a 2-by-2 experimental design testing talker (native/non-native) and verb type (restrictive/non-restrictive, e.g., "the fireman will climb the ladder", *climb* allows for object prediction or "the fireman will need the ladder", *need* does not allow for object prediction). These English sentences were spoken by either a native or Chinese accented talker. Whereas our study changed only the method of collecting data, this single change causes three immediate differences between our replication and (Porretta et al., 2020) summarized in Table 1.

Table 1. *Key Differences Between our Web-based Replication Study and Lab-based Porretta et al. (2020)*

	Porretta et al. (2020)	Our web-based replication
Eye-tracker	Eyelink 1000	Variable personal webcams
Participants	60 university students	60 Prolific participants
Data wrangling	Pre-processed	Self-wrangled

2.2 Methods

We used Gorilla Experiment Builder's eye-tracking 2 zone implemented with WebGazer.js (Anwyl-Irvine et al., 2019; Papoutsaki et al., 2016). All research materials, R data analysis, Gorilla experiment and tasks, and data are available on the Open Science Framework (OSF) (Foster & Deardorff, 2017). The study was approved by the authors' Institutional Review Board. All participants were compensated for their participation. Average completion time of the experiment was 16 minutes including a second (pilot) task that is not reported here.

2.2.1 Participants

To ensure direction comparison to Porretta et al. (2020), we tested the same number of participants, 60 (median age = 31). We recruited through Prolific (Palan & Schitter, 2018) using the same criteria: native monolingual English speakers, between the ages of 18 to 40. Prior to reaching 60 participants, 37 participants were rejected (eight failed headphone check, 23 failed eye-calibration, 5 timed-out after 90 minutes, one did not consent). As we demonstrate below, an additional 11 participants were removed during the data tidying, resulting in 49 total participants analyzed. We return to this internet data quality issue in the discussion.

2.2.2 Materials

All recordings were taken from Porretta et al. (2020). The experiment contained 250 images, 50 of which were center images and 200 that made up the 2x2 design. 99 of the images were identical to the original experiment (all 50 center images for subjects in the sentences and 49 of the visual stimuli for objects across practice, filler, and experimental items). The remaining 151 images were obtained following the same specifications of the initial study (open-source line-drawn images). Four of the images were created by lab members in-house due to not being available online. Four presentation lists were made which counterbalanced talker and verb type.

2.2.3 Procedure

Participants were recruited through Prolific. After consenting to participate, each participant did two headphone checks: a basic listening task to ensure the audio was loud enough and a dichotic pitch task (Milne et al., 2021). Next, participants did a 5-point eye-calibration set to reject participants below four successful points with a limit of three calibration attempts before rejection. On each trial (24 target, 24 filler), participants were presented with a 500-ms fixation cross followed by a 2x2 visual stimulus with an additional center image that represented the subject of the sentence. Each stimulus was previewed for 200ms. Next, participants heard either a restrictive or nonrestrictive sentence spoken with either a native accent or non-native accent while looking at the visual stimuli. Participants then answered a simple comprehension question to ensure attention. After the experimental task, participants filled out a brief questionnaire (identical to Porretta et al.'s) including age, language experience, and estimated Chinese accent experience (captured on a scale of 0-100 with a slider that starts at zero). In order to make a comparison to Porretta et al.'s reported

mean of 1.78 (SD = 0.82), accent experience was scaled to 0-30 and then log transformed with a constant of 1. Our population's mean of 0.99 (SD = 0.92), therefore, appears to be lower than that of Porretta et al.'s.

2.3 Data Analysis

In what follows, "L: + line number" (e.g., L:156-157) refer to line numbers in `AOW_r_work_flow.rmd` found on OSF. In L:33, we read in three data frames: The `task_data`, `eye_tracking_data`, and `OSF_data`. To run this, download the data folder from OSF and select task_data.csv when prompted by R after running L:33. You can load the other data frames by running the following lines. Following Figure 5, the `task_data` is made up of the experimental data and information obtained during testing; the `eye_tracking_data` is made up of eye-fixations. `task_data` is made up of a messy 97,827 rows by 111 columns, and `eye_tracking_data` is made up of an overwhelming 400,305 rows by 36 columns. As noted earlier, the data are relational. In the next 200 lines of code, we wrangle these structures into data that we can fully use, adapt, and share (see supplementary `combing_data.Rmd` for three methods on combining separate experimental files into a single data frame).

```
## ----Data Reading---
#select task_data
task_data_select<-file.choose()
task_data<-read.csv(task_data_select,header=TRUE, row.names=1)
#change for ET data
et_data_select<-sub("task_data", "et_data", task_data_select)
eyetracking_data<-read.csv(et_data_select,header=TRUE, row.names=1)
#change for OSF data
OSF_data_select<-sub("task_data", "OSF_data", task_data_select)
OSF_data<-read.csv(OSF_data_select,header=TRUE, row.names=1)
```

2.3.1 Questionnaire wrangling

Data wrangling always starts with data removal. In a VWP experiment, removal occurs at four levels: questionnaire-based, item-based, behavior-based, fixation-quality-based. Which level you start with is unimportant; we start with questionnaire-based removal and ask which participants should be excluded based on post-experiment questionnaire exclusion criteria (e.g., not an L1 English speaker and not between the ages of 18 and 40)? In L:43, we start with a clone of our behavioral data frame `task_data` and assess needed variables (`Screen.Name`, `Responses`, `Participant.Private.ID`, `Reaction.Time`(RT)). RT is kept because it allows for removing items that were unnecessarily generated from the experiment structure (i.e., getting rid of rows with 0 RT).

```
## ----Questionnaire: Clean---
cleaned_quest_data<-task_data%>%
filter(display=="questionairre",na.omit=TRUE)%>%
select(Participant.Private.ID,Screen.Name,Response,Reaction.Time)%>%
filter(Response != "",Reaction.Time!=0)%>% select(!Reaction.Time)
```

Now that we have a data frame with three columns (`Participant.Private.ID`, `Screen.Name`, `Response`), we can create tidy data with one observation per row and one variable per column. `pivot_wider()` and `pivot_longer()` offer a simple solution to this common data structure problem. Figure 6 demonstrates how experimental data (e.g., Gorilla-tasks, Psychopy, E-Prime) often require widening, whereas questionnaire data (e.g., Gorilla-questionnaires, Google forms, Qualtrix) require pivoting longer. In L:49, we pivot wider to create a single row for each participant with each question having its own column. It is much easier to come up with standards for removal in the `speaks_L2`, `age`, or `hear_impaired` columns than for the `Response` column, which would require conditional standards based on `Screen.Name`.

```
## ----Questionnaire: Tidy---
tidy_quest_data<-cleaned_quest_data%>%
group_by(Participant.Private.ID,Screen.Name)%>%
summarise_all(toString)%>%
```

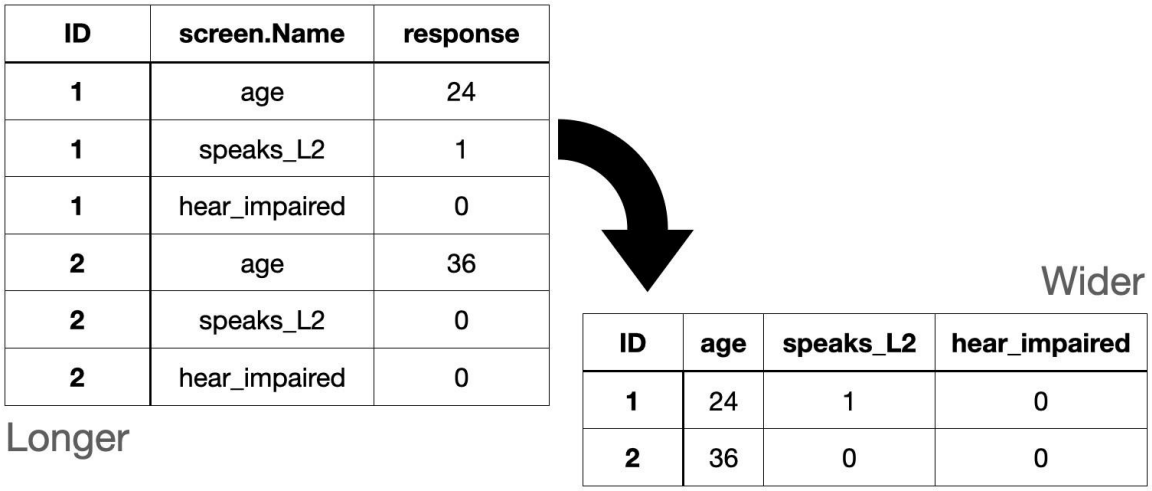

Figure 6. *Examples of long data (left) and wide data (right).*

```
53    pivot_wider(names_from=Screen.Name,values_from=Response)%>%
54    mutate(speaks_L2 =if_else(str_detect(other_languages_spoken,"German")
      &
55                        !is.na(other_languages_spoken),1,0),
56              across(c(chinese_study_duration,age,experience_chinese_accent)
      , as.numeric),
57              Participant.Private.ID = as.factor(Participant.Private.ID))%>%
58
59    select(!other_languages_spoken)
```

In L:69, we find that two participants should be removed for language expertise outside English and one for exceeding the age cut off (both predetermined values based on Porretta et al.). We can now use this data frame to filter out unqualified participants in the `Participant.Private.ID` column of the next removal stage. (See L:64-69 in `AOW_r_work_flow.rmd` for an example of helpful visualization.

```
69  ## ----Questionnaire: Filtered---
70  filtered_quest_data<-tidy_quest_data%>% filter(age<=40
71  & age>=18, #1 removed for age range
72  chinese_study_duration==0, #none removed
73  speaks_L2==0,#2 removed that speak other languages
74  language_disorder == "No") #none removed
```

2.3.2 Behavioral-task wrangling

The next cycle of data wrangling begins with the question: Which participants and items should be removed based on the behavioral results? Cleaning is similar to the questionnaire cycle, but we start from scratch with a clone of `task_data` called `experimental_cleaned` because the new question has new goals, which requires different variables. We start this cycle's implementation by filtering the participants in the behavioral-task clone with the questionnaire data from above in order to only keep those participants that qualified in the questionnaire wrangling cycle (L:77). We then remove all rows except ones related to behavioral data questions (L:78-79) and experimental items (L:80), followed by removing columns with all NAs. Lastly, to achieve tidy data, we split the visual image selection and comprehension question into two columns so that each participant has a single observation for each trial (e.g., pivot into a wider structure, L:84). Removal of columns in L:86-88 makes pivoting possible. Pivoting requires that rows do not have uniquely identifiable information outside the data columns being "widened" (This could also be achieved with the column argument of `pivot_wider`).

```
76  ##  ----Experimental  Data:  Clean  and  Tidy---
77  experimental_cleaned      <-      task_data%>%
78  filter(Participant.Private.ID %in%
79  filtered_quest_data$Participant.Private.ID)%>%
80  filter(Zone.Type == "response_button_image"|
81  Zone.Type == "response_button_text")%>%
82  filter(verb_type  ==  "Restricting"  |verb_type  ==  "NonRestricting")%>%
83  select_if(~sum(!is.na(.)) > 0)
84
85  experimental_tidy<-experimental_cleaned%>%
86    select(!c(Event.Index:Local.Date,
87  Screen.Number:Zone.Name,
88  Reaction.Time:Response.Type))%>%
89  pivot_wider(names_from = Zone.Type,values_from = Response)%>%
90  mutate(subject_img_file=center_image) #for renamed matching in next step
```

Additionally, we must load in a second data frame `OSF_data` (L:94) from the original experiment. We do this because our experiment only has the quadrants or the visual stimuli without the target, competitor, and distractor information, and later we need `SUBTLWF_obj`, which is the log frequency of the object words used in the statistical models.

```
93  ## ----OSF Data: Clean and Tidy----
94  OSF_filt<-OSF_data%>%
95    select(talker,verb_type,subject_img_file,img_1_file, img_2_file,
96  img_3_file, img_4_file,log_SUBTLWF_Obj)
```

In L:99, we filter the `OSF_data` for experimental items and use a `left_join()` based on talker condition `verb_type`, and the center visual image `subject_img_file`, which

simultaneously pulls in the variables that we need and filters out nonce items (this step could be avoided by putting these variables in the original experimental spreadsheets). Figure 7 demonstrates filtering through different types of joining.

```
98  ## ----Behavioral Data: Join OSF and Experimental Data---
99  behavioral_data<-experimental_tidy%>%left_join(OSF_filt,
100 by=c( "talker","verb_type","subject_img_file"))
```

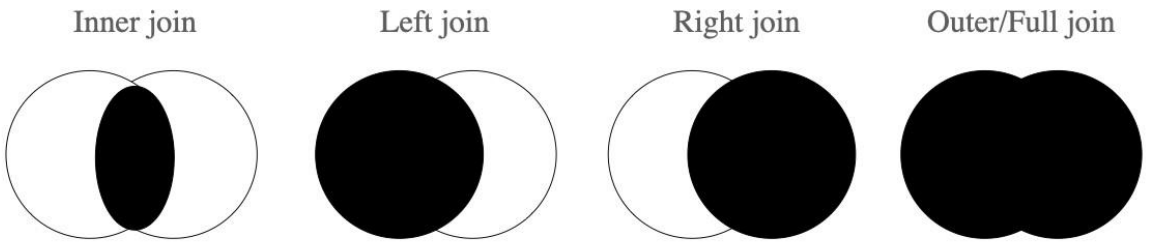

Figure 7. *Joins act as filters, determining what data to include or exclude based on commonalities and differences between data frames. Solid portions refer to what is kept.*

Now that we have the variables we need in `behavioral_data`, we can create variables for the answers being correct/incorrect for our removal process. We will do this for both the item selection (L:105) and comprehension question (L:106).

```
102 ## ----Behavioral Data: Clean and Tidy---
103 behavioral_data <-behavioral_data %>% mutate(participant =
104 as.factor(Participant.Private.ID),
105 image_incorrect= if_else(img_1_file == response_button_image
106    ,0,1), text_incorrect = if_else(response_button_text == "Yes",0,1))
```

Importantly, researchers should establish a criterion for removal prior to data collection. Because Porretta et al. (2020) did not report the criteria they used, we based our

removal on three standard deviations from the mean inaccuracy of participants/items separately, which results in three participants being removed.

```
## ----Behavioral Data: Removal Standards----
#These are in standard deviations to retain maximum amount of quality data
#We set all of these to be 3 SDs, code here is only  for your  future use
image_participant_threshold = 3
image_item_threshold = 3
text_participant_threshold = 3
text_item_threshold = 3
```

We implemented these standards through a series of modular aggregating steps. That is, we aggregated the inaccuracies of participants by adding together incorrect items by participant and item for both item selection (L:118-129) and comprehension question (L:131-142), respectively. We end here by removing the incorrect trials to prepare for the eye-tracking data wrangling (L:144-145).

```
## ----Behavioral Data: Participant and Item Removal----
#participant removal
participant_agg<-behavioral_data%>%
group_by(Participant.Private.ID)%>%
summarize(num_incorrect_image=sum(image_incorrect),
num_incorrect_text=sum(text_incorrect))%>%
mutate(mean_image_score = mean(num_incorrect_image),
sd_image_score = sd(num_incorrect_image),
mean_text_score = mean(num_incorrect_text),
```

```
sd_text_score = sd(num_incorrect_text))%>%
filter(num_incorrect_image <= mean_image_score+
  (sd_image_score*image_participant_threshold)             &
num_incorrect_text <= mean_text_score+
(sd_text_score*text_participant_threshold))
#item removal
item_agg<-behavioral_data%>%
group_by(center_image)%>%
summarize(num_incorrect_image=sum(image_incorrect),
num_incorrect_text=sum(text_incorrect))%>%
mutate(mean_image_score = mean(num_incorrect_image),
sd_image_score = sd(num_incorrect_image), mean_text_score =
mean(num_incorrect_text), sd_text_score =
sd(num_incorrect_text))%>%
filter(num_incorrect_image <= mean_image_score+
  (sd_image_score*image_item_threshold)             &
num_incorrect_text        <=        mean_text_score+
(sd_text_score*text_item_threshold))


behavioral_data <-behavioral_data%>% filter(image_incorrect
  == 0 & text_incorrect == 0)
```

One important note here is that the removal is done in parallel. That is, we removed participants and items simultaneously. If you sequentially remove participant or item first then removal results would be different in the `behaviorial_data`.

2.3.3 Eye-tracking Wrangling

Removal and adjustment of eye-tracking data is done through an exploratory lens as there is little current reference for expected results for eye-fixations and frame rate in web-based eye-tracking. However, recent work has begun to fill this gap (see: Prystauka et al.,

2023; Vos et al., 2022). Here, two questions guide our approach: How should eye-fixations be classified into quadrants in web-based eye-tracking? And, what quality of frame rate is needed to capture the effects of interest? We start by filtering out participants from the previous data sets. Here, the retained participants (L:118) and items (L:131) from the previous step are used to define what we want to keep in the `behavioral_data` (L:148-150) with the `%in%` operator.

```
147 ## ----Behavioral Data: Removing with IN Operator---
148 behavioral_data<-behavioral_data%>%
149 filter(Participant.Private.ID %in% participant_agg$Participant.
150 Private.ID & center_image %in% item_agg$center_image)%>%
151 select(-c(text_incorrect,image_incorrect,response_button_text))
```

Whereas the `et_data` is much larger than the previous data frames, the same methods are used. Selection of data can be reduced to only the time `time_elapsed`, participant `participant_id`, and eye-fixations `x_pred_normalised y_pred_normalised` (L:154-156), which is filtered by only usable fixation points (L:157), followed by variable renaming for upcoming joining of `et_data` and `behavioral_data` (L:158-159).

```
153 ## ----ET Data: Tidying and Filtering with an Inner Join---
154 et_data<-eyetracking_data%>%
155 select(time_elapsed,participant_id,spreadsheet_row,
156 type,x_pred_normalised,y_pred_normalised)%>%
157 filter(type =="prediction" )%>%
158 rename("Participant.Private.ID"="participant_id",
159 "Spreadsheet.Row"="spreadsheet_row")
```

Now that both `behavioral_data` and `et_data` are cleaned and tidy, `left_join()` (L:173) is used to create `all_data` from our `behavioral_data` and `eye_tracking` data. This data frame now has all of the eye-tracking data and behavioral-task data from the entire experiment (L:173-174). However, the data from the `et_data` only includes unclassified eye-fixations. Specifically, it includes the x and y coordinates without a link to the visual stimuli that are being viewed (see Figure 2). A Shiny app was created to dynamically explore how eye-fixations are distributed with variable amounts of removal at four crucial time points: the beginning of the sentence (-400 ms), verb onset (0 ms), object onset, and selection of visual stimuli. The app also includes dynamically calculated data loss. Figure 8 is a fixed version of the fixation points from the app (See Eye-fixations Shiny App in OSF). In the discussion, implications of removal standards based on eye-fixation alone are considered and discussed as a signal detection problem.

As displayed in Figure 8, fixations are mostly distributed at the center of the screen, indicating no looks to quadrants. Whereas this remains true for competitor items throughout the trial, target items begin to move toward visual stimuli as early as the verb onset and much more in later time frames. Crucially, however, the fixations do not always reach the actual quadrants. In analyzing the data from the Shiny app, removing data between the center point of the screen and the inner-edges of the quadrants results in ˜83.33% data loss, which is more than twice as high as previously reported for two image web-based studies (Vos et al., 2022). If we move to a more relaxed categorization, then only 6.71% of data is lost. In contrast, maximal outer-edge removal results in very little data loss (max ˜32%). When removing inner-edge eye-fixations, the choice comes down to removing signal to avoid noise in spatial ambiguity, or embrace noise to maximally retain the signal. As shown in the competitors-time 800 (upper-right) section of Figure 8, the noise is randomly distributed across quadrants just as it is early in the trial before eye-movements tend toward visual stimuli. Here, we aim to strike the balance of the signal-to-

noise trade off by removing most of the data outside the screen size and by maximally retaining inner data that shows trends. This leads us to believe that no bias would occur even if classifying data from the x, y fixation center (0.5, 0.5).

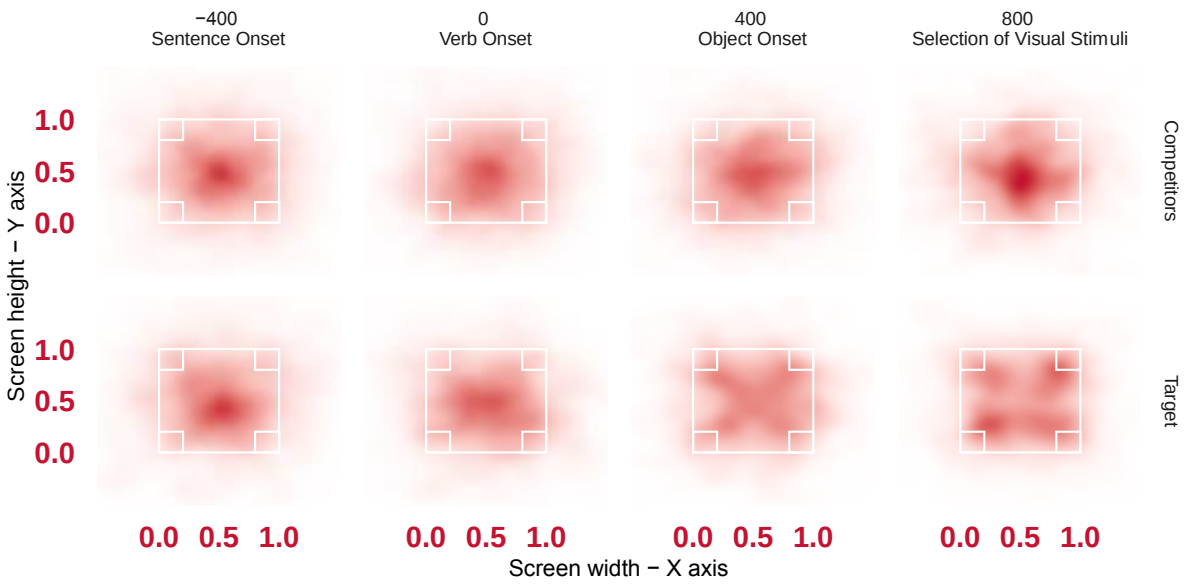

Figure 8. *Quadrant Locations and Actual Screen Sizes are Denoted with White Lines*

From L:180-190, we create a classification system based on no inner-edge removal of the eye-fixations and partial removal of outer-edge eye-fixations (the code was created with inner removal in mind so that future researchers can simply adapt the distance variable L:177, if desired). We use two types of control flow to first classify eye-fixations into quadrants and then create binary variables to link the quadrant to the visual stimuli. `case_when()` is used (L:180-190) because of the multiple conditions and because `case_when()` is only truth evaluating, meaning that it only provides a specific output in the case of something being true. For example, if we only want to classify images that are within a particular space and leave others blank, then non-binary classification like `case_when()` is optimal. In contrast, if the outcomes of a classification are binary, then `ifelse()` is an effective solution. For example, L:192-200 makes a binary decision on whether an image being viewed is the same or different

from the target (L:193), competitors (L:194-195), and distractor (L:196), separately. While complexity of implementation may vary, logically either can be used to achieve the same result in all cases with the use of operators and/or nesting.

```
## ----ET Data: Localizing Visual Stimuli----
#logically equivalent to doing full join and removing non-experimental trials.
all_data <-behavioral_data %>%
left_join(et_data, by=c("Participant.Private.ID", "Spreadsheet.Row"))

center=.5#center of screen
distance=0#distance to visual stimuli
beyond_screen=1 #distance to beyond_screen

all_data<-all_data%>%
mutate(image_viewing=
case_when(x_pred_normalised <= center-distance &
 y_pred_normalised >= center+distance ~ image_1,  x_pred_normalised >=
  center+distance &
   y_pred_normalised >= center+distance ~ image_2,
   x_pred_normalised <= center-distance &
   y_pred_normalised <= center-distance ~ image_3, x_pred_normalised >=
center+distance & y_pred_normalised <= center-distance ~ image_4))%>%
filter(!is.na(image_viewing))



all_data<-all_data %>%
mutate(target = if_else(image_viewing == img_1_file, 1, 0), comp_1 =
   if_else(image_viewing == img_2_file, 1, 0), comp_2 =
   if_else(image_viewing == img_3_file, 1, 0), dist =
   if_else(image_viewing == img_4_file, 1, 0))%>%
filter(x_pred_normalised>center-beyond_screen &
x_pred_normalised<center+beyond_screen&
y_pred_normalised>center-beyond_screen &
y_pred_normalised<center+beyond_screen)
```

In addition to more variable eye-fixations, web-based eye-tracking also has variable frame rates. Figure 9 shows a categorization of participants by median frame rate across trials.

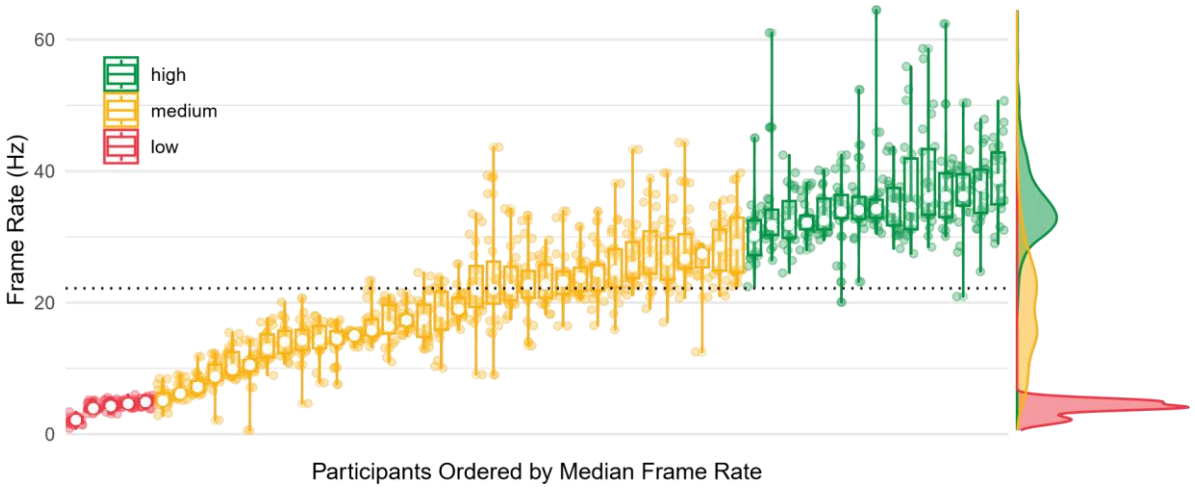

Figure 9. *Participant frame rate. Mean is marked with dotted horizontal line.*

Like other recent web-based eye-tracking studies, our mean frame rate was 20Hz (M = 22.17 Hz, SD = 11.61). Here, we remove the five participants with less than 5Hz median frame rates and create time bins by first creating a standard for removal in L:378 and a binning size (L:379). We then aggregate by participant `Participant.Private.ID`, item `subject_img_file`, and condition `verb_type talker` (L:381) in order to remove all participants that are below the predetermined median (L:381-388). Next, time bins are created by normalizing the time range for each item (L:389). Additionally, we subtracted 200 ms for human eye movements to occur and thus center the time so that 0 is always the onset of the verb of interest (this step was not explicit in Porretta et al. (2020), but we recommend future researchers always make this step explicit). After normalizing, bins are created by dividing the time `time_elapsed` by the bin size `time_binning`, rounding, then multiplying by the bin size `time_binning` (L:390), which is simply rounding items to the nearest bin size number.

```
## ----All Data: Clean and Tidy---
frame_rate_cut_off<-5
 time_binning<-50
all_data_cleaned<-all_data%>%
 group_by(Participant.Private.ID,subject_img_file,verb_type,talker)%>%
mutate(count = n(),
max_time = max(time_elapsed), frame_rate =
count/max_time*1000)%>%
ungroup()%>%
group_by(Participant.Private.ID)%>% mutate(median_frame_rate =
median(frame_rate))%>%
filter(median_frame_rate>=frame_rate_cut_off)%>%
mutate(time_elapsed=time_elapsed-object_start-200)%>%
mutate(time_elapsed_rounded=time_binning*round((time_elapsed)/
time_binning))
all_data_tidy <- all_data_cleaned%>%
filter(time_elapsed_rounded>=-400 & time_elapsed_rounded<=800)
```

Creating time bins is fundamentally discretizing a continuous scale. In any fixed set of eye-tracking data, the grain size of the time scale has an inverse relationship to the amount of data in each time bin. If you increase the bin size, you will have more data per bin, but less bins across time. Many statistical analyses can bypass the binning procedure altogether by keeping time a continuous variable. Nevertheless, for analyses that do require time bins and for visualization alone, it is worth exploring whether specific bin sizes affect a researcher's ability to capture an effect. To do this, we created a second Shiny app that is depicted in Figure 10 (see Frame Rate Shiny App in OSF), which allows the reader to explore the interactions between data removal based on participant median frame rates, changing bin sizes, and seeing output in the form of empirical logits for either linear lines or GAMM smoothed curves. Here, two crucial discoveries are made.

First, almost any arbitrary sized bin captures the effect of `verb_type`, with the caveat of the bin needing to be several sizes smaller than the window of interest. Second, nearly any frame rate of data can capture the effect outside very small frame rates of 5Hz and below. If only examining data that is 6-11Hz, the effect of `verb_type` for `talker` starts to become apparent while the accented speaker effect for `verb_type` becomes apparent between 12-17 Hz.

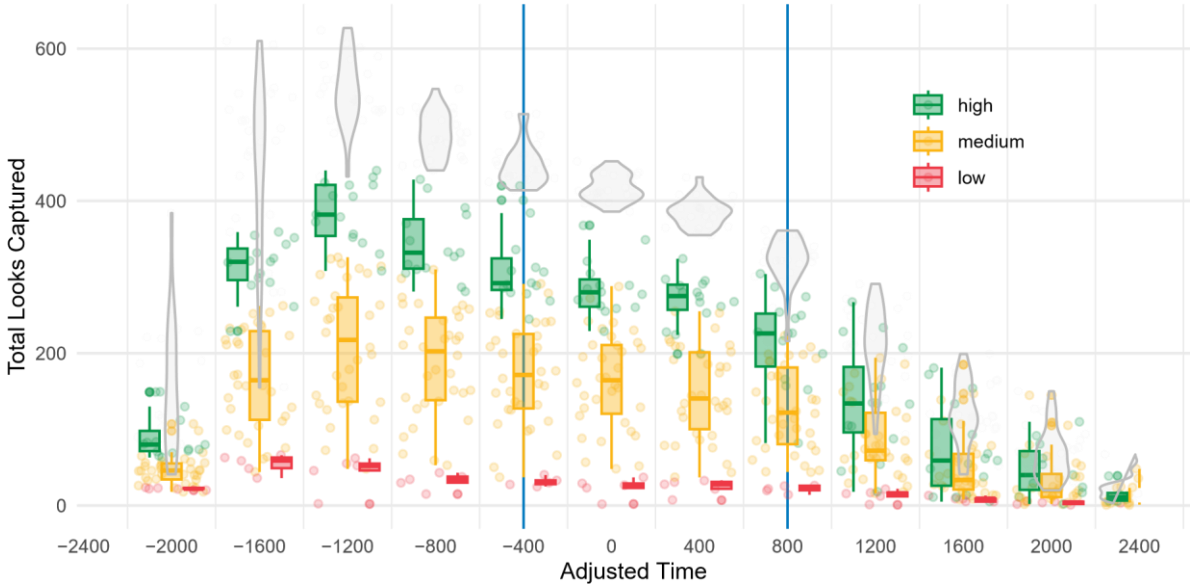

Figure 10. *Total looks per bin size.*

The last step before visualization and statistical analysis is a final tidying. Like the first wrangling that we did, we create a tidy data frame through removal. Here, all eye-fixations that are outside the window of interest (-400ms and 800ms) are removed. Now, our new tidy data is structured based on the core four. For each participant, each audio stimuli and visual stimuli set is classified by `talker` and `verb_type`. Finally, we have removed all times outside the window of interest. By tidying in this way, eye-fixations become meaningful in that each row is classified into looks to targets, competitors, and distractors, and each row is a classified eye-fixation based on a specific time, for each participant, and for varying

conditions. Between the two data frames `all_data_cleaned` and `all_data_tidy`, we have all of the behavioral data ready for any analysis or exploration that can be done.

3. Modeling ET data

In all previous steps, wrangling can be thought of as a condensing process, where the primary object is to remove, clean, and transform the data into a structure that is usable. However, once the data is put into tidy form, then the data must be transformed for specific visualizations and analyses. In this section, we think of `all_data_cleaned` and `all_data_tidy` as launching points to gain an understanding of our data[3].

We start by creating two data frames from `all_data_tidy` : `mem_data` in L:453 and `gamm_data` in L:459. In general, maximally retaining informative columns is essential to creating a usable data frame. When building models, however, it is often best to remove variables that you will not be using. This is because some models can have complications interpreting unprocessed data types (e.g., `NAs`). For `mem_data` , we start by selecting all necessary columns for the model (L:454-455). Factor type conversion occurs next (L:456). Finally, to get background information we join `tidy_quest_data` . In addition to the `mem_data` , we create `gamm_data` by simply cloning `mem_data` in L:459 and by adding a single variable needed in the GAMM models.

[3] If you wish to start from here then read in the `all_data_tidy` and `all_data_cleaned` from cleaned data on OSF.

```
## ----All Data: Preparing for Models---

mem_data<-all_data_tidy%>%

select(Participant.Private.ID,verb_type,talker,

subject_img_file,target,Trial.Number,log_SUBTLWF_Obj,

   target_obj,time_elapsed)%>%

mutate(Participant.Private.ID=as.factor(Participant.Private.ID))%>%

left_join(filtered_quest_data)


gamm_data<-mem_data%>%

mutate(Condition = paste(talker,verb_type,sep="."))
```

There are a handful of excellent papers that outline the advantages and disadvantages of different methods of eye-fixation analysis and relevant considerations for each method of analysis (Barr, 2008; Ito & Knoeferle, 2022; McMurray, 2023; Mirman et al., 2008). Here, we continue to focus on the data wrangling process and present the data wrangling steps—and decisions—needed to carry out two of the more widely used statistical analyses in the field: generalized linear mixed effect models (GLMMs) and generalized additive mixed effects models (GAMMs), which does not require the assumption of linearity. Both GLMMs and GAMMs require specific contrast coding (e.g., dummy, orthogonal) of the data before running models to get expected results. After contrast coding, all model building starts with maximal models, as justified by the design, working down to simpler models for model comparison (see Barr et al., 2013).

3.1 GLMMs

3.1.1 GLMMs: coding

For GLMMs coding, start with data type conversion (L:464-465), then re-level both `talker` (Native, Non-Native) and `verb_type` (Restrictive or Non-Restrictive) so that

`verb_type` Restrictive and `talker` Native are both set as reference levels (L:466-467). We can then rename the contrasts to improve model output readability (L:468-471) and later visualization. In L:473 through L:476, we normalize the `time_elapsed`. Lastly, we create a data frame for the accent model (L:477).

```
## ----GLMM: Leveling the Data---
mem_data$verb_type<-as.factor(mem_data$verb_type)
mem_data$talker<-as.factor(mem_data$talker)
contrasts(mem_data$verb_type)<-c(-.5,.5)
contrasts(mem_data$talker)<-c(-.5,.5)
colnames(contrasts(mem_data$talker))<- c('Native:')
rownames(contrasts(mem_data$talker))<-c("Native","NonNative")
colnames(contrasts(mem_data$verb_type))<- c('Restricting:')
rownames(contrasts(mem_data$verb_type))<-c("Non-Restricting","
    Restricting")
    mem_data$experience_chinese<-mem_data$experience_chinese_accent
    mem_data <- mem_data %>%
mutate(time_normalized =
(time_elapsed - min(time_elapsed)) / (max(time_elapsed) -
min(time_elapsed)))
accent_mem_data<-mem_data%>%filter(talker == "NonNativeMale")
```

3.1.2 GLMMs: models

Two GLMMs were built using the `lme4` package (Bates et al., 2014). Looks to the target (coded as 1, 0) served as the dependent variable. The `Main Model` included three fixed effects: `verb_type` (Restrictive or Non-Restrictive), `talker` (Native, Non-Native) and their interaction (L:509). Random intercepts for `subject_img_file`, `Participant.Private.ID`, and `time_normalized` were included, as were random slopes for `talker` and `verb_type`. The logit link function ("binomial") was specified in the model,

equivalent to modeling logit-transformed response probability with identity link function. Model comparison[4] showed preference for the full model with ANOVA comparisons ($p < .001$) and lower AIC and BIC.

```
## ----GLMM: Main Model---
glmm1_1<-glmer(target~talker*verb_type+
(talker|subject_img_file)+
(verb_type|Participant.Private.ID)+
  (1|time_normalized),
  family="binomial",data=mem_data)
summary(glmm1_1)
```

Similar to the above model, an accent only model was run on `accent_mem_data`. Model specifications are identical to `Main Model` outside of changing fixed effects to `experience_chinese` (L:540). Additionally, `talker` is removed as a random slope because `accent_mem_data` only has one `talker`: accented. Full models were shown to outperform simpler models from ANOVA comparisons ($p < .001$) and lower AIC and BIC, as well as non-convergence of simpler models.

```
## ----GLMM: Accent Model---
glmm2_1<-glmer(target~experience_chinese+
(1|subject_img_file)+
(verb_type|Participant.Private.ID)+
                (1|time_normalized),family="binomial",data=accent_mem_data)
summary(glmm2_1)
```

[4] See `AOW_r_work_flow.rmd` for all model comparisons

3.2 GAMMs

Like GLMM data, GAMM data must be first coded and prepared (L:546-559). Here, we turn variables into factors and level them at the same time (e.g., L:550-553). However, it is important to note that GAMMs do better with coded variables, L:550-552. We create `event` as a combination between conditions (L:554-555). Then we only `select()` columns necessary for the analysis (L:557-559). Lastly, we split off the accent data for the accent GAMM (L:560).

```
## ----GAMM: Leveling the Data---
gamm_data <- gamm_data %>%
mutate(
Condition = as.factor(Condition), subject_img_coded =
as.numeric(factor(subject_img_file)) - 1, talker_coded =
as.numeric(factor(talker)) - 1, verb_type_coded =
as.numeric(factor(verb_type)) - 1,
Participant.Private.ID = as.factor(Participant.Private.ID),
Event = as.factor(paste(
Participant.Private.ID,Trial.Number,sep= ".")),
experience_chinese = experience_chinese_accent)%>%
    select(Event,Participant.Private.ID,Trial.Number,verb_type_coded,
talker_coded,subject_img_coded,Condition,target,time_elapsed,
log_SUBTLWF_Obj,experience_chinese,Event)
gamm_data_accented<-gamm_data%>%filter(talker_coded == 1)
```

GAMM Models were built using the `mgcv` package (Wood, 2017). Model comparisons suggest that random intercept of `Event` significantly improved maximal model. Like the GLMM model, the GAMM models treat looks to the target (L:603) as the independent variable with dependent variables including three fixed effects: `talker_coded` (L:603), `verb_type_coded` (L:605) and their interaction (L:607). Random effects included `Event`

(L:612). Smooth terms were included for `time_elapsed` by levels of `talker_coded` (L:604), `verb_type_coded` (L:606), and `Condition` (L:608). Smooth terms allow for a non-linear relationship between `time_elapsed` and the response variable `verb_type_coded`, with a different smooth function for each level of variable. An additional smooth term for `log_SUBTLWF_Obj` (L:609) was included. Smooth terms for `time_elapsed` were included for grouping levels: `Participant.Private.ID` and `subject_image_file` (L:610-611). The logit link function ("binomial") was specified in the model, equivalent to modeling logit-transformed response probability with identity link function.

```
602 ## ----GAMM: Main Model---
603 mod1 <- bam(target ~ talker_coded +
604 s(time_elapsed, by=talker_coded) +
605 verb_type_coded +
606 s(time_elapsed, by=verb_type_coded) +
607 talker_coded:verb_type_coded +
608 s(time_elapsed, by=Condition)+
609  s(log_SUBTLWF_Obj)+
610 s(time_elapsed, Participant.Private.ID, bs="fs", m=1)+
611 s(time_elapsed, subject_img_coded, bs="fs", m=1)+
612 s(Event, bs="re"), family="binomial", data=gamm_data, discrete=TRUE, method=
613 "fREML")
614 summary(mod1)
```

The accent GAMM had identical structure to the main GAMM with the expectation of having only 1 main effect, `experience_chinese` (L:642), and removing the smoothing term leveled by `talker_coded`. `gamm_data_accented` was the data frame (L:648). Model comparisons suggest that random intercept of `Event` significantly improves in the maximum model.

```
## ----GAMM: Accent Model---
mod2 <- bam(target ~ experience_chinese +
s(time_elapsed, by=verb_type_coded) +
s(log_SUBTLWF_Obj)+ s(time_elapsed, Participant.Private.ID, bs="fs",
m=1)+ s(time_elapsed, subject_img_coded, bs="fs", m=1)+ s(Event,
bs="re"),
family="binomial", data=gamm_data_accented, discrete=TRUE, method="
    fREML")
summary(mod2)
```

3.3 Results

We observed nearly identical time course of predictive processing (Figure 11) in which restricted sentences resulted in earlier looks to the target object than nonrestrictive sentences. Further, this effect is partially reduced in accented speech in a similar manner to Porretta et al. (2020). For `ggplot()` code and data wrangling for visualizations, see `AOW_r_work_flow.rmd`.

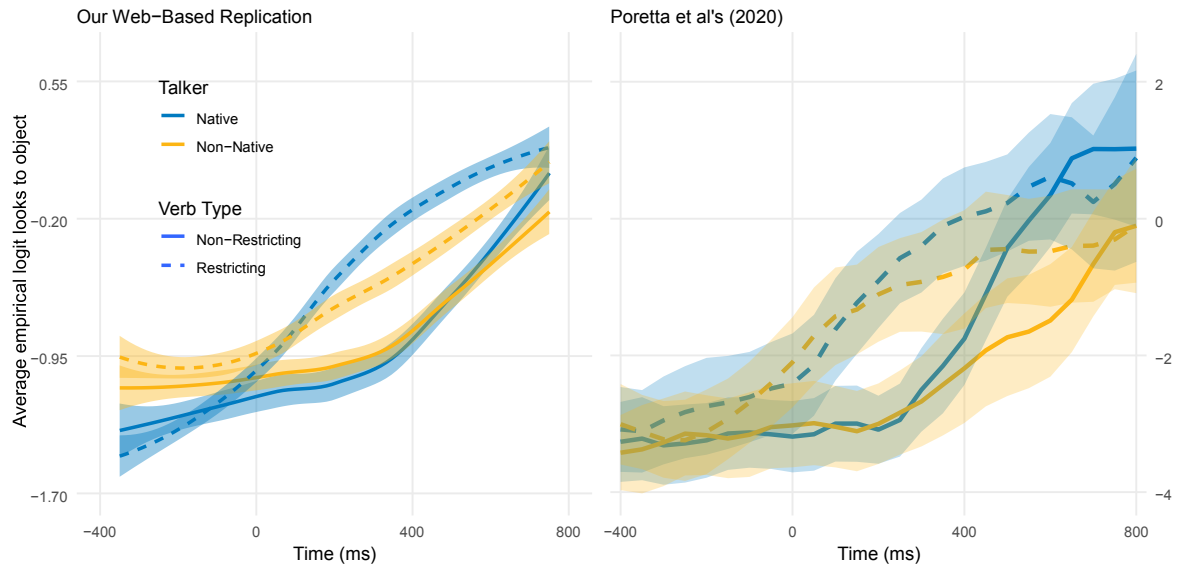

Figure 11. *Left: our data. Right: Porretta et al. (2020)*

3.3.1 GLMM Results

Results from the Main GLMM revealed a significant effect of `verb_type` ($\beta = 0.281$, SE = 0.067, z = 4.191, p < .001), indicating more looks to targets for restrictive `verb_type` over non-restrictive `verb_type` (Figure 12, left). Additionally, an interaction between speaker and verb type was found ($\beta = -0.136$, SE = 0.053, z = -2.554, p = 0.011), indicating less looks when listening to the accented speaker. Results from the Accent GLMM found null results at an alpha-level of .05 (Figure 12, right).

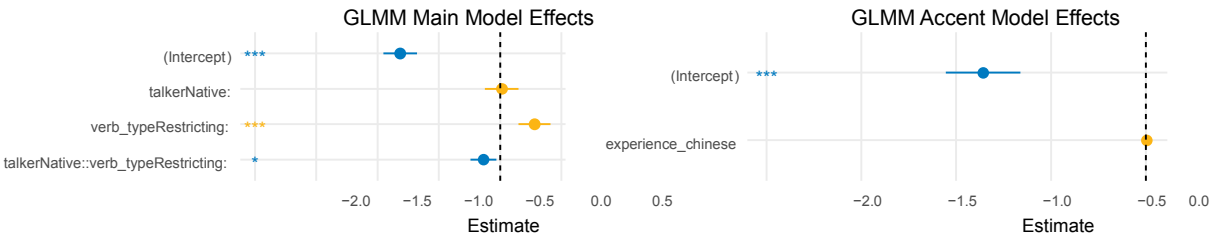

Figure 12. *Model output for parsimonious GLMM models*

3.3.2 GAMM Results

Like the GLMM modeling results, results from the Main GAMM revealed a significant effect of `verb_type` ($\beta = 0.398$, SE = 0.129, z = 3.078, p = .002), indicating more looks to targets for restrictive `verb_type` over non-restrictive `verb_type` (Figure 13, left). Results from the Accent GAMM found null results at an alpha-level of .05 (Figure 13, right).

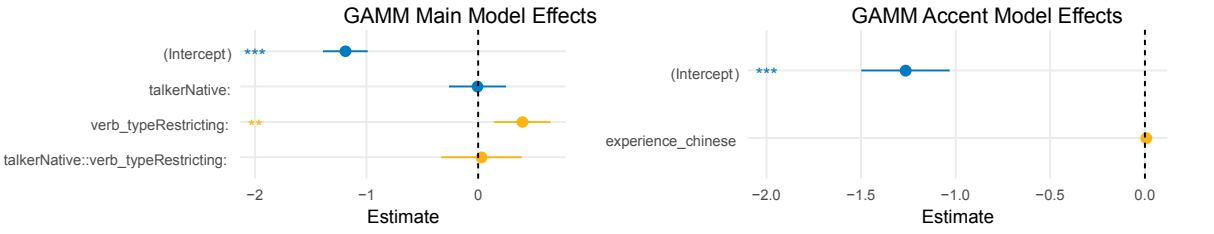

Figure 13. *Model output for parsimonious GAMM models*

4. Discussion

4.1 Web-based eye-tracking is a viable alternative to in-person eye-tracking

Like other recent web-based eye-tracking studies, our replication results indicate that web-based eye-tracking is an excellent method for not only replicating in-person eye-tracking studies (Prystauka et al., 2023; Vos et al., 2022), but also conducting novel studies. Our main models show that predictive sentence processing is modulated by restrictive and non-restrictive verb type in line with Porretta et al. (2020) and that accented speech impedes predictive processing but does not preclude it. While our accent models did not find evidence of accent-experience modulating predictive processing, our wider (non-university recruited) sample of participants also had far less experience with Chinese accents than Porretta et al. (2020). As noted earlier, our participants reported very little Chinese accent experience (range = 0-3.43, $M = 0.99$) compared to the students reported in Porretta et al. (2020) (range = 0–3.43, $M = 1.78$). There are at least two reasons for this difference. The first is that the students tested in Porretta et al. (2020) were exposed to Chinese-accented speech more as a result of being on a university campus with international students, while our crowdsourced Prolific participants had far less exposure to Chinese-accented speech in their daily lives. Another possibility is measurement error. 13 of our 49 participants reported 0 Chinese experience. One of the possible contributing factors to this may be the design of the sliding scale for reporting Chinese accent experience. The sliding scale was set to start at 0 (Gorilla pre-set setting, which can be controlled in configuration settings). It could be that some participants simply selected next to move on quickly. Future studies should clearly state the exact type of method used for capturing such data and make materials fully available to avoid this confusion for metrics that are essential for analyses. Our results are, therefore, inconclusive with respect to the accent models.

4.2 Best practices for web-based eye-tracking research

Alone, eye-fixations are meaningless. Exacting meaning for x- and y-coordinates is achieved through time, visual stimuli, and audio stimuli. These *core four* constructs correspond directly with the variables of our experiment, research questions, and data analyses. However, managing these constructs is complex. Data wrangling through lines of code knits these constructs together, gradually constructing bridges of understanding. In what follows, we summarize best practices that are essential for reproducible web-based eye-tracking studies.

Set clear exclusion criteria for participants prior to data collection. Removal of participants given language background information or demographics should be made prior to data collection, and should involve a simple filtering step at the beginning of data wrangling.

Include behavioral/attention task checks. The decisions and standards of participant and item removal should always be done before data analysis begins. We recommend removal by calculating distribution-based removal standards with median absolute deviation (Leys et al., 2013) or standard deviation with a distribution value set prior to beginning wrangling. Crucially, report what criterion you used for removal (e.g., 3 SD). We removed three participants due to low accuracy.

Ensure participant background information is accurate. As noted, we removed one participant for reporting a different age outside our preset filter and two for reporting non-monolingual status, again not in line with our preset filter. It is our experience that some Prolific users may have registered their account with inaccurate information in order to qualify for more studies (see Rodd, in press for discussion).

Include and report eye-calibration. Prior to obtaining our 60 participants, 23 potential participants failed our five-point eye-calibration. In other words, roughly one out of every five possible participants was unable to participate. We recommend using Gorilla in-person, if possible, to reduce potential participant loss.

Require a minimum frame rate greater than 5 Hz. In our study, below 5Hz seems to be 'unusable.' Whereas the research question and effect of interest will dictate the needed frame rate—consider a sentence processing study like ours which captured the native-talker predictive effects within 6-10 Hz, versus a word recognition study involving subtle voice-onset time differences which may require 20 Hz to detect differences—we echo Vos et al. (2022) and recommend researchers start by removing participants with 5 Hz median frame rate or less from their data as we did with our five participants. Removal should be reported, as well as the ranges of frame rates. In cases of more extensive removal, analyses should be run with both the removed participants and the full data to justify removing more data.

Additionally, in an exploratory attempt, we observed that device OS and age of the browser potentially explains variability between participants (see Rodd (in press) for discussion). Cut offs for types of browsers could be useful in collecting higher quality data and reduce the need to remove large amounts of participants found in other web-based eye-tracking studies (Prystauka et al., 2023).

Report all time adjustments. Report any time adjustments including the 200 ms needed to program a saccade (Matin et al., 1993) and any adjustment given a carrier phrase.

Identify a quadrant classification method. Previous web-based eye-tracking studies have shown that removal to the boundary of visual stimuli still enables the researcher to capture results even with strict standards for removal of eye-fixations (28% in Vos et al. (2022)). That is, eye-fixations outside the target areas in Figure 8 are excluded regardless of how close they are to the area (i.e., classifying web-based eye-fixation the same way that lab-based eye tracking does). However, ranges of removal at this strict standard suggest removal of up to 93.61% of the data.

Our suggestion is twofold: firstly, embrace the noise. If eye-fixations are random or equally distributed from the center, then including them will not hinder analysis. We suggest

that future research maximize retained signal, rather than maximizing removed noise. Secondly, we suggest that future research report and explore standards for maximizing signal and minimizing noise retention of eye-fixations.

Use a meaningful bin size given the research question. The amount of data per bin has an inverse relationship with the amount of bins over a period of time. Along with reporting standards for binning, we recommend that the researcher find a balance between fewer bins with more data and more bins with less data. Vos et al. (2022) and the current study used 50ms time bins. However, larger bin sizes could be useful with audio stimuli with longer duration. The crucial decision comes down to understanding the area of interest. Excluding extreme scenarios where the bin size is approaching the size of the area of interest, our data suggests that varying bin size has little effect on outcomes.

5. Conclusion

Web-based eye-tracking is here to stay, and with that comes a demand for mastering data-wrangling skills. For the first time, researchers anywhere can design an experiment, implement their analysis, and share their results openly for the cost of participant payment. Additionally, the choices of how data is treated is now up to the researcher, which leads to a need for widespread adoption of standardized practices beginning with experiment design and all throughout data analyses. Web-based eye-tracking is a powerful and accessible tool. Its convenience, cost, and reliability make it an easy choice for any researcher while the data wrangling involved may be daunting. We hope that this barrier has now been lowered with our guide through the wilds of eye-tracking data wrangling in the ***Art of Wrangling***.

Data availability statement

All data and scripts are available through OSF. All data is within the data folder of the OSF stored repository. All scripts are linked through Github. The primary script for data wrangling and analysis is `AOW_r_work_flow.Rmd`:

https://osf.io/a3e5s/?view_only=bb6015f2526f4a02bdd22dcd7449e9dd

Acknowledgements

The authors thank XXXX, XXXX, XXXX, and other members of XXXX for their help.

Competing interests declaration

The authors declare none.

References

Allopenna, P. D., Magnuson, J. S., & Tanenhaus, M. K. (1998). Tracking the time course of spoken word recognition using eye movements: Evidence for continuous mapping models. *Journal of Memory and Language, 38*(4), 419–439.

Anwyl-Irvine, A. L., Massonnié, J., Flitton, A., Kirkham, N., & Evershed, J. K. (2019). Gorilla in our midst: An online behavioral experiment builder. *Behavior Research Methods, 52*(1), 388–407.

Apfelbaum, K. S., Klein-Packard, J., & McMurray, B. (2021). The pictures who shall not be named: Empirical support for benefits of preview in the visual world paradigm. *Journal of Memory and Language, 121*, 104279.

Barr, D. J. (2008). Analyzing 'visual world' eyetracking data using multilevel logistic regression. *Journal of Memory and Language, 59*(4), 457–474.

Barr, D. J., Levy, R., Scheepers, C., & Tily, H. J. (2013). Random effects structure for confirmatory hypothesis testing: Keep it maximal. *Journal of Memory and Language, 68*(3), 255–278.

Bates, D., Mächler, M., Bolker, B., & Walker, S. (2014). Fitting linear Mixed-Effects models using lme4.

Bolibaugh, C., Vanek, N., & Marsden, E. J. (2021). Towards a credibility revolution in bilingualism research: Open data and materials as stepping stones to more reproducible and replicable research. *Bilingualism: Language and Cognition, 24*(5), 801–806.

Chen, M. C., Anderson, J. R., & Sohn, M. H. (2001). What can a mouse cursor tell us more? *CHI '01 Extended Abstracts on Human Factors in Computing Systems*.

Cooper, R. M. (1974). The control of eye fixation by the meaning of spoken language. *Cognitive Psychology, 6*(1), 84–107.

Coretta, S., Casillas, J. V., Roessig, S., Franke, M., Ahn, B., Al-Hoorie, A. H., Al-Tamimi, J., Alotaibi, N. E., AlShakhori, M. K., Altmiller, R. M., & et al. (2023). Multidimensional signals and analytic flexibility: Estimating degrees of freedom in human-speech analyses. *Advances in Methods and Practices in Psychological Science, 6*(3). https://doi.org/10.1177/25152459231162567

Degen, J., Kursat, L., & Leigh, D. D. (2021). Seeing is believing: Testing an explicit linking assumption for visual world eye-tracking in psycholinguistics. *Proceedings of the Annual Meeting of the Cognitive Science Society, 43*. https://escholarship.org/uc/item/6182t9jb

Foster, E. D., & Deardorff, A. (2017). Open science framework (osf). *Journal of the Medical Library Association, 105*(2). https://doi.org/10.5195/jmla.2017.88

Ito, A., & Knoeferle, P. (2022). Analysing data from the psycholinguistic visual-world paradigm: Comparison of different analysis methods. *Behavior Research Methods.* https://link.springer.com/article/10.3758/s13428-022-01969-3

Leys, C., Ley, C., Klein, O., Bernard, P., & Licata, L. (2013). Detecting outliers: Do not use standard deviation around the mean, use absolute deviation around the median. *Journal of Experimental Social Psychology, 49*(4), 764–766.

Marsden, E., Morgan-Short, K., Thompson, S., & Abugaber, D. (2018). Replication in second language research: Narrative and systematic reviews and recommendations for the field. *Language Learning, 68*(2), 321–391.

Matin, E., Shao, K. C., & Boff, K. R. (1993). Saccadic overhead: Information-processing time with and without saccades. *Perception & Psychophysics, 53*(4), 372–380.

McMurray, B. (2023). I'm not sure that curve means what you think it means: Toward a [more] realistic understanding of the role of eye-movement generation in the visual world paradigm. *Psychonomic Bulletin & Review, 30*(1), 102–146.

Milne, A. E., Bianco, R., Poole, K. C., Zhao, S., Oxenham, A. J., Billig, A. J., & Chait, M. (2021). An online headphone screening test based on dichotic pitch. *Behavior Research Methods, 53*(4), 1551–1562.

Mirman, D., Dixon, J. A., & Magnuson, J. S. (2008). Statistical and computational models of the visual world paradigm: Growth curves and individual differences. *Journal of Memory and Language, 59*(4), 475–494.

Mizumoto, A., & Plonsky, L. (2015). R as a lingua franca: Advantages of using r for quantitative research in applied linguistics. *Applied Linguistics, 37*(2), 284–291.

Palan, S., & Schitter, C. (2018). Prolific.ac—a subject pool for online experiments. *Journal of Behavioral and Experimental Finance, 17*, 22–27.

Papoutsaki, A., Sangkloy, P., Laskey, J., Daskalova, N., Huang, J., & Hays, J. (2016). Webgazer: Scalable webcam eye tracking using user interactions. *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, 3839–3845.

Porretta, V., Buchanan, L., & Järvikivi, J. (2020). When processing costs impact predictive processing: The case of foreign-accented speech and accent experience. *Attention, Perception, & Psychophysics, 82*(4), 1558–1565.

Prystauka, Y., Altmann, G. T., & Rothman, J. (2023). Online eye tracking and real-time sentence processing: On opportunities and efficacy for capturing psycholinguistic effects of different magnitudes and diversity. *Behavior Research Methods*.

R Core Team. (2022). R: A language and environment for statistical computing. R Foundation for Statistical Computing. Vienna, Austria.

Rodd, J. M. (in press). Moving experimental psychology online: How to obtain high quality data when we can't see our participants. *Journal of Memory and Language, 134*(104472), 104472.

Semmelmann, K., & Weigelt, S. (2017). Online webcam-based eye tracking in cognitive science: A first look. *Behavior Research Methods, 50*(2), 451–465.

Tanenhaus, M. K., Spivey-Knowlton, M. J., Eberhard, K. M., & Sedivy, J. C. (1995). Integration of visual and linguistic information in spoken language comprehension. *Science, 268*(5217), 1632–1634.

Vos, M., Minor, S., & Ramchand, G. C. (2022). Comparing infrared and webcam eye tracking in the visual world paradigm. *Glossa Psycholinguistics, 1*(1). https://doi.org/10.5070/g6011131

Wickham, H. (2014). Tidy data. *Journal of Statistical Software, 59*(10). https://doi.org/10.18637/jss.v059.i10

Wickham, H., & Grolemund, G. (2017, January). *R for data science: Import, tidy, transform, visualize, and model data* (1st ed.). O'Reilly Media. http://r4ds.had.co.nz/

Wood, S. (2017). *Generalized additive models: An introduction with R* (2nd ed.). Chapman; Hall/CRC.