

¹ Language Without Borders: A Step-by-Step Guide to Analyzing
² Webcam Eye-Tracking Data for L2 Research

³ Jason Geller¹, Yanina Prystauka², Sarah E. Colby³, and Julia R. Drouin⁴

⁴ ¹Department of Psychology and Neuroscience, Boston College

⁵ ²Department of Linguistic, Literary and Aesthetic Studies, University of Bergen

⁶ ³Department of Linguistics, University of Ottawa

⁷ ⁴Division of Speech and Hearing Sciences, University of North Carolina at Chapel Hill

⁸ Abstract

Eye-tracking has become a valuable tool for studying cognitive processes in second language acquisition and bilingualism (Godfroid et al., 2024). While research-grade infrared eye-trackers are commonly used, several factors limit their widespread adoption. Recently, consumer-based webcam eye-tracking has emerged as an attractive alternative, requiring only a personal webcam and internet access. However, webcam-based eye-tracking introduces unique design and preprocessing challenges that must be addressed to ensure valid results. To help researchers navigate these challenges, we developed a comprehensive tutorial focused on visual world webcam eye-tracking for second language research.^{**} This guide covers key preprocessing steps—from reading in raw data to visualization and analysis—highlighting the open-source R package webgazeR, freely available at: <https://github.com/jgeller112/webgazer>. ^{**}To demonstrate these steps, we analyze data collected via the Gorilla platform (Anwyl-Irvine et al., 2020) using a single-word Spanish visual world paradigm (VWP), showcasing evidence of competition both within and between Spanish and English. This tutorial aims to empower researchers by providing a step-by-step guide to successfully conduct webcam-based visual world eye-tracking studies. To follow along, please download the complete manuscript, code, and data from: https://github.com/jgeller112/L2_VWP_Webcam.

Keywords: VWP, Tutorial, Webcam eye-tracking, R, Gorilla, Spoken word recognition, L2 processing

¹ Eye-tracking technology, which has a history spanning over a century, has seen remarkable advancements. In the early days, eye-tracking often required the use of contact lenses fitted with search coils—² sometimes necessitating anesthesia—or the attachment of suction cups to the sclera of the eyes (Płużyczka,³ 2018). These methods were not only cumbersome for researchers, but also uncomfortable and invasive for⁴

5 participants. Over time, such approaches have been replaced by non-invasive, lightweight, and user-friendly
6 systems. Today, modern eye-tracking technology is widely accessible in laboratories worldwide, enabling
7 researchers to tackle critical questions about cognitive processes. This evolution has had a profound impact
8 on fields such as psycholinguistics and bilingualism, opening up new possibilities for understanding how
9 language is processed in real time (Godfroid et al., 2024).

10 **In the last decade, there has been a gradual shift towards conducting more behavioral exper-**
11 **iments online (Anderson et al., 2019; Rodd, 2024). This “onlineification” of behavioral research has**
12 **driven the development of remote eye-tracking methods that do not rely on traditional laboratory set-**
13 **tings. Allowing participants to use their own equipment from anywhere in the world opens the door**
14 **to recruiting more diverse and historically underrepresented populations [@gosling2010]. Behavioral**
15 **research has long struggled with a lack of diverse and representative samples, relying heavily on par-**
16 **ticipants who are predominantly Western, Educated, Industrialized, Rich, and Democratic (WEIRD)**
17 **(Henrich et al., 2010). Additionally, we propose adding able-bodied to this acronym (WEIRD-A) (Pe-**
18 **terson, 2021), to highlight the exclusion of individuals with disabilities who may face barriers to ac-**
19 **cessing research facilities. In language research, this issue is especially pronounced, as studies often**
20 **focus on “modal” listeners and speakers—typically young, monolingual, and neurotypical (Blasi et al.,**
21 **2022; Bylund et al., 2024; McMurray et al., 2010).**

22 **In this paper, we contribute to the growing body of research suggesting that webcam-based eye-**
23 **tracking, which is administered remotely and requires access to only a computer webcam, can increase**
24 **inclusivity and representation of the participant samples we include in research studies. Namely, by**
25 **minimizing the requirements for participants to travel to a lab, use specialized equipment, or meet**
26 **strict scheduling demands, webcam-based approaches can facilitate participation from individuals in**
27 **rural or geographically isolated areas and people with disabilities that make getting to a lab difficult.**
28 **This approach also promotes inclusion of broader sociodemographic groups that have been historically**
29 **underrepresented in cognitive and developmental research. We illustrate this by replicating a visual**
30 **world eye-tracking study with bilingual English-Spanish speaking participants (Sarrett et al., 2022)**
31 **using online methods (i.e., recruitment via Prolific.co and webcam-based eye-tracking). To facilitate**
32 **broader adoption of this approach, we also introduce our R package, webgazeR, and present a step-**
33 **by-step tutorial for analyzing webcam-based VWP data.**

Jason Geller  <https://orcid.org/0000-0002-7459-4505>

Yanina Prystauka  <https://orcid.org/0000-0001-8258-2339>

Sarah E. Colby  <https://orcid.org/0000-0002-2956-3072>

Julia R. Drouin  <https://orcid.org/0000-0003-0798-3268>

This study was not preregistered. The data and code for this manuscript can be found at https://github.com/jgeller112/L2_VWP_Webcam. The authors have no conflicts of interest to disclose. This work was supported by research start-up funds to JRD. Author roles were classified using the Contributor Role Taxonomy (CRediT; <https://credit.niso.org/>) as follows: Jason Geller: conceptualization, writing, data curation, editing, software, formal analysis; Yanina Prystauka: methodology, editing, formal analysis; Sarah E. Colby: methodology, editing; Julia R. Drouin: methodology, conceptualization, editing, funding acquisition

Correspondence concerning this article should be addressed to Jason Geller, Department of Psychology and Neuroscience, Boston College, Mcguinn Hall 405, Chestnut Hill, MA 02467-9991, USA, drjasongeller@gmail.com: jason.geller@bc.edu

34 This paper is divided into three parts. First, we introduce automated webcam-based eye-
35 tracking. Second, we review the viability of conducting VWP studies using online eye-tracking meth-
36 ods. Third, we present a detailed tutorial for analyzing webcam-based VWP data with the `webgazeR`
37 package, using our replication experiment to highlight the steps needed for preprocessing.

38 **Webcam eye-tracking with WebGazer.js**

39 There are two popular methods for online eye-tracking. One method, manual eye-tracking
40 (Trueswell, 2008), involves using video recordings of participants, which can be collected through on-
41 line teleconferencing platforms such as Zoom (www.zoom.com). Here eye gaze (direction) is manually
42 analyzed post-hoc frame by frame from these recordings. However, this method raises ethical and
43 privacy concerns, as not all participants may be comfortable having their videos recorded and stored
44 for analysis.

45 Another method, which is the focus of this paper, is automated eye-tracking or webcam eye-
46 tracking. Webcam eye-tracking generally has three requirements for the participant: (1) a personal
47 computer, tablet, or smartphone (see Chen-Sankey et al., 2023), (2) an internet connection, and (3) a
48 built-in or external camera. Gaze data is collected directly through a web browser without requiring
49 any additional software installation, making it highly accessible.

50 A popular tool for enabling webcam-based eye-tracking is WebGazer.js (Papoutsaki et al.,
51 2016)¹, an open-source, freely available, and actively maintained JavaScript library. WebGazer.js has al-
52 ready been integrated into several popular experimental platforms, including Gorilla, jsPsych, PsychoPy,
53 Labvanced, and PCIbex (Anwyl-Irvine et al., 2020; Kaduk et al., 2024; Leeuw, 2015; Peirce et al., 2019;
54 Zehr & Schwarz, 2018). Because WebGazer.js runs locally on the participant's machine, it does not store we-
55 bcam video recordings, helping alleviate ethical and privacy concerns associated with online eye-tracking.**

56 Under the hood, WebGazer.js uses machine learning to estimate gaze position in real time
57 by fitting a facial mesh to the participant and detecting the location of the eyes. At each sampling
58 point—determined by the participant's device and webcam capabilities—x and y gaze coordinates are
59 recorded. To improve accuracy, participants complete calibration and validation routines in which
60 they fixate on targets in specific locations on the screen (in some cases a manual approach is used
61 where users click on targets).

62 **Eye-tracking in the lab vs. online**

63 Several studies in psychology and psycholinguistics have evaluated the viability of WebGazer.js
64 for online research. Generally, lab-based effects can be successfully replicated in online environments
65 using WebGazer.js (Bogdan et al., 2024; Bramlett & Wiener, 2024, 2025; Özsoy et al., 2023; Prystauka
66 et al., 2024; Slim et al., 2024; Slim & Hartsuiker, 2023; Van der Cruyssen et al., 2024; Vos et al., 2022).

¹It is important to note that WebGazer.js is not the only method available. Other methods have been implemented by companies like Tobii (www.tobii.com) and Labvanced (Kaduk et al., 2024). However, because these methods are proprietary, they are less accessible and difficult to reproduce.

67 However, a critical finding across online replication studies is that effect sizes are often smaller and
68 more variable than those observed in laboratory settings (Bogdan et al., 2024; Slim et al., 2024; Slim
69 & Hartsuiker, 2023; Van der Cruyssen et al., 2024).

70 These attenuated effects likely stem from several technical limitations inherent to webcam-
71 based eye-tracking. Unlike research-grade trackers that use infrared illumination and pupil–corneal
72 reflection techniques—and can sample at rates up to 2,000 Hz with sub-degree spatial precision (0.1°
73 to 0.35°) (Carter & Luke, 2020; Hooge et al., 2024)—WebGazer.js typically operates at lower frame
74 rates, around 30 Hz (Bramlett & Wiener, 2024; Prystauka et al., 2024). Moreover, the performance
75 of the algorithm is highly dependent on ambient lighting conditions, making it more susceptible to
76 variability introduced by differences in head position, screen brightness, and background contrast.

77 There are also notable issues with the spatial and temporal accuracy of webcam-based eye-
78 tracking using WebGazer.js. Spatial precision is often lower, with average errors frequently exceeding
79 1° of visual angle (Papoutsaki et al., 2016). Temporal delays are also substantially larger, ranging from
80 200 ms to over 1000 ms (Semmelmann & Weigelt, 2018; Slim et al., 2024; Slim & Hartsuiker, 2023).
81 Additionally, recent work by Bogdan et al. (2024) has documented a systematic bias in gaze estimates
82 favoring centrally located stimuli.

83 Bringing the visual world paradigm (VWP) online

84 Despite these technical challenges, webcam-based eye-tracking has proven particularly well-
85 suited for adapting the visual world paradigm (VWP) (Tanenhaus et al., 1995; cf. Cooper, 1974) to
86 online environments.

87 In the field of language research, few methods have had as enduring an impact as the VWP.
88 Over the past 25 years, the VWP has enabled researchers to address a broad range of topics, including
89 sentence processing (Altmann & Kamide, 1999; Huettig et al., 2011; Kamide et al., 2003), spoken word
90 recognition (Allopenna et al., 1998; Dahan et al., 2001; Huettig & McQueen, 2007; McMurray et al.,
91 2002), bilingual language processing (Hopp, 2013; Ito et al., 2018; Rossi et al., 2019), the effects of
92 brain damage on language (Mirman & Graziano, 2012; Yee et al., 2008), and the impact of hearing
93 loss on lexical access [McMurray et al., 2017].

94 What makes the widespread use of the VWP particularly remarkable is the simplicity of the
95 task. In a typical VWP experiment, participants view a display of several objects, each represented
96 by a picture, while their eye movements are recorded in real time as they listen to a spoken word or
97 phrase. Although variations of the task exist—and implementations may differ depending on specific
98 research goals or design choices—the core finding remains consistent: listeners reliably shift their
99 gaze to the image corresponding to the spoken word, often before the word is fully articulated. This
100 robust effect provides compelling evidence for anticipatory or predictive processing during language
101 comprehension.

102 While eye movements are often time-locked to linguistic input, the relationship between

103 eye movements and lexical processing is not one-to-one. Lexical activation interacts with non-
104 lexical factors such as selective attention, visual salience, task demands, working memory, and prior
105 expectations—all of which can shape where and when participants look (Bramlett & Wiener, 2025;
106 Eberhard et al., 1995; Huettig et al., 2011; Kamide et al., 2003). Nonetheless, the VWP remains a
107 powerful and flexible tool for studying online language processing, offering fine-grained insights into
108 how linguistic and cognitive processes unfold moment by moment.

109 Several attempts have been made to conduct these experiments online using webcam-based eye-
110 tracking. Most online VWP replications have focused on sentence-based language processing. These
111 studies have looked at effects of set size and determiners (Degen et al., 2021), verb semantic constraint
112 (Prystauka et al., 2024; Slim & Hartsuiker, 2023), grammatical aspect and event comprehension (Vos
113 et al., 2022), and lexical interference (Prystauka et al., 2024).

114 More relevant to the current tutorial are findings from single-word VWP studies conducted
115 online. Recent research examined single-word speech perception online using a phonemic cohort task
116 (Bramlett & Wiener, 2025; Slim et al., 2024). In the cohort task, pictures were displayed randomly in
117 one of four quadrants, and participants were instructed to fixate on the target based on the auditory
118 cue. On each trial, one of the pictures was phonemically similar to the target in onset (e.g., *MILK*
119 – *MITTEN*). Slim et al. (2024) were able to observe significant fixations to the cohort compared to
120 the control condition, replicating lab-based single word VWP experiments with research grade eye-
121 trackers (e.g., Allopenna et al., 1998). However, time course differences were observed in the webcam-
122 based setting such that competition effects occurred later in processing compared to traditional, lab-
123 based eye-tracking.

124 Several factors have been proposed to explain the poor temporal performance in the VWP.
125 These include reduced spatial precision, computational demands introduced by the WebGazer.js al-
126 gorithm, slower internet connections, larger areas of interest (AOIs), and calibration quality (Boxtel
127 et al., 2024; Degen et al., 2021; Slim et al., 2024).

128 Importantly, temporal issues are not observed in every case. Work has begun to address many
129 of these challenges by leveraging updated versions of WebGazer.js and adopting different experimen-
130 tal platforms. For instance, Vos et al. (2022) reported a substantial reduction in temporal delays—
131 approximately 50 ms—when using a newer version of WebGazer.js embedded within the jsPsych
132 framework (Leeuw, 2015). Similarly, studies by Prystauka et al. (2024) and Bramlett and Wiener
133 (2024), which utilized the Gorilla Experiment Builder in combination with the improved WebGazer
134 algorithm, found timing and competition effects closely aligned with those observed in traditional lab-
135 based VWP studies.

136 While these temporal delays do present a challenge, and are at present an open issue, the gen-
137 eral findings that WebGazer.js can approximate looks to areas on the screen and replicate lab-based
138 findings underscore the potential of adapting the VWP to online environments using webcam-based
139 eye-tracking. Importantly, recent studies demonstrate that this approach can successfully capture
140 key psycholinguistic effects—such as lexical competition during single-word speech recognition—in a

141 manner comparable to traditional lab-based methods (Slim et al., 2024).

142 **Bilingual competition: A visual world webcam eye-tracking replication**

143 A goal of the present study was to conceptually replicate a study by Sarrett et al. (2022) wherein
144 they examined the competitive dynamics of second-language (L2) learners of Spanish, whose first
145 language (L1) is English, during spoken word recognition. Specifically, we investigated both within-
146 language and cross-language (L2/L1) competition using webcam-based eye-tracking.

147 It is well established that lexical competition plays a central role in language processing (Mag-
148 nuson et al., 2007). During spoken word recognition, as the auditory signal unfolds over time, multiple
149 lexical candidates—or competitors—can become partially activated. Successful recognition depends
150 on resolving this competition by inhibiting or suppressing mismatching candidates. For example, upon
151 hearing the initial segments of the word *wizard*, phonologically similar words such as *whistle* (cohort
152 competitor) may be briefly activated. As the word continues to unfold, additional competitors like *bliz-*
153 *zard* (a rhyme competitor) might also become active. For *wizard* to be accurately recognized, activation
154 of competitors such as *whistle* and *blizzard* must ultimately be suppressed.

155 One important area of exploration concerns lexical competition across languages. There is
156 growing evidence that lexical competition can occur cross-linguistically (see Ju & Luce, 2004; Spivey &
157 Marian, 1999). In a recent study, Sarrett et al. (2022) investigated whether cross-linguistic competition
158 arises in unbalanced L2 Spanish speakers—that is, individuals who acquired Spanish later in life. They
159 used carefully controlled stimuli to examine both within-language and cross-language competition in
160 adult L2 Spanish learners. Using a Spanish-language visual world paradigm, their study included two
161 critical conditions:

- 162 1. **Spanish-Spanish (within) condition:** A Spanish competitor was presented alongside the target
163 word. For example, if the target word spoken was *cielo* (sky), the Spanish competitor was *ciencia*
164 (science).
- 165 2. **Spanish-English (cross-linguistic) condition:** An English competitor was presented for the Spanish
166 target word. For example, if the target word spoken was *botas* (boots), the English competitor
167 was *border*.

168 Sarrett et al. (2022) also included a no competition condition where the Spanish-English pairs
169 were not cross-linguistic competitors (e.g., *frontera* as the target word and *botas* - boots as an unrelated
170 item in the pair). They observed competition effects in both of the critical conditions: within (e.g., *cielo*
171 - *ciencia*) and between (e.g., *botas* - *border*). Herein, we collected data to conceptually replicate their
172 pattern of findings using a webcam approach.

173 There are two key differences between our dataset and the original study by Sarrett et al. (2022)
174 worth noting. First, Sarrett et al. (2022) focused on adult unbalanced L2 Spanish speakers and posed
175 more fine-grained questions about the time course of competition and resolution and its relationship

176 with L2 language acquisition. Second, unlike Sarrett et al. (2022), who measured Spanish proficiency
177 objectively using LexTALE-esp (Izura et al., 2014)) and ran this study using participants from a Span-
178 ish college course, we relied on participant filtering on Prolific (www.prolific.co) to recruit L2 Spanish
179 speakers.

180 To conduct our online webcam replication, we used the experimental platform Gorilla (Anwyl-
181 Irvine et al., 2020), which integrates WebGazer.js for gaze tracking. We selected Gorilla because it
182 offers robust WebGazer.js integration and seems to address several temporal accuracy concerns iden-
183 tified in other platforms (Slim et al., 2024; Slim & Hartsuiker, 2023).

184 *Tutorial Overview*

185 This paper has two aims. First, we aim to provide evidence for lexical competition within and
186 across languages in L2 Spanish speakers, using webcam-based eye-tracking with WebGazer.js. While
187 there is growing interest in using VWP using webcam-based methods, lexical competition in single-
188 word L2 processing has not yet been investigated using the online version of the VWP, making this a
189 novel application. We hope that this work encourages researchers to explore more detailed questions
190 about L2 processing using webcam-based eye-tracking.

191 Second, we offer a tutorial that outlines key preprocessing steps for analyzing webcam-based
192 eye-tracking data. Building on recommendations proposed by (Bramlett & Wiener, 2024), our con-
193 tribution focuses on data preprocessing—transforming raw gaze data into a format suitable for visu-
194 alization and analysis. Here we introduce a new R package—webgazeR(Geller & Prystauka, 2024)—
195 designed to streamline and standardize preprocessing for webcam-based eye-tracking studies. We
196 believe that offering multiple, complementary resources enhances methodological transparency and
197 supports broader adoption of webcam-based eye-tracking methods. For in-depth guidance on experi-
198 mental design considerations, we refer readers to Bramlett and Wiener (2024).

199 Although Bramlett and Wiener (2024)'s tutorial provides a lot of useful code, the experiment-
200 specific nature of the code may pose challenges for newcomers. In contrast, the webgazeR package
201 offers a modular, generalizable approach. It includes functions for importing raw data, filtering and
202 visualizing sampling rates, extracting and assigning areas of interest (AOIs), downsampling and up-
203 sampling gaze data, interpolating and smoothing time series, and performing non-AOI-based analyses
204 such as intersubject correlation (ISC), a method increasingly used to explore gaze synchrony in natu-
205 ralistic paradigms (i.e., online learning) with webcam-based eye-tracking (Madsen et al., 2021).

206 We first begin by outlining the general methods used to conduct our webcam-based visual world
207 experiment. Second, we detail the data preprocessing steps needed to prepare the data for analysis
208 using webgazeR. Third, we demonstrate a statistical approach for analyzing the preprocessed data,
209 highlighting its application and implications.

210 To promote transparency and reproducibility, all analyses were conducted in R (R Core Team,
211 2024) using Quarto (Allaire et al., 2024), an open-source publishing system that enables dynamic and

212 reproducible documents. Figures, tables, and text are generated programmatically and embedded di-
213 rectly in the manuscript, ensuring seamless integration of results. To further enhance computational
214 reproducibility, we employed the `rix` package (Rodrigues & Baumann, 2025), which leverages the Nix
215 ecosystem (Dolstra & contributors, 2023). This approach captures not only the R package versions
216 but also system dependencies at runtime. Researchers can reproduce the exact computational envi-
217 ronment by installing the Nix package manager and using the provided `default.nix` file. Detailed
218 setup instructions are included in the README file of the accompanying GitHub repository. A video
219 tutorial is also provided.

220 **Method**

221 All tasks herein can be previewed here (<https://app.gorilla.sc/openmaterials/953693>). The
222 manuscript, data, and R code can be found on Github (https://github.com/jgeller112/webcam_gazeR_VWP).

223 **Participants**

224 Participants were recruited through Prolific (www.prolific.co, 2024), an online participant re-
225 cruitment platform. Our goal was to approximately double the sample size of Sarrett et al. (2022) to
226 enhance statistical power and ensure greater generalizability of the findings. However, due to practical
227 constraints and the challenges associated with online webcam eye-tracking (e.g., calibration failures)
228 and also the limited pool of bilingual Spanish speakers, we were unable to achieve the targeted usable
229 sample size. Therefore, we report the final sample based on all participants who met our predefined
230 inclusion criteria.

231 Inclusion criteria required participants to: (1) be between 18 and 36 years old, (2) be native
232 English speakers, (3) also be fluent in Spanish, and (4) reside in the United States. Criterion 1 was based
233 on findings from Colby and McMurray (2023), which suggest that age-related changes in spoken word
234 recognition begin to emerge in individuals in their 40s; thus, we limited our sample to participants
235 younger than 36. Criteria 2 and 3 ensured that we were recruiting native English speakers and those
236 fluent in Spanish to test L1 and L2 interactions. Criterion 4 matched the population of the original
237 study, which was conducted with university students in Iowa, and therefore we restricted recruitment
238 to U.S. residents.

239 After agreeing to participate, individuals were redirected to the Gorilla experiment platform
240 (www.gorilla.sc; (Anwyl-Irvine et al., 2020)). A flow diagram of participant progression through the
241 experiment is shown in Figure 1. In total, 187 participants assessed the experimental platform and con-
242 sented to be in the study. Of these, 121 passed the headphone screener checkpoint, and 111 proceeded
243 to the VWP task. Out of the 111 participants who entered the VWP, 91 completed the final surveys
244 at the end of the experiment. Among these, 32 participants successfully completed the VWP task with
245 at least 100 trials, while 79 participants did not provide adequate data for inclusion, primarily due to
246 failed calibration attempts. After applying additional exclusion criteria—namely, overall VWP task
247 accuracy below 80%, excessive missing eye-tracking data (>30%), and sampling rate < 5hz—the final

²⁴⁸ analytic sample consisted of 28 participants with usable eye-tracking data. Descriptive demographic
²⁴⁹ information for the full sample that made it to the final survey is provided in Table 1.

```
#| echo: false

knitr:::include_graphics(here::here("_manuscript", "Figures",
  "snakey_experiment.png"))
```

Figure 1

This sankey plot illustrates the flow of participants from initial consent ($N = 187$) through each stage of the study to the final analyzed sample ($N = 28$). The width of each stream is proportional to the number of participants. Detours indicate points of attrition, including failures in the headphone screener ($N = 66$) and calibration ($N = 76$). Only participants who passed all screening and calibration stages, and completed the Visual World Paradigm (VWP), were included in the final sample.

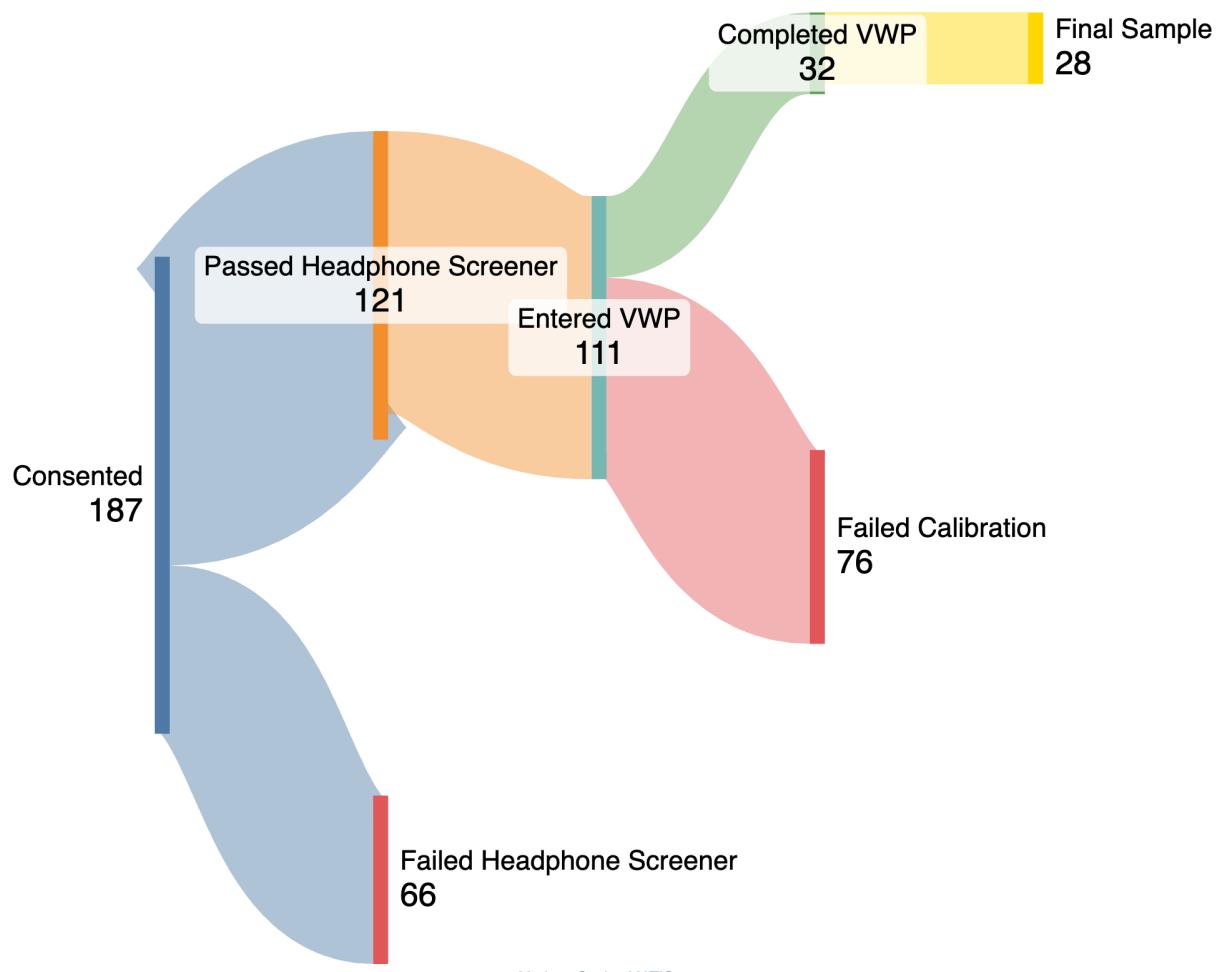


Table 1*Participant demographic variables*

Characteristic	N = 91¹
Age	(20.0, 35.0), 28.2(4.4)
Gender	
Female	42 / 91 (46%)
Male	49 / 91 (54%)
Spoken dialect	
Do not know	11 / 91 (12%)
Midwestern	19 / 91 (21%)
New England	11 / 91 (12%)
Other (please specify)	7 / 91 (7.7%)
Pacific northwest	7 / 91 (7.7%)
Pacific southwest	7 / 91 (7.7%)
Southern	21 / 91 (23%)
Southwestern	8 / 91 (8.8%)
Ethnicity	
Decline to state	1 / 91 (1.1%)
Hispanic or Latino	38 / 91 (42%)
Not Hispanic or Latino	52 / 91 (57%)
Race	
American Indian/Alaska Native	2 / 91 (2.2%)
Asian	13 / 91 (14%)
Black or African American	10 / 91 (11%)
Decline to state	7 / 91 (7.7%)
More than one race	4 / 91 (4.4%)
White	55 / 91 (60%)
Browser	
Chrome	77 / 91 (85%)
Edge	3 / 91 (3.3%)
Firefox	7 / 91 (7.7%)
Safari	4 / 91 (4.4%)
Years Speaking Spanish	(0, 35), 15(10)
% Experience Using Spanish Daily Life	25(23)

¹(Min, Max), Mean(SD); n / N (%); Mean(SD)

250 **Materials**

251 **VWP..**

252 **Items.** We adapted materials from Sarrett et al. (2022). In their cross-linguistic VWP, participants
253 were presented with four pictures and a spoken Spanish word and had to select the image that matched the
254 spoken word by clicking on it. The word stimuli for the experiment were chosen from textbooks used by
255 students in their first and second year college Spanish courses.

256 The item sets consisted of two types of phonologically-related word pairs: one pair of Spanish-
257 Spanish words and another of Spanish-English words. The Spanish-Spanish pairs were unrelated to the
258 Spanish-English pairs. All the word pairs were carefully controlled on a number of dimensions (see Sarrett
259 et al., 2022). There were three experimental conditions: (1) the Spanish-Spanish (within) condition, where
260 one of the Spanish words was the target and the other was the competitor; (2) the Spanish-English (cross-
261 linguistic) condition, where a Spanish word was the target and its English phonological cohort served as the
262 competitor; and (3) the No Competitor condition, where the Spanish word did not overlap with any other
263 word in the set. The Spanish-Spanish condition had twice as many trials as the other conditions due to the
264 interchangeable nature of the target and competitor words in that pair.

265 **Each item within a set appeared four times as the target word, resulting in a total of 240 trials**
266 **(15 sets × 4 items per set × 4 repetitions).** Each set included one Spanish-Spanish cohort pair and one
267 Spanish-English cohort pair. In the Spanish-Spanish condition, both words in the pair served as mu-
268 tual competitors—for example, *cielo* activated *ciencia*, and vice versa. This bidirectional relationship
269 yielded 120 trials for the Spanish-Spanish condition.

270 **In contrast, the Spanish-English pairs had an asymmetrical relationship: only one item in**
271 **each pair functioned as a competitor (e.g., *botas* could activate *frontera*, but *frontera* did not have a**
272 **corresponding competitor).** As a result, there were 60 trials each for the Spanish-English and No
273 Competitor conditions. Across all trials, target items were equally distributed among the four screen
274 quadrants to ensure balanced visual presentation

275 **Stimuli.** In Sarrett et al. (2022) all auditory stimuli were recorded by a female bilingual speaker
276 whose native language was Mexican Spanish and also spoke English. Stimuli were recorded in a sound-
277 attenuated room sampled at 44.1 kHz. Auditory tokens were edited to reduce noise and remove clicks. The
278 auditory tokens were then amplitude normalized to 70 dB SPL. For each target word, there were four separate
279 recordings so each instance was unique.

280 Visual stimuli were images from a commercial clipart database that were selected by a consensus
281 method involving a small group of students. All .wav files were converted to .mp3 for online data collection.
282 All stimuli can be found here: <https://osf.io/mgkd2/>.

283 **Headphone screener. Headphones were required for all participants. To ensure compliance,**
284 **we administered a six-trial headphone screening task adapted from Milne et al. (2021), which is avail-**
285 **able for implementation on the Gorilla platform.** On each trial, three tones of the same frequency and
286 duration were presented sequentially. One tone had a lower amplitude than the other two tones. Tones were

287 presented in stereo, but the tones in the left and right channels were 180 out of phase across stereo channels—
288 in free field, these sounds should cancel out or create distortion, whereas they will be perfectly clear over
289 headphones. The listener picked which of the three tones was the quietest. Performance is generally at the
290 ceiling when wearing headphones but poor when listening in the free field (due to phase cancellation).

291 **Participant background and experiment conditions questionnaire.** We had participants com-
292 plete a demographic questionnaire as part of the study. The questions covered basic demographic informa-
293 tion, including age, gender, spoken dialect, ethnicity, and race. **To gauge L2 experience, we asked partici-**
294 **pants when they started speaking Spanish, how many years of Spanish speaking experience they had,**
295 **and to provide, on a scale between 0-100, how often they use Spanish in their daily lives.**

296 **To further probe into data quality issues and get a better sense of why participants could not**
297 **make it through the experiment,** participants answered a series of questions at the end of the experiment
298 related to their personal health and environmental conditions during the experiment. These questions ad-
299 dressed any history of vision problems (e.g., corrected vision, eye disease, or drooping eyelids) and whether
300 they were currently taking medications that might impair judgment. Participants also indicated if they were
301 wearing eyeglasses, contacts, makeup, false eyelashes, or hats.

302 The questionnaire asked about natural light in the room, if they were using a built-in camera or an
303 external one (with an option to specify the brand), and their estimated distance from the camera. Participants
304 were asked to estimate how many times they looked at their phone or got up during the experiment and
305 whether their environment was distraction-free.

306 Additional questions assessed the clarity of calibration instructions, allowing participants to suggest
307 improvements, and asked if they were wearing a mask during the session. These questions aimed to gather
308 insights into personal and environmental factors that could impact data quality and participant comfort during
309 the experiment.

310 *Procedure*

311 **All tasks and questionnaires were developed using the Gorilla Experiment Builder's graphical**
312 **user interface (GUI) and integrated coding tools (Anwyl-Irvine et al., 2020). Each participant com-**
313 **pleted the study in a single session lasting approximately 45 minutes. Tasks were presented in a fixed**
314 **order: informed consent, headphone screening, the spoken word Visual World Paradigm (VWP) task,**
315 **and a set of questionnaire items. These are available to view here: <https://app.gorilla.sc/openmaterials/953693>.**

317 **Only personal computers were permitted for participation. Upon entering the study from Pro-**
318 **lific, participants were presented with a consent form. Once consent was given, participants completed**
319 **a headphone screening test. They had three attempts to pass this test. If unsuccessful by the third at-**
320 **tempt, participants were directed to an early exit screen, followed by the questionnaire. They had three**
321 **attempts to pass this test. If unsuccessful by the third attempt, participants were directed to an early**
322 **exit screen, followed by the questionnaire.**

323 **If the headphone screener was passed, participants were next introduced to the VWP task.** This

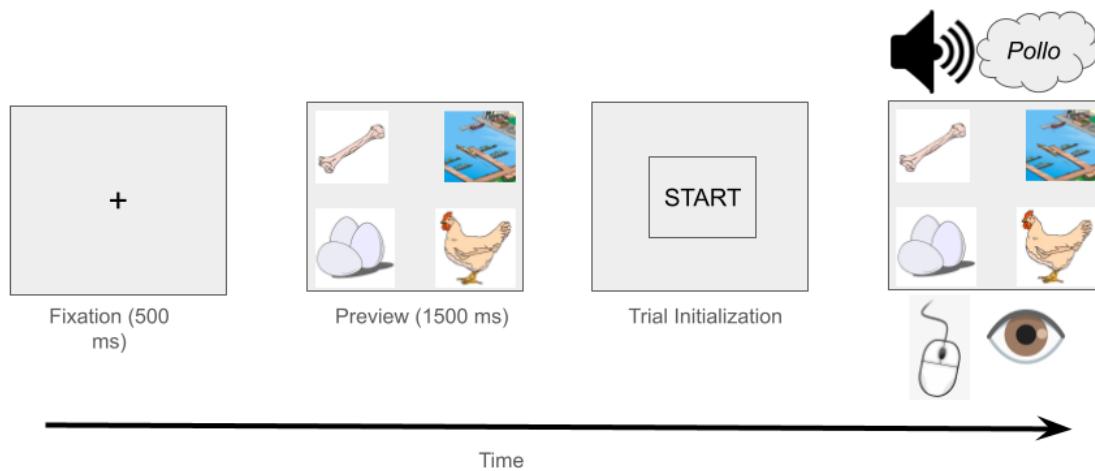
324 began with instructional videos providing specific guidance on the ideal experiment setup for eye-tracking
 325 and calibration procedures. You can view the videos here: <https://osf.io/mgkd2/>. Participants were then
 326 required to enter full-screen mode before calibration. A 9-point calibration procedure was used. Calibration
 327 occurred every 60 trials for a total of 3 calibrations. Participants had three attempts to successfully complete
 328 each calibration phase. If calibration was unsuccessful, participants were directed to an early exit screen,
 329 followed by the questionnaire.

330 In the main VWP task, each trial began with a 500 ms fixation cross at the center of the screen. This

331 was followed by a preview screen displaying four images, each positioned in a corner of the screen. After
 332 1500 ms, a start button appeared in the center. Participants clicked the button to confirm they were focused
 333 on the center before the audio played. Once clicked, the audio was played, and the images remained visible.
 334 Participants were instructed to click the image that best matched the spoken target word, while their eye
 335 movements were recorded. Eye movements were only recorded on that screen. Figure 2 displays the VWP
 336 trial sequence.

Figure 2

VWP trial schematic



337 After completing the main VWP task, participants proceeded to the final questionnaire, which in-

338 cluded questions about the eye-tracking task and basic demographic information. Participants were then
 339 thanked for their participation.

340 **Preprocessing data**

341 After the data is collected you can begin preprocessing your data. Below we highlight the steps
342 needed to preprocess your webcam eye-tracking data and get it ready for analysis. For some of this prepro-
343 cessing we will use the newly created `webgazeR` package (**v. 0.7.1**).

344 For preprocessing visual world webcam eye data, we follow seven general steps (see Figure 3):

345 1. Reading in data

346 2. Data exclusion

347 3. Combining trial- and eye-level data

348 4. Assigning areas of interest (AOIs)

349 5. Time binning

350 1. Downsampling

351 2. Upsampling (optional)

352 6. Aggregating (optional)

353 7. Visualization (optional)

354 For each of these steps, we will display R code chunks demonstrating how to perform each step with
355 helper functions (if applicable) from the `webgazeR` (Geller & Prystauka, 2024) package in R.

356 **Load packages**

357 **Package Installation and Setup.** Before proceeding, make sure to load the required packages
358 by running the code below. If you already have these packages installed and loaded, feel free to skip
359 this step. The code in this tutorial will not run correctly if any of the necessary packages are missing
360 or not properly loaded.

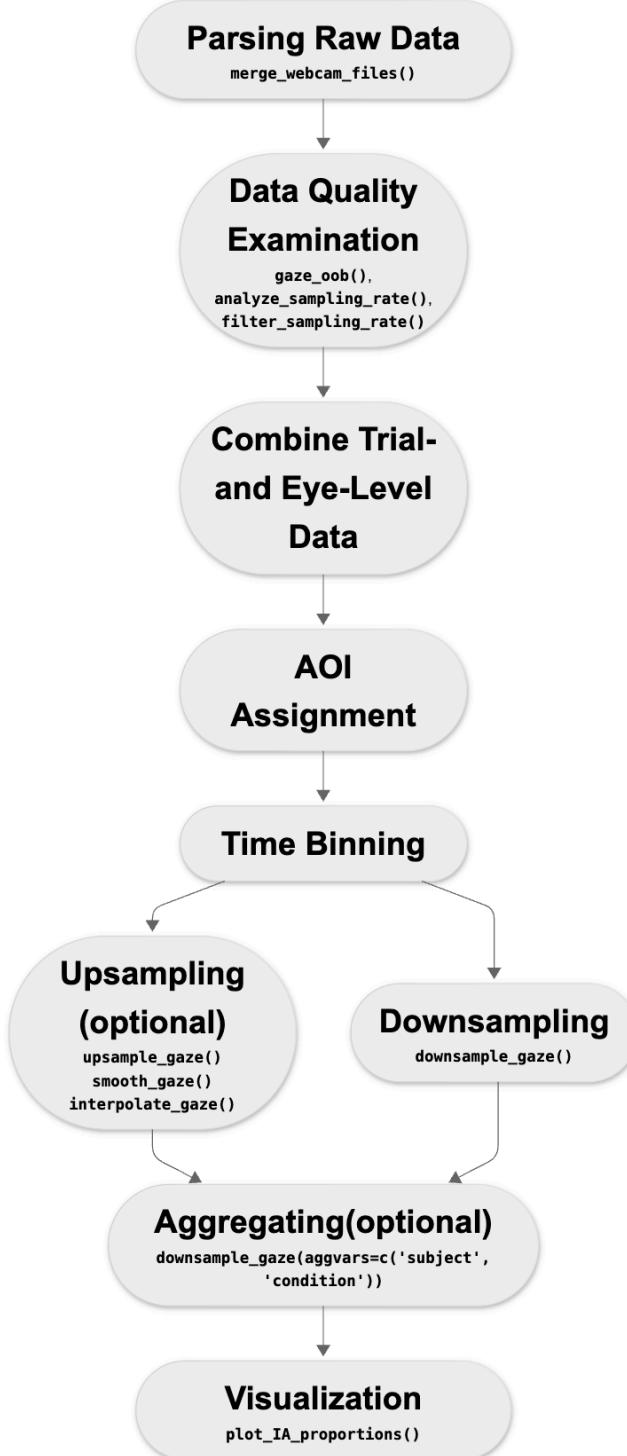
361 **webgazeR installation.** The `webgazeR` package is installed from the Github repository using the
362 `remotes` (Csárdi et al., 2024) package.

```
library(remotes) # install github repo  
  
remotes::install_github("jgeller112/webgazeR")
```

363 Once this is installed, `webgazeR` can be loaded along with additional useful packages. The following
364 code will load the required packages or install them if you do not have them on your system.

Figure 3

Preprocessing steps for webcam eye-tracking data using webgazeR functions



```
# List of required packages
required_packages <- c(
  "tidyverse",      # data wrangling
  "here",           # relative paths instead of absolute aids in reproducibility
  "tinytable",       # nice tables
  "janitor",        # functions for cleaning up your column names
  "webgazeR",        # has webcam functions
  "readxl",          # read in Excel files
  "ggokabeito",      # color-blind friendly palettes
  "flextable",        # Word tables
  "permuco",         # permutation analysis
  "foreach",          # permutation analysis
  "geomtextpath",      # for plotting labels on lines of ggplot figures
  "cowplot"          # combine ggplot figures
)
```

365 Once `webgazeR` and other helper packages have been installed and loaded the user is ready to start
 366 cleaning your data.

367 ***Reading in data***

368 **Behavioral, trial-level, data.** To process eye-tracking data you will need to make sure you have both
 369 the behavioral data and the eye-tracking data files. We have all the data needed in the repository by navigating
 370 to the L2 subfolder from the main project directory (`~/data/L2`). For the behavioral data, Gorilla produces a
 371 `.csv` file that includes trial-level information (here contained in the object `L2_data`). The files needed are
 372 called `data_exp_196386-v5_task-scf6.csv` and `data_exp_196386-v6_task-scf6.csv`. We have
 373 two files because we ran a modified version of the experiment.

374 The `.csv` files contain meta-data for each each trial, such as what picture were presented on each
 375 trial, which object was the target, reaction times, audio presentation times, what object was clicked on, etc.
 376 To load our data files into our R environment, we use the `here` (Müller, 2020) package to set a relative
 377 rather than an absolute path to our files. We read in the data files from the repository for both versions of
 378 the task and merge the files together. `L2_data` merges both `data_exp_196386-v5_task-scf6.csv` and
 379 `data_exp_196386-v6_task-scf6.csv` into one object.

```
# load in trial level data
# combine data from version 5 and 6 of the task
L2_1 <- read_csv(here("data", "L2", "data_exp_196386-v5_task-scf6.csv"))
L2_2 <- read_csv(here("data", "L2", "data_exp_196386-v6_task-scf6.csv"))
```

```
L2_data <- rbind(L2_1, L2_2) # bind the two objects together
```

380 **Eye-tracking data.** Gorilla currently saves each participant’s eye-tracking data on a per-trial basis.
 381 The raw subfolder in the project repository contains the eye-tracking files by participant for each trial
 382 individually (~/data/L2/raw). Contained in those files, we have information pertaining to each trial such as
 383 participant id, time since trial started, x and y coordinates of looks, convergence (the model’s confidence
 384 in finding a face (and accurately predicting eye movements), face confidence (represents the support vector
 385 machine (SVM) classifier score for the face model fit), and information pertaining to the the AOI screen
 386 coordinates (standardized and user-specific). The vwp_files_L2 object below contains a list of all the files
 387 contained in the folder. Because vwp_files_L2 contains trial data as well as calibration data, we remove
 388 the calibration trials and save the non-calibration to to vwp_paths_filtered_L2.

```
# Get the list of all files in the folder

# thank you to Reviewer 1 for suggesting this code
vwp_files_L2 <- list.files(here::here("data", "L2", "raw"), full.names = TRUE,
  ↵ pattern = "\\.\\.(csv|xlsx)$") %>%
# remove calibration trials
discard(~ grepl("calibration", .x))
```

389 **When data is generated from Gorilla, each trial in your experiment is saved as a separate
 390 file. To analyze the data, these individual files need to be combined into a single dataset. The
 391 merge_webcam_files() function from webgazeR** is designed for this purpose. It reads all trial-level
 392 files from a specified folder—regardless of file format (.csv, .tsv, or .xlsx)—and merges them into one cohe-
 393 sive tibble or data frame.****

394 Before using merge_webcam_files(), ensure your working directory is set to the location where
 395 the raw files are stored. The function automatically standardizes column names using clean_names(), binds
 396 the files together, and filters the data to retain only the relevant rows. Specifically, it keeps rows where the type
 397 column equals “prediction”, which are the rows that contain actual eye-tracking predictions. It also filters
 398 based on the screen_index argument: if you collected gaze data across multiple screens, you can specify one
 399 or several indices (e.g., screen_index = c(1, 4, 5)).

400 **In addition to merging and filtering, merge_webcam_files() requires the user to explicitly
 401 map critical columns—subject, trial, time, and x/y gaze coordinates. This makes the function highly
 402 flexible and robust across different experimental platforms. For instance, the function automatically
 403 renames the spreadsheet_row column to trial, and converts subject and trial into factors for compati-
 404 bility with downstream analyses.**

405 **Currently, the kind argument supports “gorilla” data, but future extensions will add sup-
 406 port for other platforms like Labvanced** (Kaduk et al., 2024), PsychoPy (Peirce et al., 2019), and**

407 **PCIbex** (Zehr & Schwarz, 2018). By explicitly allowing platform specification and flexible column mapping,
 408 `merge_webcam_files()` ensures a consistent and streamlined pipeline for preparing webcam eye-tracking data
 409 for analysis.**

410 As a general note, all steps should be followed in order due to the renaming of column names. If you
 411 encounter an error it might be because column names have not been changed.

```
setwd(here::here("data", "L2", "raw")) # set working directory to raw data folder

edat_L2 <- merge_webcam_files(vwp_files_L2, screen_index=4, col_map =
  ↵ list(subject = "participant_id", trial="spreadsheet_row",
  ↵ time="time_elapsed", x="x_pred_normalised", y="y_pred_normalised"),
  ↵ kind="gorilla")
```

412 To ensure high-quality data, we applied a set of behavioral and eye-tracking exclusion criteria prior
 413 to merging datasets. Participants were excluded if they met any of the following conditions: (1) failure
 414 to successfully calibrate throughout the experiment (fewer than 100 completed trials), (2) low behavioral
 415 accuracy (below 80%), (3) low sampling rate (below 5 Hz), or (4) a high proportion of gaze samples falling
 416 outside the display area (greater than 30%).

417 **Successful calibration is critical for reliable eye-tracking measurements, as poor calibration**
 418 **directly compromises the spatial accuracy of gaze data (Blascheck et al., 2017). Requiring a sufficient**
 419 **number of completed trials is crucial for ensuring adequate statistical power and stable individual-level**
 420 **parameter estimates, particularly in tasks with high trial-to-trial variability (Brynsaert & Stevens,**
 421 **2018). We choose 100 trials as this meant participants passed at least two calibration attempts dur-**
 422 **ing the study. Behavioral accuracy (>= 80%) was used as an additional screening measure because**
 423 **low task performance may indicate a lack of attention, misunderstanding of the task, or random re-**
 424 **sponding, all of which could undermine both the behavioral and eye-movement data quality (Bianco**
 425 **et al., 2021). Filtering based on sampling rate ensures that datasets with too few gaze samples (due to**
 426 **technical or environmental issues) are removed, as low sampling rates significantly degrade temporal**
 427 **precision and bias gaze metrics (semmlmann2018?). Finally, we excluded participants with excessive**
 428 **off-screen data (>30%) because this indicates poor gaze tracking, likely caused by head movement,**
 429 **poor lighting, or loss of face detection. At this time, there is no set guide on what constitutes accept-**
 430 **able data loss for webcam-based studies. We felt 30% was a reasonable cut-off. At the trial-level, we**
 431 **also removed incorrect trials and trials where sampling rate was < 5 Hz.**

432 **What we will do first is create a cleaned up version of our behavioral, trial-level data L2_data**
 433 **by creating an object named eye_behav_L2 that selects useful columns from that file and renames**
 434 **stimuli to make them more intuitive. Because most of this will be user-specific, no function is called**
 435 **here. Below we describe the preprocessing done on the behavioral data file. The below code pro-**
 436 **cesses and transforms the L2_data dataset into a cleaned and structured format for further analy-**
 437 **sis. First, the code renames several columns for easier access using janitor::clean_names()(Firke,**

438 **2023) function.** We then select only the columns we need and filter the dataset to include only rows
 439 where screen_name is “VWP” and zone_type is called “response_button_image”, representing the
 440 picture selected for that trial. Afterward, the function renames additional columns (tlpic to TL, trpic
 441 to TR, etc.). We also renamed participant_private_id to subject, spreadsheet_row to trial, and
 442 reaction_time to RT. This makes our columns consistent with the edat_L2 above for merging later
 443 on. Lastly, reaction time (RT) is converted to a numeric format for further numerical analysis.

444 It is important to note here that what the behavioral spreadsheet denotes as trial is not in fact the trial
 445 number used in the eye-tracking files. Thus it is imperative you use spreadsheet_row as trial number to
 446 merge the two files successfully.

```
eye_behav_L2 <- L2_data %>%
  janitor::clean_names() %>%
  # Select specific columns to keep in the dataset
  dplyr::select(participant_private_id, correct, tlpic, trpic, blpic, brpic,
  ~ condition,
    eng_targetword, targetword, typetl, typetr, typebl, typebr,
  ~ zone_name,
    zone_type, reaction_time, spreadsheet_row, response, screen_name)
  ~ %>%
  # Filter the rows where 'Zone.Type' equals "response_button_image"
  # participants clicked on preview screen so now need to filter based on screen.
  ~
  dplyr::filter(screen_name == "VWP", zone_type == "response_button_image") %>%
  # Rename columns for easier use and readability
  dplyr::rename(
    TL = tlpic,          # Rename 'tlpic' to 'TL'
    TR = trpic,          # Rename 'trpic' to 'TR'
    BL = blpic,          # Rename 'blpic' to 'BL'
    BR = brpic,          # Rename 'brpic' to 'BR'
    targ_loc = zone_name, # Rename 'zone_name' to 'targ_loc'
    subject = participant_private_id, # Rename 'participant_private_id' to
    ~ 'subject'
    trial = spreadsheet_row, # Rename 'spreadsheet_row' to 'trial'
    acc = correct,         # Rename 'correct' to 'acc' (accuracy)
    RT = reaction_time    # Rename 'reaction_time' to 'RT'
```

```
) %>%
# Convert the 'RT' (Reaction Time) column to numeric type
dplyr::mutate(RT = as.numeric(RT),
              subject = as.factor(subject),
              trial = as.factor(trial))
```

447 **Audio onset.** Because we are playing audio on each trial and running this experiment from the

448 browser, audio onset is never going to be consistent across participants. In Gorilla there is an option to
 449 collect advanced audio features (you must make sure you select this when designing the study) such as when
 450 the audio play was requested, played, and ended. We will want to incorporate this timing information into
 451 our analysis pipeline. Gorilla records the onset of the audio which varies by participant. We are extracting
 452 that in the `audio_rt_L2` object by filtering `zone_type` to `content_web_audio` and a response equal to
 453 “AUDIO PLAY EVENT FIRED”. This will tell us when the audio was triggered in the experiment. We are
 454 creating a column called (`RT_audio`) which we will use later on to correct for audio delays. Please note
 455 that on some trials the audio may not play. This is a function of the browser a participant is using and the
 456 experimenter has no control over this (see <https://support.gorilla.sc/support/troubleshooting-and-technical/technical-checklist#autoplayingsoundandvideo>). When running your experiment on a different platform,
 457 make sure you try and request this information, or at the very least acknowledge audio delay.

```
audio_rt_L2 <- L2_data %>%
janitor::clean_names() %>%
select(participant_private_id, zone_type, spreadsheet_row, reaction_time,
      ~ response) %>%
filter(zone_type == "content_web_audio", response == "AUDIO PLAY EVENT FIRED") %>%
distinct() %>%
dplyr::rename("subject" = "participant_private_id",
             "trial" = "spreadsheet_row",
             "RT_audio" = "reaction_time",
             "Fired" = "response") %>%
select(-zone_type) %>%
mutate(RT_audio = as.numeric(RT_audio))
```

459 We then merge this information with `eye_behav_L2`.

```
# merge the audio Rt data to the trial level object
trial_data_rt_L2 <- merge(eye_behav_L2, audio_rt_L2, by=c("subject", "trial"))
```

460 **Trial removal.** As stated above, participants who did not successfully calibrate 3 times or less were
 461 rejected from the experiment. Deciding to remove trials is ultimately up to the researcher. In our case, we
 462 removed participants with less than 100 trials. Let's take a look at how many participants meet this criterion
 463 by probing the `trial_data_rt_L2` object. In Table 2 we can see several participants failed some of the cal-
 464 ibration attempts and do not have an adequate number of trials. Again we make no strong recommendations
 465 here. If you decide to use a criterion such as this, we recommend pre-registering your choice.

```
# find out how many trials each participant had
edatntrials_L2 <- trial_data_rt_L2 %>%
  dplyr::group_by(subject) %>%
  dplyr::summarise(ntrials=length(unique(trial)))
```

466 Let's remove them participants with less than 100 trials from the analysis using the below code.

```
trial_data_rt_L2 <- trial_data_rt_L2 %>%
  filter(subject %in% edatntrials_bad_L2$subject)
```

467 **Low accuracy.** In our experiment, we want to make sure accuracy is high (> 80%). Again, we want
 468 participants that are fully attentive in the experiment. In the below code, we keep participants with accuracy
 469 equal to or above 80% and only include correct trials and assign it to `trial_data_acc_clean_L2`.

```
# Step 1: Calculate mean accuracy per subject and filter out subjects with mean
# accuracy < 0.8
subject_mean_acc_L2 <- trial_data_rt_L2 %>%
  group_by(subject) %>%
  dplyr::summarise(mean_acc = mean(acc, na.rm = TRUE)) %>%
  filter(mean_acc > 0.8)

# Step 2: Join the mean accuracy back to the main dataset and exclude trials with
# accuracy < 0.8
trial_data_acc_clean_L2 <- trial_data_rt_L2 %>%
  inner_join(subject_mean_acc_L2, by = "subject") %>%
  filter(acc==1) # only use accurate responses for fixation analysis
```

470 **RTs.** There is much debate on how to handle reaction time (RT) data (see Miller, 2023). Because
 471 of this, we leave it up to the reader and researcher to decide what to do with RTs. In this tutorial we leave
 472 RTs untouched.

Table 2

Participants with less than 100 trials

subject	ntrials
12102265	2
12110638	55
12110829	59
12110878	59
12110897	60
12111234	57
12111244	58
12111363	58
12111663	57
12111703	58
12111869	60
12111960	46
12112152	59
12212113	56
12213826	99
12213965	59

473 **Sampling rate.** While most commercial eye-trackers sample at a constant rate, data captured by
 474 webcams are widely inconsistent. Below is some code to calculate the sampling rate of each participant.
 475 Ideally, you should not have a sampling rate less than 5 Hz. It has been recommended you drop those
 476 values (Bramlett & Wiener, 2024) The below function `analyze_sample_rate()` calculates the sampling
 477 rate for each subject and each trial in our eye-tracking dataset (`edat_L2`). The `analyze_sample_rate()`
 478 function provides overall statistics, including the option to report mean or median (Bramlett & Wiener, 2024)
 479 sampling rate and standard deviation of sampling rates in your experiment. **Sampling rate calculations**
 480 **followed standard procedures (e.g., Bramlett & Wiener, 2024; Prystauka et al., 2024).** The function
 481 also generates a histogram of sampling rates by-subject. Looking at Figure 4, the sampling rate ranges from
 482 5 to 35 Hz with a median sampling rate of 21.56. This corresponds to previous webcam eye-tracking work
 483 (e.g., Bramlett & Wiener, 2024; Prystauka et al., 2024)

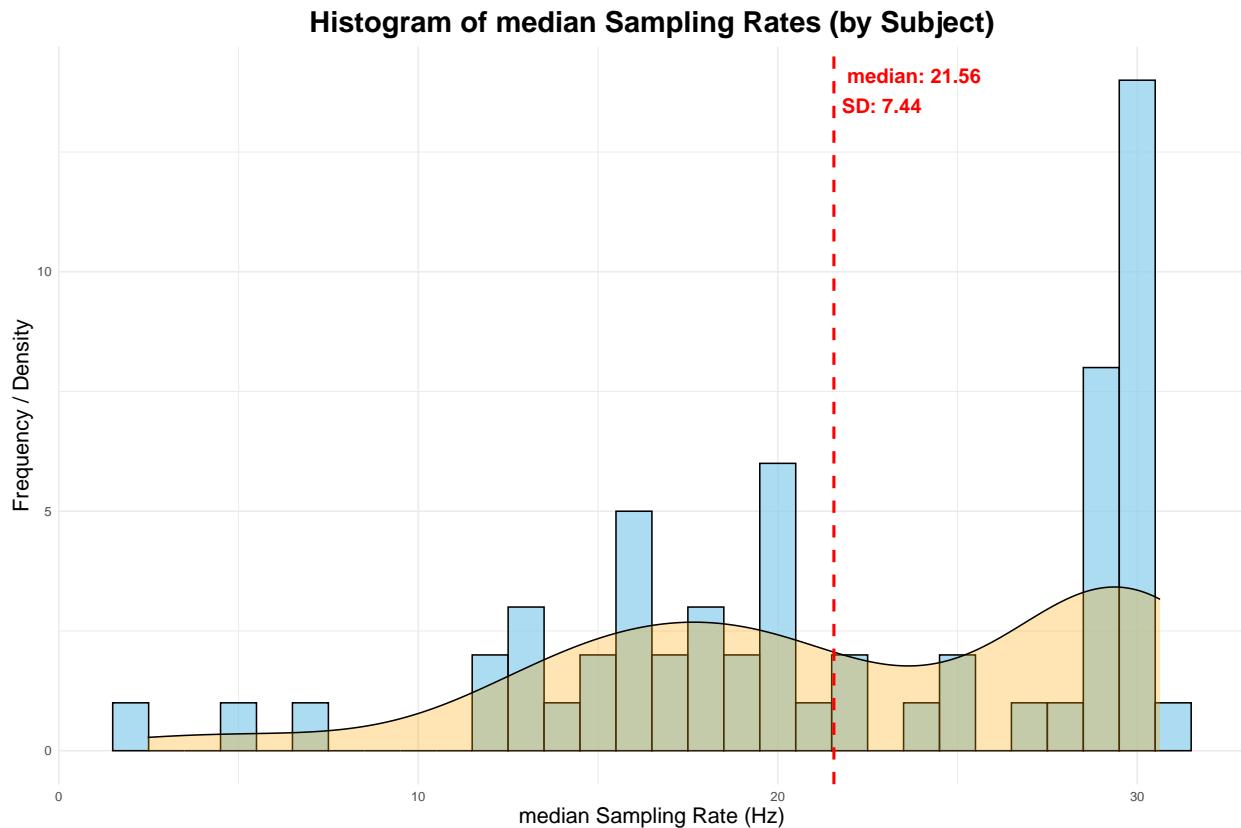
```
samp_rate_L2 <- analyze_sampling_rate(edat_L2, summary_stat="median")
```

484 Overall median Sampling Rate (Hz): 21.56

485 Overall SD of Sampling Rate (Hz): 7.44

Figure 4

Participant sampling-rate for L2 experiment. A histogram and overlayed density plot shows median sampling rate by participant. The overall median and SD is highlighted in red.



486 When using the above function, separate data frames are produced by-participants and by-trial. These
487 can be added to the behavioral data frame using the below code.

```
trial_data_L2 <- merge(trial_data_acc_clean_L2, samp_rate_L2, by=c("subject",  
↪ "trial"))
```

488 Now we can use this information to filter out data with poor sampling rates. Users can use the
489 `filter_sampling_rate()` function. The `filter_sampling_rate()` function is designed to process a
490 dataset containing participant-level and trial-level sampling rates. It allows the user to either filter out data
491 that falls below a certain sampling rate threshold or simply label it as “bad”. The function gives flexibility

492 by allowing the threshold to be applied at the participant-level, trial-level, or both. It also lets the user decide
 493 whether to remove the data or flag it as below the threshold without removing it. If `action = remove`, the
 494 function will output how many subjects and trials were removed using the threshold. We leave it up to the
 495 user to decide what to do with low sampling rates and make no specific recommendations. Here we use the
 496 `filter_sampling_rate()` function to remove trials and participants from the `trial_data_L2` object.

```
filter_edat_L2 <- filter_sampling_rate(trial_data_L2, threshold = 5,
                                         action = "remove",
                                         by = "both")
```

497 **Out-of-bounds (outside of screen).** It is essential to exclude gaze points that fall outside the
 498 screen, as these indicate unreliable estimates of gaze location. The `gaze_oob()` function quantifies
 499 how many data points fall outside these bounds, using the eye-tracking dataset (e.g., `edat_L2`) and the
 500 standardized screen dimensions—here set to (1, 1) because Gorilla recommends using standardized co-
 501 ordinates. If the `remove` argument is set to TRUE, the function applies an outer-edge filtering method
 502 to eliminate these out-of-bounds points (see Bramlett & Wiener, 2024). The outer-edge approach ap-
 503 pears to be a less biased approach based on demonstrations from Bramlett and Wiener (2024), where
 504 they showed minimal data loss compared to other approaches (e.g., inner-edge approach).

505 The function returns a summary table showing the total number and percentage of gaze points that
 506 fall outside the bounds, broken down by axis (X, Y), as well as the combined total (see Table 3). It also returns
 507 three additional tibbles: (1) missingness by-subject, (2) missingness by-trial, and (3) a cleaned dataset with
 508 all the data merged, and the problematic rows removed if specified. These outputs can be referenced in a final
 509 report or manuscript. **As shown in Figure 5, no fixation points fall outside the standardized coordinate**
 510 **range.**

```
oob_data_L2 <- gaze_oob(data=edat_L2, subject_col = "subject",
                           trial_col = "trial",
                           x_col = "x",
                           y_col = "y",
                           screen_size = c(1, 1), # standardized coordinates have
                           ↳ screen size 1,1
                           remove = TRUE)
```

511 We can use the `data_clean` tibble returned by the `gaze_oob()` function to filter out trials and sub-
 512 jects with more than 30% missing data. The value of 30% is just a suggestion and should not be used as a
 513 rule of thumb for all studies nor are we endorsing this value.

Figure 5

Looks to each quadrant of the screen

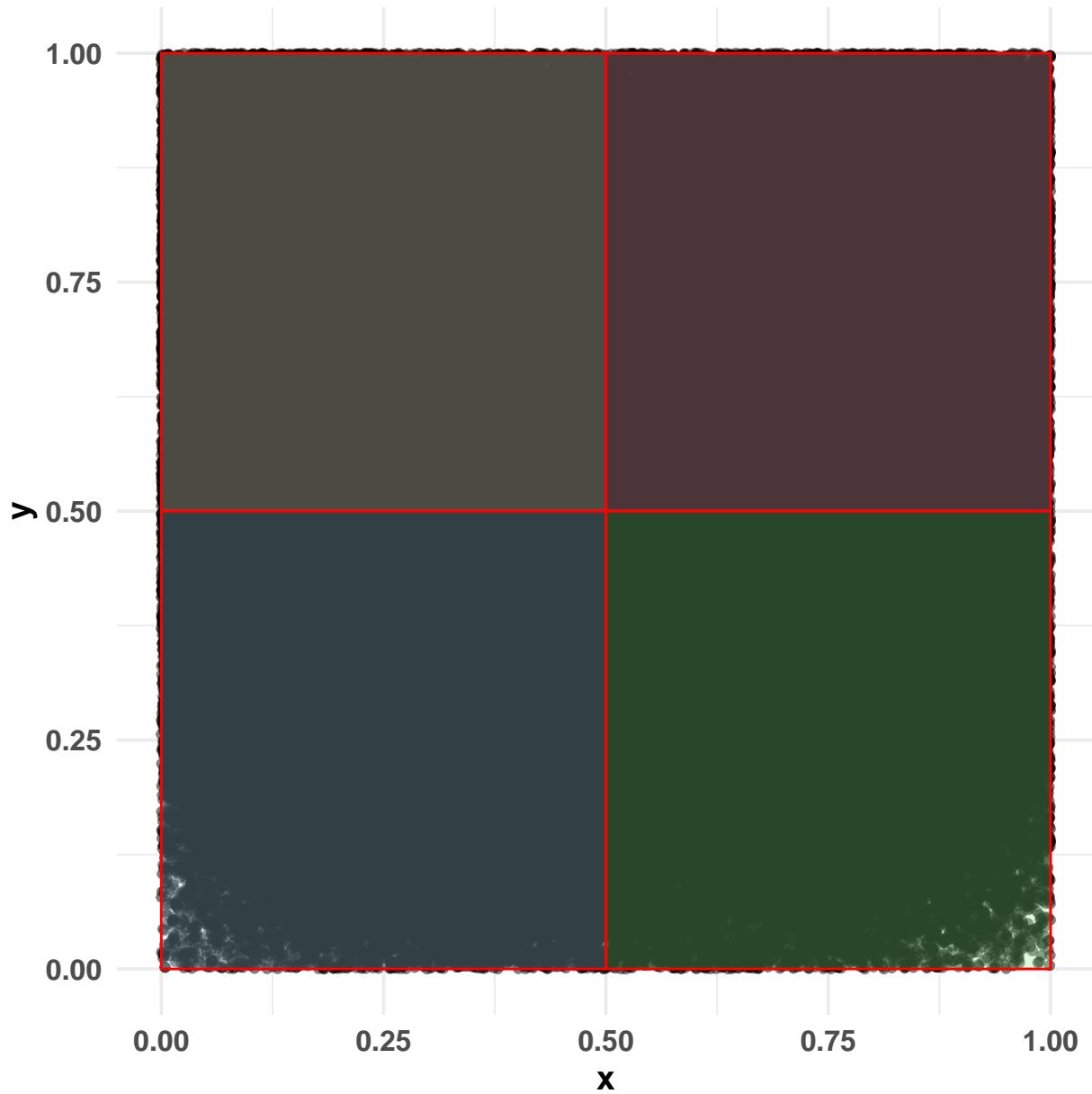


Table 3

Out of bounds gaze statistics by-participant

subject	totaltrials	totalpoints	outsidecount	subjectmissing%	xoutsidecount	youtsidecount	xoutside%	youtside%
12102265	60.00	6,192.00	1,132.00	18.28	202.00	947.00	3.26	15.29
12102286	240.00	11,765.00	354.00	3.01	267.00	181.00	2.27	1.54
12102530	240.00	9,011.00	385.00	4.27	244.00	147.00	2.71	1.63
12110559	240.00	11,887.00	415.00	3.49	194.00	221.00	1.63	1.86
12110579	178.00	5,798.00	1,061.00	18.30	696.00	435.00	12.00	7.50
12110585	240.00	13,974.00	776.00	5.55	83.00	694.00	0.59	4.97

```
# remove participants with more than 30% missing data and trials with more than
← 30% missing data
filter_oob <- oob_data_L2$data_clean %>%
  filter(trial_missing_percentage <= 30 | subject_missing_percentage <= 30)
```

514 ***Eye-tracking data***

515 **Convergence and confidence.** To ensure data quality, we removed rows with poor convergence
 516 and low face confidence from our eye-tracking dataset. The Gorilla eye-tracking output includes two
 517 key columns for this purpose: **convergence** and **face_conf** (similar variables may be available in
 518 other platforms as well). The **convergence** column contains values between 0 and 1, with lower values
 519 indicating better convergence—that is, greater model confidence in predicting gaze location and find-
 520 ing a face. Values below 0.5 typically reflect adequate convergence. The **face_conf** column reflects
 521 how confidently the algorithm detected a face in the frame, also ranging from 0 to 1. Here, values
 522 above 0.5 indicate a good model fit.

523 Accordingly, we filtered the **edat_L2**dataset to include only rows where **convergence < 0.5** and
 524 **face_conf > 0.5**, and saved the cleaned dataset as **edat_1_L2**.

```
edat_1_L2 <- filter_oob %>%
  dplyr::filter(convergence <= .5, face_conf >= .5) # remove poor convergnce and
← face confidence
```

525 **Combining eye and trial-level data.** Next, we will combine the eye-tracking data and behavioral
 526 data. In this case, we'll use merge to add the behavioral data to the eye-tracking data. This ensures that

Table 4

Quadrant coordinates in standardized space

loc	x_normalized	y_normalized	width_normalized	height_normalized	xmin	ymin	xmax	ymax
TL	0.00	0.50	0.50	0.50	0.00	0.50	0.50	1.00
TR	0.50	0.50	0.50	0.50	0.50	0.50	1.00	1.00
BL	0.00	0.00	0.50	0.50	0.00	0.00	0.50	0.50
BR	0.50	0.00	0.50	0.50	0.50	0.00	1.00	0.50

527 all rows from the eye-tracking data are preserved, even if there isn't a matching entry in the behavioral data
 528 (missing values will be filled with NA). The resulting object is called dat_L2.

```
dat_L2 <- merge(edat_1_L2, filter_edat_L2)
```

529 Areas of Interest

530 Zone coordinates

531 In the lab, we can control many aspects of the experiment that cannot be controlled online. Participants
 532 will be completing the experiment under a variety of conditions including, different computers, with
 533 very different screen dimensions. To control for this, Gorilla outputs standardized zone coordinates (labeled
 534 as x_pred_normalised and y_pred_normalised in the eye-tracking file) . As discussed in the Gorilla
 535 documentation, the Gorilla lays everything out in a 4:3 frame and makes that frame as big as possible. The
 536 normalized coordinates are then expressed relative to this frame; for example, the coordinate 0.5, 0.5 will
 537 always be the center of the screen, regardless of the size of the participant's screen. We used the normalized
 538 coordinates in our analysis (in general, you should always use normalized coordinates). However, there are
 539 a few different ways to specify the four coordinates of the screen, which are worth highlighting here.

540 **Quadrant approach.** One way is to make the AOIs as big as possible, dividing the screen into four
 541 quadrants. This approach has been used in several studies [e.g., (Bramlett & Wiener, 2024; Prystauka et al.,
 542 2024). Table 4 lists coordinates for the quadrant approach and Figure 6 shows how each quadrant looks in
 543 standardized space.

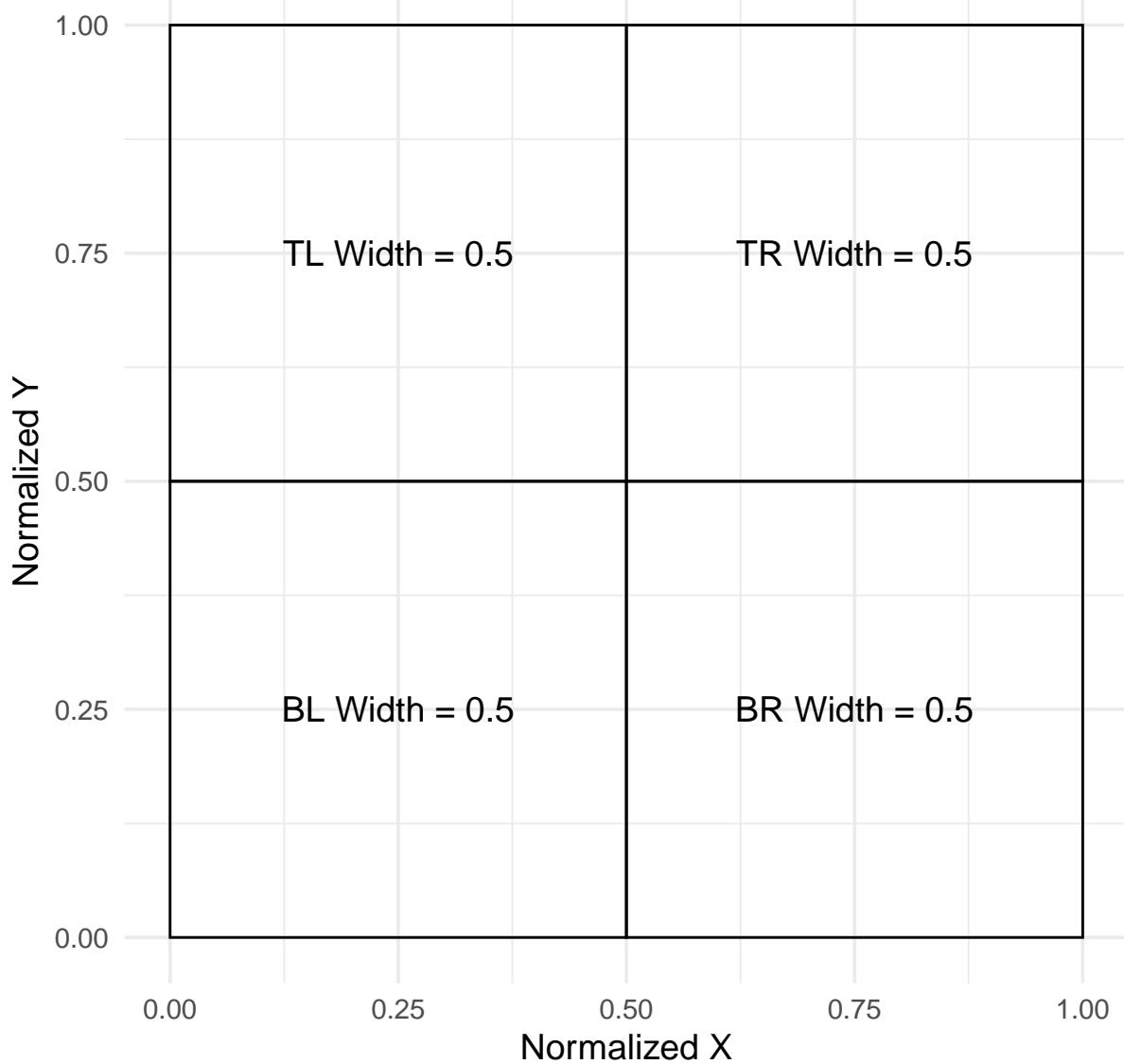
544 **Matching conditions with screen locations.** The goal of the below code is to assign condition codes
 545 (e.g., Target, Unrelated, Unrelated2, and Cohort) to each image in the dataset based on the screen location
 546 where the image is displayed (e.g., TL, TR, BL, BR).

547 For each trial, the images are dynamically placed at different screen locations, and the code maps
 548 each image to its corresponding condition based on these locations.

Figure 6

AOI coordinates in standardized space using the quadrant approach

Quadrants with Width Annotations



```
# Assuming your data is in a data frame called dat_L2
dat_L2 <- dat_L2 %>%
  mutate(
    Target = case_when(
      typetl == "target" ~ TL,
      typetr == "target" ~ TR,
      typebl == "target" ~ BL,
      typebr == "target" ~ BR,
      TRUE ~ NA_character_ # Default to NA if no match
    ),
    Unrelated = case_when(
      typetl == "unrelated1" ~ TL,
      typetr == "unrelated1" ~ TR,
      typebl == "unrelated1" ~ BL,
      typebr == "unrelated1" ~ BR,
      TRUE ~ NA_character_
    ),
    Unrelated2 = case_when(
      typetl == "unrelated2" ~ TL,
      typetr == "unrelated2" ~ TR,
      typebl == "unrelated2" ~ BL,
      typebr == "unrelated2" ~ BR,
      TRUE ~ NA_character_
    ),
    Cohort = case_when(
      typetl == "cohort" ~ TL,
      typetr == "cohort" ~ TR,
      typebl == "cohort" ~ BL,
      typebr == "cohort" ~ BR,
      TRUE ~ NA_character_
    )
  )
)
```

549 In addition to tracking the condition of each image during randomized trials, a custom function,
 550 `find_location()`, determines the specific screen location of each image by comparing it against the list
 551 of possible locations. This function ensures that the appropriate location is identified or returns NA if no
 552 match exists. Specifically, `find_location()` first checks if the image is NA (missing). If the image is NA,
 553 the function returns NA, meaning that there's no location to find for this image. If the image is not NA, the
 554 function creates a vector called `loc_names` that lists the names of the possible locations. It then attempts to

555 match the given image with the locations. If a match is found, it returns the name of the location (e.g., TL,
 556 TR, BL, or BR) of the image.

```
# Apply the function to each of the targ, cohort, rhyme, and unrelated columns
# Apply the function to each of the targ, cohort, rhyme, and unrelated columns
dat_colnames_L2 <- dat_L2 %>%
  rowwise() %>%
  mutate(
    targ_loc = find_location(c(TL = TL, TR = TR, BL = BL, BR = BR), Target),
    cohort_loc = find_location(c(TL = TL, TR = TR, BL = BL, BR = BR), Cohort),
    unrelated_loc = find_location(c(TL = TL, TR = TR, BL = BL, BR = BR),
      ↪ Unrelated),
    unrelated2_loc = find_location(c(TL = TL, TR = TR, BL = BL, BR = BR),
      ↪ Unrelated2)
  ) %>%
  ungroup()
```

557 Once we do this we can use the `assign_aoi()` function to loop through our object called
 558 `dat_colnames_L2` and assign locations (i.e., TR, TL, BL, BR) to where participants looked at on the screen.
 559 This requires the x and y coordinates and the location of our aois `aoi_loc`. Here we are using the quadrant
 560 approach. This function will label non-looks and off screen coordinates with NA. To make it easier to read
 561 we change the numerals assigned by the function to actual screen locations (e.g., TL, TR, BL, BR).

```
assign_L2 <- webgazeR::assign_aoi(dat_colnames_L2, X="x", Y="y", aoi_loc = aoi_loc)

AOI_L2 <- assign_L2 %>%

  mutate(loc1 = case_when(
    AOI==1 ~ "TL",
    AOI==2 ~ "TR",
    AOI==3 ~ "BL",
    AOI==4 ~ "BR"
  ))
```

562 In AOI_L2 we label looks to Targets, Unrelated, and Cohort items with 1 (looked) and 0 (no look)
 563 using the `case_when` function from the `tidyverse` (Wickham, 2017)

```
AOI_L2 <- AOI_L2 %>%
  mutate(
    target = case_when(loc1 == targ_loc ~ 1, TRUE ~ 0),
    unrelated = case_when(loc1 == unrelated_loc ~ 1, TRUE ~ 0),
    unrelated2 = case_when(loc1 == unrelated2_loc ~ 1, TRUE ~ 0),
    cohort = case_when(loc1 == cohort_loc ~ 1, TRUE ~ 0)
  )
```

564 The locations of looks need to be pivoted into long format—that is, converted from separate columns
 565 into a single column. This transformation makes the data easier to visualize and analyze. We use the
 566 `pivot_longer()` function from the `tidyverse` to combine the columns (Target, Unrelated, Unrelated2,
 567 and Cohort) into a single column called `condition1`. Additionally, we create another column called `Looks`,
 568 which contains the values from the original columns (e.g., 0 or 1 for whether the area was looked at).

```
dat_long_aoi_me_L2 <- AOI_L2 %>%
  select(subject, trial, condition, target, cohort, unrelated, unrelated2, time,
        x, y, RT_audio) %>%
  pivot_longer(
    cols = c(target, unrelated, unrelated2, cohort),
    names_to = "condition1",
    values_to = "Looks"
  )
```

569 We further clean up the object by first cleaning up the condition codes. They have a numeral ap-
 570 pended to them and that should be removed. We then adjust the timing in the `gaze_sub_L2_comp` object by
 571 aligning time to the actual audio onset. To achieve this, we subtract `RT_audio` from time for each trial. In
 572 addition, we subtract 300 ms from this to account for the 100 ms of silence at the beginning of each audio
 573 clip and 200 ms to account for the oculomotor delay when planning an eye movement (Viviani, 1990). Ad-
 574 ditionally, we set our interest period between 0 ms (audio onset) and 2000 ms. This was chosen based on the
 575 time course figures in Sarrett et al. (2022). It is important that you choose your interest area carefully and
 576 preferably you preregister it. The interest period you choose can bias your findings (Peelle & Van Engen,
 577 2021). We also filter out gaze coordinates that fall outside the standardized window, ensuring only valid data
 578 points are retained. The resulting object `gaze_sub_long_L2` provides the corrected time column spanning
 579 from -200 ms to 2000 ms relative to stimulus onset with looks outside the screen removed.

```
# repalce the numbers appended to conditions that somehow got added
dat_long_aoi_me_comp <- dat_long_aoi_me_L2 %>%
  mutate(condition = str_replace(condition, "TCUU-SPENG\\d*", "TCUU-SPENG")) %>%
  mutate(condition = str_replace(condition, "TCUU-SPSP\\d*", "TCUU-SPSP"))%>%
  na.omit()
```

```
# dat_long_aoi_me_comp has condition corrected

gaze_sub_L2_long <-dat_long_aoi_me_comp%>%
  group_by(subject, trial, condition) %>%
  mutate(time = (time-RT_audio)-300) %>% # subtract audio rt onset and account
  ↵ for occ motor planning and silence in audio
  filter(time >= -200, time < 2000)
```

580 **Samples to bins**

581 ***Downsampling***

582 Downsampling into larger time bins is a common practice in gaze data analysis, as it helps create
 583 a more manageable dataset and reduces noise. When using research grade eye-trackers, downsampling is
 584 an optional step in the preprocessing pipeline. However, with consumer-based webcam eye-tracking it is
 585 recommended you downsample your data so participants have consistent bin sizes (e.g., (Slim et al., 2024;
 586 Slim & Hartsuiker, 2023)). In webgazeR we included the `downsample_gaze()` function to assist with this
 587 process. We apply this function to the `gaze_sub_L2_long` object, and set the `bin.length` argument to 100,
 588 which groups the data into 100-millisecond intervals. This adjustment means that each bin now represents a
 589 100 ms passage of time. We specify `time` as the variable to base these bins on, allowing us to focus on broader
 590 patterns over time rather than individual millisecond fluctuations. There is no agreed upon downsampling
 591 value, but with webcam data larger bins are preferred (see Slim & Hartsuiker, 2023).

592 In addition, the `downsample_gaze()` allows you to aggregate across other variables, such as
 593 `condition`, `condition1`, and use the newly created `time_bins` variable, which represents the time in-
 594 tervals over which we aggregate data. The resulting downsampled dataset, output as Table 5, provides a
 595 simplified and more concise view of gaze patterns, making it easier to analyze and interpret broader trends.

```
gaze_sub_L2 <- webgazeR::downsample_gaze(gaze_sub_L2_long, bin.length=100,
  ↵ timevar="time", aggvars=c("condition", "condition1", "time_bin"))
```

596 To simplify the analysis, we combine the two unrelated conditions and average them (this is for the
 597 proportional plots).

Table 5

Aggregated proportion looks for each condition in each 100 ms time bin

condition	condition1	time_bin	Fix
TCUU-ENGSP	cohort	-200.00	0.26
TCUU-ENGSP	cohort	-100.00	0.26
TCUU-ENGSP	cohort	0.00	0.25
TCUU-ENGSP	cohort	100.00	0.25
TCUU-ENGSP	cohort	200.00	0.23
TCUU-ENGSP	cohort	300.00	0.23

```
# Average Fix for unrelated and unrelated2, then combine with the rest
gaze_sub_L2_avg <- gaze_sub_L2 %>%
  group_by(condition, time_bin) %>%
  summarise(
    Fix = mean(Fix[condition1 %in% c("unrelated", "unrelated2")], na.rm =
      TRUE),
    condition1 = "unrelated", # Assign the combined label
    .groups = "drop"
  ) %>%
  # Combine with rows that do not include unrelated or unrelated2
  bind_rows(gaze_sub_L2 %>% filter(!condition1 %in% c("unrelated",
    "unrelated2")))
```

598 The above will not include the subject variable. If you want to keep participant-level data we need
 599 to add `subject` to the `aggvars` argument.

```
# add subject-level data
gaze_sub_L2_id <- webgazeR::downsample_gaze(gaze_sub_L2_long, bin.length=100,
  timevar="time", aggvars=c("subject", "condition", "condition1", "time_bin"))
```

600 *Upsampling*

601 Users may wish to upsample their data rather than downsample it. This is standard in some
 602 preprocessing pipelines in pupillometry (Kret & Sjak-Shie, 2018) and has recently been applied to
 603 webcam-based eye-tracking data (Madsen et al., 2021). Like downsampling, upsampling standardizes

604 the time intervals between samples; however, it also increases the sampling rate, which can produce
 605 smoother, less noisy data. This is useful if you want to align webcam eye-tracking with other measures
 606 (e.g., EEG).

607 Our webgazeR package provides several functions to assist with this process. The
 608 upsample_gaze() function allows users to upsample their gaze data to a higher sampling rate (e.g.,
 609 250 Hz or even 1000 Hz). After upsampling, users can apply the smooth_gaze() function to reduce
 610 noise (webgazeR uses a n-point moving average) followed by the interpolate_gaze() function to fill
 611 in missing values using linear interpolation. Below we show you how to use the function, but do not
 612 apply to the data.

```
AOI_upsample <- AOI %>%
  group_by(subject, trial) %>%
  upsample_gaze(
    gaze_cols = c("x", "y"),
    upsample_pupil = FALSE,
    target_hz = 250)
```

```
AOI_smooth=smooth_gaze(AOI_upsample, n = 5, x_col = "x", y_col = "y",
                         trial_col = "trial", subject_col = "subject")
```

```
aoi_interp <- interpolate_gaze(deduplicated_data,x_col = "x_pred_normalised",
                                 ← y_col = "y_pred_normalised",
                                 trial_col = "trial", subject_col = "subject",
                                 ← time_col="time" )
```

613 Aggregation

614 Aggregation is an optional step. If you do not plan to analyze proportion data, and instead what time
 615 binned data with binary outcomes preserved please set the aggvars argument to “none.” This will return a
 616 time binned column, but will not aggregate over other variables.

```
# get back trial level data with no aggregation
gaze_sub_id <- downsample_gaze(gaze_sub_L2_long, bin.length=100, timevar="time",
                                ← aggvars="none")
```

617 We need to make sure we only have one unrelated value.

```
# make only one unrelated condition
gaze_sub_id <- gaze_sub_id %>%
  mutate(condition1 = ifelse(condition1=="unrelated2", "unrelated", condition1))
```

618 **Visualizing time course data**

619 To simplify plotting your time-course data, we have created the `plot_IA_proportions()` function.
 620 This function takes several arguments. The `ia_column` argument specifies the column containing
 621 your AOI labels. The `time_column` argument requires the name of your time bin column, and the
 622 `proportion_column` argument specifies the column containing fixation or look proportions. Additional
 623 arguments allow you to specify custom names for each IA in the `ia_mapping` argument, enabling you to
 624 label them as desired. In order to use this function, you must use the `downsample_gaze()` function.

625 Below, we have plotted the time-course data for each condition in Figure 7. By default, the graphs
 626 utilize a color-blind-friendly palette from the `ggokabeito` package (Barrett, 2021). However, you can set
 627 the argument `use_color = FALSE` to generate a non-colored version of the figure, where different line types
 628 and shapes differentiate conditions. Additionally, since these are `ggplot` objects, you can further customize
 629 them as needed to suit your analysis or presentation preferences.

Figure 7

Comparison of L2 competition effect in the No Competitor (a), Spanish–English (b), the Spanish–Spanish (c) conditions

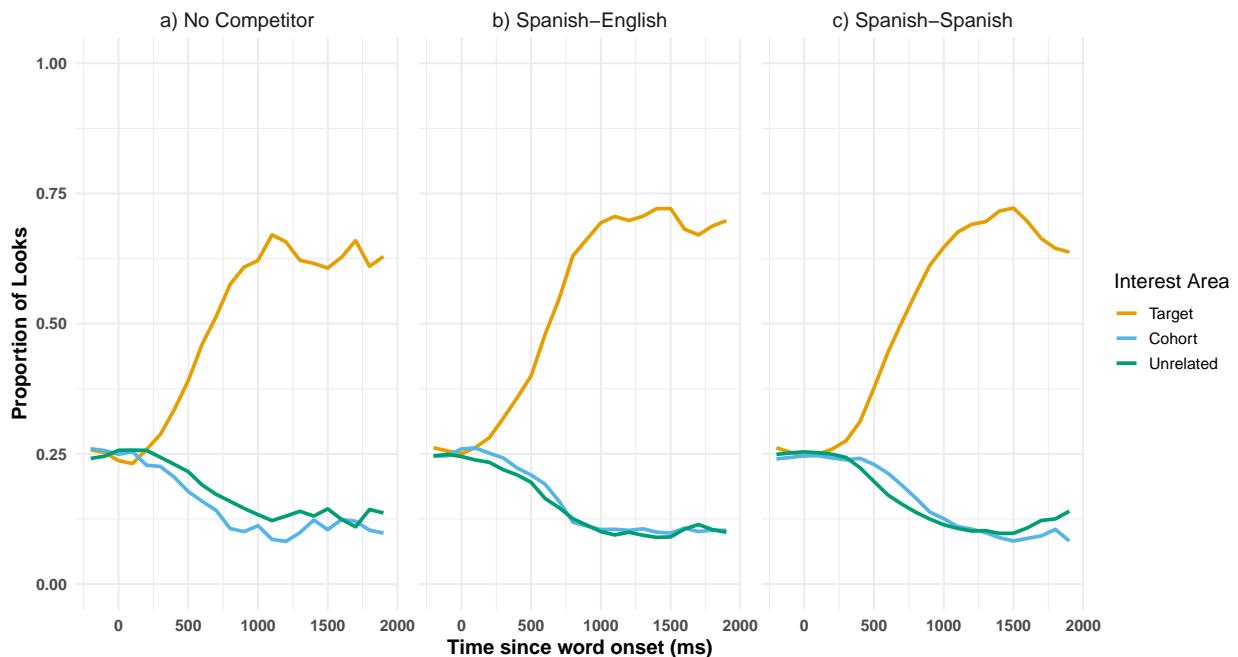


Table 6

Gorilla provided standarized gaze coordinates

loc	x_normalized	y_normalized	width_normalized	height_normalized	xmin	ymin	xmax	ymax
BL	0.03	0.04	0.26	0.25	0.03	0.04	0.29	0.29
TL	0.02	0.74	0.26	0.25	0.02	0.74	0.28	0.99
TR	0.73	0.75	0.24	0.24	0.73	0.75	0.97	0.99
BR	0.73	0.06	0.23	0.25	0.73	0.06	0.96	0.31

630 Gorilla provided coordinates

631 Thus far, we have used the coordinates representing the four quadrants of the screen. However,
 632 Gorilla provides their own quadrants representing image location on the screen. To the authors' knowledge,
 633 these quadrants have not been looked at in any studies reporting eye-tracking results. Let's examine how
 634 reasonable our results are with the Gorilla provided coordinates.

635 We will use the function `extract_aois()` to get the standardized coordinates for each quadrant on
 636 screen. You can use the `zone_names` argument to get the zones you want to use. In our example, we want the
 637 TL, BR, BL TR coordinates. We input the object from above `vwp_paths_filtered_L2` that contains all our
 638 eye-tracking files and extract the coordinates we want. These are labeled in Table 6. In Figure 8 we can see
 639 that the AOIs are a bit smaller than then when using the quadrant approach. We can take these coordinates
 640 and use them in our analysis. Looking at Figure 9, we see the data is a bit nosier than the quadrant approach,
 641 but the curves are reasonable.

```
# apply the extract_aois fucntion
aois_L2 <- extract_aois(vwp_paths_filtered_L2, zone_names = c("TL", "BR", "TR",
  ↵ "BL"))
```

```
assign_L2_gor <- webgazeR::assign_aoi(dat_colnames_L2,X="x", Y="y",aoi_loc =
  ↵ aois_L2)
```

642 Modeling data

643 After you have preprocessed the data, the next step is analysis. When analyzing VWP data there are
 644 many analytic approaches to choose from (e.g., growth curve analysis (GCA), cluster permutation analysis
 645 (CPA), generalized additive mixed models (GAMMS), logistic multilevel models, divergent point analysis
 646 (DPA), etc.), and a lot has already been written describing these methods and a lot of great tutorials exist

Figure 8

Gorilla provided standardized coordinates for the four quadrants on the screen

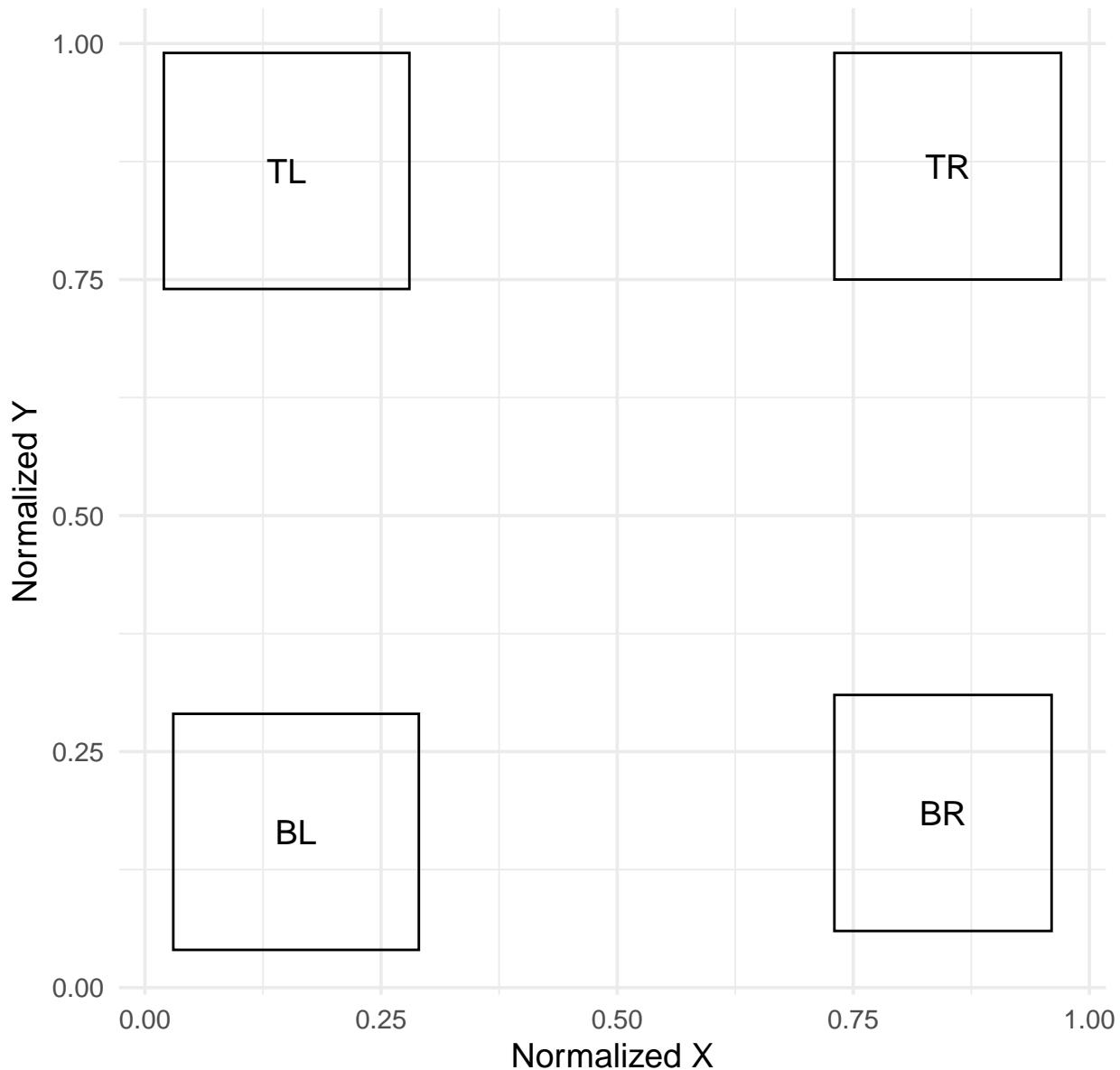
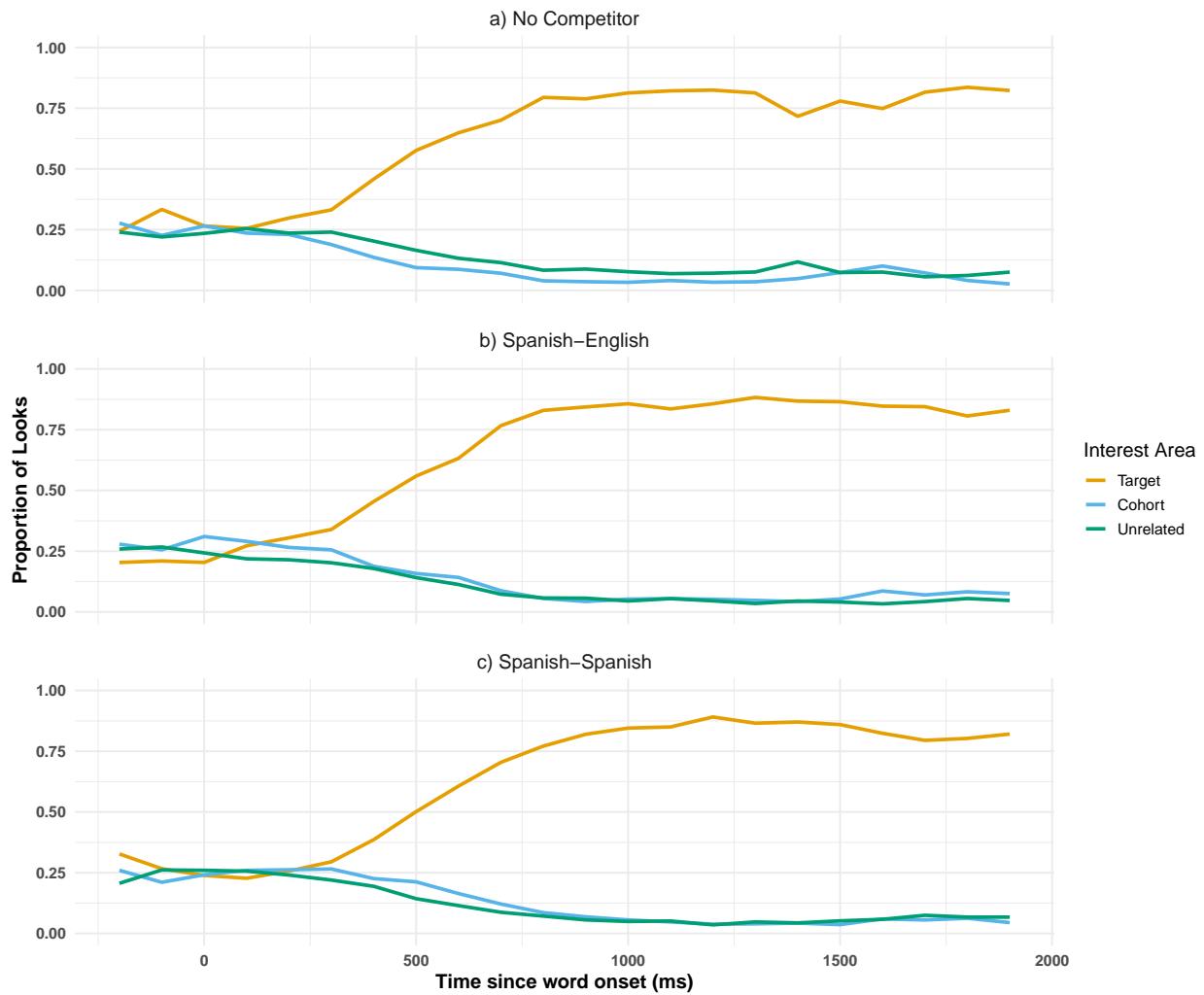


Figure 9

Comparison of competition effects with Gorilla standardized coordinates



647 showing how to apply these methods to visual world fixation data from the lab (Coretta & Casillas, 2024;
 648 see Ito & Knoeferle, 2023; Stone et al., 2021) and online (Bramlett & Wiener, 2024).

649 This paper's goal, however, is to not evaluate different analytic approaches and tell readers what they
 650 should use. All methods have their strengths and weaknesses (see Ito & Knoeferle, 2023). Nevertheless,
 651 statistical modeling should be guided by the questions researchers have and thus serious thought needs to be
 652 given to the proper analysis. In the VWP, there are two general questions one might be interested in: (1) Are
 653 there any overall difference in fixations between conditions and (2) Are there any time course differences in
 654 fixations between conditions (and/or groups).

655 With our data, one question we might want to answer is if there are any fixation differences between
 656 the cohort and unrelated conditions across the time course. One statistical approach we chose to highlight
 657 to answer this question is a cluster permutation analysis (CPA). The CPA is suitable for testing differences

658 between two conditions or groups over an interest period while controlling for multiple comparisons and
659 autocorrelation. **Given the time latency issues common in webcam-basted studies, Slim et al. (2024)**
660 **recommended using an approach like CPA.**

661 **CPA**

662 CPA is a technique that has become increasingly popular, particularly in the field of cognitive neu-
663 ropsychology, for analyzing MEG and EEG data (Maris & Oostenveld, 2007). While its adoption in VWP
664 studies has been relatively slow, it is now beginning to appear more frequently (see Huang & Snedeker, 2020;
665 Ito & Knoeferle, 2023). Notably, its use is growing in online eye-tracking studies (see Slim et al., 2024; Slim
666 & Hartsuiker, 2023; Vos et al., 2022).

667 Before we show you how to apply this method to the current dataset, we want to briefly explain what
668 CPA is. The CPA is a data-driven approach that increases statistical power while controlling for Type I errors
669 across multiple comparisons—exactly what we need when analyzing fixations across the time course.

670 The clustering procedure involves three main steps:

- 671 1. Cluster Formation: With our data, a multilevel logistic model is conducted for every data point (con-
672 dition by time). Please note that any statistical test can be run here. Adjacent data points that surpass
673 the mass univariate significance threshold (e.g., $p < .05$) are combined into clusters. The cluster-
674 level statistic, typically the sum of the t-values (or F-values) within the cluster, is computed labeled
675 as SumStatistic is output below). By clustering adjacent significant data points, this step accounts for
676 autocorrelation by considering temporal dependencies rather than treating each data point as indepen-
677 dent.
- 678 2. Null Distribution Creation: Next, the same analysis is run as in step 1. However, the analysis is based
679 on randomly permuting or shuffling the conditions within subjects. This principle of exchangeability is
680 important here, as it suggests that the condition labels can be exchanged without altering the underlying
681 data structure. This randomization is repeated n times (e.g., 1000 shuffles), and for each permutation,
682 the cluster-level statistic is computed. This step addresses the issue of multiple comparisons by con-
683 structing a distribution of cluster-level statistics under the null hypothesis, providing a baseline against
684 which observed cluster statistics can be compared. By doing so, the method controls the family-wise
685 error rate and ensures that significant findings are not simply due to chance.
- 686 3. Significance Testing: The cluster-level statistics from the observed (real) comparison is compared to
687 the null distribution we created above. Clusters with statistics falling in the highest or lowest 2.5% of
688 the null distribution are considered significant (e.g., $p < 0.05$).

689 To perform CPA, we will load in the permutes (Voeten, 2023), permuco (Frossard & Renaud,
690 2021), foreach (& Weston, 2022), and Parallel (Corporation & Weston, 2022) packages in R. Loading
691 these packages allow us to use the cluster.glmer() function to run a cluster permutation (10,000 rimes)

Table 7

Clustermass statistics for the Spanish-Spanish condition

cluster	cluster_mass	p.cluster_mass	bin_start	bin_end	t	sign	time_start	time_end	
1	236.34		0	7	13	5.48	1	500	1,100

692 across multiple system cores to speed up the process. We run a CPA on the `gaze_sub_id` object where each
 693 row in `Looks` denotes whether the AOI was fixated, with values of zero (not fixated) or one (fixated).

694 Below you find sample code to perform multilevel CPA in R (please see the Github repository for
 695 elaborated code needed to perform CPA).

```
library(permutes) # cpa
library(permuco) # cpa

total_perms <- 1000

cpa.lme <- permutes::clusterperm.glmer(Looks~ condition1_code + (1|subject) +
  ~ (1|trial), data=gaze_sub_L2_cp1, series.var=~time_bin, nperm = total_perms)
```

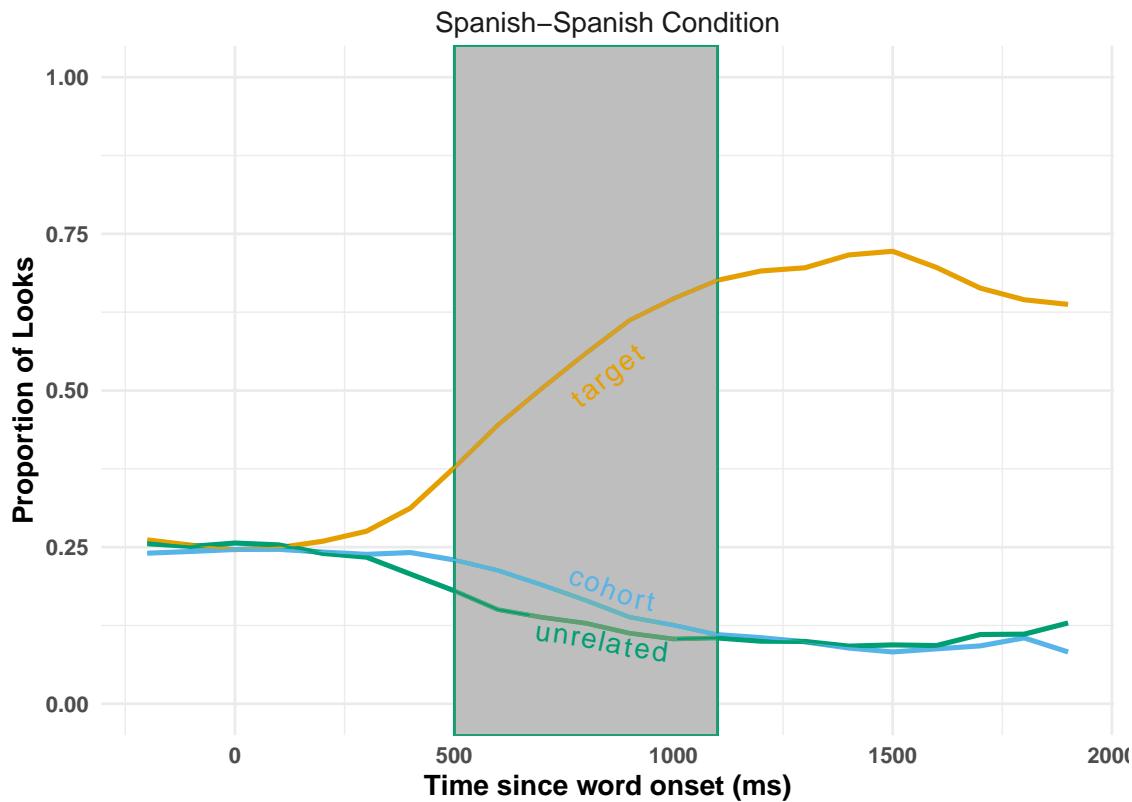
696 In the analysis for the Spanish-Spanish condition, one significant cluster was observed between
 697 500 and 1,100 ms, as indicated in the summary statistics from Table 7. The positive SumStatistic value
 698 associated with this cluster suggests that competition was greater during this time window. This result im-
 699 plies that cohorts in the Spanish-Spanish condition exhibited stronger effects or competition compared to
 700 unrelated items. In Figure 10 significant clusters are highlighted for both the Spanish-Spanish and Spanish-
 701 English conditions. Both conditions show one significant cluster. Overall, the analysis suggests that both
 702 the Spanish-Spanish and Spanish-English conditions demonstrate significant competitor effects.

703 **Effect size.** It is important to address the issue of effect sizes in the context of CPA. Calculating
 704 effect sizes for CPA is not straightforward, as the technique is designed to evaluate temporal clusters rather
 705 than individual time points. (Slim et al., 2024; but also see Meyer et al., 2021) outline three possible ap-
 706 proaches for estimating effect sizes in CPA: (1) computing the effect size within a predefined time window
 707 (often the same window used for identifying clusters), (2) calculating an average effect size across the entire
 708 cluster, and (3) reporting the maximum effect observed within the cluster. Each method has its trade-offs
 709 in terms of interpretability and comparability across studies, and the choice should be guided by theoretical
 710 considerations and the research question at hand.

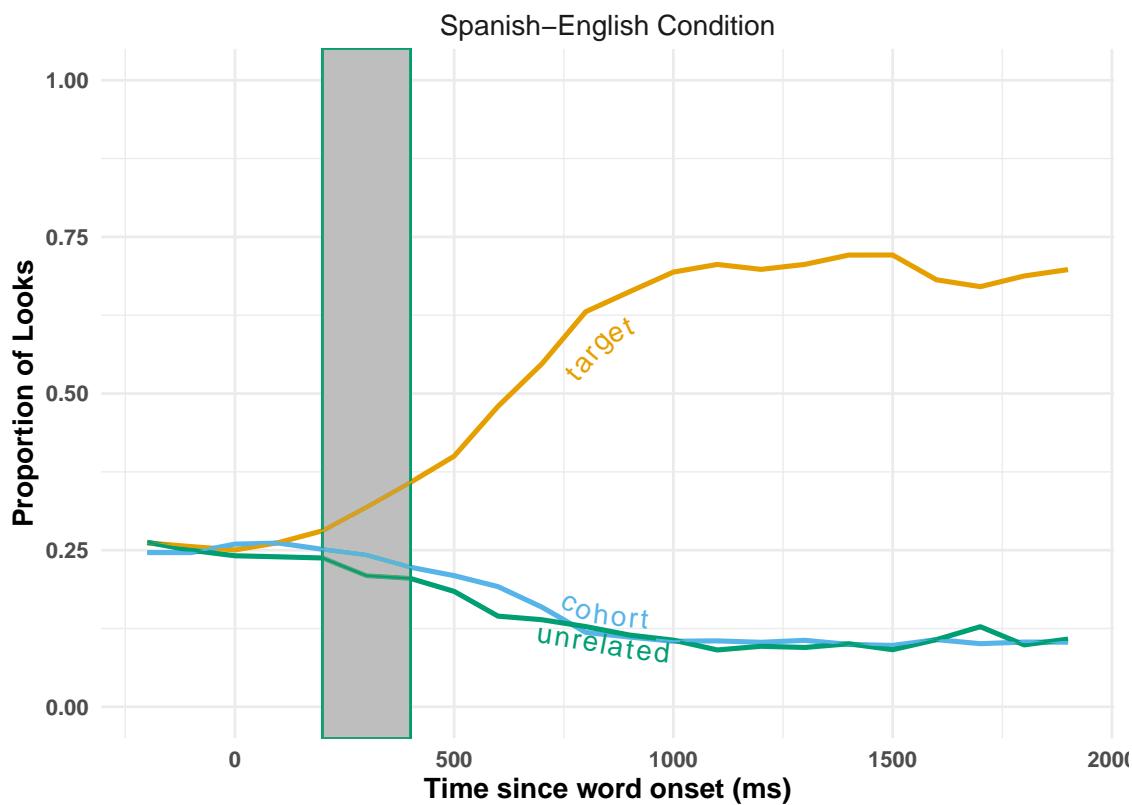
Figure 10

Average looks in the cross-linguistic VWP task over time for the Spanish-Spanish condition (a) and the Spanish-English condition (b). The shaded rectangles indicate when cohort looks were greater than chance based on the CPA.

A



B



711

Discussion

712 Webcam eye-tracking is a relatively nascent technology, and as such, there is limited guidance available
713 for researchers. To ameliorate this, we created a tutorial to assist new users of visual world webcam
714 eye-tracking, using some of the best practices available (e.g., Bramlett & Wiener, 2024). To further facilitate
715 this process, we created the `webgazeR` package, which contains several helper functions designed to
716 streamline data preprocessing, analysis, and visualization.

717 In this tutorial, we covered the basic steps of running a visual world webcam-based eye-tracking
718 experiment. We highlighted these steps by using data from a cross-linguistic VWP looking at competitive
719 processes in L2 speakers of Spanish. Specifically, we attempted to replicate the experiment by Sarrett et al.
720 (2022) where they observed within- and between L2/L1 competition using carefully crafted materials.

721 **Replication of Sarrett et al. (2022)**

722 While the main purpose of this tutorial was to highlight the steps needed to analyze webcam eye-
723 tracking data, replicating Sarrett et al. (2022) allowed us to not only assess whether within and between
724 L2/L1 competition can be found in a spoken word recognition VWP experiment online, but also provide
725 insight in how to run VWP studies online and the issues associated with it.

726 Our conceptual replication yielded highly encouraging results, revealing robust competition effects
727 both within-language (Spanish-Spanish) and across-language (Spanish-English) conditions—closely mir-
728 roring those reported by Sarrett et al. (2022). However, several key analytic, methodological, and sample
729 differences between our study and theirs warrant discussion.

730 A major analytic difference lies in how the time course of competition was examined. While Sarrett
731 et al. (2022) employed a non-linear curve-fitting approach (see McMurray et al., 2010), we used cluster-
732 based permutation analysis (CPA). This methodological distinction limits direct comparisons regarding the
733 temporal dynamics of competition. Nonetheless, the overall time course patterns align surprisingly well: our
734 CPA identified a significant cluster starting at 500 ms, while Sarrett et al. (2022) observed effects beginning
735 around 400 ms—suggesting a modest delay of approximately 100 ms in our online data. This delay is still
736 markedly smaller than in previous webcam-based studies (e.g., Semmelmann & Weigelt, 2018; Slim et al.,
737 2024), reflecting progress in online eye-tracking. That said, it's important to note that CPA is not ideally
738 suited for making precise temporal inferences about onset or offset of effects (Fields & Kuperberg, 2019; Ito
739 & Knoeferle, 2023).

740 Design differences between the studies also play a critical role. In Sarrett et al. (2022), participants
741 previewed the images in each quadrant for 1000 ms, followed by the appearance of a central red dot they
742 clicked to trigger audio playback. After selecting the target, a 250 ms inter-trial interval (ITI) preceded the
743 next trial.

744 In contrast, our sequence began with a 500 ms fixation cross (serving as the ITI), followed by a longer
745 1500 ms preview. The images then disappeared, and participants clicked a centrally placed start button to
746 initiate audio playback, at which point the images reappeared. Upon target selection, the next trial began

immediately. We also imposed a 5-second timeout for non-responses. Additionally, our study included 250 trials—fewer than the 450 in the original study²—but still more than most webcam-based research. Despite the reduced trial count, we observed parallel competition effects in both language conditions, underscoring the robustness of the findings.

Several motivations guided these design adaptations. Online testing introduces greater variability in participants' setups (e.g., device type, connection quality), so we opted for a longer preview period to enhance the likelihood of observing competition effects. Prior work suggests this can boost competition signals in the VWP (Apfelbaum et al., 2021). The start-button mechanism ensured trials began from a centralized gaze position, helping minimize quadrant-based bias. Finally, the timeout feature helped mitigate issues of inattention common in unsupervised online environments.

Participant recruitment also differed. Sarrett et al. (2022) recruited students from a Spanish language course and assessed proficiency using the LexTALE-Spanish test (Izura et al., 2014). Our participants were recruited through Prolific with more limited screening, allowing us only to filter by native language and reported experience with another language. This constraint likely contributed to differences in language profiles between samples. Whereas Sarrett et al. (2022) included L2 learners with verified proficiency, our sample encompassed a broader and more variable group of L2 speakers, with limited verification of language skills (see Table 1 for details). This broader variability may help explain the absence of a sustained cohort competition effect in our study.

In sum, while there are notable differences in methods and samples, the convergence of competition effects across both studies—within and across languages—supports the robustness of these phenomena across diverse research contexts. Still, we view these results as a promising step rather than definitive evidence. A more systematic investigation is needed to fully establish the generalizability of these effects.

Limitations

Recruitment of L2 Speakers

In this study, we used the Prolific platform to recruit L2 Spanish speakers. We specified criteria requiring participants to be native English speakers who were also proficient in Spanish, reside in the United States, and be between the ages of 18 and 36. These criteria yielded a potential recruitment pool of approximately 1,000 participants. While this number is larger than what is typically available for in-lab studies, it is still relatively limited given the overall size of the platform. Notably, English native speakers who are L2 learners of Spanish in the U.S. are not usually considered a particularly niche population, which highlights the extent of the recruitment difficulty. Participant pools are likely to be even more limited when targeting speakers of less commonly studied languages or with specific language backgrounds (e.g. heritage speakers). Moreover, Prolific currently supports only an English user interface, which makes it harder to recruit non-English speakers (Niedermann et al.,

²The curve-fitting approach used by Sarrett et al. (2022) may have required a larger number of trials to obtain reliable fits. Their study included over 400 trials, while our design was more constrained.

Table 8*Eye-tracking questionnaire items*

Question
1.Do you have a history of vision problems (e.g., corrected vision, eye disease, or drooping eyelids)?
2.Are you on any medications currently that can impair your judgement?
If yes, please list below:
4.Does your room currently have natural light?
5.Are you using the built in camera?
If no, what brand of camera are you using?
6.Please estimate how far you think you were sitting from the camera during the experiment (an arm's length from your monitor is about 20 inches (51 cm).
7.Approximately how many times did you look at your phone during the experiment?
8.Approximately how many times did you get up during the experiment?
9.Was the environment you took the experiment in distraction free?
10. When you had to calibrate, were the instructions clear?
11. What additional information would you add to help make things easier to understand?
12. Are you wearing a mask?

781 **2024; Patterson & Nicklin, 2023). For second language research in particular, researchers should be**
 782 **aware of these and other constraints (such as the limited filtering options to control for proficiency)**
 783 **and consider incorporating language background questionnaires and/or proficiency tasks directly into**
 784 **the study design. Ultimately, 181 participants signed up for the study, and recruitment proved to be**
 785 **more challenging than expected. Researchers considering similar studies should be aware of these lim-**
 786 **itations when targeting niche populations, even on large online platforms. Despite these challenges, the**
 787 **final sample was sufficient for our planned analyses and opened up the possibility to target populations**
 788 **you would be unable to capture otherwise.**

789 **Generalizability to other platforms**

790 We demonstrated how to analyze webcam eye-tracking data collected via the Gorilla platform using
 791 WebGazer.js. Although we did not validate this pipeline on other platforms that support WebGazer.js—
 792 such as PCIbex (Zehr & Schwarz, 2018), jsPsych (Leeuw, 2015), or PsychoPy (Peirce et al., 2019)—we
 793 believe the pipeline is generalizable to these and to platforms that use other gaze estimation logarithms,
 794 such as Labvanced [@kaduk2024]. To support broader compatibility, the functions in the webgazeR
 795 package are designed to work with a variety of file types—including .csv, .tsv, and .xlsx – and work
 796 with any dataset that includes five essential columns: subject, trial, x, y, and time. We also provide a
 797 helper function, `make_webgazer()`, to assist in renaming columns so your dataset can be adapted to
 798 the expected format.

Table 9

Responses to eye-tracking questions for participants who successfully calibrated vs. participants who had trouble calibrating

Question	Response	Good	Bad
1.Do you have a history of vision problems (e.g., corrected vision, eye disease, or drooping eyelids)?	No	65.71	64.29
1.Do you have a history of vision problems (e.g., corrected vision, eye disease, or drooping eyelids)?	Yes	34.29	35.71
2.Are you on any medications currently that can impair your judgement?	No	100.00	98.21
2.Are you on any medications currently that can impair your judgement?	Yes	0.00	1.79
4.Does your room currently have natural light?	No	40.00	26.79
4.Does your room currently have natural light?	Yes	60.00	73.21
5.Are you using the built in camera?	No	14.29	8.93
5.Are you using the built in camera?	Yes	85.71	91.07
9.Was the environment you took the experiment in distraction free?	No	11.43	3.57
9.Was the environment you took the experiment in distraction free?	Yes	88.57	96.43

799 **We encourage researchers to test this pipeline in their own studies and report any issues or sug-
800 gestions on our GitHub repository. We are committed to improving webgazeR and welcome feedback
801 that will make the package more flexible, user-friendly, and adaptable to a wider range of experimental
802 platforms.**

803 ***Power***

804 While we successfully demonstrated competition effects similar to Garrett's study, we did not conduct
805 an a priori power analysis nor was it our intention. With webcam eye-tracking, it has been recommended
806 running twice the number of participants from the original sample, or powering the study to detect an effect
807 size half as large as the original (Slim & Hartsuiker, 2023; Van der Cruyssen et al., 2024). We did attempt
808 to increase our sample size 2x, but were unable to recruit enough participants through Prolific. However,
809 our sample size is similar to the lab based study. Regardless, researchers should be aware of this and plan
810 accordingly.

811 We strongly urge researchers to perform power analyses and justify their sample sizes (Lakens, 2022).
812 While tools like G*Power (Faul et al., 2007) are available for this purpose, we recommend power simulations
813 using Monte Carlo or resampling methods on pilot or sample data (see Prystauka et al., 2024; Slim & Hart-
814 suiker, 2023). Several excellent R packages, such as `mixedpower` (Kumle et al., 2021) and `SIMR` (Green &
815 MacLeod, 2016) make such simulations straightforward and accessible.

816 *****Recommendations and ways forward*****

817 While our findings support the promise of webcam eye-tracking for language research, several chal-
818 lenges remain that researchers should consider. One of the most significant issues is data loss due to poor
819 calibration. In our study, we excluded approximately 75% of participants due to calibration failure. These
820 attrition rates are in line with some previous reports (e.g., Slim & Hartsuiker, 2023), though others have
821 found substantially lower rates (Bramlett & Wiener, 2025; Prystauka et al., 2024). With this valuation, it is
822 important to understand the factors that lead to better quality data.

823 To address this, we included a post-task questionnaire assessing participants' setups and their experi-
824 ences with the experiment. These questions, included in Table 8, provide insights that informed the following
825 recommendations, which we also base on our experimental design and personal experience.

826 In our experimental design, participants were branched based on whether they successfully com-
827 pleted the experiment or failed calibration at any point. Table 9 highlights the comparisons between good
828 and poor calibrators. For the sake of brevity, we will discuss some recommendations based on questionnaire
829 responses and personal experience that will hopefully improve research using webcam eye-tracking.

830 *****Prioritize external webcams*****

831 Our data suggest that participants using external webcams were significantly more likely to complete
832 the calibration successfully than those using built-in laptop cameras. External webcams typically offer higher

833 resolution and frame rates—both critical for accurate gaze estimation (Slim & Hartsuiker, 2023) Researchers
834 should, whenever possible, encourage participants to use external webcams and may consider administering
835 a brief pre-experiment questionnaire to screen for webcam type and exclude low-quality setups.

836 *****Optimize environmental conditions*****

837 Poor calibration was often reported in environments with natural light. Ambient lighting introduces
838 variability that can degrade tracking performance. We recommend that researchers instruct participants to
839 complete studies in rooms with consistent artificial lighting and minimal glare or shadows.

840 In addition to lighting, *****head movement and distance from the screen***** are critical for achieving
841 reliable eye-tracking. Excessive movement or leaning in and out of the camera's view can disrupt the face
842 mesh tracking used by WebGazer.js. Participants should be advised to remain still and maintain a consistent,
843 moderate distance from the screen—approximately 50–70 cm, depending on their camera setup. We asked
844 individuals to provide an approximate distance from their screens, (arms length) but it is not clear how
845 accurate this is. Providing clear guidance (e.g., via an instructional video) may help mitigate these issues
846 and improve overall tracking fidelity).

847 A different platform, Labvanced [@kaduk2024], for example, offers additional eye-tracking func-
848 tionality including a virtual chinrest to ensure head movement is restricted to an acceptable range and warns
849 users if they deviate from this range. Together this might make for a better eye-tracking experience with less
850 data thrown out. This should be investigated further.

851 *****Conduct a priori power analysis***

852 *****To ensure adequate statistical power, researchers should conduct a priori power analyses either***
853 ***via GUI like GPower or perform Monte Carlo simulations/resampling on pilot data. This step is particularly***
854 ***important for online studies, where sample variability can be higher than in controlled lab environments. To***
855 ***this point, you will have to over-enroll your study due to the high attrition rate to reach your target goal, so***
856 ***please plan accordingly.*****

857 *****Collect detailed post-experiment feedback*****

858 *****Gathering detailed feedback about participants' setups—such as webcam type, browser, lighting***
859 ***conditions, and perceived ease of use—can provide valuable information about what contributes to successful***
860 ***calibration. These insights can inform more effective participant instructions and refined inclusion criteria***
861 ***for future studies.***

862 By implementing these strategies, researchers can improve the quality and consistency of data col-
863 lected through webcam-based eye-tracking. These recommendations aim to maximize the utility and repro-
864 ducibility of remote eye-tracking research, particularly in language processing contexts.**

865 **Conclusions**

866 This work highlights the steps required to process webcam eye-tracking data, demonstrating the
867 potential of webcam-based eye-tracking for robust psycholinguistic experimentation. By providing a stan-
868 dardized pipeline for processing eye-tracking data, we aim to give researchers a clear and practical path for
869 collecting and analyzing visual world webcam eye-tracking data. **An interactive demo of the preprocess-
870 ing pipeline—using data from a monolingual visual world paradigm (VWP)—is available at the webgazeR
871 website (https://jgeller112.github.io/webgazeR/vignettes/webgazeR_vignette.html), where users can explore
872 the code and workflow firsthand.**

873 Moreover, our findings demonstrate the feasibility of conducting high-quality online experiments,
874 paving the way for future research to address more nuanced questions about L2 processing and language
875 comprehension more broadly. Additionally, further refinement of webcam eye-tracking methodologies could
876 enhance data precision and extend their applicability to more complex experimental designs. This is an
877 exciting time for eye-tracking research, with its boundaries continuously expanding. We eagerly anticipate
878 the advancements and possibilities that the future of webcam eye-tracking will bring.

879 **References**

- 880 Allaire, J. J., Teague, C., Scheidegger, C., Xie, Y., Dervieux, C., & Woodhull, G. (2024). *Quarto* (Version
881 1.6) [Computer software]. <https://doi.org/10.5281/zenodo.5960048>
- 882 Allopenna, P. D., Magnuson, J. S., & Tanenhaus, M. K. (1998). *Tracking the time course of spoken word
883 recognition using eye movements: Evidence for continuous mapping models* (pp. 419–439).
- 884 Altmann, G. T. M., & Kamide, Y. (1999). Incremental interpretation at verbs: Restricting the domain of
885 subsequent reference. *Cognition*, 73(3), 247–264. [https://doi.org/10.1016/S0010-0277\(99\)00059-1](https://doi.org/10.1016/S0010-0277(99)00059-1)
- 886 Anderson, C. A., Allen, J. J., Plante, C., Quigley-McBride, A., Lovett, A., & Rokkum, J. N. (2019). The
887 MTurkification of Social and Personality Psychology. *Personality & Social Psychology Bulletin*, 45(6),
888 842–850. <https://doi.org/10.1177/0146167218798821>
- 889 Anwyl-Irvine, A. L., Massonnié, J., Flitton, A., Kirkham, N., & Evershed, J. K. (2020). Gorilla in our
890 midst: An online behavioral experiment builder. *Behavior Research Methods*, 52(1), 388–407. <https://doi.org/10.3758/s13428-019-01237-x>
- 891 Apfelbaum, K. S., Klein-Packard, J., & McMurray, B. (2021). The pictures who shall not be named: Em-
892 pirical support for benefits of preview in the visual world paradigm. *Journal of Memory and Language*,
893 121, 104279. <https://doi.org/10.1016/j.jml.2021.104279>
- 894 Barrett, M. (2021). *Ggokabeito: 'Okabe-ito' scales for 'ggplot2' and 'ggraph'*. <https://CRAN.R-project.org/package=ggokabeito>
- 895 Bianco, R., Mills, G., Kerangal, M. de, Rosen, S., & Chait, M. (2021). Reward enhances online participants'
896 engagement with a demanding auditory task. *Trends in Hearing*, 25, 23312165211025941. <https://doi.org/10.1177/23312165211025941>
- 897 Blascheck, T., Kurzhals, K., Raschke, M., Burch, M., Weiskopf, D., & Ertl, T. (2017). Visualization of Eye
898 Tracking Data: A Taxonomy and Survey. *Computer Graphics Forum*, 36(8), 260–284. <https://doi.org/>

902 10.1111/cgf.13079

903 Blasi, D. E., Henrich, J., Adamou, E., Kemmerer, D., & Majid, A. (2022). Over-reliance on english hinders
904 cognitive science. *Trends in Cognitive Sciences*, 26(12), 1153–1170. <https://doi.org/10.1016/j.tics.2022.09.015>

905 Bogdan, P. C., Dolcos, S., Buetti, S., Lleras, A., & Dolcos, F. (2024). Investigating the suitability of
906 online eye tracking for psychological research: Evidence from comparisons with in-person data using
907 emotion–attention interaction tasks. *Behavior Research Methods*, 56(3), 2213–2226. <https://doi.org/10.3758/s13428-023-02143-z>

908 910 Boxtel, W. S. van, Linge, M., Manning, R., Haven, L. N., & Lee, J. (2024). Online eye tracking for aphasia:
911 A feasibility study comparing web and lab tracking and implications for clinical use. *Brain and Behavior*,
912 14(11), e70112. <https://doi.org/10.1002/brb3.70112>

913 Bramlett, A. A., & Wiener, S. (2024). The art of wrangling. *Linguistic Approaches to Bilingualism*.
914 [https://doi.org/https://doi.org/10.1075/lab.23071.bra](https://doi.org/10.1075/lab.23071.bra)

915 Bramlett, A. A., & Wiener, S. (2025). Individual differences modulate prediction of Italian words based on
916 lexical stress: a close replication and LASSO extension of Sulpizio and McQueen (2012). *Journal of*
917 *Cultural Cognitive Science*, 9(1), 55–81. <https://doi.org/10.1007/s41809-024-00162-6>

918 Brysbaert, M., & Stevens, M. (2018). Power analysis and effect size in mixed effects models: A tutorial.
919 *Journal of Cognition*, 1(1). <https://doi.org/10.5334/joc.10>

920 Bylund, E., Khafif, Z., & Berghoff, R. (2024). Linguistic and geographic diversity in research on second
921 language acquisition and multilingualism: An analysis of selected journals. *Applied Linguistics*, 45(2),
922 308–329. <https://doi.org/10.1093/applin/amad022>

923 Carter, B. T., & Luke, S. G. (2020). Best practices in eye tracking research. *International Journal of Psy-*
924 *chophysiology*, 155, 49–62. <https://doi.org/10.1016/j.ijpsycho.2020.05.010>

925 Chen-Sankey, J., Elhabashy, M., Gratale, S., Geller, J., Mercincavage, M., Strasser, A. A., Delnevo, C. D.,
926 Jeong, M., & Wackowski, O. A. (2023). Examining Visual Attention to Tobacco Marketing Materials
927 Among Young Adult Smokers: Protocol for a Remote Webcam-Based Eye-Tracking Experiment. *JMIR*
928 *Research Protocols*, 12, e43512. <https://doi.org/10.2196/43512>

929 Colby, S. E., & McMurray, B. (2023). Efficiency of spoken word recognition slows across the adult lifespan.
930 *Cognition*, 240, 105588. <https://doi.org/10.1016/j.cognition.2023.105588>

931 Cooper, R. M. (1974). The control of eye fixation by the meaning of spoken language: A new methodol-
932 ogy for the real-time investigation of speech perception, memory, and language processing. *Cognitive*
933 *Psychology*, 6(1), 84–107. [https://doi.org/10.1016/0010-0285\(74\)90005-X](https://doi.org/10.1016/0010-0285(74)90005-X)

934 Coretta, S., & Casillas, J. V. (2024). A tutorial on generalised additive mixed effects models for bilingualism
935 research. *Linguistic Approaches to Bilingualism*. <https://doi.org/10.1075/lab.23076.cor>

936 Corporation, M., & Weston, S. (2022). *doParallel: Foreach parallel adaptor for the 'parallel' package*.
937 <https://CRAN.R-project.org/package=doParallel>

938 Csárdi, G., Hester, J., Wickham, H., Chang, W., Morgan, M., & Tenenbaum, D. (2024). *Remotes: R pack-*
939 *age installation from remote repositories, including 'GitHub'*. <https://CRAN.R-project.org/package=remotes>

941 Dahan, D., Magnuson, J. S., & Tanenhaus, M. K. (2001). Time course of frequency effects in spoken-word

- 942 recognition: Evidence from eye movements. *Cognitive Psychology*, 42(4), 317–367. <https://doi.org/10.1006/cogp.2001.0750>
- 943
- 944 Degen, J., Kursat, L., & Leigh, D. D. (2021). Seeing is believing: Testing an explicit linking assumption
945 for visual world eye-tracking in psycholinguistics. *Proceedings of the Annual Meeting of the Cognitive
946 Science Society*, 43.
- 947 Dolstra, E., & contributors, T. N. (2023). *Nix* (Version 2.15.3) [Computer software]. <https://nixos.org/>
- 948 Eberhard, K. M., Spivey-Knowlton, M. J., Sedivy, J. C., & Tanenhaus, M. K. (1995). Eye movements as a
949 window into real-time spoken language comprehension in natural contexts. *Journal of Psycholinguistic
950 Research*, 24(6), 409–436. <https://doi.org/10.1007/BF02143160>
- 951 Faul, F., Erdfelder, E., Lang, A.-G., & Buchner, A. (2007). G*Power 3: A flexible statistical power analysis
952 program for the social, behavioral, and biomedical sciences. *Behavior Research Methods*, 39(2), 175–
953 191. <https://doi.org/10.3758/BF03193146>
- 954 Fields, E. C., & Kuperberg, G. R. (2019). Having your cake and eating it too: Flexibility and power with
955 mass univariate statistics for ERP data. *Psychophysiology*. <https://doi.org/10.1111/psyp.13468>
- 956 Firke, S. (2023). *Janitor: Simple tools for examining and cleaning dirty data*. <https://CRAN.R-project.org/package=janitor>
- 957
- 958 Frossard, J., & Renaud, O. (2021). *Permutation tests for regression, {ANOVA}, and comparison of signals:
959 The {permuco} package*. 99. <https://doi.org/10.18637/jss.v099.i15>
- 960 Geller, J., & Prystauka, Y. (2024). *webgazeR: Tools for processing webcam eye tracking data*. <https://github.com/jgeller112/webgazeR>
- 961
- 962 Godfroid, A., Finch, B., & Koh, J. (2024). Reporting Eye-Tracking Research in Second Language Ac-
963 quisition and Bilingualism: A Synthesis and Field-Specific Guidelines. *Language Learning*, n/a(n/a).
964 <https://doi.org/10.1111/lang.12664>
- 965 Green, P., & MacLeod, C. J. (2016). SIMR: an R package for power analysis of generalized linear mixed
966 models by simulation. *Methods in Ecology and Evolution*, 7(4), 493–498. <https://doi.org/10.1111/2041-210X.12504>
- 967
- 968 Henrich, J., Heine, S. J., & Norenzayan, A. (2010). Most people are not WEIRD. *Nature*, 466(7302), 29.
969 <https://doi.org/10.1038/466029a>
- 970 Hooge, I. T. C., Hessels, R. S., Niehorster, D. C., Andersson, R., Skrok, M. K., Konklewski, R., Stemplewski,
971 P., Nowakowski, M., Tamborski, S., Szkulmowska, A., Szkulmowski, M., & Nyström, M. (2024). Eye
972 tracker calibration: How well can humans refixate a target? *Behavior Research Methods*, 57(1), 23.
973 <https://doi.org/10.3758/s13428-024-02564-4>
- 974 Hopp, H. (2013). Grammatical gender in adult L2 acquisition: Relations between lexical and syntactic
975 variability. *Second Language Research*, 29(1), 33–56. <https://doi.org/10.1177/0267658312461803>
- 976 Huang, Y., & Snedeker, J. (2020). Evidence from the visual world paradigm raises questions about unac-
977 cusativity and growth curve analyses. *Cognition*, 200, 104251. <https://doi.org/10.1016/j.cognition.2020.104251>
- 978
- 979 Huettig, F., & McQueen, J. M. (2007). The tug of war between phonological, semantic and shape information
980 in language-mediated visual search. *Journal of Memory and Language*, 57(4), 460–482. <https://doi.org/10.1016/j.jml.2007.02.001>
- 981

- 982 Huettig, F., Rommers, J., & Meyer, A. S. (2011). Using the visual world paradigm to study language pro-
983 cessing: a review and critical evaluation. *Acta Psychologica*, 137(2), 151–171. <https://doi.org/10.1016/j.actpsy.2010.11.003>
- 985 Ito, A., & Knoeferle, P. (2023). Analysing data from the psycholinguistic visual-world paradigm: Compar-
986 ison of different analysis methods. *Behavior Research Methods*, 55(7), 3461–3493. <https://doi.org/10.3758/s13428-022-01969-3>
- 988 Ito, A., Pickering, M. J., & Corley, M. (2018). Investigating the time-course of phonological prediction in
989 native and non-native speakers of english: A visual world eye-tracking study. *Journal of Memory and*
990 *Language*, 98, 1–11. <https://doi.org/10.1016/j.jml.2017.09.002>
- 991 Izura, C., Cuetos, F., & Brysbaert, M. (2014). Lextale-Esp: a test to rapidly and efficiently assess the Spanish
992 vocabulary size. *PSICOLOGICA*, 35(1), 49–66. <http://hdl.handle.net/1854/LU-5774107>
- 993 Ju, M., & Luce, P. A. (2004). Falling on sensitive ears: Constraints on bilingual lexical activation. *Psycho-
994 logical Science*, 15(5), 314–318. <https://doi.org/10.1111/j.0956-7976.2004.00675.x>
- 995 Kaduk, T., Goeke, C., Finger, H., & König, P. (2024). Webcam eye tracking close to laboratory standards:
996 Comparing a new webcam-based system and the EyeLink 1000. *Behavior Research Methods*, 56(5),
997 5002–5022. <https://doi.org/10.3758/s13428-023-02237-8>
- 998 Kamide, Y., Altmann, G. T. M., & Haywood, S. L. (2003). The time-course of prediction in incremental
999 sentence processing: Evidence from anticipatory eye movements. *Journal of Memory and Language*,
1000 49(1), 133–156. [https://doi.org/10.1016/S0749-596X\(03\)00023-8](https://doi.org/10.1016/S0749-596X(03)00023-8)
- 1001 Kret, M. E., & Sjak-Shie, E. E. (2018). Preprocessing pupil size data: Guidelines and code. *Behavior
1002 Research Methods*, 1–7. <https://doi.org/10.3758/s13428-018-1075-y>
- 1003 Kumle, L., Võ, M. L.-H., & Draschkow, D. (2021). Estimating power in (generalized) linear mixed models:
1004 An open introduction and tutorial in R. *Behavior Research Methods*, 53(6), 2528–2543. [https://doi.org/10.3758/s13428-021-01546-0](https://doi.org/
10.3758/s13428-021-01546-0)
- 1006 Lakens, D. (2022). Sample size justification. *Collabra: Psychology*, 8(1). [https://doi.org/10.1525/collabra.33267](https://doi.org/10.1525/collabra.
1007 33267)
- 1008 Leeuw, J. R. de. (2015). jsPsych: A JavaScript library for creating behavioral experiments in a Web browser.
1009 *Behavior Research Methods*, 47(1), 1–12. <https://doi.org/10.3758/s13428-014-0458-y>
- 1010 Madsen, J., Júlio, S. U., Gucik, P. J., Steinberg, R., & Parra, L. C. (2021). Synchronized eye movements
1011 predict test scores in online video education. *Proceedings of the National Academy of Sciences*, 118(5),
1012 e2016980118. <https://doi.org/10.1073/pnas.2016980118>
- 1013 Magnuson, J. S., Dixon, J. A., Tanenhaus, M. K., & Aslin, R. N. (2007). The Dynamics of Lexical Com-
1014 petition During Spoken Word Recognition. *Cognitive Science*, 31(1), 133–156. [https://doi.org/10.1080/03640210709336987](https://doi.org/10.1080/
1015 03640210709336987)
- 1016 Maris, E., & Oostenveld, R. (2007). Nonparametric statistical testing of EEG- and MEG-data. *Journal of
1017 Neuroscience Methods*, 164(1), 177–190. <https://doi.org/10.1016/j.jneumeth.2007.03.024>
- 1018 McMurray, B., Samelson, V. M., Lee, S. H., & Tomblin, J. B. (2010). Individual differences in online
1019 spoken word recognition: Implications for SLI. *Cognitive Psychology*, 60(1), 1–39. [https://doi.org/10.1016/j.cogpsych.2009.06.003](https://doi.org/10.
1020 1016/j.cogpsych.2009.06.003)
- 1021 McMurray, B., Tanenhaus, M. K., & Aslin, R. N. (2002). Gradient effects of within-category phonetic vari-

- 1022 ation on lexical access. *Cognition*, 86(2), B33–B42. [https://doi.org/10.1016/S0010-0277\(02\)00157-9](https://doi.org/10.1016/S0010-0277(02)00157-9)
- 1023 Meyer, M., Lamers, D., Kayhan, E., Hunnius, S., & Oostenveld, R. (2021). Enhancing reproducibility in
1024 developmental EEG research: BIDS, cluster-based permutation tests, and effect sizes. *Developmental
1025 Cognitive Neuroscience*, 52, 101036. <https://doi.org/10.1016/j.dcn.2021.101036>
- 1026 Microsoft, & Weston, S. (2022). *ForEach*: Provides foreach looping construct. [https://CRAN.R-project.org/
1027 package=foreach](https://CRAN.R-project.org/package=foreach)
- 1028 Miller, J. (2023). Outlier exclusion procedures for reaction time analysis: The cures are generally worse
1029 than the disease. *Journal of Experimental Psychology: General*, 152(11), 3189–3217. [https://doi.org/
1030 10.1037/xge0001450](https://doi.org/10.1037/xge0001450)
- 1031 Milne, A. E., Zhao, S., Tampakaki, C., Bury, G., & Chait, M. (2021). Sustained pupil responses are
1032 modulated by predictability of auditory sequences. *The Journal of Neuroscience*, 41(28), 6116–6127.
1033 <https://doi.org/10.1523/JNEUROSCI.2879-20.2021>
- 1034 Mirman, D., & Graziano, K. M. (2012). Individual differences in the strength of taxonomic versus the-
1035 matic relations. *Journal of Experimental Psychology: General*, 141(4), 601–609. [https://doi.org/
1036 10.1037/a0026451](https://doi.org/10.1037/a0026451)
- 1037 Müller, K. (2020). *Here*: A simpler way to find your files. <https://CRAN.R-project.org/package=here>
- 1038 Niedermann, J. P., Sucholutsky, I., Marjeh, R., Çelen, E., Griffiths, T., Jacoby, N., & Rijn, P. van. (2024).
1039 Studying the Effect of Globalization on Color Perception using Multilingual Online Recruitment and
1040 Large Language Models. *Proceedings of the Annual Meeting of the Cognitive Science Society*, 46(0).
1041 <https://escholarship.org/uc/item/4hs755zz>
- 1042 Özsöy, O., Çiçek, B., Özal, Z., Gagarina, N., & Sekerina, I. A. (2023). Turkish-german heritage speakers'
1043 predictive use of case: Webcam-based vs. In-lab eye-tracking. *Frontiers in Psychology*, 14, 1155585.
1044 <https://doi.org/10.3389/fpsyg.2023.1155585>
- 1045 Papoutsaki, A., Sangkloy, P., Laskey, J., Daskalova, N., Huang, J., & Hays, J. (2016). *Webgazer: Scalable
1046 webcam eye tracking using user interactions*. 38393845.
- 1047 Patterson, A. S., & Nicklin, C. (2023). L2 self-paced reading data collection across three contexts: In-person,
1048 online, and crowdsourcing. *Research Methods in Applied Linguistics*, 2(1), 100045. [https://doi.org/
1049 10.1016/j.rmal.2023.100045](https://doi.org/10.1016/j.rmal.2023.100045)
- 1050 Peelle, J. E., & Van Engen, K. J. (2021). Time stand still: Effects of temporal window selection on eye
1051 tracking analysis. *Collabra: Psychology*, 7(1), 25961. <https://doi.org/10.1525/collabra.25961>
- 1052 Peirce, J., Gray, J. R., Simpson, S., MacAskill, M., Höchenberger, R., Sogo, H., Kastman, E., & Lindeløv,
1053 J. K. (2019). PsychoPy2: Experiments in behavior made easy. *Behavior Research Methods*, 51(1), 195–
1054 203. <https://doi.org/10.3758/s13428-018-01193-y>
- 1055 Peterson, R. J. (2021). We need to address ableism in science. *Molecular Biology of the Cell*, 32(7), 507–510.
1056 <https://doi.org/10.1091/mbc.E20-09-0616>
- 1057 Płużyczka, M. (2018). The First Hundred Years: a History of Eye Tracking as a Research Method.
1058 *Applied Linguistics Papers*, 25/4, 101–116. [http://cejsh.icm.edu.pl/cejsh/element/bwmeta1.element.
desklight-98576d43-39e3-4981-8c1c-717962cf29da](http://cejsh.icm.edu.pl/cejsh/element/bwmeta1.element.
1059 desklight-98576d43-39e3-4981-8c1c-717962cf29da)
- 1060 Prystauka, Y., Altmann, G. T. M., & Rothman, J. (2024). Online eye tracking and real-time sentence pro-
1061 cessing: On opportunities and efficacy for capturing psycholinguistic effects of different magnitudes and

- 1062 diversity. *Behavior Research Methods*, 56(4), 3504–3522. <https://doi.org/10.3758/s13428-023-02176-4>
- 1063 R Core Team. (2024). *R: A language and environment for statistical computing* (Version 4.4.2). R Foundation for Statistical Computing. <https://www.R-project.org/>
- 1065 Rodd, J. M. (2024). Moving experimental psychology online: How to obtain high quality data when we
1066 can't see our participants. *Journal of Memory and Language*, 134, 104472. <https://doi.org/10.1016/j.jml.2023.104472>
- 1068 Rodrigues, B., & Baumann, P. (2025). *Rix: Reproducible data science environments with 'nix'*. <https://docs.ropensci.org/rix/>
- 1070 Rossi, E., Krass, K., & Kootstra, G. J. (2019). *Psycholinguistic Methods in Multilingual Research* (pp. 75–
1071 99). John Wiley & Sons, Ltd. <https://doi.org/10.1002/9781119387725.ch4>
- 1072 Sarrett, M. E., Shea, C., & McMurray, B. (2022). Within- and between-language competition in adult second
1073 language learners: Implications for language proficiency. *Language, Cognition and Neuroscience*, 37(2),
1074 165–181. <https://doi.org/10.1080/23273798.2021.1952283>
- 1075 Semmelmann, K., & Weigelt, S. (2018). Online webcam-based eye tracking in cognitive science: A first
1076 look. *Behavior Research Methods*, 50(2), 451–465. <https://doi.org/10.3758/s13428-017-0913-7>
- 1077 Slim, M. S., & Hartsuiker, R. J. (2023). Moving visual world experiments online? A web-based replication
1078 of Dijkgraaf, Hartsuiker, and Duyck (2017) using PCIbex and WebGazer.js. *Behavior Research Methods*,
1079 55(7), 3786–3804. <https://doi.org/10.3758/s13428-022-01989-z>
- 1080 Slim, M. S., Kandel, M., Yacovone, A., & Snedeker, J. (2024). Webcams as windows to the mind? A direct
1081 comparison between in-lab and web-based eye-tracking methods. *Open Mind*, 8, 1369–1424. https://doi.org/10.1162/opmi_a_00171
- 1083 Spivey, M. J., & Marian, V. (1999). Cross talk between native and second languages: Partial activation of
1084 an irrelevant lexicon. *Psychological Science*, 10(3), 281–284. <https://doi.org/10.1111/1467-9280.00151>
- 1085 Stone, K., Lago, S., & Schad, D. J. (2021). Divergence point analyses of visual world data: applications
1086 to bilingual research. *Bilingualism: Language and Cognition*, 24(5), 833–841. <https://doi.org/10.1017/S1366728920000607>
- 1088 Tanenhaus, M. K., Spivey-Knowlton, M. J., Eberhard, K. M., & Sedivy, J. C. (1995). Integration of visual
1089 and linguistic information in spoken language comprehension. *Science (New York, N.Y.)*, 268(5217),
1090 1632–1634. <http://www.ncbi.nlm.nih.gov/pubmed/7777863>
- 1091 Trueswell, J. C. (2008). *Using eye movements as a developmental measure within psycholinguistics* (I. A.
1092 Sekerina, E. M. Fernández, & H. Clahsen, Eds.; pp. 73–96). John Benjamins Publishing Company.
1093 <https://doi.org/10.1075/lald.44.05tru>
- 1094 Van der Cruyssen, I., Ben-Shakhar, G., Pertzov, Y., Guy, N., Cabooter, Q., Gunschera, L. J., & Verschuere,
1095 B. (2024). The validation of online webcam-based eye-tracking: The replication of the cascade effect,
1096 the novelty preference, and the visual world paradigm. *Behavior Research Methods*, 56(5), 4836–4849.
1097 <https://doi.org/10.3758/s13428-023-02221-2>
- 1098 Viviani, P. (1990). Eye movements in visual search: cognitive, perceptual and motor control aspects. *Reviews
1099 of Oculomotor Research*, 4, 353–393.
- 1100 Voeten, C. C. (2023). *Permutest: Permutation tests for time series data*. <https://CRAN.R-project.org/package=permutes>

- 1102 Vos, M., Minor, S., & Ramchand, G. C. (2022). Comparing infrared and webcam eye tracking in the Visual
1103 World Paradigm. *Glossa Psycholinguistics*, 1(1). <https://doi.org/10.5070/G6011131>
- 1104 Wickham, H. (2017). *Tidyverse: Easily install and load the 'tidyverse'*. <https://CRAN.R-project.org/package=tidyverse>
- 1106 Yee, E., Blumstein, S., & Sedivy, J. C. (2008). Lexical-semantic activation in broca's and wernicke's aphasia:
1107 Evidence from eye movements. *Journal of Cognitive Neuroscience*, 20(4), 592–612. <https://doi.org/10.1162/jocn.2008.20056>
- 1109 Zehr, J., & Schwarz, F. (2018). *PennController for internet based experiments (IBEX)*. <https://doi.org/10.17605/OSF.IO/MD832>
- 1110