

UC Irvine

UC Irvine Electronic Theses and Dissertations

Title

Exploring User Interaction with Modern CAPTCHAs

Permalink

<https://escholarship.org/uc/item/6rc989dw>

Author

SEARLES, ANDREW

Publication Date

2024

Peer reviewed|Thesis/dissertation

UNIVERSITY OF CALIFORNIA,
IRVINE

Exploring User Interaction with Modern CAPTCHAs

DISSERTATION

submitted in partial satisfaction of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

in Computer Science

by

Andrew Searles

Dissertation Committee:
Professor Gene Tsudik, Chair
Professor Ardalan Amiri Sani
Professor Alfred Chen

2024

DEDICATION

To science and humanity, we will discover more about reality.

TABLE OF CONTENTS

	Page
LIST OF FIGURES	vi
LIST OF TABLES	viii
ACKNOWLEDGMENTS	x
VITA	xi
ABSTRACT OF THE DISSERTATION	xiii
1 Introduction	1
1.1 Personal Contributions	5
1.2 Dissertation Structure	7
2 An Empirical Study & Evaluation of Modern CAPTCHAs	8
2.1 Introduction	10
2.2 Research Questions & Main Findings	12
2.3 Website Inspection	14
2.3.1 Results and analysis	15
2.3.2 Potential limitations	16
2.4 User Study	18
2.4.1 Choice of CAPTCHAs	20
2.4.2 Direct vs. contextualized settings	21
2.4.3 Timeline and compensation	22
2.4.4 Ethical considerations	23
2.4.5 User study implementation	23
2.4.6 Potential limitations	24
2.5 Results & Analysis	26
2.5.1 Solving times	26
2.5.2 Preferences analysis	27
2.5.3 Direct vs. contextualized setting	28
2.5.4 Effects of demographics	29
2.5.5 Accuracy of CAPTCHAs	30
2.6 Measuring User Abandonment	37
2.7 Related Work	38

2.7.1	Comparison of methodologies	38
2.7.2	Detailed comparisons	39
2.7.3	Summarized comparisons	41
2.8	Abandonment measurement	45
2.9	Questions asked in User Study	46
2.10	Statistical Analysis of Solving Times	47
2.11	CAPTCHA Solving Times for Other Demographic Features	49
2.12	Summary & Future Work	50
3	Understanding reCAPTCHA v2 via a Large-Scale Live User Study	52
3.1	Introduction	54
3.2	Background	57
3.3	The User Study	59
3.3.1	The Setting	59
3.3.2	Justification	60
3.3.3	The Website	62
3.3.4	Directory Crawler	63
3.3.5	Logistics & Data Cleaning	63
3.3.6	Post-Study Survey	64
3.3.7	Ethical Considerations	64
3.4	Results & Analysis	65
3.4.1	University Demographics	65
3.4.2	reCAPTCHA v2 Dashboard Data	66
3.4.3	Solving Time	67
3.4.4	Survey Results	73
3.5	Comparison with Related Work	77
3.6	Discussion	80
3.6.1	Cost Analysis	80
3.6.2	Security Analysis	82
3.6.3	reCAPTCHA v2	82
3.6.4	reCAPTCHA v3	85
3.7	Statistical Testing	85
3.8	Workflow and Additional Figures	90
3.8.1	Account Creation	90
3.8.2	Password Recovery	91
3.9	Network Analysis of reCAPTCHA v2	94
3.9.1	Page load Latency	95
3.9.2	Checkbox Click Overhead	96
3.9.3	reCAPTCHA Image load Overhead	97
3.9.4	Image Solution Verification Overhead	97
3.9.5	reCAPTCHA Expiration Overhead	98
3.10	Summary	99

4	Exploring CAPTCHA-induced Task Abandonment with QUITCHA	101
4.1	Introduction	103
4.2	Research Questions & Results	107
4.3	System Design & Implementation	110
4.3.1	Dataset	110
4.3.2	Challenges	111
4.3.3	Back-End	112
4.3.4	Front-End	113
4.3.5	Direct vs Contextualized	113
4.3.6	QUITCHA	114
4.3.7	Survey	115
4.4	Study Logistics	116
4.4.1	Timeline and Logistics	116
4.4.2	Pilot Study & The MTurk Botnet	117
4.4.3	Ethical Considerations	118
4.5	Results and Analysis	118
4.5.1	Data Logistics and Cleaning	118
4.5.2	Timing results	119
4.5.3	Checkbox Solution Time	119
4.5.4	Image Selection Solution Timing	122
4.5.5	Accuracy of Image Selection	131
4.5.6	Statistical Testing Methodology	134
4.5.7	Quit Analysis	135
4.5.8	Survey Analysis	136
4.6	Related Work	142
4.6.1	Detailed Comparisons	143
4.7	Accuracy, Statistical, Timing, and Survey Figures	145
4.8	Summary	149
5	Final Remarks	150
	Bibliography	152

LIST OF FIGURES

	Page
2.1 Discrete distribution of discovered CAPTCHAS (full data available in the accompanying dataset).	17
2.2 reCAPTCHA [33, 32, 34]	19
2.3 Arkose Labs [23]	19
2.4 Geetest [26]	19
2.5 hCAPTCHA [29]	19
2.6 Distorted text CAPTCHAS	20
2.7 Solving times for various types of CAPTCHAS. Boxes show the middle 50% of participants, and whiskers show the filtered range. Black vertical lines show the median.	32
2.8 Participant-reported preference scores for different types of CAPTCHAS, sorted from highest to lowest.	33
2.9 CAPTCHA solving times for direct (D) vs. contextualized (C) user study settings. The horizontal axis shows solving time in seconds, quantized into one-second buckets, and the vertical axis shows number of participants.	34
2.10 Effects of age in CAPTCHA solving time. The horizontal axis shows the age and the vertical axis shows the solving time. The red line shows the linear fit of the data points and the green line shows the average solving time per age.	34
2.11 Effects of device type.	35
2.12 Effects of typical Internet use.	36
2.13 Effects of Gender.	49
2.14 Effects of Education Level.	50
3.1 reCAPTCHA v2 checkbox CAPTCHA [33]	58
3.2 reCAPTCHA v2 Image Labeling Task CAPTCHA [32]	59
3.3 Timing results in bins of .1 seconds	67
3.4 Image timing results in bins of .2 seconds	69
3.5 Preference score for checkbox only scenario	75
3.6 Preference score for checkbox+image scenario	75
3.7 Word cloud from feedback on checkbox	76
3.8 Word cloud from feedback on image	77
3.9 Kruskal-Wallis results for checkbox attempts	87
3.10 Kruskal-Wallis results for total attempts and educational level	88
3.11 Kruskal-Wallis results for total attempts and major	89

3.12	Initial login page	90
3.13	Initial Account Creation Page	90
3.14	Account creation form	91
3.15	AC form after clicking submit	91
3.16	Password Recovery form	92
3.17	Password Recovery form after clicking submit	93
4.1	Distribution of image types in the dataset [28]	110
4.2	Side by side comparison of reCAPTCHA v2 and QUITCHA	111
4.3	Front-End workflow for participants	112
4.4	Challenge completion heuristic	115
4.5	QUITCHA completion heuristic	115
4.6	The Account Creation Form	116
4.7	Checkbox solution time in .2 second bins	120
4.8	A comparison of checkbox solution time by context	121
4.9	Image selection solution duration by context	123
4.10	Solving time groups separated by true type.	125
4.11	Solving time groups separated by quit	126
4.12	Solving time distribution across attempts separated by quitting (Image solution duration in seconds on x-axis)	128
4.13	Inter-selection time groups by true image type	131
4.14	Accuracy across true types	133
4.15	Accuracy across false types	133
4.16	Accuracy across quitters	134
4.17	Age distribution of participants	137
4.18	Effect of priming on opinion about CAPTCHA	139
4.19	Factors behind CAPTCHAS task abandonment	140
4.20	Effect of solution confidence in abandonment	141
4.21	Accuracy across context	145
4.22	Accuracy across subsequent challenges	145
4.23	ANOVA results comparing true type	146
4.24	ANOVA results for inter selection averages across true types	147
4.25	Solving time groups separated by accuracy.	148
4.26	Word Cloud from Task Completion Motivation	148

LIST OF TABLES

	Page
2.1 Summary of research questions and main findings.	14
2.2 Summary of demographic data for the 1,400 participants of the main user study.	22
2.3 Humans vs. bot solving time (seconds) and accuracy (percentage) for different CAPTCHA types.	31
2.4 Agreement for distorted text CAPTCHAS.	31
2.5 Methodology and details of previous CAPTCHA-related user studies.	43
2.6 Comparison of results from prior user studies evaluating CAPTCHAS: audio (A), behavior (B), distorted text (DT), game (G), honeypot (HP), image (I), math (M), service (S), slider (SL), video (V) and newly-proposed (New). Some studies used non-unique (NU) participants or MTurk (MT). * denotes reimplemented CAPTCHA types.	44
2.7 Abandonment in contextualized setting (\$0.75 payment)	45
2.8 Abandonment in contextualized setting (\$1.50 payment)	45
2.9 Abandonment in direct setting (\$0.30 payment)	46
2.10 Abandonment in direct setting (\$0.60 payment)	46
2.11 Questions in user study	47
3.1 Google’s reCAPTCHA dashboard data	66
3.2 Agglomerated solving time for reCAPTCHA Mode	67
3.3 Checkbox solving time in seconds for each service	68
3.4 Image solving time in seconds for each service	68
3.5 Total solving time in seconds for each service	70
3.6 Solving time for number of checkbox attempts	71
3.7 Solving time for number of image attempts	71
3.8 Total solving time for different educational levels	72
3.9 Total solving time for various majors	73
3.10 SUS Scores for reCAPTCHA v2	74
3.11 Comparison with results from prior user studies evaluating reCAPTCHA v2: checkbox (C), image (I), total (T). Mean in seconds	78
3.12 Humans vs. bot solving time (seconds) and accuracy (percentage) for reCAPTCHA v2.	82
3.13 Notation Summary	94
3.14 reCAPTCHA API Calls during page load	95

3.15	recaptcha.html load network overhead	96
3.16	recaptcha.html load latency	96
3.17	reCAPTCHA API Calls after checkbox click	97
3.18	reCAPTCHA API Calls for image load	97
3.19	reCAPTCHA API Calls for correct image solution	98
3.20	reCAPTCHA API Calls for reCAPTCHA expiration	98
3.21	Summary of reCAPTCHA Network Overhead	99
4.1	Summary of research questions and main findings.	108
4.2	Image selection solution timing overview	122
4.3	Image selection solution time by context	122
4.4	Image selection solution time by the true type	124
4.5	Image selection solution time by the false type	125
4.6	Image selection solution time separated by quitting	126
4.7	Image selection solution time by the heuristic result	126
4.8	Image selection solution time by the challenge attempt number	127
4.9	Overview of the data from selection time	129
4.10	Inter selection time by the true type	130
4.11	Inter selection time by separated by heuristic result	130
4.12	Inter selection time separated by quit	130
4.13	Overview of selections per 14k challenges	132
4.14	Percentage of true/false type selected	132
4.15	Comparison of related work results. Time in average seconds.	144

ACKNOWLEDGMENTS

First and foremost, I would like to thank my Ph.D. advisor, Professor Gene Tsudik, for teaching me how to do research and being an amazing mentor in life, computer science, and security. Thank you again for this wonderful opportunity to work and live at the University of California Irvine.

Also, I would like to thank my committee members, Professor Ardalan Amiri Sani and Professor Qi Alfred Chen. Thanks for teaching me so much about computer science and security over the years and for helping and providing insight into accomplishing my PhD.

I would like to express gratitude to my early mentors, Christian Duncan, Michael, and Kyle Levi. I would not have begun this journey without their help and recommendations for admissions.

Special thanks to SPROUT, the Security and Privacy Research Outfit, for being such cool labmates throughout the years: Yoshimichi Nakatsuka, Ivan De Oliveira Nunes, Norrathep Rattanaivanon, Ercan Ozturk, Seo Yeon Hwang, Sashidhar Jakkamsetti, Benjamin Turner, Youngil Kim, Renascence Tarafder Prapty, Elina Van Kempen, Pavel Frolikov, Isita Bagayatkar and the mighty Gene Tsudik. Thanks for being great people to work with and train with.

Thank you to my family and friends for being supportive and proud of my endeavors. Keep up the journey of self-growth and improvement!

Portion of Chapter 2 is a reprint of the material as it appears in the Proceedings of the 32nd USENIX Security Symposium (USENIX Security 2023) [94], used with permission from the USENIX Association. The co-authors listed in this publication are Doctor Yoshimichi Nakatsuka, Doctor Ercan Ozturk, Doctor Andrew Paverd, Ai Enkoji, and Professor Gene Tsudik.

Financial support was provided by Gene Tsudik in conjunction with the University of California, Irvine and UCI School of ICS.

VITA

Andrew Searles

EDUCATION

Doctor of Philosophy in Computer Science University of California, Irvine	2024 <i>Irvine, California</i>
Master of Science in Computer Science University of California, Irvine	2022 <i>Irvine, California</i>
Bachelor of Science in Computer Science Quinnipiac University	2016 <i>Hamden, Connecticut</i>

RESEARCH EXPERIENCE

Graduate Research Assistant University of California, Irvine	2020–2024 <i>Irvine, California</i>
--	---

TEACHING EXPERIENCE

TA for Programming In $C++$ (ICS45C)	Spring 2024
TA for Computer & Network Security (CS 134)	Fall 2021
TA for Spreadsheets for Problem-Solving. (I&C SCI 7)	Spring 2021
TA for Computer Networks (CS 132/EECS 148)	Winter 2021
TA for Computer & Network Security (CS 134)	Fall 2020
TA for Computer & Network Security (CS 134)	Spring 2020
TA for Computer Networks (CS 132/EECS 148)	Winter 2020
TA for Boolean Algebra & Logic (ICS 6B)	Fall 2019
University of California, Irvine	<i>Irvine, California</i>

PROFESSIONAL EXPERIENCE

Cyber Security Intern / Associate Activision Central Tech	Summer 2022, Summer & Winter 2023 <i>Irvine, California</i>
Application Security Intern Blizzard Entertainment	Summer 2021 <i>Irvine, California</i>

Computer Security Software Development Intern
Edwards Lifesciences

Summer 2020
Irvine, California

Software Engineering Intern
MITRE

2014–2016
Burlington, Massachusetts

REFEREED CONFERENCE PUBLICATIONS

An Empirical Study & Evaluation of Modern CAPTCHAs
USENIX Security Symposium
Aug 2023

REFEREED JOURNAL PUBLICATIONS

Observing CAPTCHAS “in the Wild”
;login: Online
Aug 2023

PAPERS IN SUBMISSION OR UNDER REVIEW

Understanding reCAPTCHA_{v2} via a Large-Scale Live User Study
TBD
2024

EILID: Execution Integrity for Low-end IoT Devices
TBD
2024

Exploring CAPTCHA-Induced Task Abandonment with QUITCHA
TBD
2024

SOFTWARE

QUITCHA <https://github.com/sprout-uci/QUITCHA/>
Querying Unending Image Tasks to Create Human Annoyance. It is designed to measure fine-grained image selection CAPTCHA interaction via a full stack implementation of a mocked-up version of reCAPTCHA_{v2}.

ABSTRACT OF THE DISSERTATION

Exploring User Interaction with Modern CAPTCHAs

By

Andrew Searles

Doctor of Philosophy in Computer Science

University of California, Irvine, 2024

Professor Gene Tsudik, Chair

Currently, Turing tests (notably in the form of CAPTCHAs) are used as a claimed barrier against bots on the Internet. During the last 20+ years, both CAPTCHAs and attacks against them have evolved in sophistication, in the course of a seemingly endless “arms race”. Many problems that were considered as being hard for AI are now trivial to solve automatically, such as OCR and image labeling. Consequently, most types of modern CAPTCHAs have become ineffective. This prompts a natural question: Is it possible to remotely determine whether an interaction is performed by a human or a computer?

In this dissertation, we conduct three large-scale user studies of modern CAPTCHAs. The first investigates comparative performance of ten popular CAPTCHA types with 1,400 MTurk participants. The second study is a long-term (over a year long) experiment with Google’s reCAPTCHA v2 deployed in a real-world scenario with over 3,600 live and unaware participants. Finally, the third study – with over 1,400 participants – focuses on CAPTCHA-induced activity abandonment. Results from the three studies show that CAPTCHAs:

- are solved slower by humans than bots on average.
- have consumed billions of hours of human time.
- are generally disliked and cause users to quit.
- should be deprecated.

Chapter 1

Introduction

In 1950, Alan Turing wrote *Computing Machinery and Intelligence*, proposing the original question, “Can machines think?” [103]. Turing envisioned the imitation game, in which an interrogator would ask questions to accurately classify two distinct parties (A and B) who were deceptively trying to appear to be the other party. One of these parties would be played by a computer and the other by a human. However, the capabilities of computers have drastically evolved since the 1950s with an exponential increase in global computing power, availability, and interconnectedness. Computers are now an integral part of how humans think of, and solve problems.

In 2002, Von Anh et al. proposed the Completely Automated Public Turing Test to Tell Computers and Humans Apart (CAPTCHA), as a program that can generate and grade tests that: A) Most humans can pass but B) Current computer programs can not pass [41]. Von Anh claimed that “CAPTCHAS are the ultimate tool for stealing cycles from people:” as, “malicious, intelligent programmers” would have to solve hard artificial intelligence (AI) problems to bypass “security” measures. Original CAPTCHAS presented optical character recognition (OCR) problems, which asked users to transcribe images of written text to typed

digital text. The invention of CAPTCHAS introduced a feedback loop; if enough humans can accurately solve hard AI problems with a computer recording all their interactions, enough of these recordings could be used with machine learning to train a computer to solve the exact same problem. Thus, recording solutions of tests that satisfy property A (Most humans can pass) would inevitably lead to the falsification of property B (Current computer programs can not pass).

In 2008, Von Anh et al. realized this feedback loop with the invention of reCAPTCHA. The premise was that not only could reCAPTCHA steal cycles from intelligent programmers breaking it but the regular users could be turned into a free labor force in the name of security. Not long after, in late 2009, Google acquired reCAPTCHA [11] and, by 2010, the reCAPTCHA website reported over 100 million distinct daily users [12].

In 2010, Bursztein et al. [53] posed the question: How Good are Humans at Solving CAPTCHAS? The same paper [53] performed a large-scale two-part evaluation: (1) a user study with over 1,100 unique participants from Amazon Mechanical Turk (MTurk) [22] and (2) a study of underground CAPTCHA-breaking farms which are sweatshop-like operations where humans are paid to solve CAPTCHAS [85] as a service for bots. [53] showed that CAPTCHAS were often more difficult or took longer to solve than was expected. There was a loose correlation between time-to-annoyance and abandonment, with higher abandonment rates observed for CAPTCHAS that took longer to solve. The same study also showed several demographic trends, e.g., users outside the US typically took longer to solve English-language CAPTCHA schemes. However, since this study, the CAPTCHA ecosystem has changed substantially: new CAPTCHA types emerged, input methods evolved, and Web use boomed.

To date, there have been many CAPTCHA user studies [94, 53, 49, 65, 91, 104, 83, 77, 66, 78, 61, 60, 72, 102, 95]. They focused on various aspects, such as solving performance, usability, introduction of novel CAPTCHAS, user abandonment, and comparisons of CAPTCHA types. In this dissertation, we thoroughly investigate all of these aspects and more to answer the

following important question:

Should CAPTCHAS be used for security?

We explore user interaction with modern CAPTCHAS via three large-scale user studies with over 6000 unique participants and over 100,000 CAPTCHA solutions.

- The first study (An Empirical Study & Evaluation of Modern CAPTCHAS) asked five main research questions: (1) How long do human users take to solve different types of CAPTCHAS? (2) What CAPTCHA types do users prefer? (3) Does experimental context affect solving time? (4) Do demographics affect solving time? (5) Does experimental context influence abandonment? A 1,400 participant study was conducted on MTurk to answer these questions, with participants solving 10 CAPTCHAS each. Results show that: (1) Humans are slower than bots for known attacks on CAPTCHA types. (2) Users' preference is not fully correlated with CAPTCHA solving time. (3) Experimental context significantly influences CAPTCHA solving times. (4) Age impacts on solving time. (5) There are high abandonment rates due to CAPTCHA-related tasks.
- The second study (Understanding reCAPTCHA v2 via a Large-Scale Live User Study) investigates reCAPTCHA v2 using a real-world account creation and password recovery service with over 3,600 unbiased (unwitting) participants solving over 9,000 reCAPTCHA v2 challenges. This study makes three main contributions: (1) A comprehensive quantitative analysis of reCAPTCHA v2 solving time. (2) An in-depth qualitative analysis of reCAPTCHA v2 usability. (3) A detailed discussion of the cost and security of reCAPTCHA v2. Results show that: (1) Solving time is influenced by number of attempts, service/website setting, educational level, and field of study. (2) Using the SUS scale, the usability of reCAPTCHA v2 is rated 77/100 (checkbox only) and 59/100 (checkbox+image). (3) In terms of cost, we estimate that – during over 13 years of its deployment – 819 million hours of human time has been spent on reCAPTCHA. In addition,

Google has potentially profited USD \$888 billion from cookies and \$8.75-32.3 billion USD per each sale of its total labeled data set. In terms of security, reCAPTCHA v2 is susceptible to click-jacking (a blatant vulnerability) [73], trivial implementation of large-scale automation attacks [99], weak security premise of fallback (image challenge) [62, 99, 75]. Furthermore, it uses privacy-invasive tracking cookies (for security) [99]. We conclude that reCAPTCHA v2 and similar reCAPTCHA technology should be deprecated.

- The third study (Exploring CAPTCHA-Induced Task Abandonment with QUITCHA) asked five main questions:
 1. What is inside the “black box” of image labeling tasks?
 2. What makes users quit, i.e., what factors (if any) influence session abandonment induced by image labeling captchas?
 3. What behavioral conclusions stem from fine-grained event logging?
 4. How long do users take to solve reCAPTCHA v2-like image labeling tasks?
 5. What factors influence solving time and accuracy?

Results show that:

1. QUITCHA produces fine-grained timing and accuracy features that exhibit statistically significant trends.
2. Repeated image selection challenges produce significant quitting in the 26 - 41% range. Multiple features influence dropout rates.
3. Results show that exact accuracy, selections, clicks, timing, and function tracing allow for the recreation of user interaction and exhibit statistically significant results across certain features
4. We provide a large-scale fine-grained image solution dataset and analysis tool-set that quantifies various solution-related timing events.

5. New features are discovered with statistically significant trends.

We present an open-source alternative to reCAPTCHA v2 if the website or service operators wish to label their datasets and keep their user's data private.

After much investigation, it is clear that modern CAPTCHAS should not be used as a security tool. The original CAPTCHA paper was titled: CAPTCHAS: How Lazy Cryptographers Do AI. Ultimately, there is no place for laziness in security; this only leads to failure. CAPTCHAS should be deprecated as a security service for the following reasons:

- Defining property B of CAPTCHAS [41] (Current computer programs can not pass) has been broken for decades. A false sense of security is worse than no security at all.
- The nature of the feedback loop: if computers record humans solving a problem, enough recordings can be used to make a program that does the same thing. Using "hard" AI problems (that are easy for humans) will never provide any provable security.
- They come at an immense cost of human time to train AI models, which profit large corporations and do not pay users for their labor.
- There are secure alternatives for rate limiting, such as CACTI [86] and Scrappy [43].

1.1 Personal Contributions

Research is rarely an individual effort; in the case of this thesis, I was lucky to work with great researchers who taught me how to do research. Here are my contributions to the following projects:

- The first study (An Empirical Study & Evaluation of Modern CAPTCHAS) served as my introduction to research authorship. I started as a lowly process implementing portions of

code and discussing experimental design. The idea was partly conceived by Gene Tsudik, Andrew Paverd, Ercan Ozturk, and Yoshimichi Nakatsuka. However, publishing a paper is no easy task, and through many iterations, I eventually became involved in all aspects of the project, from coding to data analysis and writing. Gene Tsudik, Ercan Ozturk, and Andrew Paverd were great mentors to me in this project. Yoshimichi Nakatsuka was a great mentor and partner in this project, which was undoubtedly completed due to his hard work and collaborative nature. Through consistent iterative growth and effort, I became the first author by pushing this project over the finish line through much adversity.

- The second study (Understanding reCAPTCHA v2 via a Large-Scale Live User Study) was a planned case study envisioned by Gene Tsudik. I performed technical implementation of the reCAPTCHA v2 monitoring with the help of the CS IT department as the study was conducted on the school's account creation service, which the IT Department maintains. I also performed the data analysis and wrote a large majority of the paper. This was also an opportunity for me to lead and mentor Renasence Tarafder Prapty, who was responsible for the survey portion of the study (implementation and writing).
- The third study (Exploring CAPTCHA-Induced Task Abandonment with QUITCHA) was my own idea contextualized by the previous work; we wanted to discover what was truly happening inside the CAPTCHA black box. The entire stack of QUITCHA was 95% implemented by me, with a couple of features implemented by Renasence Tarafder Prapty. I mainly conducted all data analysis and writing, however my co-authors helped a lot with finalizing text. Gene Tsudik advised and mentored this project.

Ultimately, I could not have done this research without the help of my co-authors. To them, I express my deepest gratitude for helping me on this journey of growth and discovery.

1.2 Dissertation Structure

Following the Introduction, Chapter 2 presents the first result: An Empirical Study & Evaluation of Modern CAPTCHAS. It provides a great view into the landscape of modern CAPTCHAS and background information surrounding the state of CAPTCHA research. The second result, Chapter 3, presents an in-depth case study of reCAPTCHA_{v2} and the quantitative and qualitative results that help evaluate the cost and security of reCAPTCHA_{v2}. The third and final result, Exploring CAPTCHA-induced Task Abandonment with QUITCHA, Chapter 4, demonstrates what kind of data CAPTCHAS can obtain and how CAPTCHAS influence abandonment. Finally, we conclude with closing remarks in Chapter 5.

Chapter 2

An Empirical Study & Evaluation of Modern CAPTCHAs

Abstract

For nearly two decades, CAPTCHAS have been widely used as a means of protection against bots. Throughout the years, as their use grew, techniques to defeat or bypass CAPTCHAS have continued to improve. Meanwhile, CAPTCHAS have also evolved in terms of sophistication and diversity, becoming increasingly difficult to solve for both bots (machines) and humans. Given this long-standing and still-ongoing arms race, it is critical to investigate how long it takes legitimate users to solve modern CAPTCHAS, and how they are perceived by those users.

In this work, we explore CAPTCHAS *in the wild* by evaluating users' solving performance and perceptions of *unmodified currently-deployed* CAPTCHAS. We obtain this data through manual inspection of popular websites and user studies in which 1,400 participants collectively solved 14,000 CAPTCHAS. Results show significant differences between the most popular types of CAPTCHAS: surprisingly, solving time and user perception are not always correlated. We performed a comparative study to investigate the effect of *experimental context* – specifically the difference between solving CAPTCHAS directly versus solving them as part of a more natural task, such as account creation. Whilst there were several potential confounding factors, our results show that experimental context could have an impact on this task, and must be taken into account in future CAPTCHA studies. Finally, we investigate CAPTCHA-induced user task *abandonment* by analyzing participants who start and do not complete the task.

2.1 Introduction

Automated bots pose a significant challenge for, and danger to, many website operators and providers. Masquerading as legitimate human users, these bots are often programmed to scrape content, create accounts, post fake comments or reviews, consume scarce resources, or generally (ab)use other website functionality intended for human use [76, 58]. If left unchecked, bots can perform these nefarious actions at scale. CAPTCHAS are a widely-deployed defense mechanism that aims to prevent bots from interacting with websites by forcing each user to perform a task, such as solving a challenge [24]. Ideally, the task should be straightforward for humans, yet difficult for machines [105].

The earliest CAPTCHAS asked users to transcribe random distorted text from an image. However, advances in computer vision and machine learning have dramatically increased the ability of bots to recognize distorted text [112, 63, 70], and by 2014, automated tools achieved over 99% accuracy [67, 97]. Alternatively, bots often outsource solving to CAPTCHA *farms* – sweatshop-like operations where humans are paid to solve CAPTCHAS [85]. In light of this, CAPTCHAS have changed and evolved significantly over the years. Popular CAPTCHA tasks currently include object recognition (e.g., “select squares with...”), parsing distorted text, puzzle solving (e.g., “slide the block...”), and user behavior analysis [67, 97]. It is therefore critical to understand and quantify how long it takes legitimate users to solve current CAPTCHAS, and how these CAPTCHAS are perceived by users.

Several prior research efforts have explored CAPTCHA solving times, e.g., [53, 49, 65, 91, 104, 60]. For example, over a decade ago, Bursztein et al. [53] performed a large-scale user study, using over 1,100 unique participants from Amazon Mechanical Turk (MTurk) [22] as well as CAPTCHA farms. Their results showed that CAPTCHAS were often more difficult or took longer to solve than was expected. There was a loose correlation between time-to-annoyance and abandonment, with higher abandonment rates observed for CAPTCHAS that took longer

to solve. The same study also showed several demographic trends, e.g., users outside the US typically took longer to solve English-language CAPTCHA schemes. However, since this study, the CAPTCHA ecosystem has changed substantially: new CAPTCHA types emerged, input methods evolved, and Web use boomed.

More recently, Feng et al. [60] used a similar methodology, with 202 participants, to study the usability of their newly-proposed senCAPTCHA in comparison to text, audio, image, and video-based CAPTCHAS. They found that senCAPTCHA outperformed the alternatives, both in terms of solving time and user preference. They used Securimage [88], a free open-source PHP script, to generate text and audio CAPTCHAS, and they implemented their own image and video CAPTCHAS.

Building upon and complementing prior work, this chapter evaluates CAPTCHAS *in the wild* – specifically, the solving times and user perceptions of *unmodified* (i.e., not re-implemented) *currently-deployed* CAPTCHA types. We first performed a manual inspection of 200 popular websites, based on the Alexa Top websites list [36], to ascertain: (1) *how many* websites use CAPTCHAS, and (2) *what types* of CAPTCHAS they use. Next, we conducted a 1,000-participant user study using Amazon MTurk, wherein each participant was required to solve 10 different types of CAPTCHAS. We collected information about participants’ CAPTCHA solving times, relative preferences for CAPTCHA types, types of devices used, and various demographic information.

One notable aspect of our user study is that we attempted to measure the impact of experimental context on participants’ CAPTCHA solving times. Half of the participants were directly asked to solve CAPTCHAS, whilst the other half were asked to create accounts, which involved solving CAPTCHAS as part of the task. The latter setting was designed to measure CAPTCHA solving times *in the context* of a typical web activity.

One inherent limitation of any user study, especially when using MTurk, is that we cannot

ensure that all participants who begin the study will complete it. All of our results should therefore be interpreted as referring to *users who are willing to solve* CAPTCHAS, rather than users in general.

Indeed, having noted that some participants began but did not complete our main study, we conducted a secondary MTurk study specifically designed to quantify how many users abandon their intended web activity when confronted with different types of CAPTCHAS. We believe that CAPTCHA-induced *user abandonment* is an important – yet understudied – consideration, since every abandoned task (e.g., purchase, account creation) represents a potential loss for the website.

To facilitate reproducibility and enable further analysis, we provide the entire anonymized data-set collected during our user studies, along with our analysis code.¹

2.2 Research Questions & Main Findings

We now present our research questions and summarize our main findings. Table 4.1 shows how our findings relate to prior work at a high level, with detailed comparisons in Section 2.7.

RQ1: How long do human users take to solve different types of captchas? Specifically, we aimed to measure solving times for CAPTCHAS that users are likely to encounter (e.g., those used on popular websites). Our results align with previous findings [53, 60, 49] in showing that there are significant differences in mean solving times between CAPTCHA types. For comparison, we also identified the current fastest attacks on each type of CAPTCHA (Table 3.12).

RQ2: What Captcha types do users prefer? In order to understand users’ relative preference for various types of CAPTCHAS, we asked participants to rate all CAPTCHA types

¹<https://github.com/sprout-uci/captcha-study>

on a Likert scale of 1–5, from least to most enjoyable. Our results show that there are marked differences in participants’ preferences, with average preference scores ranging from 2.76 to 3.94. Our results also show that average solving time is *not fully correlated* with participants’ preferences, which means that other factors, beyond the amount of time required to solve a CAPTCHA, influence participants’ preferences. Our analysis of data from prior studies [78, 60, 102] shows that their data supports this finding (even if they do not discuss it explicitly).

RQ3: Does experimental context affect solving time? Specifically, we aimed to quantify the difference in solving times between the setting where participants are directly tasked with solving CAPTCHAS versus the setting in which participants solve CAPTCHAS as part of a typical web activity, such as user account creation. We therefore ran two separate versions of our main user study: *direct* and *contextualized*, which we describe in detail in Section 2.4.2. Whilst there were several potential confounding factors in our study, our results show that experimental context could have an impact on CAPTCHA user studies, with the difference in mean solving times as high as 57.5% in our study.

RQ4: Do demographics affect solving time? We analyzed different self-reported metrics including age, gender, country of residence, education, Internet usage, device type and input method. In line with prior results [53], we found that all types of CAPTCHAS take longer for older participants. Specifically, [53] reported an increase in solving time for text-based CAPTCHAS of 0.03 seconds per year of participant age. Our results show an even stronger dependence with an average increase across all CAPTCHA types of 0.09 seconds per year. Additionally, [53] showed that participants with a PhD solved CAPTCHAS faster than all other educational groups. In contrast, our results show that our participants’ self-reported level of education does not correlate with their solving times.

RQ5: Does experimental context influence abandonment? Specifically, we aimed to quantify the extent to which abandonment within a CAPTCHA user study is influenced by i) experimental context, and ii) the amount of compensation offered. For different combinations

of the above variables, we found that between 18% and 45% of participants abandoned the study after the presentation of the first CAPTCHA. Only one prior CAPTCHA user study [53] disclosed their observed rate of abandonment, which is similar to that observed in our study. Overall, participants in the contextualized setting were 120% more likely to abandon than their peers in the direct setting. This connection between experimental context and user abandonment is a new finding.

Table 2.1: Summary of research questions and main findings.

	Findings supporting prior work	Findings contradicting prior work	New findings on Captchas
RQ1: How long does it take humans to solve different types of captchas?	Solving time across CAPTCHA types has a large degree of variance. [53, 60, 49]		
RQ2: What Captcha types do users prefer?	Solving time is not correlated with user preference. [78, 60, 102]		
RQ3: Does experimental context affect solving time?			Solving time is heavily influenced by experimental context, with differences in means up to 57.5%.
RQ4: Do demographics affect solving time?	Age has an effect on solving time. [53]	Self-reported education does not correlate with solving time. [53]	
RQ5: Does experimental context influence abandonment?	High abandonment rates observed in CAPTCHA user studies. [53]		Experimental context directly affects the rate of abandonment.

2.3 Website Inspection

To understand the landscape of modern CAPTCHAS and guide the design of the subsequent user study, we manually inspected the 200 most popular websites from the Alexa Top Website list [36]. Where applicable, we use the terminology from the taxonomy proposed by Guerar et al. [68].

Our goal was to imitate a normal user’s web experience and trigger CAPTCHAS in a natural setting. Although CAPTCHAS can be used to protect any section or action on a website, they are often encountered during user account creation to prevent bots creating accounts. Thus, for each website, we investigated the process of creating an account (wherever available). Of the inspected websites, 185 had some type of account creation process, and we could successfully create accounts on 142 websites. Distinct domains operated by the same organization (e.g., `amazon.com` and `amazon.co.jp`) were counted separately. We visited each website twice: once with Google Chrome in incognito mode, and once with the Tor browser over the Tor network [37]. We used incognito mode to avoid websites changing their behavior based on cookies presented by our browser. We used Tor since anecdotal evidence suggests Tor users are asked to solve CAPTCHAS more frequently and with greater difficulty than non-Tor users. If no CAPTCHAS were displayed, we searched the page source for the string “CAPTCHA” (case insensitive).

Ethical considerations: Based on the Guidelines for Internet Measurement Activities [54], we did not engage in malicious behavior which may trigger additional CAPTCHAS. We used only manual analysis to avoid various challenges that arise from automated website crawling.

2.3.1 Results and analysis

Figure 2.1 shows the distribution of CAPTCHA types we observed during our inspection. The most prevalent types were:

reCAPTCHA [33, 32, 34] was the most prevalent, appearing on 68 websites (34% of the inspected websites). This is a Google-owned and operated service that presents users with “click” tasks, which include behavioral analytics and may potentially result in an image challenge. reCAPTCHA allows website operators to select a difficulty level, ranging from “easiest for users” to “most secure”.

Slider-based CAPTCHAS appeared on 14 websites (7%). These typically ask users to slide a puzzle piece into a corresponding empty spot using a drag interaction. The timing and accuracy is checked for bot-like behavior.

Distorted Text CAPTCHAS appeared on 14 websites (7%). We observed differences in terms of text type, color, length, masking, spacing, movement, and background. Text type varied in several ways: 2D or 3D, solid or hollow, font, and level of distortion. Certain CAPTCHAS used masking, i.e., lines or shapes obscured parts of the letters.

Game-based CAPTCHAS appeared on 9 websites (4.5%). These present users with dynamic games and compute a risk profile from the results. For example, users are asked to rotate an image or select the correctly oriented image.

hCAPTCHA [29] appeared on 1 website. This is a service provided by Intuition Machines, Inc. that was recently adopted by Cloudflare [90] and is gaining popularity.

Invisible captchas were found on 12 websites (6%). These websites did not display any visible CAPTCHAS, but contained the string “CAPTCHA” in the page source.

Other Captchas found during our inspection included: a CAPTCHA resembling a scratch-off lottery ticket; a CAPTCHA asking users to locate Chinese characters within an image; and a proprietary CAPTCHA service called “NuCaptcha” [6].

2.3.2 Potential limitations

Choice of website list: There are several lists of “popular” websites that could be used for this type of study, including the Alexa Top Website list [36], Cisco Umbrella [2], Majestic [8], TRANCO [89], Cloudflare Radar [3], and SecRank TopDomain [109]. These lists vary because of the differences in the methodology used to identify and rank websites. Following

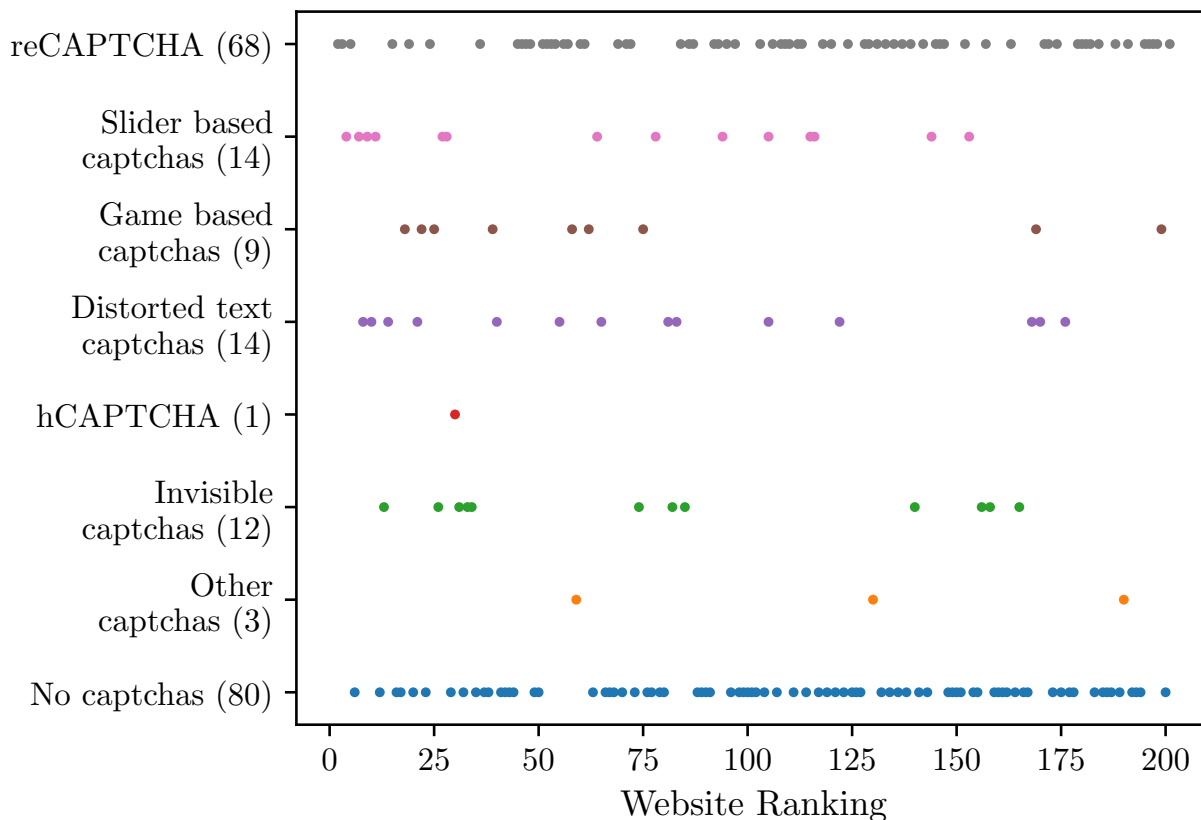


Figure 2.1: Discrete distribution of discovered CAPTCHAS (full data available in the accompanying dataset).

the work of Bursztein et al. [53] and the recommendation of Scheitle et al. [93], we used the Alexa list.

Number of inspected websites: Since our website inspection was a manual process, we could only inspect the top 200 websites. This may also introduce a degree of systemic bias towards the types of CAPTCHAS used on the most popular websites. However, we specifically chose these websites because they are visited by large numbers of users.

Lower bound: Since we did not exercise all possible functionality of every website, it is possible that we might not have encountered all CAPTCHAS. Therefore, our results represent a lower bound, while the actual number of deployed CAPTCHAS may be higher. Nevertheless, we believe that we identified the most prevalent CAPTCHA types across all inspected websites.

Timing: Web page rankings change on the daily basis and CAPTCHAS shown by the same service may change. Given that our inspection was performed at a particular point in time, the precise results will likely change if the inspection were repeated at a different point in time. However, as explained above, we believe that the identified set of CAPTCHA types is representative of currently-deployed CAPTCHAS.

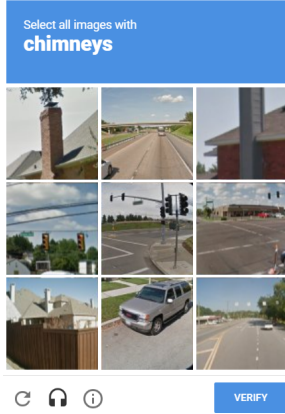
Other types of captchas: We only inspected mainstream websites (i.e., those that would appear on a top websites list). This means that there could be CAPTCHAS that are prevalent on other types of websites (e.g., on the dark web) but are not included in our study. However, studying these *special-purpose* CAPTCHAS might require recruiting participants who have prior experience solving them, which was beyond the scope of our study.

Impact of limitations: The above limitations could have had an impact on the set of CAPTCHA types we identified and subsequently used in our user study. However, we have high confidence that the CAPTCHA types we identified are a realistic sample of those a real user would encounter during typical web browsing. For instance, BuiltWith [24] has analyzed a dataset of 673 million websites and identified 15.2 million websites that use CAPTCHAS. reCAPTCHA accounts for 97.3% and hCAPTCHA for a further 1.4%. The CAPTCHA types used in our study therefore account for over 98% of CAPTCHAS in this large-scale dataset.

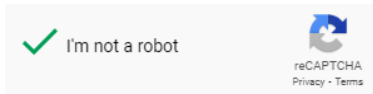
2.4 User Study

Having identified the relevant CAPTCHA types, we conducted a 1,000 participant online user study to evaluate real users' solving times and preferences for these types of CAPTCHAS. Our study was run using Amazon MTurk and can be summarized into the following four phases:

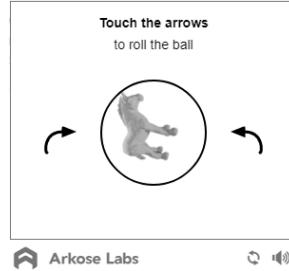
1. **Introduction:** Participants were first given an overview of the study and details of the



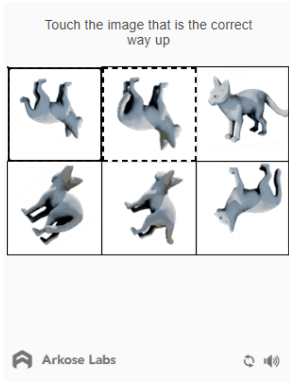
(a) Image Task CAPTCHA [32]



(b) v2 checkbox CAPTCHA [33]



(a) Rotation CAPTCHA



(b) Orientation selection



Figure 2.4: Geetest [26]

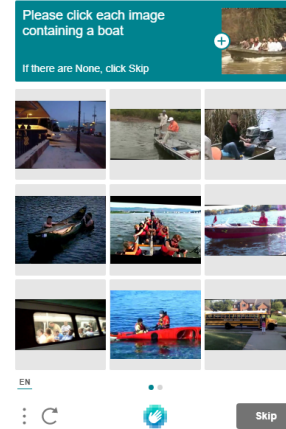


Figure 2.5: hCAPTCHA [29]

Figure 2.2: reCAPTCHA [33, 32, 34]

Figure 2.3: Arkose Labs [23]

tasks to complete.

2. Pre-study questions: All participants were then asked to provide demographic information by answering the pre-study questions shown in Table 2.11 in Section 2.9.

3. Tasks: Participants were asked to complete tasks, which included solving exactly ten CAPTCHAS, presented in random order. Unless otherwise stated, each CAPTCHA was *unique* (i.e., freshly generated per participant). Participants had to solve each CAPTCHA in order to progress to the next step, thus preventing them from speeding through the study.

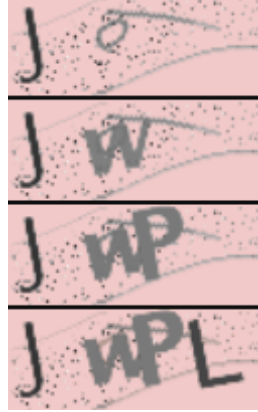
4. Post-study question Finally, participants were asked questions about the CAPTCHAS they had just solved. The exact questions and possible answers are shown in Table 2.11 in Section 2.9.



(a) Xinhuanet CAPTCHA [10]



(b) 360.cn CAPTCHA [1]



(c) jrj.com CAPTCHA [5]

Figure 2.6: Distorted text CAPTCHAS

2.4.1 Choice of Captchas

Based on our website inspection (Section 2.3), we selected the following ten types of CAPTCHAS:

- Two reCAPTCHA v2 CAPTCHAS: one with the setting *easiest for users* and the other with *most secure*. Note that we do not have control over whether the user is shown an image-based (Figure 3.2) challenge in addition to the click-based (Figure 3.1) task.
- Two game-based CAPTCHAS from Arkose Labs [23]: one required using arrows to rotate an object (Figure 2.3a) and the other required selecting the upright object (Figure 2.3b).
- Two hCAPTCHAs [29]: one with easy and one with difficult settings (Figure 2.5).
- One slider-based CAPTCHA from Geetest [26]: we selected Geetest because it was used on several of the inspected websites and offers a convenient API (Figure 2.4).
- Three types of distorted text CAPTCHAS (Figure 2.6): (a) the *simple* version had four unobscured characters, (b) the *masked* version had five characters and included some masking effects, and (c) the *moving* version contained moving characters.

These form a representative sample of CAPTCHAS we encountered in our website inspection. Although hCAPTCHA only appeared once, we included it since it is an emerging image-

based approach, which claims to be the largest independent CAPTCHA service [30].

2.4.2 Direct vs. contextualized settings

We initially hypothesized that we would observe a difference in behavior depending on experimental context. In order to evaluate this, we designed two settings of the study: 500 participants completed the *direct setting*, whilst the other 500 completed the *contextualized setting*. In both settings, each participant solved exactly ten CAPTCHAS in random order.

Direct setting: This setting was designed to match previous CAPTCHA user studies, in which participants are directly asked to solve CAPTCHAS. The MTurk study title was “CAPTCHA User Study” and the instructions in the first phase informed users that their task was to solve CAPTCHAS. In the second phase, in addition to the basic demographic information, participants were asked about their experience with and perception of CAPTCHAS; see Table 2.11 in Section 2.9. In the third phase, participants were shown ten CAPTCHAS in random order. The fourth phase was the same for both settings.

Contextualized setting: This setting was designed to measure CAPTCHA solving behavior *in the context* of a typical web activity. We selected the task of user account creation, as this often includes solving a CAPTCHA. The MTurk study title was “Account Creation User Study” and the first and second phases did not mention CAPTCHAS. In the third phase, participants were asked to complete ten typical user account creation forms, each displaying a CAPTCHA *after* the participant clicked submit, as is often the case on real websites. This sequencing allowed us to precisely measure the CAPTCHA solving time in isolation from the rest of the account creation task. The account creation task was a basic web form asking for a randomized subset of: name, email address, phone number, password, and address. To avoid collecting personally identifiable information, participants were provided with synthetic information at each step. Each page also included a large banner clearly stating not to enter

Table 2.2: Summary of demographic data for the 1,400 participants of the main user study.

Age	Residence	Education	Gender	Device Type	Input Method	Internet Use
30 - 39 (531)	USA (985)	Bachelors (822)	Male (832)	Computer (1301)	Keyboard (1261)	Work (860)
20 - 29 (403)	India (240)	Masters (243)	Female (557)	Phone (74)	Touch (125)	Web surf (397)
40 - 49 (271)	Brazil (50)	High school (210)	Non-Binary (11)	Tablet (25)	Other (14)	Education (87)
50 - 59 (106)	Italy (27)	Associate (98)				Gaming (30)
≥ 60 (58)	UK (24)	PhD (24)				Other (26)
18 - 19 (31)	Other (74)	No degree (3)				

any personal information. The fact that we were specifically measuring CAPTCHA solving time was only revealed to participants after they completed the first three phases.

2.4.3 Timeline and compensation

The primary study ran for two months with a total of 1,000 distinct participants.² Participants were initially paid \$0.30 for completing the direct version and \$0.75 for the contextualized version, as the latter involved a larger workload. After completing the study, we realized we may have unintentionally under-compensated participants,³ since the median HIT completion time was 4.4 and 11.5 minutes for direct and contextualized versions. We therefore retroactively doubled all participants’ compensation to \$0.60 and \$1.50, which equates to approximately \$7.80 – \$8.20 per hour.

²To the best of our knowledge, all participants were distinct. We configured Amazon MTurk to only allow unique accounts to participate.

³In terms of US federal minimum wage.

2.4.4 Ethical considerations

This user study was duly approved by the Institutional Review Board (IRB) of the primary authors' organization. No sensitive or personally identifiable information was collected from participants. We used the pseudonymous MTurk worker IDs only to check that participants were unique.

Since the contextualized setting did not inform participants of the actual aim of the study beforehand, two additional documents were filed and approved by the IRB: (1) *“Use of deception/incomplete disclosure”* and (2) *“Waiver or Alteration of the Consent”*. After each participant completed the contextualized setting, we disclosed the study's actual goal and asked whether they gave us permission to use their data. No data were collected from participants who declined.

2.4.5 User study implementation

The realization of the user study included a front-end webpage and a back-end server. The front-end was a single HTML page that implemented the four phases described above. To prevent any inconsistencies, participants were prevented from going back to a previous phase or retrying a task once they had progressed. Timing events were captured with millisecond precision using the native JavaScript `Date` library. Timing events were recorded at several points for each CAPTCHA: request, serve, load, display, submit, and server response. We measured *solving time* as the time between a CAPTCHA being displayed and the participant submitting a solution, as is done in prior CAPTCHA user studies [53, 49, 61, 113, 48, 72, 78, 91, 104, 65, 77, 83, 66]. Depending on the type of CAPTCHA, this might include multiple rounds or attempts.

We used Amazon MTurk to recruit participants, host the front-end, and collect data. While

most types of CAPTCHAS shown by the front-end were served from their respective providers, distorted text CAPTCHAS were not available from a third-party provider, as these are usually hosted by the websites themselves. We therefore set up our own back-end server to serve distorted text CAPTCHAS. Specifically, we downloaded a total of 1,000 unique distorted text CAPTCHAS of three different types, and stored these in a local MongoDB [16] database. We used a Node.js [17] server to retrieve and serve CAPTCHAS from the database. Every participant was served one text CAPTCHA of each type, and each unique text CAPTCHA was served to three different participants.

Table 2.2 shows the demographic information of the participants who completed the study. The demographics of the two subgroups who completed direct and contextualized studies are very similar to each other.

2.4.6 Potential limitations

Use of MTurk: Webb et al. [107] reported several potential concerns regarding the quality of data collected from MTurk. Of their six criteria, our study did not implement two: consent quiz (1) and examination of qualitative responses (2), which we acknowledge as a limitation. The remaining four criteria can be either evaluated through collected data or are not an issue for our study. Eligibility (3) and attention check (4) can be verified via the accuracy of text-based CAPTCHA responses, which confirm that nearly all of our participants were focused and provided correct data. Response time (5) was within our expected range. Study completion (6) was not an issue, since each participant had to complete every CAPTCHA to proceed.

Bots and farms: Similarly, Chmielewski et al. [57] reported a decrease in data quality, citing bot and farm activity. However, Moss and Litman [84] subsequently used several bot-detection measures to evaluate whether bots could be contaminating MTurk data, and found

no evidence of bot activity. Every participant who completed our study solved ten modern CAPTCHAS, which although possible, would be more difficult for bots. Since we configured MTurk to only allow one completion per MTurk account, farm activity was also limited. Therefore, we are reasonably confident that our results are not influenced by bots or farms.

Choice of captchas: One consequence of using the CAPTCHA types we identified in Section 2.3 is that our user study results are not directly comparable with those from prior CAPTCHA user studies. In general, it is difficult to directly compare such studies, as even if the same *types* of CAPTCHAS are studied, different implementations may be used e.g., reCAPTCHA and hCAPTCHA are both image-based CAPTCHAS, but could give different results.

Unmodified captchas: In order to maximize the level of realism in our study, we used existing unmodified CAPTCHAS. We therefore did not have fine-grained control over the precise behavior of these CAPTCHAS, nor the ability to obtain more fine-grained measurements of participants' accuracy or performance beyond overall solving time. However, like previous studies, we consider overall solving time to be the most important measurable quantity.

Invalid inputs: Unfortunately, the input field for the CAPTCHA preference question in our post-study questionnaire was a free text field rather than a pull-down menu. This allowed some participants to provide preference scores outside the requested 1-5 range. We therefore excluded invalid preference scores from 163 participants.⁴

Abandonment: Since we did not record how many participants began our main study, we cannot precisely quantify the rate of abandonment. To investigate this further, we performed an additional abandonment-focused study (Section 2.6), where we observed a 30% abandonment rate. We can therefore assume a similar abandonment rate for our main study. Whilst

⁴However, we have high confidence that these participants did not provide incorrect or rushed responses during the rest of the study because their average accuracy in text-based CAPTCHAS was similar to the study-wide average. We therefore retained their measurements in other sections.

the impact of this level of abandonment is unclear, it could potentially affect the ecological validity of our results, as the participants who were willing to complete the study may not be representative of all users.

Confounding factors: There were several differences between our direct and contextualized settings, some of which may be confounding factors when comparing these two groups. For example, participants in the contextualized setting had to do more work, so their attention or focus might have been reduced during CAPTCHA solving. Differences in compensation or participants’ perceived benefit of completing the task (i.e., creating an account vs. solving a CAPTCHA) may have affected motivation or likeliness to abandon the task.

2.5 Results & Analysis

This section presents the user study results. Unless otherwise indicated, results are based on the full set of participants.

2.5.1 Solving times

This subsection addresses **RQ1:** *How long do human users take to solve different types of CAPTCHAS?* Figure 2.7 shows the the distribution of solving times for each CAPTCHA type. We observed a small number of extreme outliers where the participant likely switched to another task before returning to the study. We therefore filtered out the highest 50 solving times per CAPTCHA type, out of 1,000 total.

For reCAPTCHA, the selection between image- or click-based tasks is made dynamically by Google. Whilst we know that 85% and 71% of participants (easy and hard setting) were shown a click-based CAPTCHA, the exact task-to-participant mapping is not revealed

to website operators. We therefore assume that the slowest solving times correspond to image-based tasks. After disambiguation, click-based reCAPTCHA had the lowest median solving time at 3.7 seconds. Curiously, there was little difference between easy and difficult settings.

The next lowest median solving times were for distorted text CAPTCHAS. As expected, simple distorted text CAPTCHAS were solved the fastest. Masked and moving versions had very similar solving times. For hCAPTCHA, there is a clear distinction between easy and difficult settings. The latter consistently served either a harder image-based task or increased the number of rounds. However, for both hCAPTCHA settings, the fastest solving times are similar to those of reCAPTCHA and distorted text. Finally, the game-based and slider-based CAPTCHAS generally yielded higher median solving times, though some participants still solved these relatively quickly (e.g., < 10 seconds).

With the exception of reCAPTCHA (click) and distorted text, we observed that solving times for other types have a relatively high variance. Some variance is expected, especially since these results encompass all input modalities across both direct and contextualized settings. However, *relative differences in variances* indicate that, while some types of CAPTCHAS are consistently solved quickly, most have a range of solving times across the user population. The full statistical analysis of our solving time results is presented in Section 2.10.

2.5.2 Preferences analysis

This subsection addresses **RQ2**: *What CAPTCHA types do users prefer?* Figure 2.8 shows participants' CAPTCHA preference responses after completing the solving tasks. The CAPTCHA types are sorted from most to least preferred by overall preference score, which is calculated by summing the numeric scores. Since easy and difficult settings of hCAPTCHA are visually indistinguishable, we could only ask participants for one preference.

As expected, participants tend to prefer CAPTCHAS with lower solving times. For example, reCAPTCHA (click) has the lowest median solving time and the highest user preference. However, surprisingly, this trend does not seem to hold for game-based and slider-based CAPTCHAS, since these received some of the highest preference scores, even though they typically took longer than other types. This suggests that factors beyond solving time could be contributing to participants' preference scores. Notably, no single CAPTCHA type is either universally liked or disliked. For example, even the top-rated click-based reCAPTCHA, was rated 1 or 2 by 18.9% of participants. Similarly, over 31.0% rated hCAPTCHA 4 or 5, although it had the lowest overall preference score.

2.5.3 Direct vs. contextualized setting

This subsection addresses **RQ3**: *Does experimental context affect solving time?* Figure 2.9 shows histograms of CAPTCHA solving times for participants in the direct vs. contextualized settings. In every case except one, the mean solving time is lower in the direct setting. In most cases, the distribution from the contextualized setting has more participants with longer solving times, i.e., a longer tail.

The largest statistically significant difference is in reCAPTCHA (easy click), where the mean solving time grows by 1.8 seconds (57.5%). Second is Arkose (rotation), where it grows by 10 seconds (56.1%). Across all CAPTCHA types, the average increase from direct to contextualized is 26.7%. Similarly, the mean solving time for reCAPTCHA (easy image) increased by 63.6% in the contextualized setting showing the largest increase. However this was not statistically significant. This is likely due to the skew of participants in direct and contextualized versions receiving image-challenges, which is controlled by Google. Easy images were shown to 8.9% of contextualized and to 17.2% of direct setting participants, while hard images were shown to 25.5% and 30% respectively, resulting in different population sizes.

On the other hand, hCAPTCHA (difficult), which has the highest median solving time overall, showed no significant difference in mean solving time between direct and contextualized settings. This may be attributable to the difficulty of solving this type of CAPTCHA, regardless of the setting.

Results of Kruskal-Wallis tests confirm that there are statistically significant differences in mean solving times for all CAPTCHA types ($p < 0.001$) except Geetest, reCAPTCHA (image) and hCAPTCHA (difficult). While there were several potential confounding factors in our study, these results suggest that experimental context can have a significant impact on participants' CAPTCHA solving times, and must therefore be taken into account in the design of future user studies.

2.5.4 Effects of demographics

This subsection addresses **RQ4**: *Do demographics affect solving time?* We analyzed how demographic characteristics in our study correlate with CAPTCHA solving times. For some characteristics, such as education and gender, we did not observe large differences in CAPTCHA solving times (see Figures 2.13 and 2.14 in Section 2.11).

Effects of age

Figure 2.10 shows the effect of participants' age on solving time. The green line is the average solving time for each age, and the red line is a linear fit minimizing mean square error. For all types, except reCAPTCHA (easy image), there is a trend of younger participants having lower average solving times. This agrees with prior results [53] and is especially noticeable in hCAPTCHA, Arkose (selection), and Geetest.

Effects of device type

Figure 2.11 shows the effect of device type. Although there are some differences in median between device types for certain CAPTCHA types, the Kruskal-Wallis test shows that the differences in means are mostly not statistically significant. The only statistically significant differences are in distorted text CAPTCHAS ($p < 0.02$) and reCAPTCHA (hard click) ($p < 0.01$), where participants who used computers had a lower mean solving time compared to those using phones. Interestingly, we found a statistically significant difference between participants who used physical keyboards and those who used touch input for the simple and masked distorted text CAPTCHAS ($p < 0.02$), as well as reCAPTCHA (hard click) ($p < .001$), reCAPTCHA (easy click) ($p < .05$), and Arkose (selection) ($p < .003$). We found no statistically significant difference in mean solving times for moving distorted text.

Effects of typical Internet use

Figure 2.12 shows the relationship between participants' self-reported dominant Internet usage patterns and their CAPTCHA solving times. The Kruskal-Wallis test shows some initial evidence for statistically significant differences between participants who use the Internet primarily for work and those who use it for other purposes ($p < 0.05$). The former were typically slower than the latter in 8 out of 12 CAPTCHAS. However, some categories do not have a sufficient number of participants, thus further investigation is recommended.

2.5.5 Accuracy of Captchas

Table 3.12 contrasts our measured human solving times and accuracy against those of automated bots reported in the literature. Interestingly, these results suggest that bots *can* outperform humans, both in terms of solving time and accuracy, across all these CAPTCHA

types. As mentioned in Section 2.4.6, our decision to use unmodified real-world CAPTCHAS means we only have accuracy results for a subset of CAPTCHA types (e.g., neither Geetest nor Arkose provide accuracy information). For the same reason, our accuracy results also include participants who only partially completed the study.

reCAPTCHA: The accuracy of image classification was 81% and 81.7% on the easy and hard settings respectively. Surprisingly, the difficulty appeared not to impact accuracy.

hCAPTCHA: The accuracy was 81.4% and 70.6% on the easy and hard settings respectively. This shows that, unlike reCAPTCHA, the difficulty has a direct impact on accuracy.

Distorted Text: We evaluated *agreement* among participants as a proxy for accuracy. As each individual CAPTCHA was served to three separate participants, we measured agreement between any two or more participants. We also observed that agreement increases dramatically (20% on average) if responses are treated as case insensitive, as shown in Table 2.4.

Table 2.3: Humans vs. bot solving time (seconds) and accuracy (percentage) for different CAPTCHA types.

Captcha Type	Human		Bot	
	Time	Accuracy	Time	Accuracy
reCAPTCHA (click)	3.1-4.9	71-85%	1.4 [99]	100% [99]
Geetest	28-30	N/A	5.3 [108]	96% [108]
Arkose	18-42	N/A	N/A	N/A
Distorted Text	9-15.3	50-84%	<1 [115]	99.8% [67]
reCAPTCHA (image)	15-26	81%	17.5 [75]	85% [75]
hCAPTCHA	18-32	71-81%	14.9 [74]	98% [74]

Table 2.4: Agreement for distorted text CAPTCHAS.

	Average Agreement	Average Agreement (case insensitive)
Simple	84%	93%
Masked	50%	73%
Moving	62%	90%
Total	65%	85%

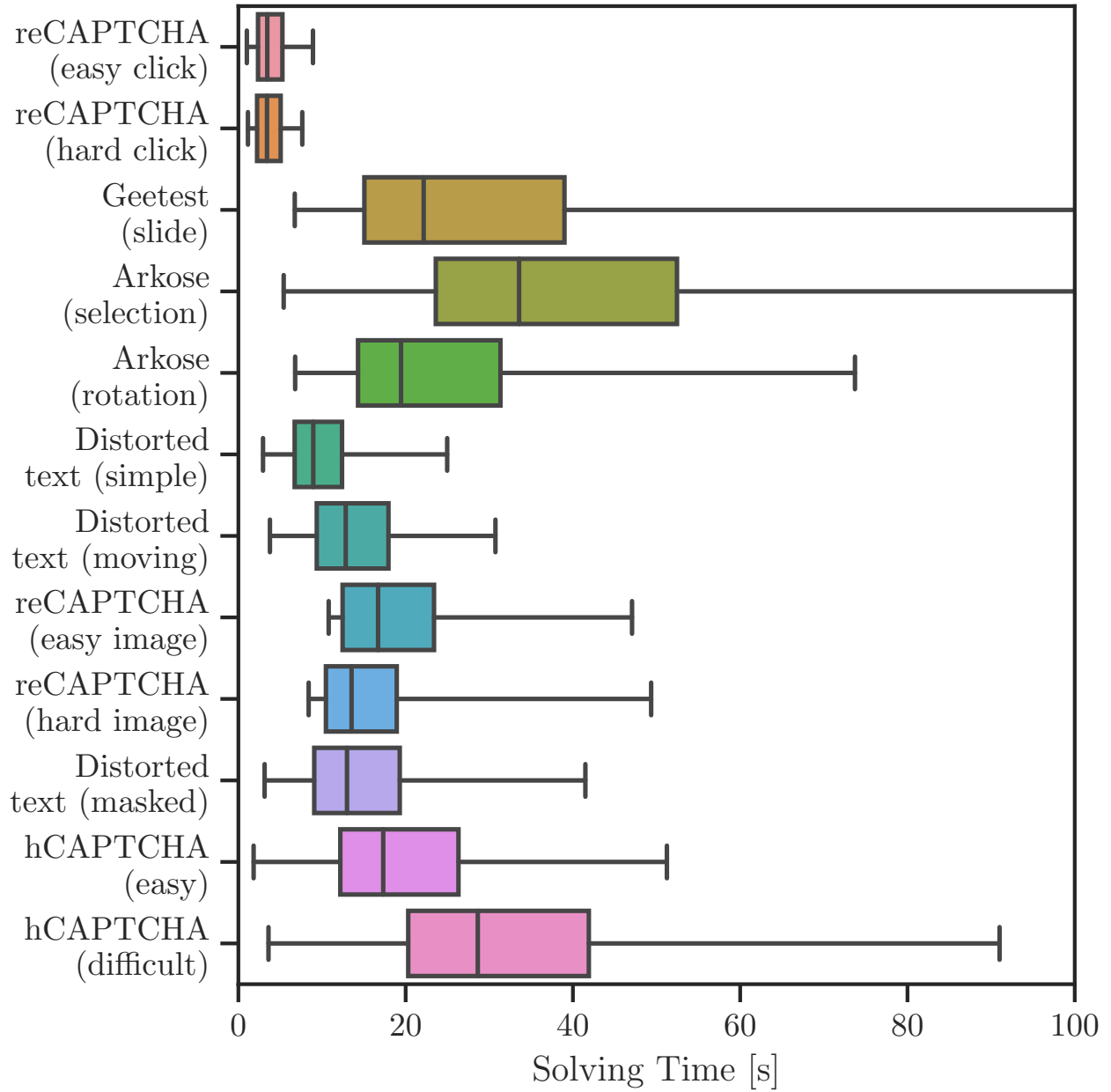


Figure 2.7: Solving times for various types of CAPTCHAS. Boxes show the middle 50% of participants, and whiskers show the filtered range. Black vertical lines show the median.

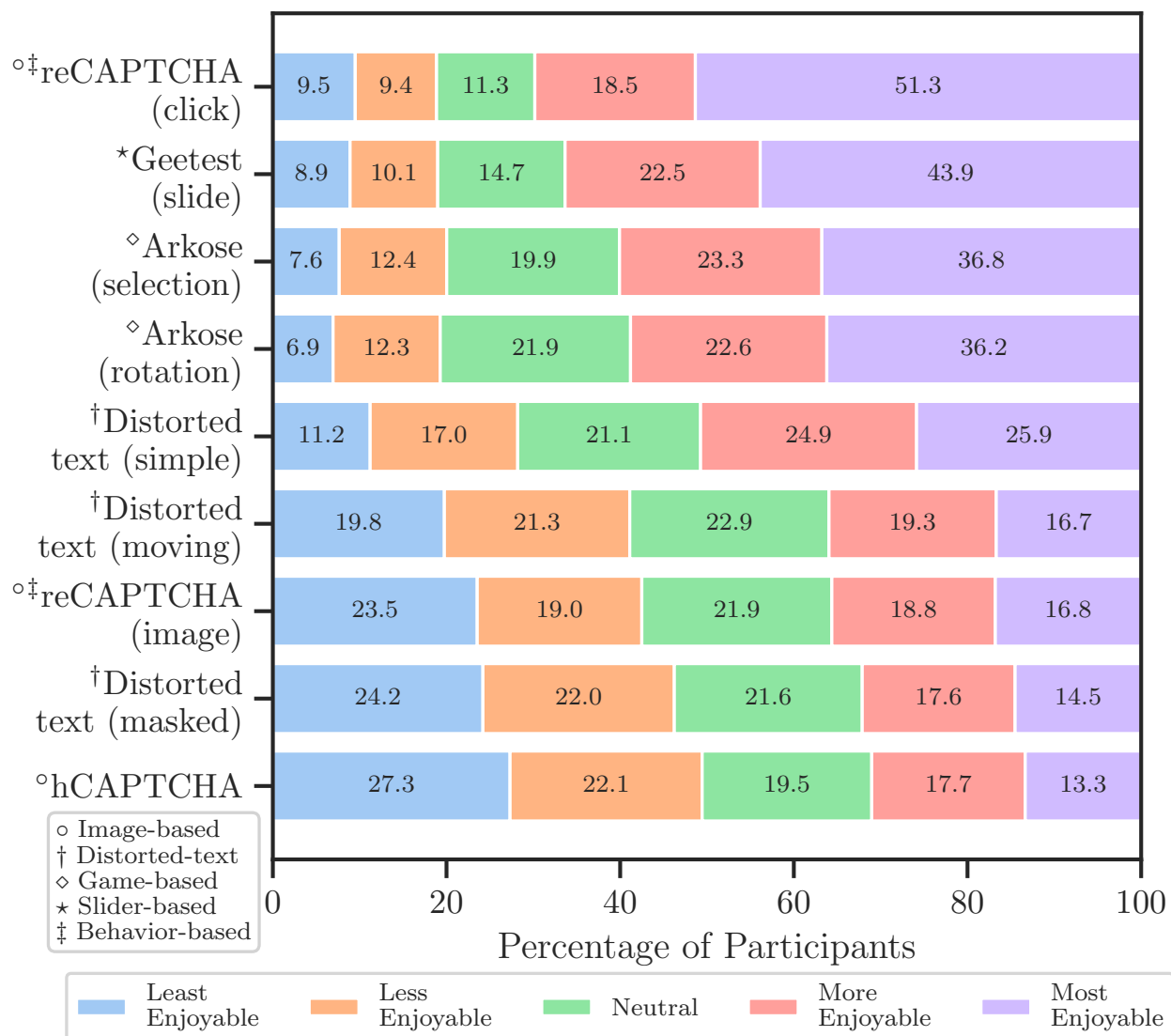


Figure 2.8: Participant-reported preference scores for different types of CAPTCHAs, sorted from highest to lowest.

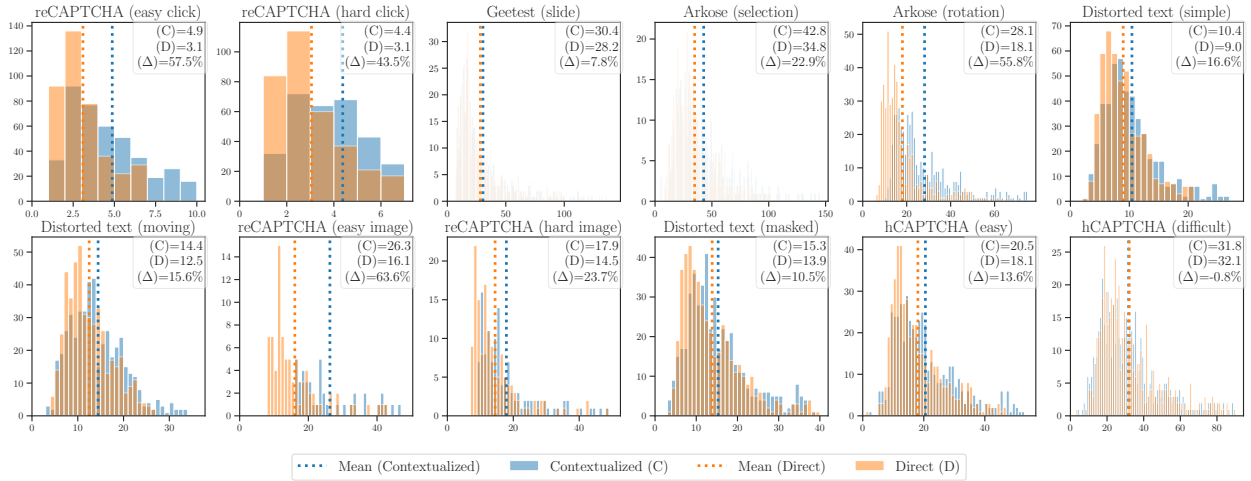


Figure 2.9: CAPTCHA solving times for direct (D) vs. contextualized (C) user study settings. The horizontal axis shows solving time in seconds, quantized into one-second buckets, and the vertical axis shows number of participants.

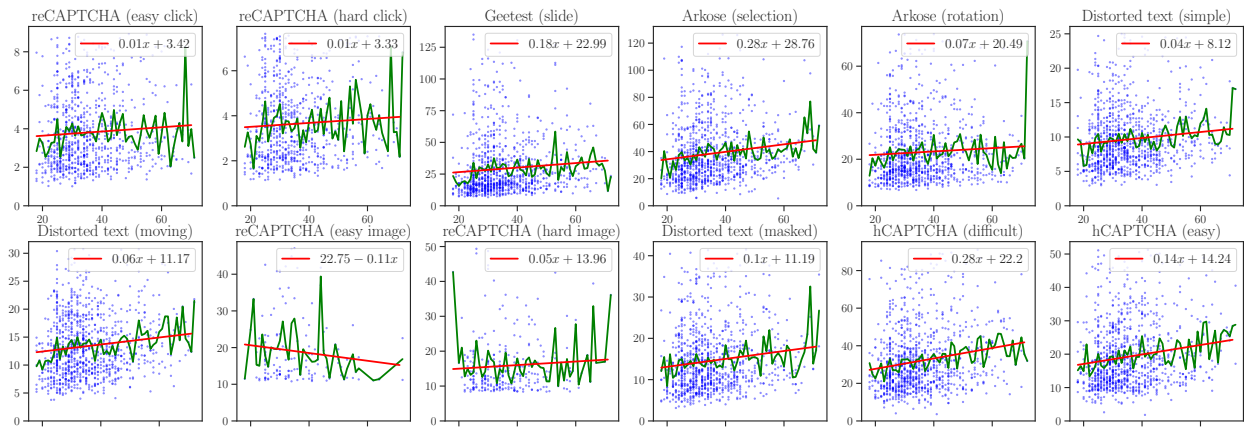


Figure 2.10: Effects of age in CAPTCHA solving time. The horizontal axis shows the age and the vertical axis shows the solving time. The red line shows the linear fit of the data points and the green line shows the average solving time per age.

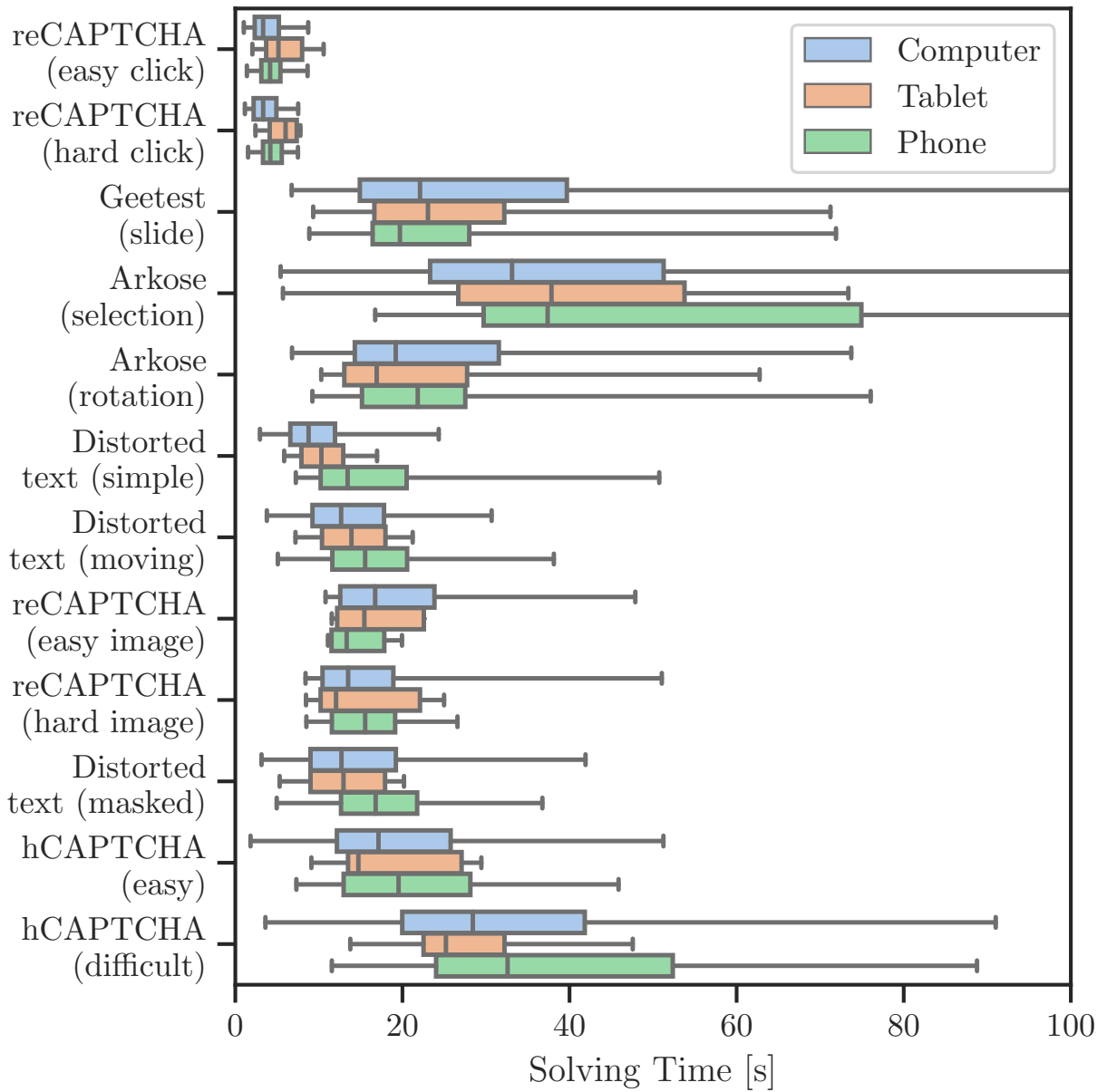


Figure 2.11: Effects of device type.

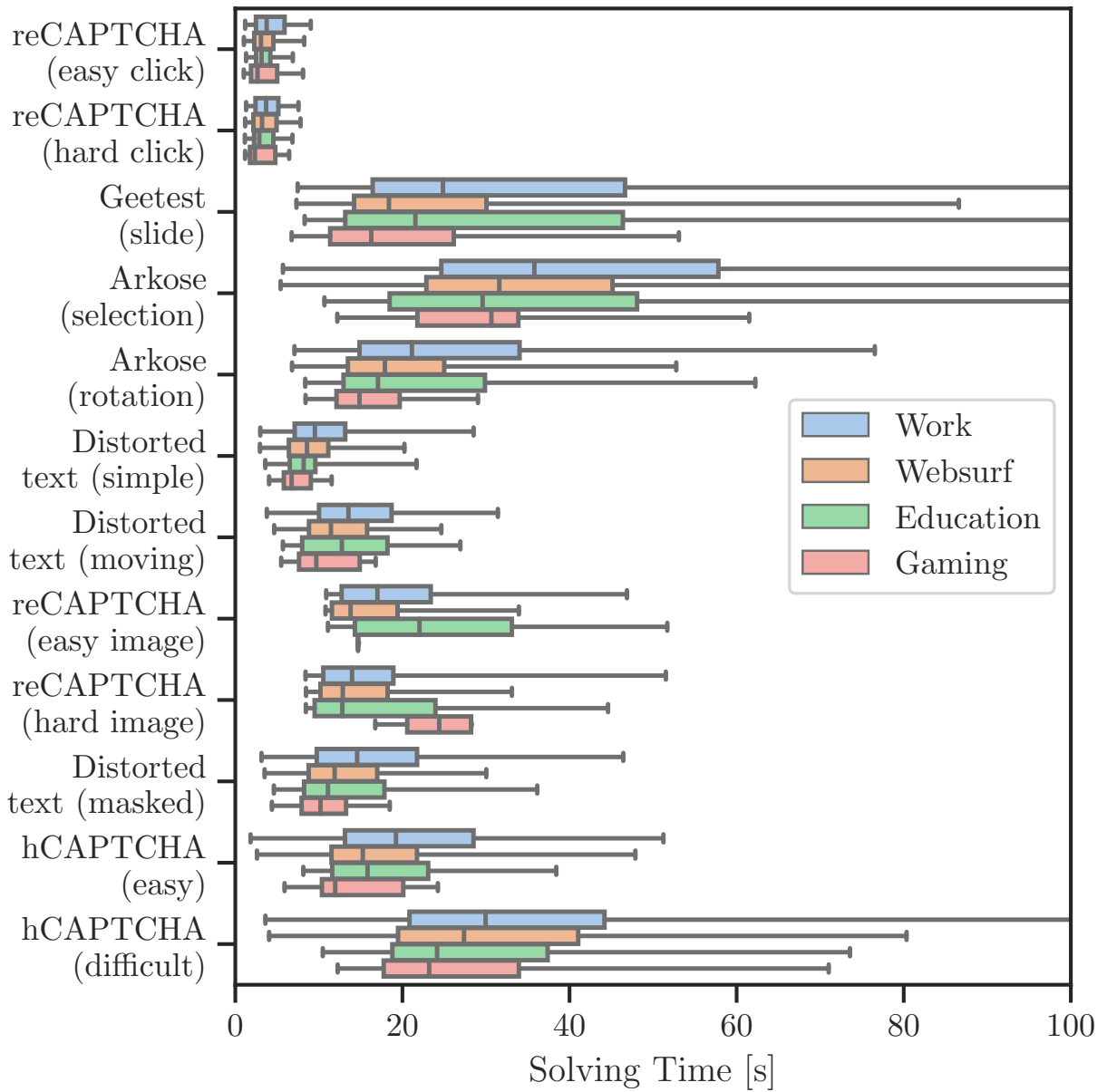


Figure 2.12: Effects of typical Internet use.

2.6 Measuring User Abandonment

This subsection addresses **RQ5**: *Does experimental context influence abandonment?* Upon completion, we observed that the number of CAPTCHAS solved during our study exceeded what would be expected based on the number of participants who completed the study. We hypothesized that this was due to participants starting but not completing the study. To measure this behavior, we conducted a second user study that collected timestamps between CAPTCHAS, regardless of whether the entire study was completed. We measured: (1) how many participants started the task; (2) how many abandoned the task when solving a CAPTCHA; and (3) if so, at which task and CAPTCHA.

This abandonment-focused study consisted of four groups, each with 100 unique participants. Two groups were presented with the direct setting and the other two with the contextualized setting (see Section 2.4.2). We hypothesized that the amount of compensation might also impact abandonment, so we doubled the compensation for one of the groups in each setting. The studies were run sequentially to avoid prospective participants simply picking the higher-paying study.

We summarize the key findings below, and present the full results in Tables 2.7, 2.8, 2.9, and 2.10 in Section 2.8. Out of a total of 574 participants who started the study, 174 abandoned prior to completion (i.e., 30% abandonment rate). Several observations can be made: First, in the direct setting, 25% of the participants who ultimately abandoned the study did so before solving the first CAPTCHA, but this rose to nearly 50% in the contextualized setting. Second, doubling the pay halved the abandonment rate for the contextualized setting (as expected), but increased it by 50% in the direct setting. Third, participants in the contextualized setting were 120% more likely to abandon than those in the direct setting. Fourth, in the contextualized setting, participants at the higher compensation level solved CAPTCHAS faster than those at the lower compensation level (21.5% decrease

in average solving time across all CAPTCHA types). Interestingly, in the direct setting, participants at the higher compensation level solved CAPTCHAS *slower* than those at the lower compensation level (27.4% *increase* in average solving time across all CAPTCHA types). Finally, some CAPTCHA types (e.g., Geetest) exhibited higher rates of abandonment than others.

This initial investigation strongly motivates the need for further exploration of CAPTCHA-induced abandonment. Although we studied the impact of compensation and experimental context, there may be other reasons behind abandonment, such as: CAPTCHA type, CAPTCHA difficulty, and expected duration of study. Nevertheless, the trend of average users' unwillingness to solve a CAPTCHA during account creation (even for monetary compensation) is a relevant finding for websites that choose to protect account creation (and/or account access) using CAPTCHAS.

2.7 Related Work

CAPTCHAS are a well-studied topic, with several prior studies investigating both existing and novel CAPTCHA schemes.

2.7.1 Comparison of methodologies

Table 2.5 summarizes the key methodological aspects of prior CAPTCHA user studies, from which the following observations can be made:

- Most prior research has focussed on distorted text and newly-proposed CAPTCHA schemes.
- MTurk and proprietary websites have been the norm across CAPTCHA user studies (except DevilTyper [72]).

- Whilst almost all studies measured solving time, there is a bifurcation in terms of accuracy measurements: studies evaluating their own CAPTCHA schemes or reimplementing existing schemes typically have direct access to accuracy results, whereas those evaluating unmodified deployed CAPTCHAS can only measure quantities such as agreement.
- Most studies measured demographics and ratings or preferences. Some studies also measured workload, open response (perceptions), and perceived usability.

2.7.2 Detailed comparisons

We present detailed comparisons of our methodology and results with three representative prior CAPTCHA studies.

Bursztein et al. [53] presented the first large-scale study on human CAPTCHA solving performance. Focussing on distorted text and audio CAPTCHAS, they used both MTurk and an underground CAPTCHA-solving service to measure solving time and accuracy. In terms of solving times, they found that it took on average 9.8 and 28.4 seconds to solve distorted text and audio CAPTCHAS respectively. Although we did not evaluate audio CAPTCHAS (as we did not observe these in our website inspection), our results for distorted text CAPTCHAS broadly agree at 12.5 on average. Similarly to our study, they used *agreement* between participants as a proxy for accuracy. For distorted text, they observed 71% agreement, which is in line with our observation of 75% when averaging case sensitive and insensitive versions (see Table 2.4).

Feng et al. [60] presented senCAPTCHA, a new CAPTCHA type using orientation sensors designed specifically for mobile devices with small screens. They evaluated its security against brute-force and ML-based attacks, and its usability through two usability studies totalling 472 participants. The second user study compared senCAPTCHA against text-, audio-, image-, and video-based CAPTCHAS, some of which were reimplemented for the

study. senCAPTCHA had the lowest median solving time (5.02 seconds), followed by image (9.6), video (10.08), text (11.93), and audio (47.07). With the exception of click-based reCAPTCHA, it can be extrapolated that senCAPTCHA would have a lower solving time than the other CAPTCHA types in our study. In terms of preferences, most participants in their study preferred senCAPTCHA. Out of the CAPTCHA types in our study, senCAPTCHA most closely resembles the game-based CAPTCHAS, which supports our finding that game-based CAPTCHAS are generally preferred over text and image-based CAPTCHAS (see Figure 2.8).

Tanthavech and Nimkoompai [102] performed a 40-participant user study, measuring solving time for five CAPTCHA types: click-based reCAPTCHA, text-, game-, math-based, and a newly-proposed invisible CAPTCHA, which is essentially a honeypot for bots. In terms of solving times, their distorted text measurement (12 seconds) is in the middle of our observed range (9-15 seconds), which is expected since it closely resembles our *masked* type of distorted text. Similarly, their click-based reCAPTCHA measurement (3.1 seconds) is on the boundary of our range (3.1-4.9), which suggests they may have configured the “easier for users” setting. Their game-based CAPTCHA appears to have a lower solving time than ours, but this is likely due to the type of game. We did not observe or evaluate any math-based CAPTCHAS. They also asked participants several post-study questions about the five CAPTCHA types. Interestingly, their participants “enjoyed” the game-based CAPTCHA more than reCAPTCHA (click), which is the inverse of our findings (see Figure 2.8), but may again be due to the different types of game.

Overall, where our study measured similar quantities to prior work, our findings broadly agree. However, there is still a high degree of diversity in the sets of quantities measured in each study (e.g., types of CAPTCHAS, effect of experimental context), suggesting that a plurality of studies are needed to understand the full CAPTCHA landscape.

2.7.3 Summarized comparisons

In addition, Table 3.11 presents a summarized comparison of our results with those of other prior studies.

Solving Time: Overall, the average solving time in our study ranged from 3.6 to 42.7 seconds per CAPTCHA, which is a larger range than that observed by Bursztein et al. [53] in 2010 (9.8 – 28.4 seconds) but is similar to the 2019 study by Feng et al. [60] (medians ranging from 5.0 to 47.1 seconds). Although direct comparison of solving times is not always meaningful, even for the same CAPTCHA type (e.g., due to differing implementations or difficulty settings), we can identify a few trends. Firstly, our measured solving times for the three types of distorted text CAPTCHAS (9-15 seconds⁵) are within the range of observations from prior studies (6-20 seconds). We can therefore use this as a reference point for comparisons. Secondly, with the exception of behavior-based CAPTCHAS, we observed that all other CAPTCHA types took longer than distorted text. Without considering newly-proposed CAPTCHA types, this trend is consistent across most prior studies (with the exception of [60] and [102]). Thirdly, although we do not evaluate any newly-proposed CAPTCHA types, the times reported for these by other studies are typically faster than most of the CAPTCHA types in our study, suggesting that there is scope for developing new CAPTCHA types with lower solving times. Finally, even in comparison to newly-proposed schemes, the behavior-based CAPTCHAS (e.g. reCAPTCHA click) appear to have the lowest solving times overall.

Accuracy: For the case-sensitive setting, we observed a relatively broad range of accuracy (i.e., agreement) measurements for distorted text (50-84%). However, in the case-insensitive setting, our accuracy range narrows to 73-93%, which more closely aligns with prior studies, which have reported distorted text accuracies in the range 71-96%. This suggests that both participants and prior studies have focussed on the case-insensitive setting. In terms of

⁵Unless otherwise stated, measurements refer to average solving time.

deployed CAPTCHAS, [53] reported an accuracy of 93% for distorted text CAPTCHAS used by EBay in 2010. This is higher than for the image-based CAPTCHAS we measured (71-81%), suggesting that the latter may have increased in difficulty.

Security: Table 3.12 shows a comparison of our results to prior security analyses. Automated attacks on various CAPTCHA schemes have been quite successful [115, 55, 101, 64, 79, 67, 50, 42, 92, 52, 96, 100, 59, 75, 74, 45, 108, 80, 99]. The bots' accuracy ranges from 85-100%, with the majority above 96%. This substantially exceeds the human accuracy range we observed (50-85%). Furthermore the bots' solving times are significantly lower in all cases, except reCAPTCHA (image), where human solving time (18 seconds) is similar to the bots' (17.5 seconds). However, in the contextualized setting, human solving time rises to 22 seconds, indicating that in this more natural setting, humans are slightly slower than bots.

Table 2.5: Methodology and details of previous CAPTCHA-related user studies.

	Captcha types	Delivery medium	Measurements	Survey methods	Captcha source	Compensation (USD per # CAPTCHAS)
Ours	Text, Image, Game, Slider, Behavior	MTurk	Time, Agreement, Accuracy, Abandonment, Context	Demographics, Preference	Alexa	\$0.30-\$1.50 per 10
[53]	Text, Audio	MTurk, Website	Time, Agreement	Demographics	Alexa	\$0.02-\$0.50 per 24-39
[83]	DCG Captcha	MTurk	Time, Accuracy	Demographics, SUS	Newly proposed	\$0.50 per 4
[77]	reCAPGen Audio	MTurk	Time, Accuracy	Demographics, Rating/Preference	Newly proposed	\$4.00 per 60
[66]	3D/2D Text	MTurk	Time, Accuracy	Demographics, SUS	[82, 111, 110]	\$1.00 per 30
[72]	Text	MTurk, DevilTyper	Time, Accuracy, Abandonment	None	Major websites	\$0.03 per 15 (MTurk), 30.00 per 1.4 mil
[49]	Text, Audio, Interface	Website	Time, Accuracy	Demographics, Preference	Alexa	None
[61]	Text	Website	None	Demographics, Rating/Preference	Newly proposed	None
[65]	Jigsaw puzzle	Website	Time, Accuracy	Demographics, Preference	Newly proposed	None
[78]	Text, Game, NoBot	Website	Time	Workload, Perceptions, Preference	None	None
[60]	SenCAPTCHA, Text, Image, Audio, Video	MTurk	Time	Demographics, Preference, SUS	Newly proposed, [88]	\$1.25 per 9-15
[102]	Text, Behavior, Invisible, Game, Math	Unknown	Time	Demographics, Preference	None	None
[91]	Sketcha	MTurk	Time, Accuracy	Demographics	Newly proposed	\$0.05-\$0.30 per 10-12

Table 2.6: Comparison of results from prior user studies evaluating CAPTCHAS: audio (A), behavior (B), distorted text (DT), game (G), honeypot (HP), image (I), math (M), service (S), slider (SL), video (V) and newly-proposed (New). Some studies used non-unique (NU) participants or MTurk (MT). * denotes reimplemented CAPTCHA types.

	Unique users	Captchas solved	Average solving time (seconds)	Average accuracy
Ours	1,400 (MT)	14,000	9-15 (DT), 15-32 (I), 18-42 (G), 29 (SL), 3.1-4.9 (B)	50-84% (DT), 71-81% (I), 71-85% (B)
[53]	1,100-11,800 (MT)	318,000	9.8 (DT), 28.4 (A), 22.4 (S)	71% (DT), 31% (A), 93% (ebay DT)
[83]	120	480	8.5-16 (New), 17-47 (Attacks)	16-100% (New)
[77]	79	4,740	9.6 (New)	78.2% (New)
[66]	120	3,600	10 (3D-DT), 6.2-6.7 (DT)	84% (3D-DT), 92-96% (DT)
[72]	5,000 (NU), 44 (MT)	1.4 mil, 7,500	8.5-12 (DT)	79%-89% (DT)
[49]	162, 14 (Interface)	2,350	9.9 (DT), 50.9 (Blind DT), 22.8 (A)	80% (DT), 39-43% (A)
[61]	210	210	None	None
[65]	100	300	4.9-6.4 (New)	78%-87.5% (New)
[78]	87	261	20 (DT), 29 (G), 70 (NoBot)	None
[60]	436	4,920	12 (DT), 47 (A), 9.6 (I*), 5 (New), 12 (V*)	None
[102]	40	200	12 (DT), 0 (HP), 3.1 (B), 8.2 (G [114]), 4.1 (M [71])	None
[91]	558 (NU)	14,302	35 (New)	42%-88% (New)

2.8 Abandonment measurement

Tables 2.7, 2.8, 2.9, and 2.10 show the results from four groups of participants from the secondary study which aimed to measure abandonment. Columns represent the order of CAPTCHAS shown, while rows represent the CAPTCHA type. Cell values represent the number of MTurkers who abandoned.

Table 2.7: Abandonment in contextualized setting (\$0.75 payment)

	1	2	3	4	5	6	7	8	9	10	Total
reCAPTCHA (easy)	5	0	0	0	2	0	0	0	0	0	7
Geetest (slide)	3	1	2	1	3	0	0	1	1	1	13
Arkose (selection)	8	2	0	1	1	0	0	0	0	0	12
Arkose (rotation)	2	1	1	0	1	1	0	0	0	0	6
Distorted text (simple)	2	1	0	0	0	2	1	0	0	0	6
Distorted text (moving)	0	1	2	1	1	0	1	0	1	0	7
reCAPTCHA (difficult)	5	0	1	1	0	0	0	0	0	0	7
Distorted text (masked)	4	2	1	0	0	0	0	0	0	0	7
hCAPTCHA (easy)	2	2	2	0	1	0	0	0	0	0	7
hCAPTCHA (difficult)	4	1	2	1	0	0	1	0	0	0	9
Total	35	11	11	5	9	3	3	1	2	1	81

Table 2.8: Abandonment in contextualized setting (\$1.50 payment)

	1	2	3	4	5	6	7	8	9	10	Total
reCAPTCHA (easy)	2	1	0	0	0	0	0	0	0	0	3
Geetest (slide)	4	0	0	0	0	1	0	1	2	0	8
Arkose (selection)	1	2	0	0	0	1	0	0	0	0	4
Arkose (rotation)	4	0	1	0	0	0	0	1	0	0	6
Distorted text (simple)	2	0	0	1	0	0	0	0	0	0	3
Distorted text (moving)	1	1	1	0	0	1	1	0	0	0	5
reCAPTCHA (difficult)	2	1	0	0	0	0	0	0	0	0	3
Distorted text (masked)	1	2	0	0	0	0	0	0	0	0	3
hCAPTCHA (easy)	1	1	0	0	0	0	0	0	0	0	2
hCAPTCHA (difficult)	0	0	1	0	0	0	0	0	1	0	2
Total	18	8	3	1	0	3	1	2	3	0	39

Table 2.9: Abandonment in direct setting (\$0.30 payment)

	1	2	3	4	5	6	7	8	9	10	Total
reCAPTCHA (easy)	0	0	0	1	0	0	0	0	0	0	1
Geetest (slide)	1	1	0	0	1	0	1	2	0	0	6
Arkose (selection)	2	1	1	0	0	1	0	0	0	0	5
Arkose (rotation)	0	0	0	0	0	1	0	0	0	0	1
Distorted text (simple)	0	0	0	0	0	0	0	0	0	0	0
Distorted text (moving)	0	0	0	0	0	1	1	0	0	0	2
reCAPTCHA (difficult)	0	0	0	1	0	0	0	0	0	0	1
Distorted text (masked)	0	0	0	0	0	0	1	0	0	0	1
hCAPTCHA (easy)	1	1	0	1	0	0	0	0	0	0	3
hCAPTCHA (difficult)	1	0	0	1	0	0	0	0	0	0	2
Total	5	3	1	4	1	3	3	2	0	0	22

Table 2.10: Abandonment in direct setting (\$0.60 payment)

	1	2	3	4	5	6	7	8	9	10	Total
reCAPTCHA (easy)	0	0	0	0	0	0	0	0	0	0	0
Geetest (slide)	4	3	2	0	3	5	0	0	2	0	19
Arkose (selection)	0	0	1	0	0	0	0	0	0	0	1
Arkose (rotation)	1	0	0	2	1	0	0	0	0	0	4
Distorted text (simple)	0	0	0	0	0	0	0	0	0	0	0
Distorted text (moving)	1	0	0	0	0	0	0	0	1	0	2
reCAPTCHA (difficult)	0	0	0	0	0	0	0	0	1	0	1
Distorted text (masked)	2	0	0	0	0	0	0	0	0	0	2
hCAPTCHA (easy)	0	1	0	1	0	0	0	0	0	0	2
hCAPTCHA (difficult)	0	0	0	0	1	0	0	0	0	0	1
Total	8	4	3	3	5	5	0	0	4	0	32

2.9 Questions asked in User Study

Table 2.11 shows the exact questions that were asked to the participants during the pre- and post-study questionnaire.

Table 2.11: Questions in user study

Question	Possible Answers
<i>Pre-study questions</i>	
Age	18 - 100
Gender	Male, Female, Non-binary
What is your country of residence?	<i>[selected from list of countries]</i>
What is your highest level of Education?	No formal education, High School, Associate, Bachelor's, Master's, Doctorate
Which of the following most closely describes the majority of your Internet use?	Work, Education, Browsing the Web, Gaming, Other
Which device type are you using for this survey?	Phone, Computer (Desktop / Laptop), Tablet
Which input method are you using for this survey?	Touchscreen, Keyboard, Other
<i>[Only in the direct setting:]</i> Are you familiar with the purpose of CAPTCHAs?	Yes, No
<i>Post-study question</i>	
On a scale of 1-5, how enjoyable was solving the following CAPTCHA types? (1 being the least, and 5 – the most, enjoyable). If the CAPTCHA type wasn't shown to you please put a 0 in that place. Note: You may not have seen the exact images shown, they are templates designed to represent different CAPTCHA types.	<i>[single digit]</i>

2.10 Statistical Analysis of Solving Times

To confirm the validity of our conclusions, we conducted several standard tests on the measured solving times. We used the Holm-Bonferroni method to adjust for family-wise error in our statistical tests.

- First, we performed the *Shapiro-Wilk normality test* with a null hypothesis that solving times adhere to a normal distribution. For all CAPTCHA types, results showed that we can reject the null hypothesis ($p < 0.001$).
- Second, we ran a *skewness test* with a null hypothesis that the skewness of the sample population is the same as that of a corresponding normal distribution. For all CAPTCHA types, results allowed us to reject the null hypothesis in favor of the alternative: the

distribution of solving times is skewed ($p < 0.001$).

- Third, we used the *tailedness test* with a null hypothesis that the kurtosis of the sample population is the same as that of a normal distribution. Results showed that, for all except distorted text (moving), the samples were drawn from a population that has a heavy-tailed distribution ($p < 0.001$).

Since solving times are: (1) not normally distributed, and (2) heavy tailed, we selected the *Brown Forsythe test* to compare the equality of variance between different types of CAPTCHAS. Results show that these distributions do not have equal variance, thus confirming our observations in Section 2.5.1. Given the result of the Brown Forsythe test, we selected the *Kruskal-Wallis test* to test the equality of mean. For two pairs: reCAPTCHA (easy image) - hCAPTCHA (easy) and reCAPTCHA (easy click) - (hard click), we didn't see any statistical evidence that the means differ. For the remainder, this test showed strong statistical evidence that the means differ ($p < 0.05$ between masked and moving distorted text and $p < 0.001$ for all other combinations).

2.11 Captcha Solving Times for Other Demographic Features

Figures 2.13 and 2.14 show participants' solving times analyzed across other demographic features.

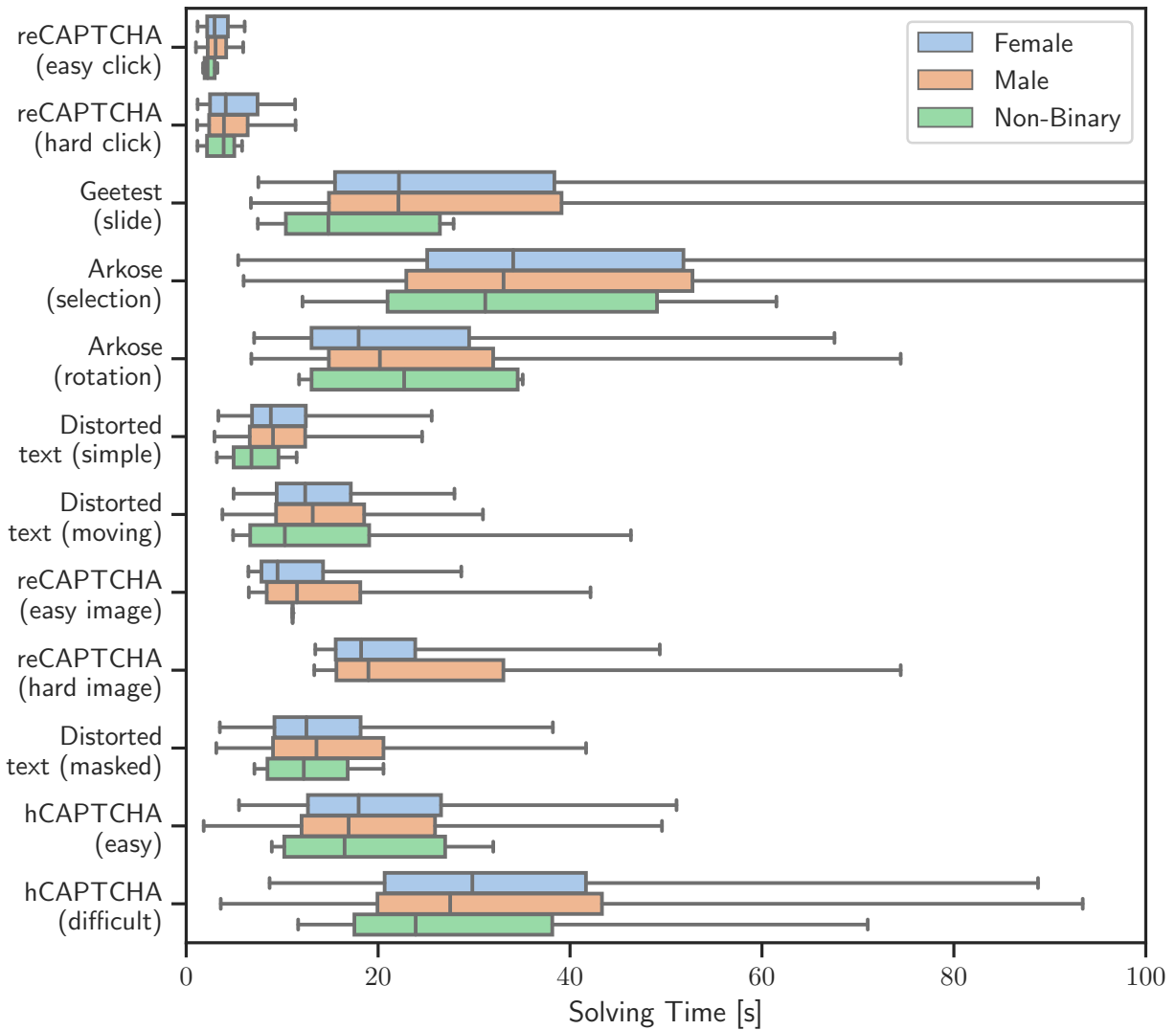


Figure 2.13: Effects of Gender.

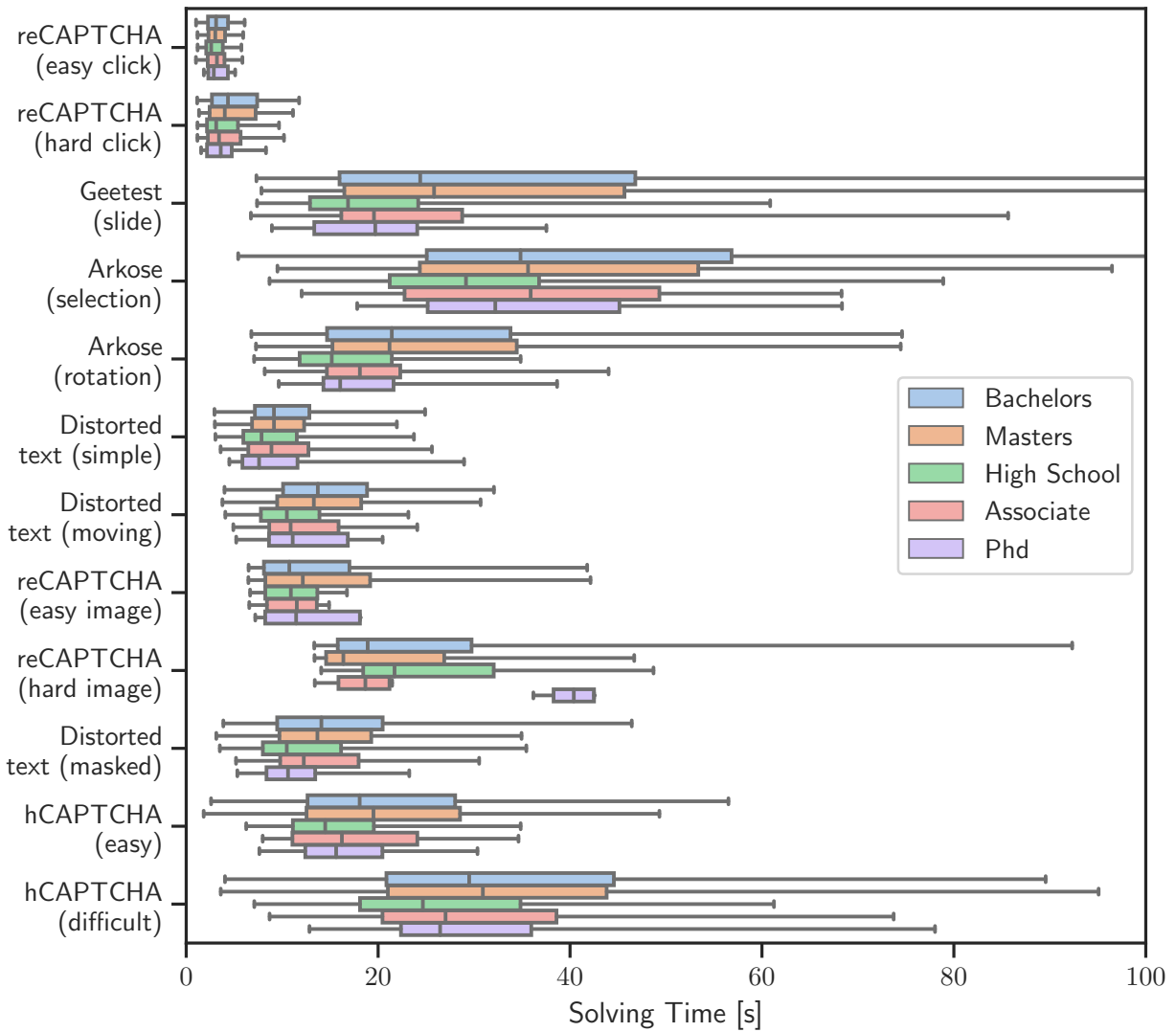


Figure 2.14: Effects of Education Level.

2.12 Summary & Future Work

This chapter explores currently-deployed CAPTCHAs via inspection of 200 popular websites and a series of user studies totalling 1,400-participants. For the research questions we posed at the outset, our results:

RQ1: show that there are significant differences in mean solving times between CAPTCHA types.

RQ2: show that users' preference is not fully correlated with CAPTCHA solving time.

RQ3: show that experimental context significantly influences CAPTCHA solving times.

RQ4: confirm the previously-reported effects of age on solving time.

RQ5: confirm the high rates of abandonment due to CAPTCHA-related tasks and identify that experimental context impacts abandonment.

We anticipate several directions for future work, including obtaining detailed measurements through a controlled user study, and further investigating the causes of abandonment.

Chapter 3

Understanding reCAPTCHA v2 via a Large-Scale Live User Study

Abstract

Since about 2003, CAPTCHAS have been widely used as a barrier against bots, while simultaneously annoying great multitudes of users worldwide. As their use grew, techniques to defeat or bypass CAPTCHAS kept improving, while CAPTCHAS themselves evolved in terms of sophistication and diversity, becoming increasingly difficult to solve for both bots and humans. Given this long-standing and still-ongoing arms race, it is important to investigate usability, solving performance, and user perceptions of modern CAPTCHAS. In this work, we do so via a large-scale (over 3,600 distinct users) 13-month real-world user study and post-study survey. The study, conducted at a large public university, was based on a live account creation and password recovery service with currently prevalent CAPTCHA type: reCAPTCHA v2.

Results show that, with more attempts, users improve in solving checkbox challenges. For website developers and user study designers, results indicate that the website context directly influences (with statistically significant differences) solving time between password recovery and account creation. We consider the impact of participants' major and education level, showing that certain majors exhibit better performance, while, in general, education level has a direct impact on solving time. Unsurprisingly, we discover that participants find image challenges to be annoying, while checkbox challenges are perceived as easy. We also show that, rated via System Usability Scale (SUS), image tasks are viewed as "OK", while checkbox tasks are viewed as "good".

We explore the cost and security of reCAPTCHA v2 and conclude that it has an immense cost and no security. Overall, we believe that this study's results prompt a natural conclusion: *reCAPTCHA v2 and similar reCAPTCHA technology should be deprecated.*

3.1 Introduction

Many types of Internet-based activities and services require verification of human presence, e.g., ticket sales, reservations, and account creation. Left unchecked, bots will gobble up most resources available through such activities: they are much faster and way more agile than any human or a group thereof. This problem is not new: the first seminal step to combat it took place in 2003 when von Ahn et al. [105] proposed CAPTCHA as an automated test that is supposed to be easy for humans to pass, yet difficult or impossible for computer programs (aka bots) at the time. The key conjecture underlying the CAPTCHA concept is that, if a computer program successfully solved CAPTCHAS, then the same program could be repurposed to solve some computationally hard AI problem.

This seemed to be a win-win situation: either CAPTCHAS attest to genuine human presence or they spur a significant advance in AI technology. Furthermore, CAPTCHAS were touted as a tool for the common good, since human-based solutions helped with difficult (for computers) and useful tasks, such as recognizing blurred text that confounded OCR algorithms, or labeling photos with names of objects appearing in them in order to aid image classification.

Another major advance occurred in 2007 when von Ahn et al. introduced reCAPTCHA [106]. reCAPTCHA was designed to reuse challenge results as a form of human-based data labeling for advancing machine learning. Google acquired reCAPTCHA in late 2009 [11] and, by June 2010, it was reported that reCAPTCHA had over 100 million distinct daily users [12]. Assuming that this number stayed constant since 2010 (though it most likely grew significantly), over half a trillion reCAPTCHAs have been solved in the meantime. This collectively amounts to an immense human cost.

However, almost from the start, an “arms race” began between bot and CAPTCHA developers. Most early CAPTCHA types were based on recognition of distorted text. Unfortunately, as a consequence of rapid advances in machine learning and computer vision, bots evolved to

quickly and accurately recognize and classify distorted text [112, 63, 70], reaching over 99% accuracy by 2014 [67, 97]. To this end, in 2012 Google switched from distorted text to image classification, using images from the Google Street View project [87]. This transition ended in 2014 with the introduction of reCAPTCHA v2 [13], which uses a two-step process: (1) a combination of behavioral analysis and a simple checkbox, and (2) image classification tasks as a fallback for users who fail the checkbox challenge [33]. By 2016, both (1) and (2) were defeated with a high degree of accuracy by bots [98].

Regardless of its diminished efficacy, reCAPTCHA remains to be the prevalent CAPTCHA type on the Internet [24], deployed on over 13 million websites in 2023. It is therefore important to periodically evaluate and quantify its impact in terms of usability, solving performance, and user perceptions.

Several prior CAPTCHA user studies explored solving performance, e.g., [94, 53, 49, 65, 91, 104, 83, 77, 66, 78, 61, 60, 72, 102]. Also, [83, 66, 60] looked into usability of CAPTCHAS via the well-known SUS scale. [94, 61, 78, 49, 102, 65, 77] studied user preferences related to CAPTCHA types. However, only two recent (2019/2023) user studies [102, 94] involved reCAPTCHA v2. However, [102] had relatively few participants (40), used unclear methodology, and did not consider usability. [94] presents interesting comparison points discussed in Section 3.5. Most other user studies [83, 77, 66, 61, 65, 60] were conducted on newly proposed (and therefore, mocked-up) CAPTCHA types.

Furthermore, many previous CAPTCHA studies [94, 53, 83, 77, 66, 72, 60] were conducted on Amazon Mechanical Turk (MTurk) [22], which exhibits data quality issues [107]. Also, all these studies involved some bias, since participants were informed about study goals, i.e., they were selected based on their willingness to solve CAPTCHAS, for a certain monetary reward.

The above discussion motivates the work presented in this chapter, the centerpiece of which is

a large-scale (> 3,600 participants) 13-month IRB-approved user study of reCAPTCHA_{v2}. The study was conducted using a live account creation (and password recovery) service with unaware participants who, for the most part, have never before used this service. Results of the study yield some interesting observations that might be of interests to CAPTCHA designers as well as websites using (or considering the use of) CAPTCHAS.

Main contributions of this work are:

- A comprehensive quantitative analysis of solving time and how it relates to certain dimensions. In particular, this is the first study to obtain multiple solving attempts per person. It shows that form-specific checkbox solving time improves with more attempts, with the first attempt being 35% slower than the 10th, shown in Tables 3.6 and 3.7. We also show statistically significant changes in checkbox solving time based on the type of service, with password recovery being faster, as shown in Tables 3.3, 3.4 and 3.5. With respect to educational level¹, there is a direct trend from freshmen (slowest) to seniors (fastest) at solving reCAPTCHA_{v2} as shown in Table 3.8. In terms of participants' major (field of study), there were minor trends with statistical significance of technical (aka STEM) majors solving time being faster than that of non-technical majors, as shown in Table 3.9.
- An in-depth qualitative analysis of reCAPTCHA_{v2} usability for both checkbox-only and checkbox-and-image combination. Results demonstrate that 40% of participants found the image version to be annoying (or very annoying), while 10% found the checkbox version annoying. SUS data shows that image results have a mid-score of 58, while checkbox has a score of 78, with 90 being the highest score observed. Based on the open-ended feedback represented in a *word cloud*, participants' most frequent term for the checkbox version was “**easy**” and, for the image version – “**annoying**”.
- A detailed discussion of the cost and security of reCAPTCHA_{v2} (Section 3.6). Our

¹In the American undergraduate system, “freshmen” are 1st-year students, “sophomore” – 2nd, “junior” – 3rd, and “senior” – 4th.

security analysis shows a blatant vulnerability [73], the ease of implementing large-scale automation [99], usage of privacy invasive tracking cookies [99], and weakness of security premise of fallback (image challenge) [62]. Our cost analysis investigates total human time spent solving reCAPTCHA_{v2}, human labor, network traffic, electricity usage, potential profits and the corresponding environmental impact. There have been at least 512 billion reCAPTCHA_{v2} sessions, taking 819 million hours, which translates into at least \$6.1 billion USD in free wages. Traffic resulting from reCAPTCHA_{v2} consumed 134 Petabytes of bandwidth, which translates into about 7.5 million kWhs of energy, corresponding to 7.5 million pounds of CO₂ pollution.

Organization: Section 3.2 provides some background on current CAPTCHA types and System Usability Scale (SUS). Then, Section 3.3 describes the methodology, design, ethics, and implementation of the user study. Next, Section 3.4 presents the results and their analysis. Then, Section 3.5 contextualizes our results against previous user studies. Next, Section 3.6 presents the cost and security analysis. Section 3.10 concludes the chapter.

3.2 Background

A recent survey by Guerar et al. [68] is a comprehensive overview of the current CAPTCHA landscape. It proposes a ten-group classification to encompass all current and emerging schemes: Text-based, Image-based, Audio-based, Video-based, Game-based, Slider-based, Math-based, Behavior-based, Sensor-based, and Liveliness-detection. It also discusses usability, attack resilience, privacy, and open challenges for each class. Since this chapter focuses on behavior and image-based CAPTCHAS (Which are used in reCAPTCHA_{v2}), we summarize them below. For the rest, we refer to [68].

Image-based captchas typically require users to perform an image classification task,

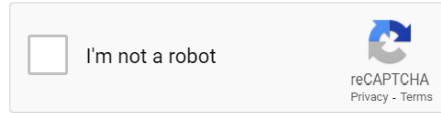


Figure 3.1: reCAPTCHA v2 checkbox CAPTCHA [33]

such as selecting images that match the accompanying written description. Most popular instances are hCAPTCHA [29] and reCAPTCHA [32] version 2 onward. The difficulty of these CAPTCHAs is associated with that of computer vision-based image classification. At the time of the introduction of these CAPTCHAs, corresponding problems were not easily solvable by machines. However as computer vision research advanced, attacks on image-based CAPTCHAs became more successful. Concrete attacks include [75, 74, 45, 108, 80, 99], some of which report success rates of 85% for reCAPTCHA and 96% for hCAPTCHA.

Behavior-based (or invisible) captchas are newer: they either require users to click a box (e.g., “I am not a robot”), or are completely invisible/transparent to the user. Instead of a visual challenge, they rely on client-side scripts and other opaque techniques to collect, in the background, historical behavioral information about the user. This information is sent to the CAPTCHA provider, which uses various heuristic-based techniques to identify bot-like behavior. For instance, Google’s popular No-CAPTCHA reCAPTCHA: *“actively considers a user’s entire engagement with the CAPTCHA – before, during, and after – to determine whether that user is a human”* [33]. Sivakorn, et al. [98], evaluate reCAPTCHA risk analysis system and determine that Google tracks cookies, browsing history, and browser environment, e.g., canvas rendering, user-agent, screen resolution and mouse. [98] also showed that legitimate cookies can be automatically farmed to attack reCAPTCHA v2 with 100% success on a large scale.



Figure 3.2: reCAPTCHA v2 Image Labeling Task CAPTCHA [32]

3.3 The User Study

Recall that the goals of the user study are to measure solving times, error rates, and user perceptions of reCAPTCHA v2, the currently prevalent CAPTCHA type.

3.3.1 The Setting

This study was conducted continuously over the period of roughly 13 months in the 2022-2023 time-frame. It took place on a campus of a large American University, though the scope was limited to one specific school. The term *school* denotes an organizational entity that includes two or more academic departments. The university contains a number of such schools, e.g., School of Engineering, School of Law, and School of Humanities.

The specific school hosting our study is called SICS: *School of Information & Computer Sciences*. SICS includes several departments, all somehow related to Computer Science. SICS offers a number of fairly typical undergraduate (BS) and graduate (MS and PhD) programs.

For many years, SICS requires for every person, who for the first time, enrolls in any SICS

course, to create a SICS-specific user account via the school’s web interface. A typical scenario is that a student who enrolls in at least one SICS course in their entire university career, would create a SICS account **only once**. Consequently, a student who wants to create a SICS account has not previously engaged in SICS account creation, meaning that they have no knowledge of the workflow involved, and no expectations of either seeing or not seeing CAPTCHAS as part of the process.

This motivates the key feature of our user study: introduction (insertion) of reCAPTCHA_{v2} into the SICS account management workflow. This actually involves two separate services: (1) account creation for new users, and (2) password recovery for users with existing accounts. This was accomplished with the much-appreciated help and cooperation of the SICS IT Department.

As mentioned earlier, the study ran for about 13 months. This is because we wanted to include as many distinct users as possible. Since the yearly academic calendar has multiple terms, we aimed to catch the beginning of each term (and a week or so prior to it), since this is the time when the bulk of new account creation and password recovery activity typically takes place.

3.3.2 Justification

We now discuss the rationale for the user study setting. Clearly, an ideal and comprehensive CAPTCHA user study would be as inclusive as possible, comprising a true cross-section of the world population. Whereas, our study targeted participants are (mostly) university students, including undergraduates who range from incoming (freshmen) to graduating (seniors), as well as graduate students enrolled in a variety of programs (MS, MA, MBA, MFA, JD, MD, PhD). The latter are split among so-called *professional* degree programs, e.g., MBA, JD, MD, and some MS/MA, while others are in regular degree programs, e.g., PhD, MFA, and some

MS/MA. Such participants are surely are not representative of the world, or even national, user population. Nonetheless, we conjecture that data stemming from this admittedly narrow population segment is useful, since it reflects an “optimistic” perception of CAPTCHAS. This is because young and tech-savvy users represent the most agile populations segment and the one most accustomed to dealing with CAPTCHAS, due to their heavy Internet use. Thus, by studying various (not generally positive) impact factors of CAPTCHAS, we prefer to err on the side of the population that is intuitively the least allergic to CAPTCHA use.

Some reasons for our study setting are fairly obvious. In particular, it would have been very challenging, if not impossible, to convince any other organization to introduce CAPTCHAS into its service workflow, or to allow us to collect data about their current CAPTCHA use. Alternatively, one could imagine approaching Google and requesting access to the centralized reCAPTCHA_{v2} service. This would have been ideal since it would give us access to a huge number of diverse reCAPTCHA_{v2} users worldwide. Indeed, we attempted to do this. However Google’s legal team denied our request to gain access to large-scale data from reCAPTCHA_{v2}. There is very likely a natural counter-incentive for Google (or any other CAPTCHA provider) to cooperate with outside researchers in a user study, since doing so might reveal certain negative aspects of the service. Another possibility would have been to create our own brand new service and use CAPTCHA to guard access to it, thus hoping to attract prospective users of broad demographics. While theoretically plausible, doing so would be prohibitively time and effort-consuming for academic researchers.

Finally, even with our somewhat narrow target demographic of university students, the user study could have been more latitudinal, i.e., it could span multiple universities in various parts of the world. This would have yielded more valuable results across political, cultural and linguistic boundaries. However, this would have been a massive effort requiring careful coordination with, and participation of, both researchers and IT departments in each university.

3.3.3 The Website

The SICS website used in the study is hosted within the university network. In order to create a SICS account, a user must first login to the campus VPN with their university account. This allows us to claim, with high confidence, that all collected data stemmed from real human users, who are, for the most part, students (see Section 3.3.5 below).

The back-end is a basic PHP server that serves HTML and JavaScript. It is maintained by SICS IT department. The account creation service includes a form requesting basic student information, e.g., name and student ID. The password recovery service includes a form requesting existing account information. In both cases, reCAPTCHA_{v2} was initially hidden and rendered after clicking the submit button. Basic website workflows for account creation and password recovery are described in Section 3.8.

All timing events were measured using JavaScript native Date library, which has millisecond precision. JavaScript was used to block form submission, such that an initial timing event is recorded and a reCAPTCHA_{v2} is rendered simultaneously. Initially, a behavior-based click box is presented. In order to solve it, a user clicks the checkbox sending data to Google reCAPTCHA_{v2} site. It either approves the request or presents an image-based challenge. Upon reCAPTCHA_{v2} validation, a second timing event is captured and the form is submitted.

Solving time is thus comprised of the time interval starting from CAPTCHA rendering until the client browser receives a successful validation response from Google reCAPTCHA_{v2} service. (This includes image challenges and failed solution attempts.) Upon successful form submission, the IT database stores these two timestamps along with the form information.

3.3.4 Directory Crawler

Recall that the study involved unwitting participants, i.e., unaware of both existence and purpose of the study. In order to subsequently obtain demographic information about each participant, we created a JavaScript crawler that automatically searches the university directory using email addresses. This directory is publicly available from both inside and outside the university network. Information gathered by the crawler includes major and college education level (freshman, sophomore, junior, senior, or graduate) of each participant.

3.3.5 Logistics & Data Cleaning

In total, the SICS IT department supplied 9,169 instances of account creation and password recovery with reCAPTCHA_{v2} solving time data. The original form data was larger, since it included errors, such as incomplete forms and incorrect values. Each record (form) has the following fields: database ID, date and time, student ID, email address, service, and timing. Starting with 9,169 instances, we filtered results using the directory crawler, labeling entries with student IDs that were not found and correcting student IDs with minor typos. A total of 229 entries were labeled as none for student ID and 295 student ID typos were corrected.

Successful form submissions have certain constraints, e.g., field formatting. If a person enters erroneous data that does not fit the constraints, they still have to solve a reCAPTCHA_{v2} before the form is submitted. Cases of multiple submissions occurred because of unsuccessful attempts to enter form data. For some entries, there were small typos, though mixed with temporal evidence they were correctable.

28 records were removed, since each had solving time of > 60 seconds which adds a high degree of variance. We ended up with 9,141 valid records of which 8,915 correspond to 3,625 unique participants. 226 entries, labeled as none for student ID, are not included among the

unique participants, attempts, educational level, and major analysis. Of the 8,915,231 form submissions correspond to 52 unique non-students (i.e., faculty or staff) and are not included in the educational level and major analysis. For the purposes of the educational level and major analysis, 3,573 unique students completed 8,631 reCAPTCHA v2 challenges.

3.3.6 Post-Study Survey

After the completion of the study, we randomly selected and contacted, by email, 800 participants in order to solicit feedback on their reCAPTCHA v2 experience via a survey (a Google form). In the end, a total of 108 completed the survey. The incentive was an \$5 Amazon gift card. The survey collected answers to SUS questions regarding both checkbox and image CAPTCHAs. It also collected information about (more detailed) demographics, frequency and nature of internet usage, as well as preferences and opinions about checkbox and image CAPTCHAs.

3.3.7 Ethical Considerations

The user study was duly approved by the university’s Institutional Review Board (IRB). Collection of student email addresses for recruitment and demographic analysis purposes was also explicitly approved. Since prospective participants were not pre-informed of their participation in the study, two additional documents were filed and approved by the IRB: (1) *“Use of deception/incomplete disclosure”* and (2) *“Waiver or Alteration of the Consent”*. Study participants who completed the post-study survey were compensated US\$5 for about 5 minutes of their time. This was also IRB-approved.

No personally identifiable information (PII) was used in the demographics analysis.

After the completion of the study, **all** participants were informed, by email, of their partici-

pation and the purpose of the study. They were also informed that some basic demographic information about them that was collected via campus directory lookup.

3.4 Results & Analysis

This section presents the results of the user study based on the live service experiment. We consider both quantitative (solving time) and qualitative (SUS, rating, feedback) data to provide a comprehensive analysis of reCAPTCHA v2 usability.

3.4.1 University Demographics

Student population of the university is large and diverse. We use university demographics, because students from multiple departments who take any SICS course create accounts. Thus, demographics about SICS students would not be enough. Moreover, the university does not maintain or provide SICS-specific demographics.

According to recent statistics, the total number of students is $\approx 36,000$ of whom 54% are female, 44.6% are male, plus 1.4% are non-binary or unstated. In terms of ethnicity, the rough breakdown is: 34% Asian, 24% Hispanic, 17% international, 15.44% White, 2.23% Black, and 7.25% other ethnic groups. The split between undergraduate and graduate students is 78.10% to 21.9%.

As far as the educational level, freshmen constitute 14% of the student body, sophomores – 15%, juniors – 21%, and seniors – 28%. The rest ($\approx 22\%$) are graduate students. Interestingly, the age range of the student population is very wide, ranging from under 18 to over 64. Nonetheless, the majority (82%) fall into the 18 – 24 age range.

3.4.2 reCAPTCHA v2 Dashboard Data

Google provides reCAPTCHA v2 analytic data for website operators via a dashboard [27]. With it, website operators can generate a key-pair necessary for implementing reCAPTCHA v2 on a web page. Difficulty setting can also be chosen on the dashboard. We used the “easy” setting in all experiments. The admin console allows for data to be downloaded in CSV format with the following fields per day:

no CAPTCHAs, Passed CAPTCHAs, Failed CAPTCHAs, Total Sessions, Failed Sessions, Average Score, and Average Response Time.

Table 3.1: Google’s reCAPTCHA dashboard data

no CAPTCHAs (checkbox)	7629
Passed CAPTCHAs (Image)	1890
Failed CAPTCHAs (Image)	143
Total Sessions	9538
Failed Sessions	19
Image accuracy	92.96%
Behavior accuracy	79.98%

Average score and response time are highly sparse and only appear on days with over 400 total sessions. Table 3.1 shows a sum for all days when data was collected over the entire study period. The image accuracy of 93% is computed as:

$$\frac{(\#passed\ CAPTCHAS)}{(\#passed\ CAPTCHAS + \#failed\ CAPTCHAS)}$$

The behavioral accuracy of 80% is computed as:

$$\frac{(\#of\ CAPTCHAS)}{(total\ \#sessions)}$$

Notably, there are 9,538 CAPTCHA sessions reported by the admin console data, while we were supplied with 9,169 sessions, meaning that 369 form submissions has incomplete data

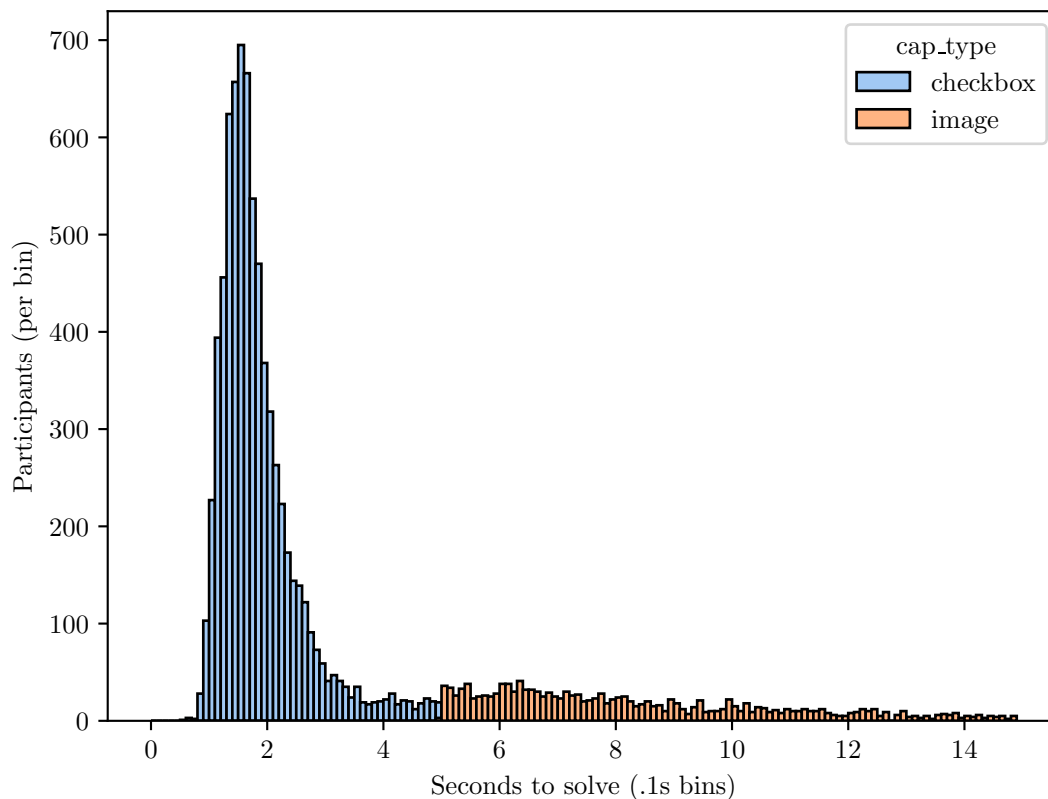


Figure 3.3: Timing results in bins of .1 seconds

or resulted in an error. This is likely due to incomplete sessions, e.g., refreshing before validation, or other form submission errors.

Table 3.2: Agglomerated solving time for reCAPTCHA Mode

mode	Count	Mean	Median	Std	Var	Max	Min
behavior	7334	1.85	1.67	0.71	0.50	4.99	0.51
image	1807	10.3	8.20	6.54	42.8	59.8	4.99
total	9141	3.53	1.83	4.50	20.3	59.8	0.51

3.4.3 Solving Time

Solving time for reCAPTCHA v2 is measured from the initial display to the successful verification. Data for solving time is split based on behavioral accuracy of 80% in Table 3.1.

Since all tasks require a checkbox and some also require an image task, we assume that the 80% fastest solving times correspond to checkbox interactions. This split is also noted in the recent work by Searles, et. al [94]. All timing for image-based results is therefore a combination of check-box and image tasks.

Table 3.2 shows the results of 7,334 behavior and 1,807 images based on this split. The mean solving time for behavioral CAPTCHAS is 1.85 seconds, while the image mean solving time is 10.3 seconds. The latter corresponds to a notable 557% increase.

Looking at Figure 3.3, there is a sharp drop-off in solving time starting around 2, and ending at 5, seconds: it hits a low and then goes back up slightly. The split point for image and behavior is about 5 seconds, which matches the drop-off point, thus strengthening the accuracy of the split. Figure 3.4 shows timing results after the image split. Notably, image and checkbox data follow similar patterns of distribution.

Solving time can also be partitioned into the following dimensions, based on collected data: Service, Attempts, Educational Level, and Major. For a description of the statistical methods used see Section 3.7.

Table 3.3: Checkbox solving time in seconds for each service

Service	Count	Mean	Median	Std	Var	Max	Min
Password Reset	2654	1.67	1.51	0.65	0.42	4.99	0.51
Account Creation	4680	1.96	1.76	0.71	0.51	4.97	0.86

Table 3.4: Image solving time in seconds for each service

Service	Count	Mean	Median	Std	Var	Max	Min
Password Reset	332	10.4	8.01	6.59	43.5	43.5	5.01
Account Creation	1475	10.3	8.23	6.53	42.7	59.8	4.99

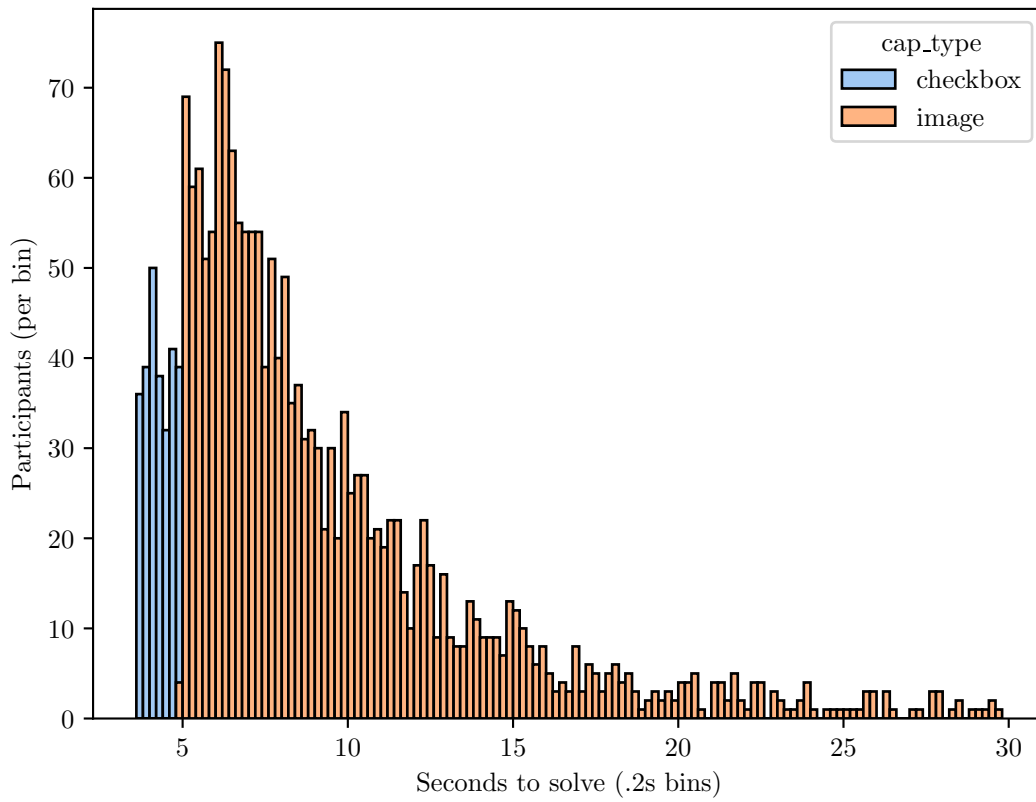


Figure 3.4: Image timing results in bins of .2 seconds

Table 3.5: Total solving time in seconds for each service

Service	Count	Mean	Median	Std	Var	Max	Min
Password Reset	2986	2.63	1.58	3.56	12.7	43.5	0.51
Account Creation	6155	3.97	2.00	4.84	23.4	59.8	0.86

Services

As mentioned earlier, the website had two services that invoked reCAPTCHA_{v2}: password recovery and account creation. Tables 3.3, 3.4, and 3.5 show results from these two CAPTCHA interactions. There were 6,155 account creation, and 2,986 password recovery, form submissions. Notably, for behavioral results, the Kruskal-Wallis test shows statistically significant differences between account creation and password recovery with a $p = 1.1e^{-115}$. Students who interacted with the account creation service solved behavioral CAPTCHAs 17% slower than those who interacted with the password recovery service. Additionally, 50% more time was spent solving reCAPTCHA_{v2} during account creation than during password recovery. Total results are also statistically significant with $p = 6.7e^{-162}$. However, since 90% of students who interacted with the latter have already interacted with the account creation service, these results may be conflated by multiple prior attempts. For the image case, the Kruskal-Wallis Test yielded no statistically significant results.

Attempts

Interestingly, some participants submitted forms multiple times. For behavior-based challenges, the average number of attempts was 3.52, and 1.73 for image-based ones. Tables 3.6 and 3.7 show timing results over multiple attempts. The highest number of attempts was 37 for behavior-based challenges and 20 for image-based ones. Behavioral results from the Kruskal-Wallis test in Figure 3.9 show that there is a statistically significant difference between the first and subsequent attempts ($p < .001$). While for the second attempt there is a

Table 3.6: Solving time for number of checkbox attempts

Attempt	Count	Mean	Median	Std	Var	Max	Min
1	2888	2.02	1.80	0.73	0.54	4.97	0.94
2	1293	1.84	1.67	0.65	0.42	4.97	0.62
3	751	1.80	1.63	0.66	0.44	4.95	0.80
4	513	1.73	1.55	0.63	0.40	4.89	0.78
5	371	1.73	1.57	0.70	0.49	4.92	0.89
6	272	1.61	1.47	0.58	0.34	4.57	0.84
7	212	1.67	1.52	0.65	0.43	4.90	0.64
8	167	1.66	1.52	0.65	0.43	4.65	0.84
9	127	1.60	1.48	0.57	0.33	4.09	0.88
10	112	1.56	1.44	0.63	0.39	4.97	0.85
11	94	1.63	1.41	0.76	0.57	4.90	0.88
12	67	1.61	1.46	0.68	0.46	4.47	0.51
13	52	1.58	1.37	0.70	0.49	4.49	0.96
14	37	1.53	1.45	0.63	0.40	4.62	0.92
15	28	1.51	1.41	0.56	0.31	3.88	0.88

Table 3.7: Solving time for number of image attempts

Attempt	Count	Mean	Median	Std	Var	Max	Min
1	1264	10.5	8.36	6.60	43.5	58.9	4.99
2	260	10.9	8.16	7.47	55.8	55.5	5.00
3	93	9.30	8.16	4.09	16.7	29.2	5.00
4	45	10.0	7.77	8.41	70.7	59.8	5.21
5	25	8.76	7.48	4.56	20.8	26.4	5.12
6	15	7.26	6.06	2.33	5.44	12.3	5.18

statistically significant difference ($p < .001$) between all other attempts except the third. In general, this data shows that checkbox solving time decreases with more attempts, meaning that humans improve at solving checkbox challenges.

We observe an interesting behavioral phenomena whereby participants react faster when they know what to expect. However, average image results show a slight increase on the second attempt, while subsequent attempts decrease. This may be attributed to reCAPTCHA v2 presenting a more difficult challenge on the second attempt. Image results from the Kruskal-Wallis test show no statistically significant differences between image attempts. This is likely

due to the drop-off in the number of participants who solved multiple image challenges.

Table 3.8: Total solving time for different educational levels

Level	Count	Mean	Median	Std	Var	Max	Min
Freshmen	773	5.15	2.33	5.69	32.4	56.2	0.95
Sophomore	1681	4.33	2.05	5.47	29.9	59.8	0.91
Junior	2246	3.09	1.77	3.84	14.7	45.4	0.51
Senior	2745	2.85	1.71	3.62	13.1	43.9	0.64
Graduate	1186	3.82	1.97	4.83	23.3	50.0	0.91

Educational Level

Educational level was obtained via the website crawler, as described in Section 3.3.4. Table 3.8 present data for different educational levels. In terms of statistical significance, Figure 3.10 shows statistically significant differences in total solving time for all educational levels. In terms of total time, freshmen are the slowest – 80% slower than seniors. There is a direct trend from freshman to seniors showing a reduction in solving time. Similarly, there is a trend of the total ratio of image to checkbox challenges.

Majors

Majors of the study participants (i.e., disciplines they study) were obtained through the website crawler, as described in Section 3.3.4. Table 3.9 presents solving times for participants with various majors. Although there are 62 majors in total, Table 3.9 only shows 22 majors. This is because each of the remaining 40 majors had < 20 reCAPTCHA_{v2} sessions. As the Kruskal-Wallis test in Figure 3.11 shows, only 8 majors had statistically significant differences in terms of checkbox solving behavior. Among these, Computer Science had the lowest, and Informatics – the highest, total average solving time.

Table 3.9: Total solving time for various majors

Major	Count	Mean	Median	Std	Var	Max	Min
CmptSci	3185	3.19	1.75	4.05	16.4	44.5	0.62
CSE	950	3.51	1.81	4.23	17.9	42.0	0.64
Undclrd	850	4.47	2.03	6.02	36.2	59.8	0.95
SW Engr	796	3.38	1.75	4.06	16.5	45.4	0.51
MCS	404	3.65	2.08	4.32	18.7	38.1	0.91
DataSci	362	3.98	2.02	4.55	20.7	41.2	1.03
IN4MATX	287	4.14	1.89	6.29	39.5	50.9	1.01
BIM	226	3.79	1.97	3.91	15.3	25.5	0.89
GameDes	186	4.38	1.86	7.03	49.4	56.2	0.77
Math	147	3.50	1.89	4.11	16.9	28.9	1.00
MofData	131	3.63	1.94	3.92	15.3	25.6	1.03
EngrCpE	106	3.93	1.98	4.31	18.6	20.1	0.98
PSW ENG	97	3.32	1.93	3.33	11.1	21.3	0.91
Bus Adm	89	2.85	1.83	2.55	6.52	13.1	0.88
CSGames	85	2.43	1.61	2.60	6.77	18.4	0.88
BusEcon	78	3.57	2.06	3.76	14.1	21.3	0.95
Bio Sci	75	3.90	1.87	4.80	23.0	23.0	0.99
Stats	65	3.70	1.68	4.02	16.2	23.9	1.11
Cog Sci	44	2.96	2.02	2.81	7.90	16.8	0.97
Net Sys	39	4.55	2.07	8.81	77.6	50.0	1.33
Engr ME	34	4.08	2.16	4.18	17.5	16.1	1.39
Psych	32	2.14	1.65	1.49	2.21	7.65	1.05

3.4.4 Survey Results

We now discuss the study results pertaining to usability, preferences, and opinions about reCAPTCHA v2. An interactive version of the google form we used is available at [31]. 800 randomly selected study participants were contacted by email, with the goal of obtaining at least 100 respondents. In the end, a total of 108 completed the survey. Two solving scenarios are considered:

Checkbox only Only the checkbox challenge: after clicking the checkbox, no image challenge was served. This applies to 42 participants.

Checkbox+image Both checkbox and image challenges: after clicking the checkbox, an

Table 3.10: SUS Scores for reCAPTCHA_{v2}

Solving Scenario	reCAPTCHA Type	SUS Score
Checkbox only	Checkbox	78.51
Checkbox+image	Checkbox	76.21
Checkbox+image	Image	58.90

image challenge was served. This applies to 66 participants.

System Usability Scale (SUS) Score Analysis

Table 3.10 reports the SUS score for both scenarios. Results from individual SUS statements are not analyzed, since they do not provide meaningful information [47, 51].

SUS checkbox scores are: 78.51 for checkbox only, and 76.21 for checkbox+image. Referring to adjective scaling [47], the usability level for checkbox in both scenarios is “Good”. We thus conclude that for checkbox, the SUS score and the usability level do not vary depending on the solving scenario, i.e., whether or not an image challenge is served afterwards. On the other hand, the SUS score of image is 58.90 and the usability level is “OK”. This difference is likely influenced by the difficulty of the task, since clicking a checkbox is surely much simpler than classifying an image. We observed that solving image challenges takes 557% longer than checkbox.

Preference Analysis

Participants were asked to provide a rating using a Likert scale. Figure 3.5 and Figure 3.6 show the preferences in both scenarios. The rating of checkbox is: 3.62/5 for checkbox only, and 3.68/5 for checkbox+image, scenario. Similar to the SUS score, the rating of checkbox is independent of the solving scenario. The rating of image is appreciably lower, at 2.84/5.

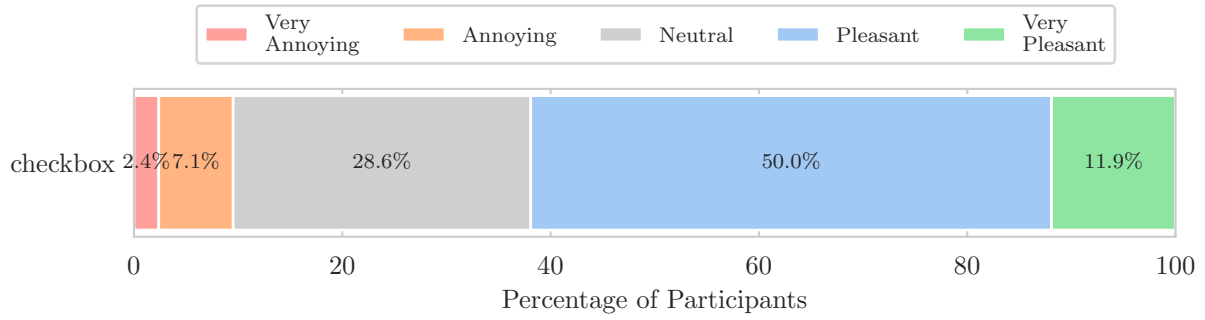


Figure 3.5: Preference score for checkbox only scenario

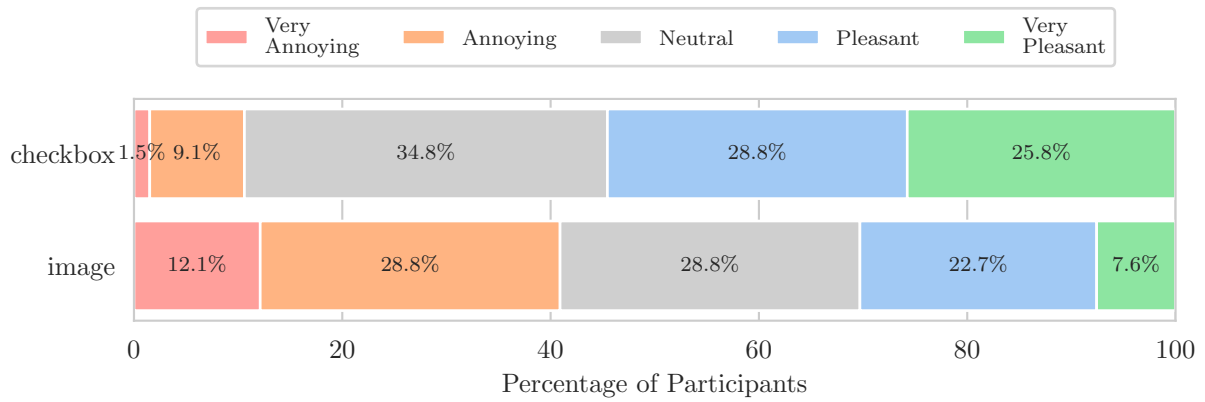


Figure 3.6: Preference score for checkbox+image scenario

are 1.7 to 26 times slower. For image reCAPTCHA_{v2}, our mean solving time is 10 seconds, which is 3.3 times slower than the fastest, and 5 times faster than the slowest, previously reported results. The fast solving time may be related to the trend noted in [94, 53] of age influencing solving time: younger participants seem to solve faster than older ones. Since our population is mostly university students (aged 18 – 25), our results re-confirm this trend.

Table 3.11: Comparison with results from prior user studies evaluating reCAPTCHA_{v2}: checkbox (C), image (I), total (T). Mean in seconds

Study	Unique users	reCAPTCHA _{v2} s solved	Mean	Accuracy
Ours	3,625	9,141	10.4 (I), 1.85 (C), 3.53 (T)	93% (I), 80% (C)
[94]	1,400	2,800	14-26 (I), 3.1-4.9 (C)	71-81% (I), 71-85% (C)
[102]	40	40	3.1 (T)	None

To the best of our knowledge, only two prior efforts studied reCAPTCHA_{v2}: [94] and [102]. Table 3.11 shows a direct comparison of the results. However, [102] provides a very limited data set of ($n = 40$) reCAPTCHA_{v2}, containing only total times. Whereas, [94], provides the following points of comparison:

1. Amazon Mturk *vs* “real world” participants
2. Participant awareness *vs* unawareness of the study existence and purpose
3. Mock *vs* real account creation;
4. Preferences/Rating

Webb et al. [107] reported several points of concern about the quality of data collected from MTurk [22]. Our data and results are derived from a real-world scenario of actual users creating real accounts for a real service. However, since both this work and [94] implement reCAPTCHA_{v2} in a similar way, some interesting conclusions can be drawn regarding the efficacy of Mturk data. Mturk users in [94] solved easy checkbox challenges 1.7 – 2.7 times slower than our participants. They also solved easy image challenges 1.6 – 2.6 times slower than our participants. Another consideration is network speed, since MTurkers

were participating in the [94] study over the Internet. In contrast, our study was conducted with most² participants being in close network proximity. Therefore, it would explain why Mturk results are slower since they can originate anywhere in the world, according to demographics reported in [94]. This may also skew our results to be faster than the actual total reCAPTCHA_{v2} solving time.

[94] showed that participants' awareness of the true purpose of the study could alter solving times. The solving time of participants who thought that they were participating in an account creation study was up to 57.5% slower than those who knew that they were participating in a study about solving CAPTCHAS. Account creation solving times (in seconds) for easy reCAPTCHA_{v2} of [94] are 4.9 for checkbox, and 26.3 for image. In contrast, our results are 2.02 for checkbox and 10.5 for image (on the first attempt). This translates into an average of 2.5 times slowdown for both challenge types. On the first attempt, our participants show the slowest mean and the least awareness (no study information presented) for checkbox challenges across significant groups ($n = 2,888$), and our results show that solving time improves with subsequent attempts. Whereas, in [94] participants solved 10 CAPTCHAS (among them 2 reCAPTCHA_{v2}s) in sequence, which could lower the timing due to the repeated attempts bias.

Also, [94] observed a lot of task abandonment, which might be due to the mocked-up (fake) account creation in that study. This is unlike our case where participants must create accounts due to the SICS school-wide policy. Thus, they must complete the form with successful post-validation by the back-end server. (In other words, abandonment is not an option).

[94] did not validate form information during account creation form submission beyond checking form field constraints, which could significantly alter the user study experience.

²Recall that VPN use was required to create an account or recover a password, thus taking part in our study. Although most participants were on campus, some were remote. The exact number of the latter is unknown.

Since our high average multiple attempts per participant of 3.52 for checkbox and 1.73 for image was likely due to failed post-validation by the back-end server.

Study participants in [94] rated reCAPTCHA v2 on a Likert scale, from “least enjoyable” to “most enjoyable”. Results showed that checkbox was the most enjoyable out of all CAPTCHAs, while image challenges were the least so. The term “enjoyable” is synonymous with pleasant (the opposite of “annoying”), which presents a point of comparison. Our results in Figures 3.5 and 3.6 are very similar in terms of positive and negative responses, thereby confirming results of [94]

3.6 Discussion

3.6.1 Cost Analysis

We now attempt to quantify various costs incurred by global use of reCAPTCHA on the internet. In particular we want to estimate the total time spent solving reCAPTCHAs, the overall amount of human labor, network traffic (bandwidth), power consumption and the consequent environmental impact. Note that, in the informal analysis below, we consider all estimates to be generous lower bounds.

Given that historic average solving time for distorted-text CAPTCHAs (same type used by reCAPTCHA v1) was 9.8 seconds and the conservative rate of 100 million reCAPTCHAs per day [12], 980 million seconds per day were spent solving reCAPTCHA v1s. For reCAPTCHA v1, it lived from 2009-2014 for 5 years amounting to 183 billion reCAPTCHA v1 sessions, taking 1.79 trillion seconds, or 497 million hours of human time spent solving reCAPTCHA v1. Given that the US federal minimum wage is \$7.5, this roughly yields \$3.7 billion in free wages.

Given that average solving time for all reCAPTCHA v2 sessions is 3.53 seconds and the conservative rate of 100 million reCAPTCHAs per day [12], 353 million seconds per day are spent solving reCAPTCHA v2s. For reCAPTCHA v2, it has been 9 years amounting to 329 billion reCAPTCHA v2 sessions taking 1.16 trillion seconds, or 322 million hours of human time spent solving reCAPTCHA v2. Given that the US federal minimum wage is \$7.5, this roughly yields \$2.4 billion in free wages.

Assuming un-cached scenarios from our technical analysis (see Section 3.9), network bandwidth overhead is 408 KB per session. This translates into 134 trillion KB or 134 Petabytes (194 x 1024 Terrabytes) of bandwidth. A recent (2017) survey [46] estimated that the cost of energy for network data transmission was 0.06 kWh/GB (Kilowatt hours per Gigabyte). Based on this rate, we estimate that 7.5 million kWh of energy was used on just the network transmission of reCAPTCHA data. This does not include client or server related energy costs. Based on the rates provided by the US Environmental Protection Agency (EPA) [38] and US Energy Information Administration (EIA) [25], 1 kWh roughly equals 1-2.4 pounds of CO2 pollution. This implies that reCAPTCHA bandwidth consumption alone produced in the range of 7.5-18 million pounds of CO2 pollution over 9 years.

In total from reCAPTCHA v1 and reCAPTCHA v2: There have been at least 512 billion reCAPTCHA sessions taking 2.95 trillion seconds, or 819 million hours, which is at least \$6.1 billion USD in free wages. Out of the 329 billion reCAPTCHA v2 sessions, (Our rate of 20%) at least 65.8 billion would have been image challenges, while 263.2 million would have been checkbox challenges. Thus 250 billion challenges would have resulted in labeled data. According to Google, the value of 1,000 items of labeled data is in the \$35-129 USD range [21], which would be worth at least \$8.75-32.3 billion USD per each sale.

Lastly, we look into the economics of tracking cookies, another main by-product of reCAPTCHA. Tracking cookies play an ever-increasing role in the rapidly growing online advertisement market. According to Forbes [9], digital ad spending reached over \$491 billion

globally in 2021, and more than half of the market (51%) heavily relied on third-party cookies for advertisement strategies [4]. The expenditure on third-party audience data (collected using tracking cookies) in the United States reached from \$15.9 billion in 2017 to \$22 billion in 2021 [7]. More concretely, the current average value life-time of a cookie is €2.52 or \$2.7 [81]. Given that there have been at least 329 billion reCAPTCHA v2 sessions, which created tracking cookies, that would put the estimated value of those cookies at \$888 billion dollars.

3.6.2 Security Analysis

In the following subsections 3.6.3 and 3.6.4, we discuss different attacks that have been performed successfully against reCAPTCHA. We consider behavior-based, image, and audio challenges in reCAPTCHA v2 and reCAPTCHA v3. Table 3.12 shows a direct comparison of the time and accuracy for humans and bots.

Table 3.12: Humans vs. bot solving time (seconds) and accuracy (percentage) for reCAPTCHA v2.

Type	Human				Bot	
	Time	Acc	Time	Acc	Time	Acc
checkbox	1.85	80%	3.1-4.9 [94]	85% [94]	1.4 [99]	100% [99]
image	10.4	93%	16-26 [94]	81% [94]	17.5 [75]	85% [75]

3.6.3 reCAPTCHA v2

reCAPTCHA v2 presents three different types of captcha challenges to the users: behavior-based (checkbox) challenge, image challenge, and audio challenge. Unfortunately, each of these captcha types has been proven vulnerable to attacks.

Checkbox Challenge

With the introduction of reCAPTCHA v2, came a new serious vulnerability in the form of click-jacking [73]. Adversaries can make "trustworthy" users generate g-recaptcha-responses, which can be automatically used to pass challenges, ultimately making a Bot's job infinitely easier!

Sivakorn, et al. [99] perform an in-depth analysis of the risk analysis system of reCAPTCHA and implement an attack to manipulate it. Based on this analysis and implementation:

1. Google primarily uses tracking cookies in the risk analysis system.
2. At least 63,000 valid cookies can be automatically created per day per IP address.
3. 9 days after a cookie creation, checkbox attempts using the cookie will succeed.
4. 52,000-59,000 checkbox challenges can be solved with 100% accuracy per day per IP address.
5. The average solution time is 1.4 seconds with 100% accuracy, shown in Table 3.12.

Given the blatant vulnerability [73], ease of implementing large-scale automation [99], and usage of privacy invasive tracking cookies reCAPTCHA v2 checkbox presents itself as a complete vulnerability disguised as a security tool. Google was previously sued 22.5 million for **secretly** adding tracking cookies to apple users devices [15]. It can be concluded that the true purpose of reCAPTCHA v2 is as a tracking cookie farm for advertising profit masquerading as a security service.

Image Challenge

Image-labeling challenges have been around since 2004 with the introduction of Image Recognition CAPTCHAS by Chew et al. [56]. 6 years later, in 2010 Fritsch et al. [62] published an attack that beat the prevalent image CAPTCHAS of the time with 100% accuracy. At this

point, it could be concluded that image recognition was no longer difficult to solve automatically with a computer. However in 2014 with the introduction of reCAPTCHA_{v2}, the fall-back security method was an image challenge, which had been proven insecure 4 years prior. The idea is that if your cookies aren't valuable enough then reCAPTCHA_{v2} would present an image labeling task. This wouldn't make sense as a security service, yet it would make sense given that obtaining labeled image data is highly valuable and is even sold by Google [21]. The conclusion can be extended that the true purpose of reCAPTCHA_{v2} is a free image-labeling labor and tracking cookie farm for advertising and data profit masquerading as a security service.

Consequently, [99] and [75] investigate and successfully implement automated solutions to reCAPTCHA_{v2}'s image labeling task. In 2016, [99] showed that a plethora of automated services, including Google's own Google Reverse Image Search (GRIS), could be used to automatically complete reCAPTCHA_{v2}'s image labeling tasks. [99] also implemented its own easy solver with 70.8% accuracy at 19.2 seconds per reCAPTCHA_{v2} image labeling task. In 2020, [75] also showed that many automated services, including Google's own Google Cloud Vision, could be used to automatically complete reCAPTCHA_{v2}'s image labeling tasks with reasonable speed and accuracy. [75] similarly implemented an attack, achieving a high level of speed (17.5 seconds) and accuracy (85%), shown in Table 3.12.

Audio Challenge

As part of reCAPTCHA_{v2}, Google introduced accessibility options allowing users to use audio CAPTCHAs, instead of image-based ones. Unsurprisingly, these audio CAPTCHAs introduce an accessibility side-channel, especially apparent due to advances in speech-to-text technology.

In 2017, Bock, et al. [50] introduced an automated system called *unCaptcha* which can solve

audio challenges with 85.15% accuracy and 5.42 seconds average solving time. Similar to other attacks [50] uses Google’s own voice recognition technology as a means to break audio challenges.

3.6.4 reCAPTCHA v3

reCAPTCHA v3 was introduced in 2018 [14] proposing the returning of a score, which website developers could use to decide whether to prompt with a challenge or perform some other action. Challenges types served by reCAPTCHA v3 are the same as reCAPTCHA v2. Also, there is no discernible difference between reCAPTCHA v2 and reCAPTCHA v3 in terms of appearance or perception of image challenges and audio challenges. Hence, attacks targeting reCAPTCHA v2 image/audio challenges are also applicable for those of reCAPTCHA v3. However, assuming that the risk analysis system was updated from reCAPTCHA v2 to reCAPTCHA v3, breaking behavior-based challenges of reCAPTCHA v3 might require new techniques. In 2019, Akrou, et al. [44] presented a reinforcement learning (RL) based attack breaking reCAPTCHA v3’s behavior-based challenges, obtaining high scores (.9+), with 97% accuracy and only requiring 2,000 data points as a training set.

3.7 Statistical Testing

For the sake of statistical validity, we apply a series of standard statistical tests to solving times. We run all of the following statistical tests on both image and checkbox data separately. Statistical methods were applied using python’s `scipy` [35] library. With a null hypothesis that solving times adhere to a normal distribution, we performed the *Shapiro-Wilk normality test*. For both image and checkbox cases, results showed that we can reject the null hypothesis ($p < 0.001$). With a null hypothesis that the skewness is the same as

that of a corresponding normal distribution, we ran the timing data with `skewtest`. For both image and checkbox, results reject the null hypothesis in favor of the alternative: the distribution of solving times is skewed ($p < 0.001$) to the right. With a null hypothesis that the kurtosis is the same as that of a normal distribution, we used the *tailedness test*. For both image and checkbox, results show the samples were drawn from a population that has a heavy-tailed distribution ($p < 0.001$). We used the *Brown Forsythe test* to compare equality of variance between image and checkbox, which shows that they do not exhibit equal variance. We used the *Kruskal-Wallis test* with the Holm-Bonferroni method to adjust for family-wise error in order to test the equality of mean between modes, services, attempts, majors, and educational level. Significant results are included in Figures 3.9, 3.10, and 3.11.

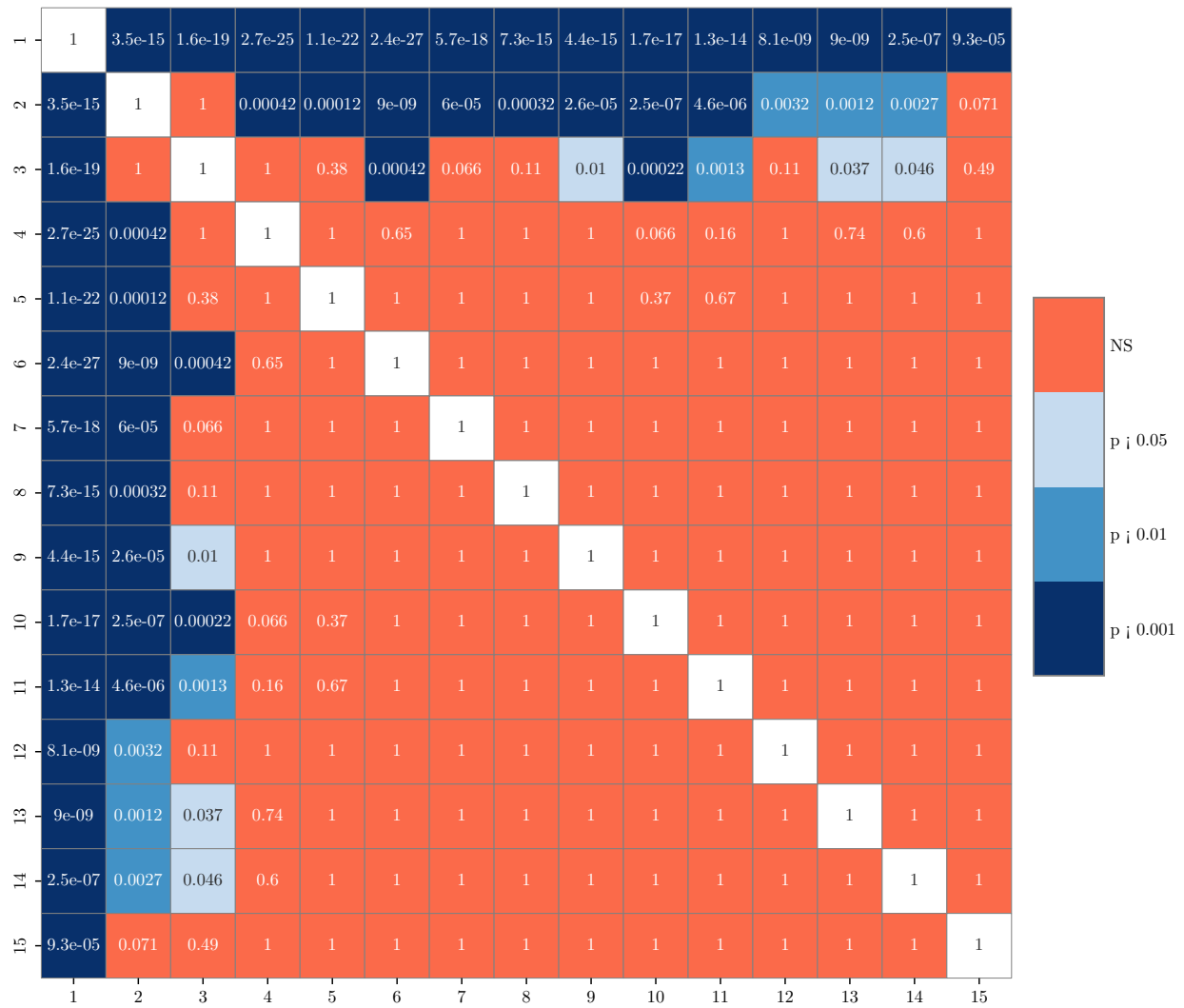


Figure 3.9: Kruskal-Wallis results for checkbox attempts

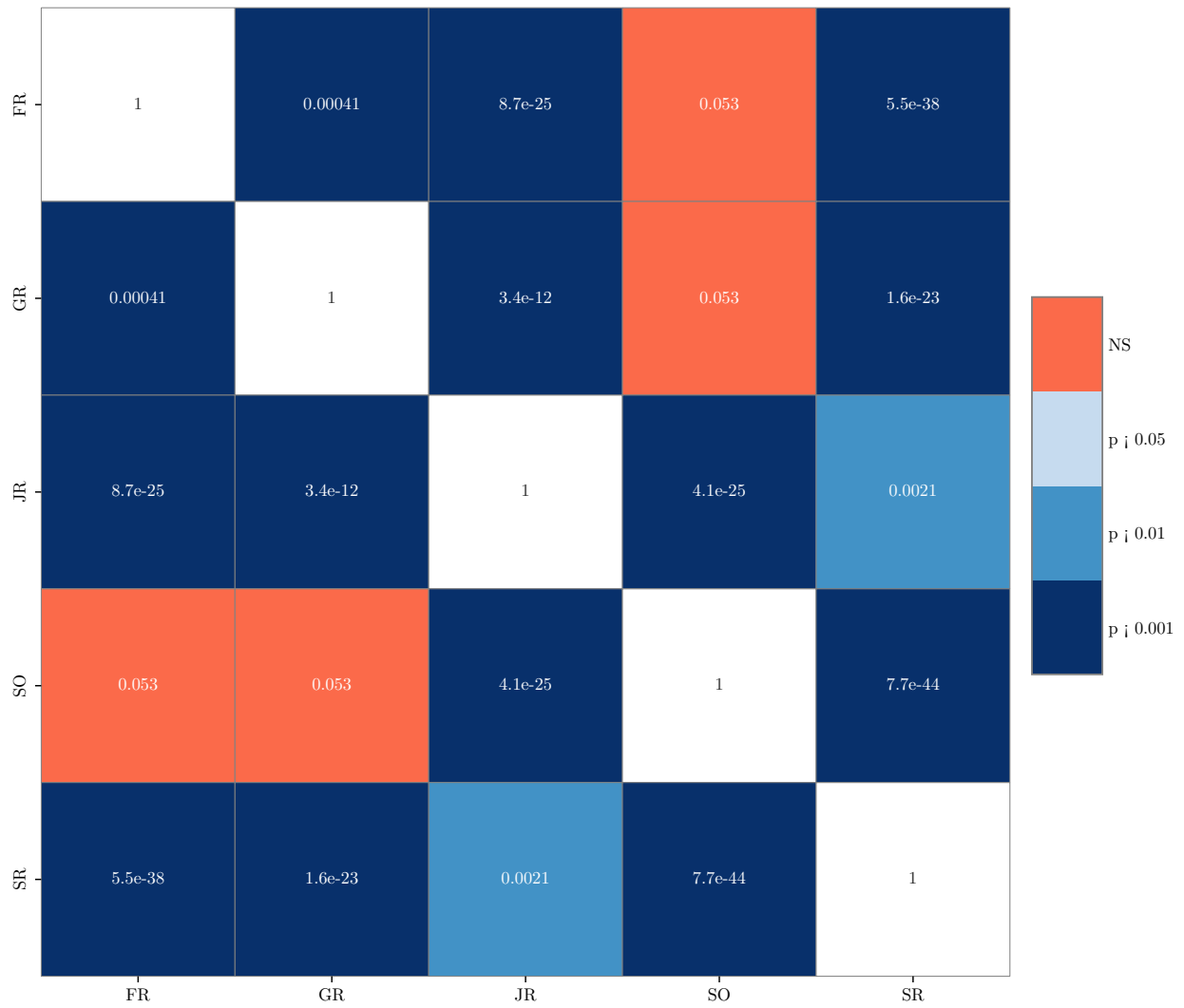


Figure 3.10: Kruskal-Wallis results for total attempts and educational level

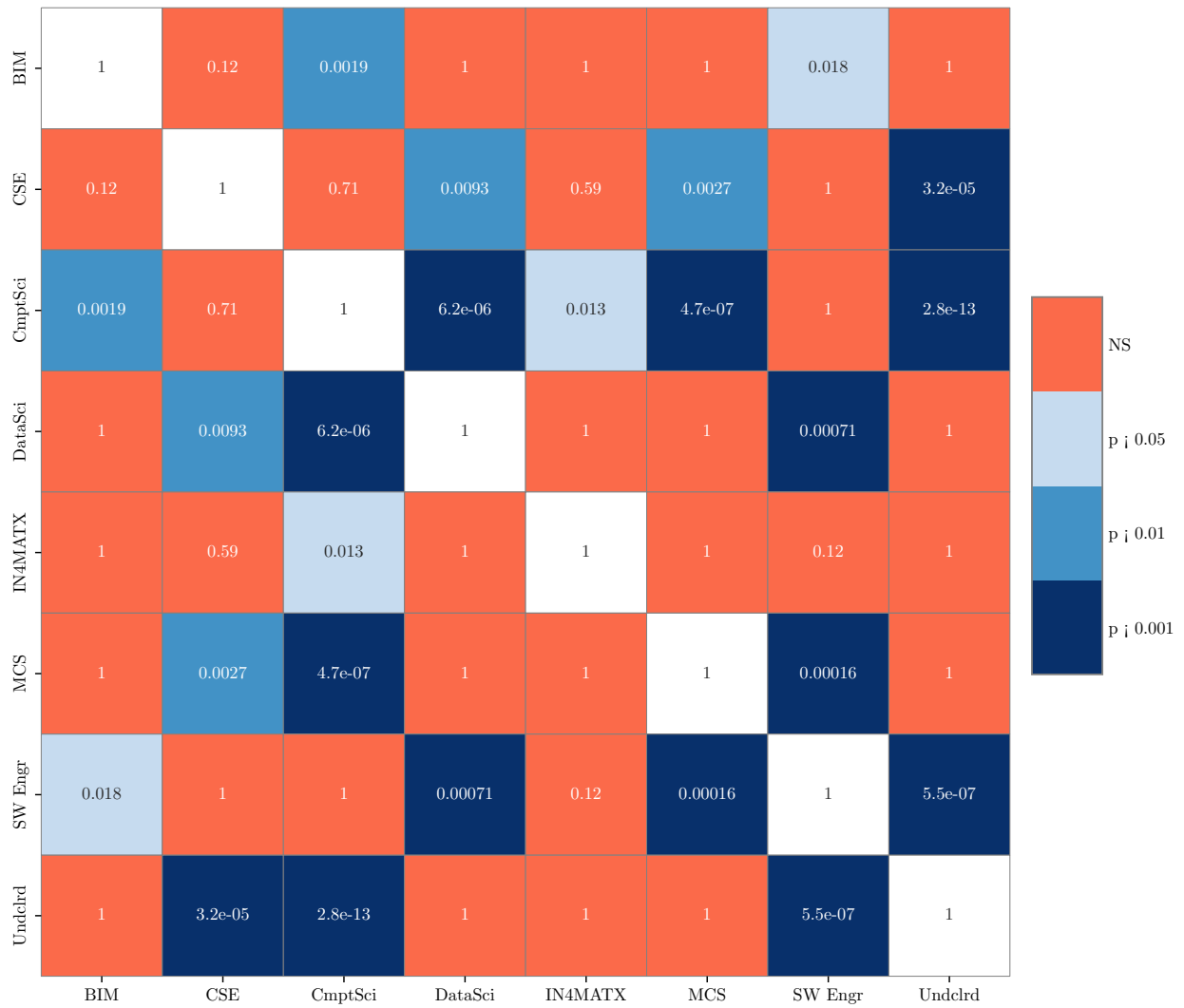


Figure 3.11: Kruskal-Wallis results for total attempts and major

3.8 Workflow and Additional Figures

In this Section we show basic workflows for account creation and password recovery processes that participants followed in the user study.

3.8.1 Account Creation

Figures 3.12, 3.13, 3.14, 3.15 constitute the workflow of the account creation process.

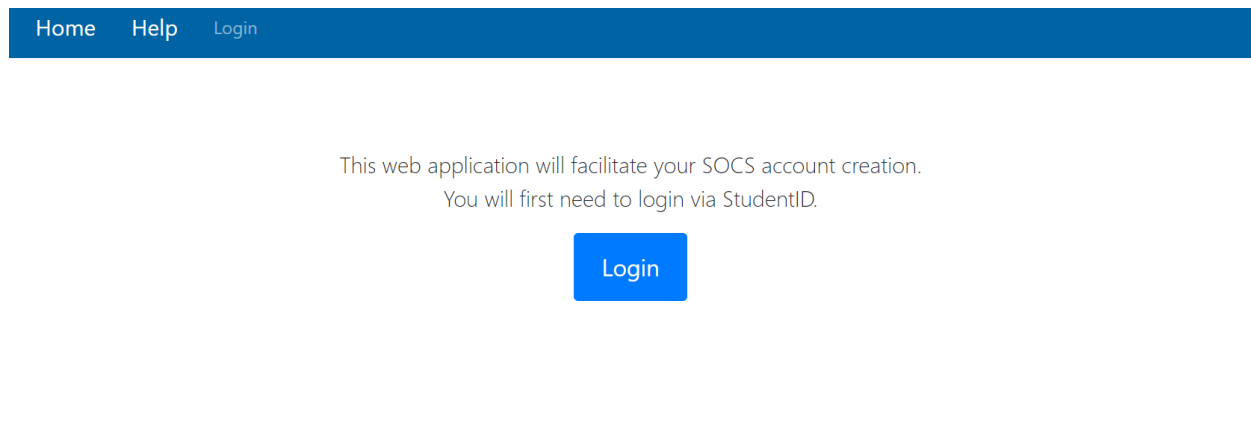


Figure 3.12: Initial login page

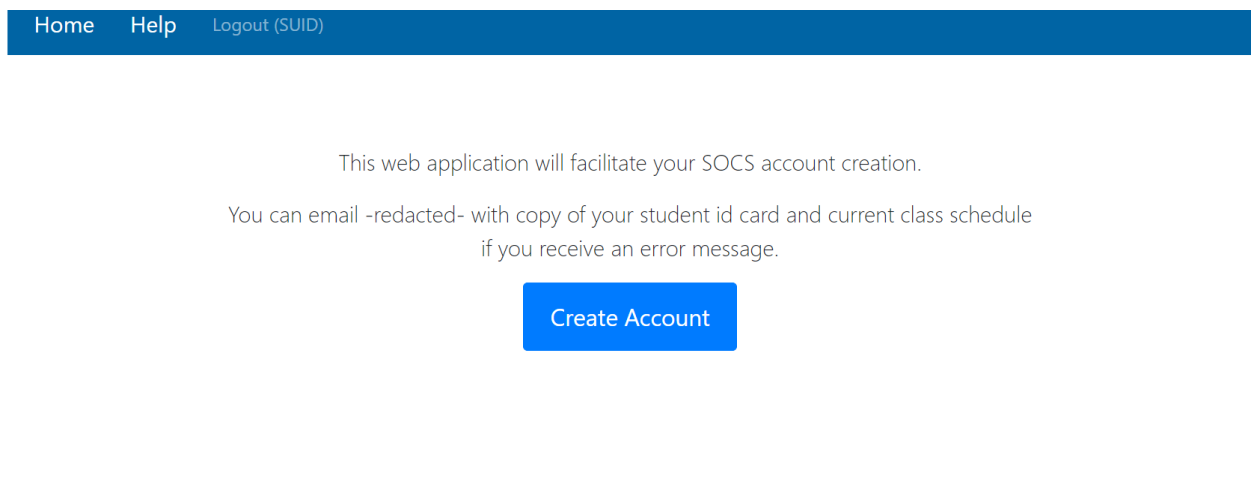


Figure 3.13: Initial Account Creation Page

Home Help Logout (jbond)

Account Creation

Now that you have logged in with your Email, please verify the information below and provide your University Student Number or Employee ID.

Name

Email

Affiliation

Student/Employee ID

Figure 3.14: Account creation form

Home Help Logout (jbond)

Account Creation

Now that you have logged in with your Email, please verify the information below and provide your University Student Number or Employee ID.

Name

Email

Affiliation

Student/Employee ID


I'm not a robot  reCAPTCHA
Privacy - Terms

Figure 3.15: AC form after clicking submit

3.8.2 Password Recovery

Figures 3.16 and 3.17 present the workflow of the password recovery process.

Self service password
 Question
 Email

redacted

Reset your password

Choose a question and answer it to reset your password. This requires that you have already [register an answer](#).

Your password must conform to the following constraints:

- Minimum length: 8
- Maximum length: 32
- Minimum number of lowercase characters: 1
- Minimum number of uppercase characters: 1
- Minimum number of digits: 1
- Minimum number of special characters: 1
- Forbidden characters (in addition to '*' and blank spaces): -_
- Your new password may not be the same as your old or be based on a dictionary word.
- Your new password may not be the same as your login or contain any two of the same consecutive letters.

Login	<input type="text" value="Login"/>
Question	<input type="text" value="What is your favorite ice cream flavor?"/>
Answer	<input type="text" value="Answer"/>
New password	<input type="password" value="New password"/>
Confirm	<input type="password" value="Confirm"/>

Figure 3.16: Password Recovery form

🏠 Self service password
🔔 Question
✉ Email

redacted

Reset your password

i Choose a question and answer it to reset your password. This requires that you have already [register an answer](#).

Your password must conform to the following constraints:

- Minimum length: 8
- Maximum length: 32
- Minimum number of lowercase characters: 1
- Minimum number of uppercase characters: 1
- Minimum number of digits: 1
- Minimum number of special characters: 1
- Forbidden characters (in addition to "*" and blank spaces): -_
- Your new password may not be the same as your old or be based on a dictionary word.
- Your new password may not be the same as your login or contain any two of the same consecutive letters.

Login	<input type="text" value="Login"/>
Question	<input type="text" value="What is your favorite ice cream flavor?"/>
Answer	<input type="text" value="Answer"/>
New password	<input type="password" value="New password"/>
Confirm	<input type="password" value="Confirm"/>

I'm not a robot

reCAPTCHA
Privacy - Terms

Figure 3.17: Password Recovery form after clicking submit

3.9 Network Analysis of reCAPTCHA v2

This Section contains a high-level technical analysis of reCAPTCHA. It has been considered in [99], which described the display method and workflow of reCAPTCHA with the emphasis on security aspects. Whereas, our goal is to: (1) determine various overhead factors incurred whenever a web-page uses reCAPTCHA, and (2) investigate reCAPTCHA’s automation detection capability. To this end, we performed black box program and network traffic analyses for common usage scenarios. We used two simple web pages for this purpose:

- Baseline page without any CAPTCHAS. This page is called *simple.html*.
- A page similar to the baseline page, except with an additional reCAPTCHA. This page is called *recaptcha.html*. As evident from the figure, integrating reCAPTCHA into a web page is very easy and straightforward.

These pages were visited using Google Chrome browser [18] and each usage scenario was repeated at least ten times. Browsing was performed in both guest and normal (profile logged-in) modes. Relevant information in the format of a *.har* file was collected for each scenario using Chrome DevTools.

The rest of this section describes the findings. Notations are summarized in Table 3.13.

Table 3.13: Notation Summary

Notation	Description
g1	https://www.google.com/recaptcha
g2	https://www.gstatic.com/recaptcha/releases/vkGiR-M4noX1963Xi_DB0Jel
g3	https://www.gstatic.com/recaptcha/api2/
g4	https://www.google.com/recaptcha/api2/
g5	https://fonts.gstatic.com/s/roboto/v18/
dv	different values

3.9.1 Page load Latency

Table 3.14 shows additional API calls made while loading *recaptcha.html* webpage.

Table 3.14: reCAPTCHA API Calls during page load

Request URL	Content-Length (B)
g1/api.js	554
g2/recaptcha__en.js	166822
g4/anchor?ar=[dv]	27864 (average)
g2/styles__ltr.css	24605
g2/recaptcha__en.js	166822
g3/logo_48.png	2228
g4/webworker.js?hl=[dv]	112
g2/recaptcha__en.js	166822
g4/bframe?hl=[dv] &v=[dv]&k=[dv]	1141-1145
g2/styles__ltr.css	24605
g2/recaptcha__en.js	166822
Network Overhead	254.01 KB-316.64KB

There are also 2-to-6 calls to g5 for downloading various web fonts. Content length for each of these calls is 15340, 15344, and 15552 bytes. Even though multiple calls are made to download *recaptcha__en.js* and *styles__ltr.css*, only the first call downloads the file, if necessary. These observations are taken into account when computing network overhead in Table 3.14.

Moreover, *api.js*, *recaptcha__en.js*, *styles__ltr.css*, *logo_48.png*, and web fonts are often served from the cache. Table 3.14 provides an upper bound on network overhead for page load. Average network overhead is computed by extracting actual network transmission during page load from collected *.har* files. Table 3.15 shows the results.

We investigated load latency using Chrome DevTools, pingdom.com [19], and webpagetest.com [20]. Table 3.16 presents the results. Latency computed using Chrome DevTools is the highest since Chrome DevTools determines the load time of *simple.html* and *recaptcha.html* in the same network where the concerned web pages are hosted. Observation shows that load

Table 3.15: recaptcha.html load network overhead

Scenario	Page Name	Page Size(KB)
First load	simple.html	0.631KB
First load	recaptcha.html	408.5KB
Network Overhead		407.869KB
Subsequent loads	simple.html	0.241 KB
Subsequent loads	recaptcha.html	29.56 KB
Network overhead		29.319 KB

latency increases as the distance between the user and the hosted webpage decreases (in terms of hops).

Table 3.16: recaptcha.html load latency

Measurement Tool	Page Name	load Time
Chrome DevTools	simple.html	51.16ms
Chrome DevTools	recaptcha.html	425.81ms
Time Overhead		374.65ms, 732.31%
pingdom.com	simple.html	375ms
pingdom.com	recaptcha.html	796ms
Time Overhead		471ms, 125.6%
webpagetest.org	simple.html	814.22ms
Subsequent Loads	recaptcha.html	2074.78ms
Latency		1260.56ms, 154.82%

3.9.2 Checkbox Click Overhead

Table 3.17 shows additional API calls made after checkbox is clicked. In this scenario, image CAPTCHA is not served to the user.

In some cases, only the first two calls are made. Even when other calls are made, files are normally served from the cache, so there is no network traffic. Files are downloaded only in the first-ever attempt to solve reCAPTCHA in a given client browser. Table 3.17 depicts upper and lower bounds for the network overhead.

Table 3.17: reCAPTCHA API Calls after checkbox click

Request URL	Content-Length (B)
g4/reload?k=[dv]	23844.67 (average)
g4/userverify?k=[dv]	580.56 (average)
g3/refresh_2x.png	600
g3/audio_2x.png	530
g3/info_2x.png	665
g5/[font].woff2	15552
Network Overhead	24.43 KB-41.77KB

Table 3.18: reCAPTCHA API Calls for image load

Request URL	Content-Length (B)
g4/reload?k=[dv]	24439.16667 (average)
g3/refresh_2x.png	600
g3/audio_2x.png	530
g3/info_2x.png	665
g4/payload?p=[dv]	39589.45455 (average)
Network Overhead	64.03 KB-96.72KB

3.9.3 reCAPTCHA Image load Overhead

Table 3.18 shows additional API calls made when checkbox is clicked and an image CAPTCHA is loaded. It also provides the upper bound and the lower bound of the network overhead due to these calls.

In some cases, two calls are made to g5 to download web fonts; content length is 15340 and 15552 bytes, respectively. Also, refresh_2x.png, audio_2x.png, info_2x.png, and web fonts are often served from the cache instead of being downloaded.

3.9.4 Image Solution Verification Overhead

Table 3.19 shows additional API calls made when an image CAPTCHA solution is verified. In case of a correct solution, only the third call from Table 3.19 requires network transmission

Table 3.19: reCAPTCHA API Calls for correct image solution

Case	Request URL	Content-Length (B)
Both	g3/refresh_2x.png	600
Both	g3/audio_2x.png	530
Both	g4/userverify?k=[dv]	595.88
Both	g3/info_2x.png	665
Wrong Solution	g4/payload?p=[dv]	40922.167 (average)
Correct Solution Network Overhead		0.6KB
Wrong Solution Network Overhead		41.58KB

Table 3.20: reCAPTCHA API Calls for reCAPTCHA expiration

Request URL	Content-Length (B)
g4/anchor?ar=[dv]	27864 (average)
g2/styles__ltr.css	24605
g2/recaptcha__en.js	166822
g3/logo_48.png	2228
g4/webworker.js?hl=[dv]	112
g2/recaptcha__en.js	166822
g4/bframe?hl=[dv] &v=[dv]&k=[dv]	1141-1145
g2/styles__ltr.css	24605
g2/recaptcha__en.js	166822
Network Overhead	29KB

and thus incurs network overhead. In case of a wrong solution, the last call from Table 3.19 is made, which requires network transmission and adds to network overhead. In both cases, other calls are usually served from the cache. In some instances, when a wrong solution occurs, only the third and fifth calls from Table 3.19 are made.

3.9.5 reCAPTCHA Expiration Overhead

Table 3.20 shows additional API calls made after a reCAPTCHA solution expires. Only the first and seventh calls (g4/anchor and g4/bframe) require network transmission and are considered in network overhead. Other calls are served from the cache.

Table 3.21: Summary of reCAPTCHA Network Overhead

Scenario	Network Overhead(KB)
First time Page Load	408.5
Subsequent Page Loads	29.319
Checkbox Click	24.43-41.77
Image Load	64.03-96.72
Image Correct Solution Verification	0.6
Image Wrong Solution Verification & New Image load	41.58
Solution Expiration	29

Summary: Results of evaluating network overhead for various reCAPTCHA usage scenarios are summarized in Table 3.21. As evident from these results, using reCAPTCHA incurs considerable network and timing overhead.

3.10 Summary

Over 13 years passed since reCAPTCHA's initial appearance and its current prevalence is undeniable. It is thus both timely and important to investigate its usability. This chapter presents a real-world user study with over 3,600 unbiased (unwitting) participants solving over 9,000 reCAPTCHA v2 challenges. We explore four new dimensions of reCAPTCHA v2 solving time: # of attempts, service type, as well as educational level and major. Results show that:

- Participants improve in terms of solving time with more attempts, for checkbox challenges.
- The service/website setting is an important consideration for researchers and web developers, since it has a statistically significant effect on solving time.
- Educational level directly impacts solving time.
- There were minor trends with statistical significance of participants with technical (STEM) majors solving time being faster than that of others.

In terms of usability, the post-study survey results show that the checkbox challenge gets an average SUS score of 77. This is considered to be acceptable and preferred by many participants over the image challenge, which has an average SUS score of 59. Notably, participants found the image challenge to be annoying.

In terms of cost, we estimate that – during over 13 years of its deployment – 819 million hours of human time has been spent on reCAPTCHA, which corresponds to at least \$6.1 billion USD in wages. Traffic resulting from reCAPTCHA consumed 134 Petabytes of bandwidth, which translates into about 7.5 million kWhs of energy, corresponding to 7.5 million pounds of CO2. In addition, Google has potentially profited \$888 billion USD from cookies and \$8.75-32.3 billion USD per each sale of their total labeled data set.

In terms of security reCAPTCHA_{v2} presents:

- Click-jacking (a blatant vulnerability) [73]
- Trivial implementation of large-scale automation attacks [99]
- Weakness of security premise of fallback (image challenge) [62, 99, 75]
- Usage of privacy invasive tracking cookies (for security) [99]

Ultimately, given these points it can be concluded that reCAPTCHA_{v2} presents no real security.

Given that: (1) reCAPTCHA_{v2} is negatively perceived by most users, (2) its immense cost, and (3) its susceptibility to bots, our results prompt a natural conclusion:

reCAPTCHA_{v2} and similar reCAPTCHA technology should be deprecated.

Chapter 4

Exploring CAPTCHA-induced Task Abandonment with QUITCHA

Abstract

For well over 20 years various types of CAPTCHAS have been ostensibly serving as a barrier against bot activity on the Web while simultaneously annoying great numbers of people worldwide. While early CAPTCHAS claimed positive societal benefits, such as unscrambling hard-to-parse text, broader benefits (if any) of modern CAPTCHAS are unclear mainly because they operate in a black box manner, thus no longer being “public”.

Despite many prior studies and explorations of various CAPTCHA types, fairly little attention has been paid to CAPTCHA-induced activity (session) abandonment by human users and factors contributing to it. To this end, we investigate the frequency of, and the reasons for, user abandonment when confronted with repeated CAPTCHA tasks. To do so, we construct a public-domain tool, called QUITCHA, and use it to perform a large-scale ($> 1,400$) user study on the Amazon Mturk platform. Study results yield a large scale dataset of image selection solutions shedding some light on the inner workings of the “black box” of image labeling tasks. The dataset is comprised of over $14k$ solutions: $75k$ image selections and $75k$ clicks, with over 20 columns of features. Analysis of this dataset shows significant trends surrounding CAPTCHA-induced abandonment, true solving time, image types, attempts, accuracy, and selections. Curiously, in the course of this study, we also discover an MTurk-based botnet.

4.1 Introduction

For over twenty years, CAPTCHAS (Completely Automated Public Turing test to tell Computers and Humans Apart) are serving as barriers to bot activity on the web. CAPTCHAS were first proposed by Von Ahn et al.[105] in 2003 as a way to distinguish bots from humans by using challenging AI problems. The main idea behind them is the creation of tasks easily solvable by human, while being very difficult for machines. This seems to be a win-win situation: either CAPTCHAS confirm human presence or they advance AI. Also, CAPTCHAS solved by humans could be used to improve machine learning models performing similar tasks, e.g., (1) Identifying letters or digits from low quality images to help Optical character recognition (OCR) models, (2) Identifying stop signs and traffic signals from roadside images to improve object detection models.

reCAPTCHA [106] is the first and most prominent CAPTCHA provider [24]. reCAPTCHA was proposed in 2007 and the resulting technology was acquired by Google in 2009. It used solutions of distorted text CAPTCHAS to digitize archives of The New York Times and books from Google Books [69]. However, due to rapid progress in computer vision and machine learning, bots became capable of easily solving distorted text CAPTCHAS [112, 63, 70]. By 2014, bots achieved over 99% accuracy for distorted text CAPTCHAS [67, 97]. As a result, this CAPTCHA type became obsolete. In 2012 reCAPTCHA switched to image selection tasks using images from the Google Street View project [87]. Then in 2014, reCAPTCHA_{v2} was introduced to minimize the level of human involvement needed to verify human presence [13]. reCAPTCHA_{v2} is a two-step process: (1) a combination of behavioral analysis and a simple checkbox click as the initial task, and (2) image classification tasks whenever behavioral analysis suspects potential bot behavior [33].

Meanwhile, techniques to defeat or bypass CAPTCHAS kept improving. Both behavioral analysis and image classification CAPTCHAS tasks were shown to be insecure [73, 98, 75].

Nonetheless, reCAPTCHA v2 is still the most widely used CAPTCHA type. Interestingly, image selection CAPTCHAS tasks were first broken in 2010, long before their adoption in reCAPTCHA v2. The first image selection CAPTCHA was introduced by Chew, et al. [56] in 2004. Then, in 2010, Fritsch, et al. [62] presented attack that broke image CAPTCHAS prevalent at the time with 100% accuracy.

Based on this history, it becomes clear that there is no solid security premise underlying image classification/labeling tasks. Rather, the real reason they persist is the profitability of labeled images and their role in training various AI models. This commercialization comes with the black box approach and removes the “**public Turing test**” aspect from CAPTCHAS.

To-date, there have been many CAPTCHA user studies [94, 53, 49, 65, 91, 104, 83, 77, 66, 78, 61, 60, 72, 102, 95]. They focused on various aspects, such as solving performance, usability, introduction of novel CAPTCHAS, user abandonment, and comparisons of CAPTCHA types. Bursztein et al. [53] conducted a large-scale user study involving over 318,000 CAPTCHA tasks solved by Amazon MTurkers [22] and an underground CAPTCHA-breaking service. Results showed that CAPTCHAS are often more difficult or time-consuming to solve than anticipated. Moreover, the study revealed various demographic patterns, such as non-native English speakers taking longer and being less accurate with English-centric CAPTCHAS. [53] also observed a loose correlation between time-to-annoyance and abandonment, i.e., CAPTCHAS with longer solving time are prone to higher abandonment rates.

More recently, Searles, et al. [94] performed a large-scale user study with 1,400 MTurkers. It evaluated users’ solving performance and perceptions of popular modern CAPTCHAS types. The study found that users’ perceptions and solving times were significantly different for various types of CAPTCHAS. Notably, perception did not always correlate with solving time. On the other hand, experimental context had significant impact on solving performance. The study also investigated the effects of experimental context and compensation on CAPTCHA-induced task abandonment. Moreover, it demonstrated that humans are slower than bots in

solving modern CAPTCHAS.

Furthermore, some prior work [83, 66, 60] investigated usability of CAPTCHAS using the well-known SUS scale. Other results [94, 61, 78, 49, 102, 65, 77] studied user preferences related to various CAPTCHA types.

Popular image CAPTCHAS, such as reCAPTCHA v2, function as a black box. reCAPTCHA v2 dashboard only provides the total number of CAPTCHAS and the percentage of image-based tasks. However, it does not specify whether a particular task involved an image-based task, or only a checkbox click, or whether the solution for a particular task was correct. Previous work [94, 102] that studied solving performance of reCAPTCHA v2 lacked fine-grained analytic data. As a result, these studies made assumptions about which tasks were actually image-based, and the solving time of image tasks had to be approximated with a small data sample.

Moreover, with respect to CAPTCHA-induced task abandonment, prior studies [53, 94] only investigated a very small subset of potential factors, such as solving time, experimental context, and compensation. They did not explore the factors leading to task abandonment, such as: why people quit, when they quit, how they quit, etc.

Motivated by the above, we design QUITCHA— an open-source full-stack image labeling CAPTCHA tool. It serves a custom image CAPTCHAS that resemble reCAPTCHA v2 image selection tasks. For this purpose, we use images scraped from actual reCAPTCHA v2 image challenges [28]. QUITCHA aims to discourage users from completing CAPTCHAS by presenting them with an seemingly endless series of image tasks, regardless of whether they have correctly solved the previous one(s). As part of this process, detailed event logging information is collected to measure user behavior, CAPTCHA solving time, and factors influencing abandonment. In practice, users must correctly solve 10 CAPTCHAS, in addition to CAPTCHAS they solved incorrectly. At the end of the task, or if a user quits and then

returns¹, a survey is presented. This survey includes questions about factors that motivated them to quit or continue solving, as well as their opinions on CAPTCHAS in general and their usability.

QUITCHA was deployed on MTurk, and 1,000 distinct MTurkers were recruited for the user study. Beyond previously mentioned factors, another goal of the study was to investigate the effect of the experimental context. To this end, 500 participants were pre-informed that they would take part in a study related to CAPTCHAS, while the remaining 500 were told that they would participate in a study on account creation. Using QUITCHA’s fine-grained event logging, we also collected data about users who started – and did not finish – the task. As a result, we obtained data from 1,457 unique participants in total: 1,000 who finished, and 457 who abandoned the study. In the longer term, besides results presented in this chapter, QUITCHA might benefit website providers by allowing them to craft their own labeled image sets, while still providing a means of rate limiting.

Research contributions of this work are discussed in detail in Section 4.2 below.

Organization: Section 4.2 introduces research questions and summarizes our main findings. Then, Section 4.3 discusses the design and implementation of QUITCHA components: Dataset, Challenges, Back End, and Front End. Next, Section 4.4 describes the timeline, logistics, and ethics of the study, followed by Section 4.5 which presents the results and their analysis. Then, Section 4.6 contextualizes our results with respect to previous user studies and Section 4.8 concludes the chapter.

¹Either by closing and reopening the page or refreshing it.

4.2 Research Questions & Results

We now present the research questions and summarize our main results. Table 4.1 shows how these results relate to prior work at a high level, with detailed comparisons in Section 4.6.

RQ1: What is inside the “black box” of image labeling tasks?

At best, main commercial CAPTCHA providers (CPs) currently only provide coarse-grained metrics (daily agglomerate) about solutions of CAPTCHAS. At worst, CPs provide no metrics at all for free versions by hiding user data behind an expensive paywall. For website operators, CPs do not provide a way to correlate data between users and their CAPTCHA solutions. CPs favor security-by-obscurity by not releasing exact data. The issue with this obscurity is lack of any solid foundation for any claimed security. The problem space has been reduced to: whoever has a large-scale corpus of labeled data can solve image CAPTCHAS with machine learning models. This motivates this study in part as a means to provide an open source image-labeling CAPTCHA, which can provide real time fine grained analysis for free. Results from this work show that QUITCHA captures and presents at least 40 times more data points with ≥ 20 more features than reCAPTCHA v2. Furthermore, QUITCHA provides high quality analysis for identifying statistical trends. Our results peek into the black box created by popular CAPTCHA services, such as reCAPTCHA v2. Ultimately, QUITCHA allows website operators to have an alternative means of image labeling CAPTCHAS without surrendering their users’ data to Google.

RQ2: What makes users quit, i.e., what factors (if any) influence session abandonment induced by image labeling captchas?

Prior studies [94, 53], reported some degree of abandonment. However neither study was designed to make participants quit or to measure the exact mechanisms of (and factors contributing to) the quitting process. We seek to capture this behavior with the design and implementation of QUITCHA: Querying Unending Image Tasks to Create Human Annoy-

ance. In it, we confront a user with many image tasks (10 minimum) in order to see if and when the sequence of challenges would cause abandonment. We collected detailed data from 758 participants who quit at different stages of the study and found that those who quit were less accurate in solving image challenges and took longer to interact with them. However, such participants made less number of attempts to solving image CAPTCHAS, as compared to participants who completed the entire task.

Table 4.1: Summary of research questions and main findings.

	Findings Confirming prior work	Findings Con- tradicting prior work	New findings
RQ1: What data is underneath the black box of image labeling tasks?			New fine-grained timing and accuracy events and features that exhibit statistically significant trends.
RQ2: What makes humans quit? Specifically, what factors (if any) influence image labeling (captcha) related abandonment?	(Repeated) CAPTCHAS cause users to quit.		Accuracy, interaction time, and the number of attempts impact quitting related behavior.
RQ3: What behavioral conclusions can be drawn from fine grained event logging?			Exact accuracy, selections, clicks, timing, and function tracing allow for re-creation of user interaction. All exhibit statistically significant results across certain features.
RQ4: How long do human users actually take to solve reCAPTCHA v2 like image labeling tasks?	Results help contextualize prior coarse-grained black box image labeling timings.		A large scale fine-grained image solution dataset and analysis tool set that quantifies various solution related timing events.
RQ5: What factors (if any) influence the solving time and accuracy?	Experimental context influences solving time.	Age does not impact solving time.	New features that exhibit statistically significant trends.

RQ3: What behavioral conclusions stem from fine-grained event logging? Prior work focused on coarse-grained events, e.g., total time to solve a CAPTCHA. In contrast,

QUITCHA tracks every relevant function call associated with the solving process. We create a fine-grained event logging system that records a list of all actions, relevant action data, and time when it occurred. This allows us to observe and recreate the exact interaction between users and QUITCHA. This rich data is then used to assess behavioral phenomena with a thorough multi-dimensional analysis. Results demonstrate several trends with statistical significance. We introduce 5 key timing measurements: checkbox solution time, image selection solution time, inter-click selection time, network delay, and reaction time. Comparing users we find that the true image type, quitting, experimental context, and attempts all impact (solution and selection) accuracy and timing.

RQ4: How long do users actually take to solve reCAPTCHA_{v2}-like image labeling tasks? Prior work [102, 94] explored reCAPTCHA_{v2} and approximated data only for a very small number of image challenge solutions. In contrast, this work seeks to fill that gap by mocking up reCAPTCHA_{v2} using QUITCHA. Instead of guessing whether total solution time is indicative of image challenges, QUITCHA’s fine-grained event logging allows us to quantify exact solution timings. These timings show: (1) time it takes to solve image challenges, (2) users’ accuracy in solving them, as well as (3) time it takes to select each image. We present a large-scale database that can be expanded into 4 separate granularities: total QUITCHA, single challenges, selection/un-selection events, and clicks. We observe 1,000 – – \geq 75,000 data points with 20 distinct features.

RQ5: What factors influence solving time and accuracy? Prior work [94, 53] showed that factors influencing solving time include age and experimental context. Fine-grained data collected in this QUITCHA-based study allows us to evaluate these (and other) trends via a combinatorial cross-dimensional analysis. Results show that accuracy and solving time are influenced by quitting, experimental context, attempt/sequence, and image type.

4.3 System Design & Implementation

We now present QUITCHA: Querying Unending Image Tasks to Create Human Annoyance. It is designed to measure fine-grained image selection CAPTCHA interaction via a full stack implementation of a mocked-up version of reCAPTCHA v2. The system stack consists of: a dataset, challenges, database, server, web page, analysis, and data processing scripts.

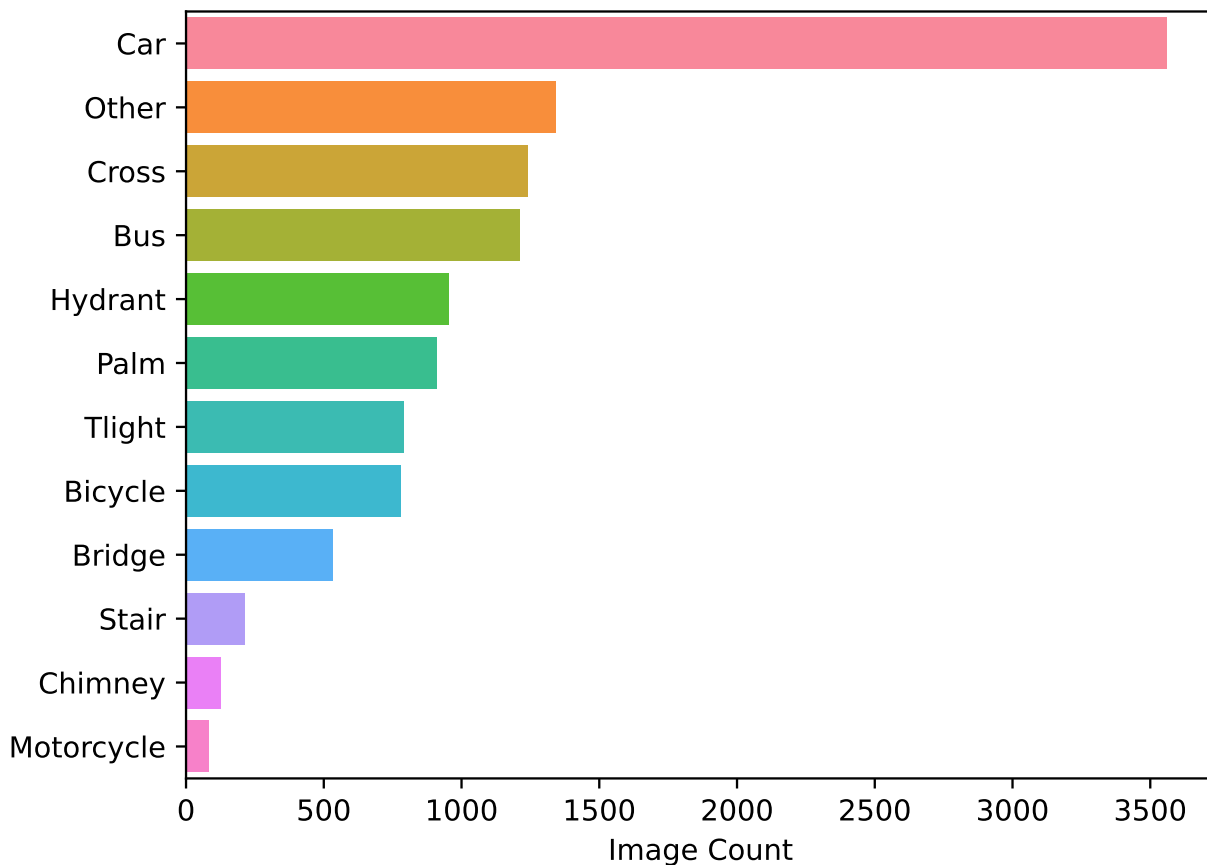


Figure 4.1: Distribution of image types in the dataset [28]

4.3.1 Dataset

The dataset [28] is comprised of roughly 12,000 images used in Google’s reCAPTCHA v2 collected by category. The dataset is free and open-sourced under the CC0: Public Domain

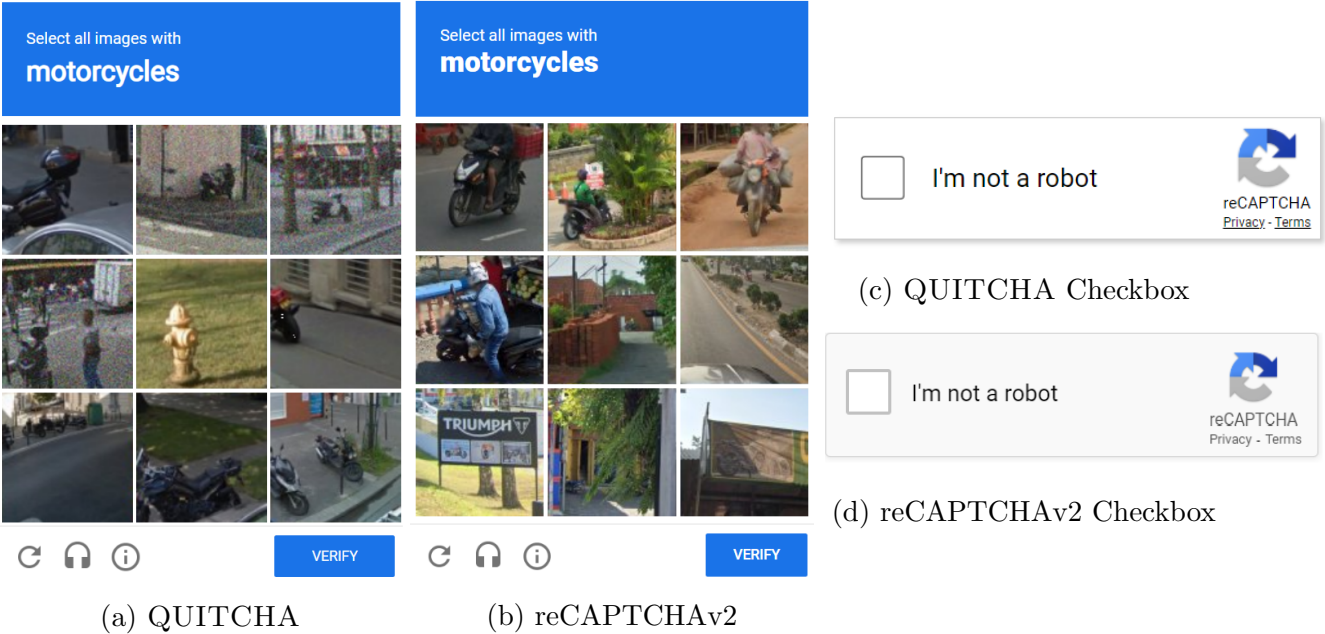


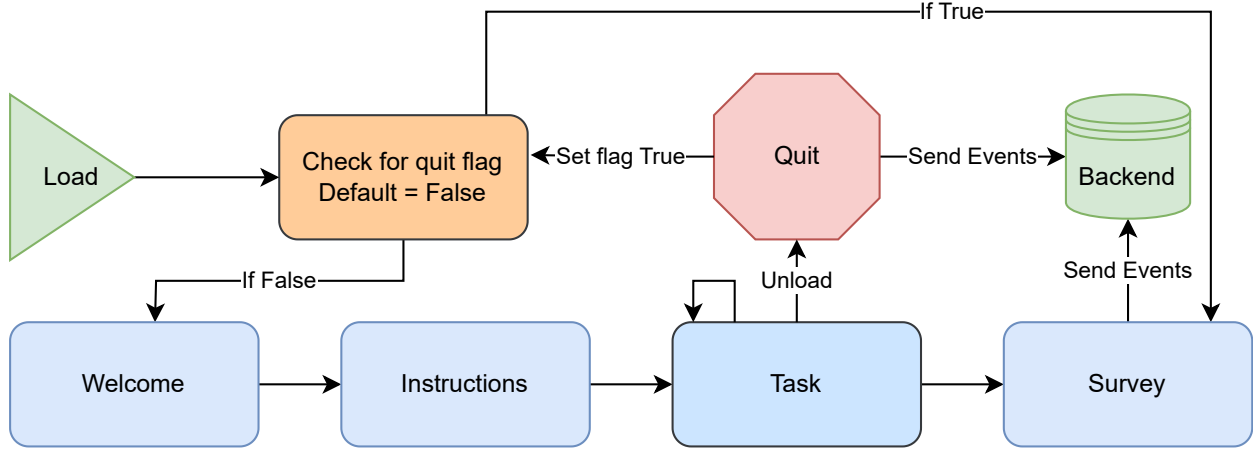
Figure 4.2: Side by side comparison of reCAPTCHA v2 and QUITCHA

License. The image categories are: Bicycle, Bridge, Bus, Car, Chimney, Crosswalk, Hydrant, Motorcycle, Other, Palm, Stair, and Traffic Light. Figure 4.1 shows the distribution (by count) of different image types in the dataset.

4.3.2 Challenges

The challenges are created in json format from the dataset using python scripts. Initially, the dataset is processed into a json dictionary, where each image type is a key and the values are json objects, containing a unique tag, filename, extension, and image data encoded in utf-8. Next we take permutations of the pairs of image types: 12 image types corresponds to 132 pair permutations. The pair corresponds to a (true type, false type), which is presented to users as “Please Select the ‘true type’s”. For each pair of image types, 60 challenges with 9 images each are created. The amount of true type images present is varied from 3 to 8, with the corresponding false type images from 1 to 6. For each true and false type pair, 10 challenges are created taking random samples from the whole dataset. This results in the

Figure 4.3: Front-End workflow for participants



total of 7,920 distinct challenges. The challenges are then loaded into our database using a python script.

4.3.3 Back-End

The back-end is hosted on a computing cluster operated by the authors' organization. A virtual machine hypervisor hosts an Ubuntu operating system upon which the back-end is deployed. The server is secured using a PKCS #1 SHA-256 RSA certificate using TLS 1.3 and HTTPS. We use MongoDB, express, and NodeJS for the database and server stack. The server REST interface is simple, with two interfaces for get and post. The get interface fetches a random challenge from the database using aggregate sample and serves it. The post interface takes an event list, validates that it is json, and writes it to a file along with some network information. No validation of responses is done on the back-end.

4.3.4 Front-End

The front-end is composed of a single HTML file, with embedded javascript and css. This constraint is due to MTurk, which only allows upload of a single HTML file. The workflow for the front-end is shown in Figure 4.3. Four study pages shown are: Welcome, Instructions, Task, and Survey:

1. Welcome page corresponds to the study information sheet (SIS), which participants are allowed to preview. The SIS provides all relevant study information as required by our organization’s Institutional Review Board (IRB).
2. Instructions page describes how to complete the task page and survey.
3. Task page varies based on the experimental context. The direct version just presents a QUITCHA, while the contextualized version presents an account creation task, on which submitting the form renders a QUITCHA. In either case, once QUITCHA task is completed the workflow transitions to the survey.
4. If a participant quits at any time, their data is captured for measuring the total drop out rates, behavior relevant to quitting, and presentation of a survey. If a participant quits during the tasks phase (after completing some tasks) the next time they open the study, they are presented with a survey. There are two versions of the survey: (1) one for those who complete the task, and (2) one for those who quit. For those who quit and do not return within the study HIT window, a secondary HIT is created to give them the opportunity to explain why they quit.

4.3.5 Direct vs Contextualized

Prior work [94, 95] determined that experimental context affects CAPTCHA solving time. Following these findings, we design the study in two versions of the front-end experience: participants were split in half and were shown direct and contextualized versions, respectively.

Direct setting

Participants were informed of the presence of CAPTCHAS from the task title in MTurk, which was: “CAPTCHA User Study”. Also, instructions included the purpose of the study as solving a single CAPTCHA. The task portion of the front-end workflow only contained the QUITCHA checkbox. The survey portion did not vary based on context.

Contextualized setting

Participants were not informed of the presence of CAPTCHAS in this setting. The task title in MTurk was: “Account Creation User Study”. Instructions include no references to CAPTCHAS and only mentioned the account creation task. The task portion of the front-end workflow contained an account creation form shown in Figure 4.6. The create button required for the fields to contain text, though no specific text was required. If the fields contained text the create button rendered the QUITCHA checkbox.

4.3.6 QUITCHA

Similar to reCAPTCHA v2, QUITCHA has two parts: a checkbox with “I am not a robot” and an image challenge. The QUITCHA checkbox is shown side-by-side with the real reCAPTCHA v2 in Figure 4.2. Upon clicking the checkbox, a user is presented with a series of image selection challenges described in Section 4.3.2. We define the heuristic for completing a single image selection challenge in Figure 4.4. The challenge heuristic is called upon clicking verify, it stores the result and checks the completion heuristic. We define the heuristic for completing the QUITCHA in Figure 4.5.

Alternatively if a participant quits after 2 attempts, the next time they return to the study web page, the survey is presented. Completing the QUITCHA task triggers the survey

Figure 4.4: Challenge completion heuristic

```
if (s_tt >= tt - 1) and (s_ft < 2):
    return true
else
    return false
where
tt = number of true type images in the challenge
s_tt = number of selected true type images
s_ft = number of selected false type images
```

Figure 4.5: QUITCHA completion heuristic

```
if (tr >= fr + 10):
    start survey
where
tr = count of true challenge solution results
fr = count of false challenge solution results
```

portion of the front-end workflow. A side-by-side comparison of image selections is shown in Figure 4.2.

4.3.7 Survey

The survey includes questions about the participant's age, their opinion about CAPTCHAS, their confidence in the correctness of CAPTCHAS solutions, satisfaction with the time spent on tasks, reasons for quitting or motivations for completing the tasks, and how many attempts they would be willing to make if guaranteed success after a certain number of attempts. Furthermore, the survey contains questions from the well-known System Usability Scale (SUS) [51] to assess usability of QUITCHA.

Figure 4.6: The Account Creation Form


PLEASE ENTER THE INFORMATION PROVIDED BELOW.
DO NOT ENTER YOUR PERSONAL INFORMATION!

Username: User123
Password: 123456789

Create an Account

User123
.....
.....

Create

I'm not a robot  reCAPTCHA
Privacy - Terms

4.4 Study Logistics

4.4.1 Timeline and Logistics

The study was conducted using Amazon’s Mechanical Turk [22] (MTurk) labor platform. MTurk allows requesters to publish Human Intelligence Tasks (HITs). We published our HIT as a webpage described in Section 4.3.4. The study included 1,000 completed HITs and lasted from March to May of 2024. As mentioned earlier, 500 were shown the direct, and 500 – the contextualized, version. The study was conducted in as a sequence of successive batches of 100 participants. Data from 1,457 unique participants was collected by our server throughout the duration of the study, meaning that 457 participants quit, i.e., did not complete the HIT. For participants who quit after solving at least one image challenge, we created a secondary MTurk HIT containing just the survey to see why they quit. It is

described in Section 4.3.7.

For the task portion, the median time per challenge was 7.7 seconds, and the median number of attempts was 10, which means a 77-second total median time for the entire QUITCHA. The median time for the survey was 104 seconds. Thus, 181 seconds was the median time to complete the whole study. Each participant was compensated US\$0.50, which corresponds to \$9.97/hr and exceeds the US Federal Minimum Wage of \$7.25/hr [40].

4.4.2 Pilot Study & The MTurk Botnet

We performed an initial pilot study as part of implementation testing and data gathering for subsequent construction of QUITCHA data processing scripts. During this study, the port for the server was (unintentionally) closed by default, which created an interesting situation:

Since server did not serve QUITCHAs during this time, participants were unable to progress or submit the completed HIT. However in just one day there were 100 HIT submissions, containing **completely empty** form data. Our javascript code checks that all required fields are present before allowing submission and – without tasks – participants could not progress to the survey. This means that the HITs must have been submitted automatically, using javascript. Lo and behold, upon rejecting the HITs we received a sequence of very similar emails from the corresponding Mturkers, within seconds of each other. These emails all came from email addresses that all adhered to the following format:

(first name)(last name)##@(email provider)

We strongly believe that the pilot study thus discovered a large scale bot-net scam operating on the Amazon’s MTurk platform. The scam works in two parts. (1) First, the scammer sends its bots to complete as many HITs as possible just by bypassing the pages with ‘crowd-form’.submit(). Second, the scammer sends email – before and after the HIT is rejected –

claiming to have encountered server issues. All email messages are almost identical, with slight capitalization and spacing differences.

4.4.3 Ethical Considerations

The user study was duly approved by the university’s Institutional Review Board (IRB). Since prospective participants were not pre-informed of the nature of the study, two additional documents were filed and approved by the IRB: (1) *“Use of deception/incomplete disclosure”* and (2) *“Waiver or Alteration of the Consent”*.

4.5 Results and Analysis

4.5.1 Data Logistics and Cleaning

We collected a total of 1,808 database entries associated with 1,457 unique Mturk participants. 1000 participants submitted a HIT through Mturk, while the remaining 457 did not submit a HIT and completely abandoned the original HIT. Interestingly we find that a large group of Mturkers (260) were able to submit the HIT without interacting with our QUITCHA. This means that out of our 1000 submitted HITs, only 740 are attributable to completing the QUITCHA. Furthermore, it highlights the horrible data quality of Mturk and the large amount of automation and scripting that is likely occurring (26%), since they had to have used programmatic means to bypass the QUITCHA. Out of the 457, who did not submit a HIT, 204 started the QUITCHA but quit at some point. 50 participants quit on QUITCHA render, and the remaining 203 participants quit at the welcome page. Out of the 254, we are able to obtain survey responses from 85 of them through our follow-up survey study. In total we end up with 944 participants, who at least clicked the checkbox.

After processing the data from the 944 participants, we end up with 14,231 valid image solutions.

4.5.2 Timing results

Five key timing measurements emerge from the fine-grained event logging:

- checkbox solution time: time for a participant to click on the checkbox after it is rendered, which occurs either immediately on the task page (direct) or after filling the form (contextualized).
- image selection solution time: time between challenge rendering and user clicking verify.
- solution inter-click selection time: time taken between selection/un-selection clicks.
- network delay: self-explanatory.
- reaction time: time between challenge rendering and first selection click.

We discuss each measurement below.

4.5.3 Checkbox Solution Time

Since we do not implement any emulation of reCAPTCHA v2s behavioral analysis system, checkbox solution time is the time taken to check the box from rendering. A total of 944 checkbox interactions were measured with a mean solving time of 3.2 seconds, as shown in Figure 4.7.

In terms of RQ5, we find statistically significant differences in solving behavior depending on experimental context. Results for direct and contextualized groups are shown in Figure 4.8. In the latter, it takes users longer (on average) to react and click on the checkbox. These results re-confirm the trends theorized and identified in prior work [94, 95].

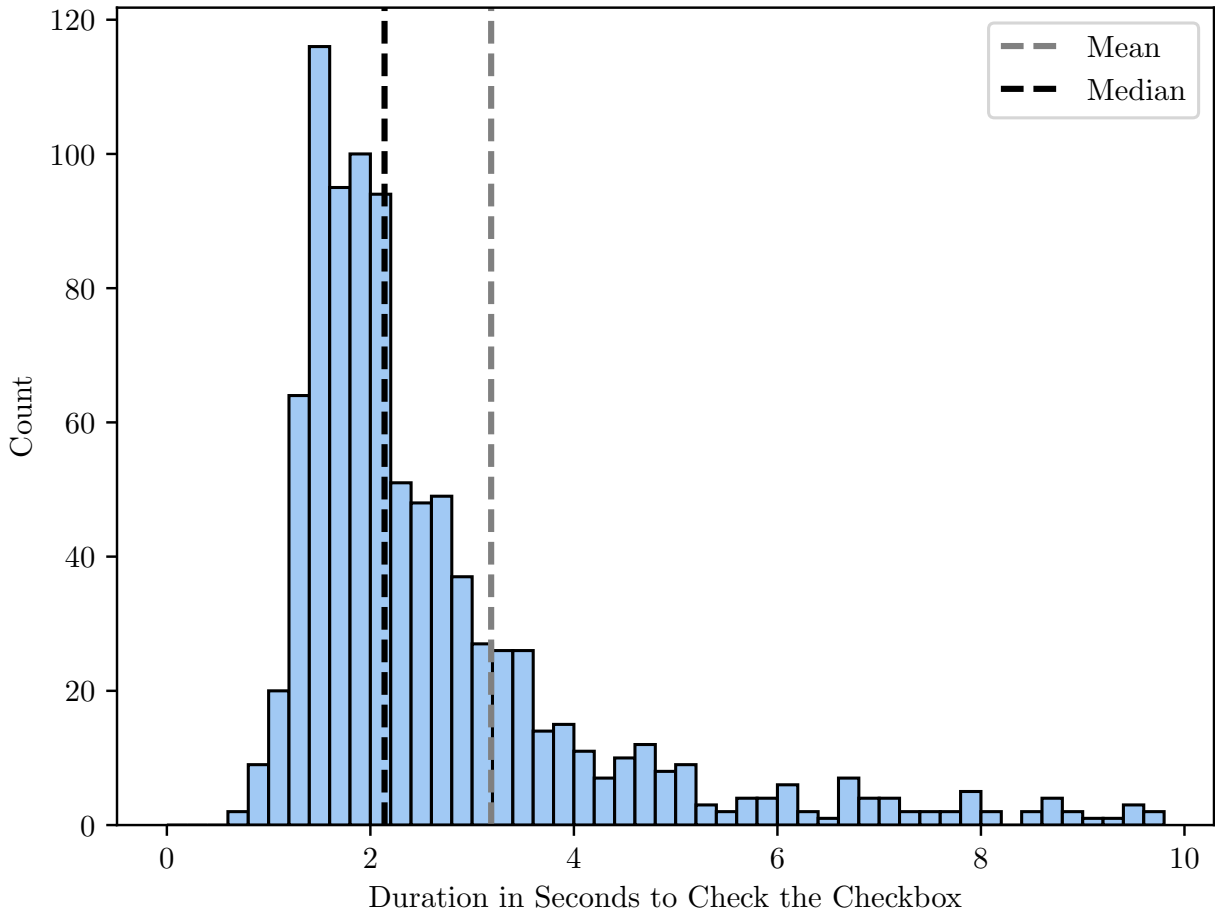


Figure 4.7: Checkbox solution time in .2 second bins

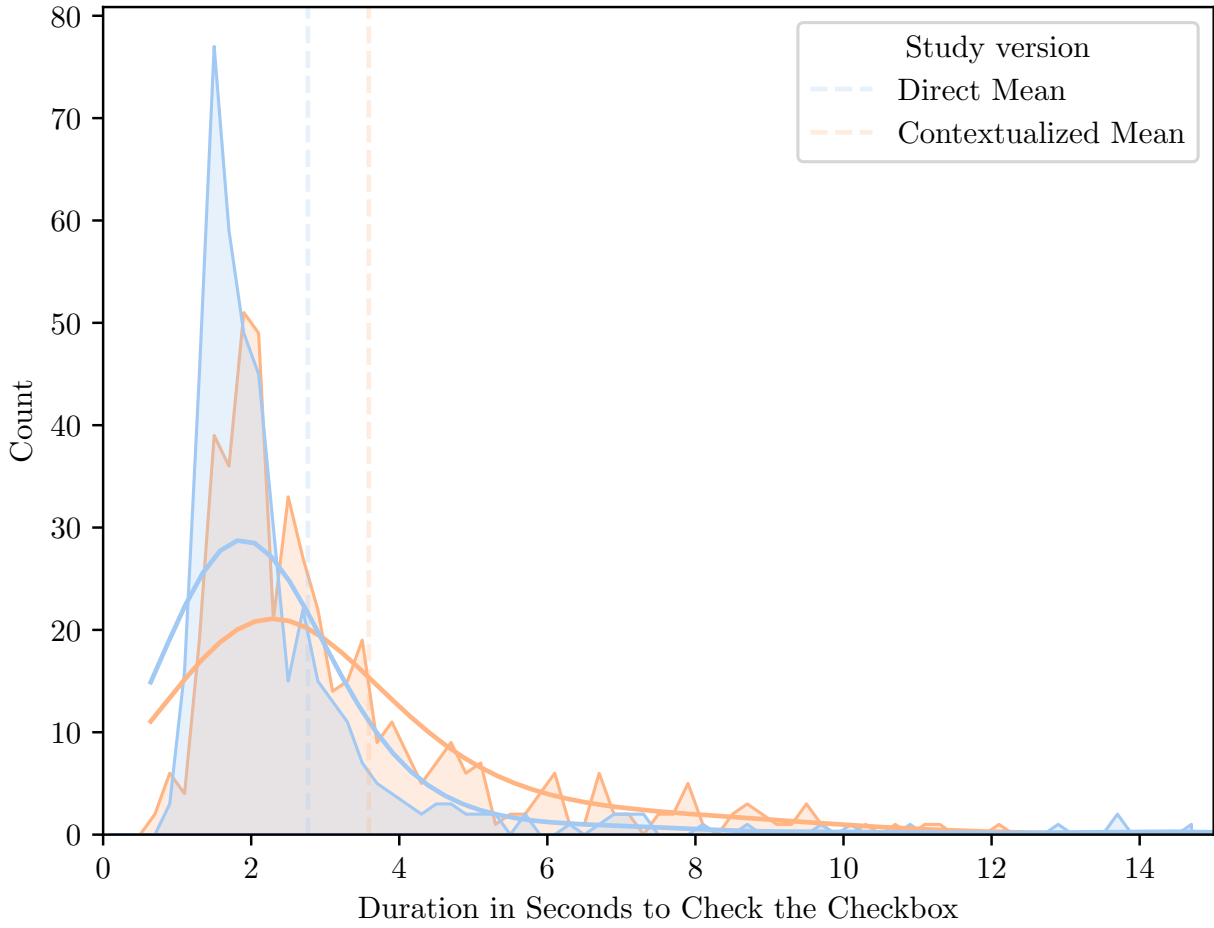


Figure 4.8: A comparison of checkbox solution time by context

Table 4.2: Image selection solution timing overview

count	14041
mean	9.34
median	7.73
std	6.02
var	36.23
max	59.18
min	0.67

Table 4.3: Image selection solution time by context

context	count	mean	median	std	var	max	min
direct	7632	9.55	7.87	6.12	37.44	59.16	1.25
contextualized	6409	9.09	7.56	5.89	34.69	59.18	0.67

4.5.4 Image Selection Solution Timing

We measure all aspects of user interaction during image-solving, including: clicks, selections, accuracy, true/false type, and attempts. Out of 14,231 image solutions, we consider 14,041 for analysis after removing outliers – solving time > 60 seconds. Selection duration is defined as time between rendering and user submitting verify. The mean and median times to complete a single challenge were 9.3 and 7.7 seconds, respectively. Further details are shown in Table 4.2. The median time for network delay was 1.1 seconds.

In terms of image selection solution timings, we see a large amount of statistical significance across various features of the challenges and the study. In Figure 4.9 and Table 4.3 statistically significant ($p < .001$) results show that it actually takes participants slightly less time on average in the contextualized version.

As far as True and False types, we observe statistically significant results mainly for the former. Figure 4.23 and Table 4.4 show that fire hydrants and crosswalks take the shortest time to solve, with the largest difference being $\approx 25\%$ slower. It can be concluded that the type of image to be selected impacts solving performance. However the type of images

Comparison of Solving Time For Single Image Challenges by Context

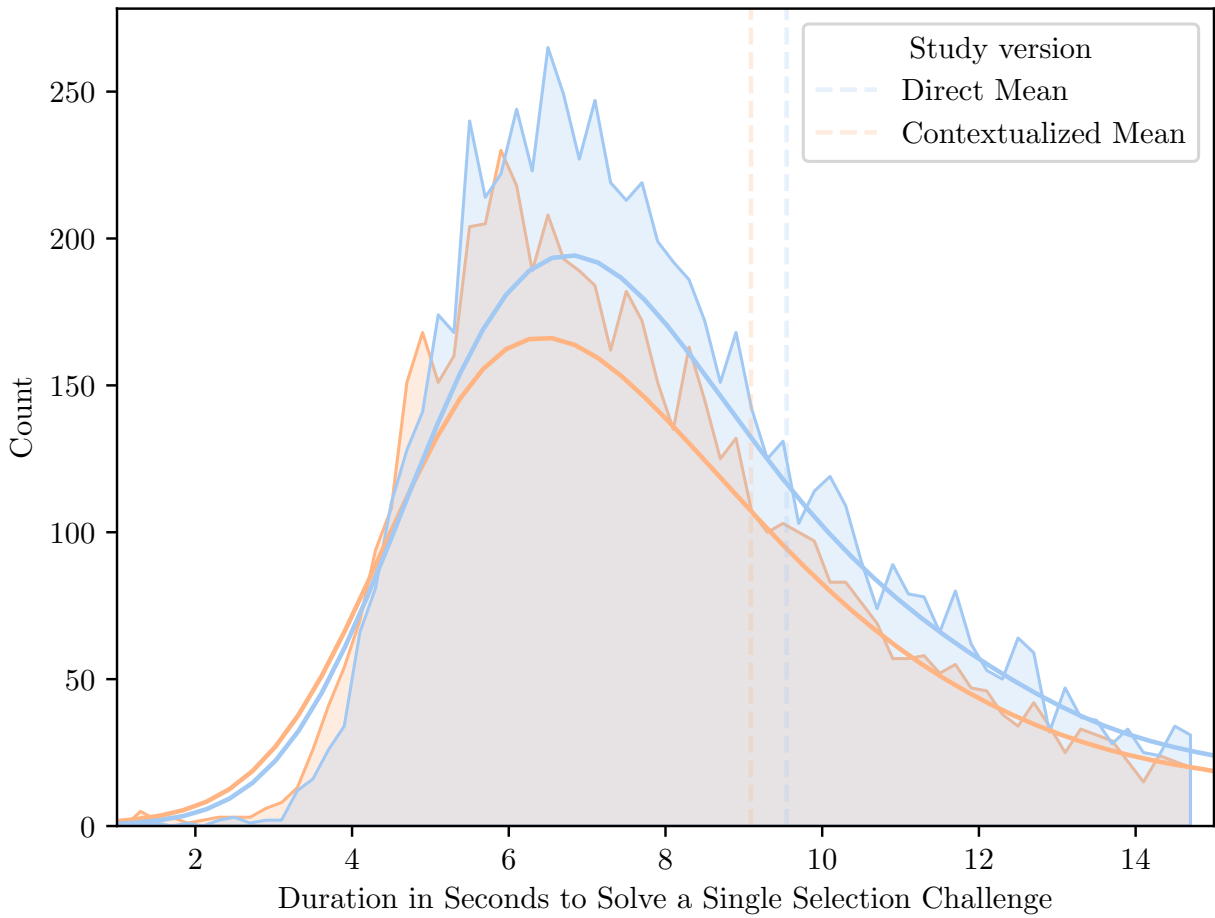


Figure 4.9: Image selection solution duration by context

Table 4.4: Image selection solution time by the true type

True Type	count	mean	median	std	var	max	min
Bicycle	1181	9.24	7.56	6.36	40.45	58.33	1.35
Bridge	1203	9.94	8.52	5.56	30.92	58.92	1.25
Bus	1199	9.04	7.47	6.19	38.35	58.45	1.59
Car	1245	9.51	8.06	5.81	33.74	55.15	1.25
Chimney	1157	9.80	8.06	5.86	34.35	56.28	2.06
Crosswalk	1193	8.54	7.16	5.46	29.77	58.47	1.84
Hydrant	1181	8.04	6.61	5.06	25.62	58.97	1.50
Motorcycle	1163	9.22	7.70	5.64	31.86	54.09	1.81
Other	1101	9.76	7.82	6.76	45.71	53.24	2.22
Palm	1136	9.33	7.71	5.91	34.96	58.59	1.32
Stair	1139	9.82	7.99	6.84	46.78	59.18	0.67
Traffic Light	1143	9.89	8.23	6.33	40.12	54.08	3.30

that are used as the false type does not have a very generalized effect although there is some significance for hydrants and palm trees.

Table 4.6 and Figure 4.11 show timing results for users who quit (True) and users who completed (False). Statistical significance ($p < .001$) is also observable between participants who quit and those who completed. In general it took less time for the latter. Spending more time on challenges may increase a persons likelihood of to quitting during image selection challenges.

There are also features that exhibit no statistical significance. Figure 4.25 and Table 4.7 show that there are no statistically significant differences in mean solving time based on accuracy results.

Figure 4.12 and Table 4.8 show that there are no statistically significant trends based only on attempts and image solution duration. Lack of statistical significance differs from prior works [95], while the overall trend remains the same: as attempts increase, solution time dips slightly. Interestingly, Figure 4.12 shows that users who quit were less likely to have more attempts.

Comparison of Solving Time For Single Image Challenges by the Correct Type

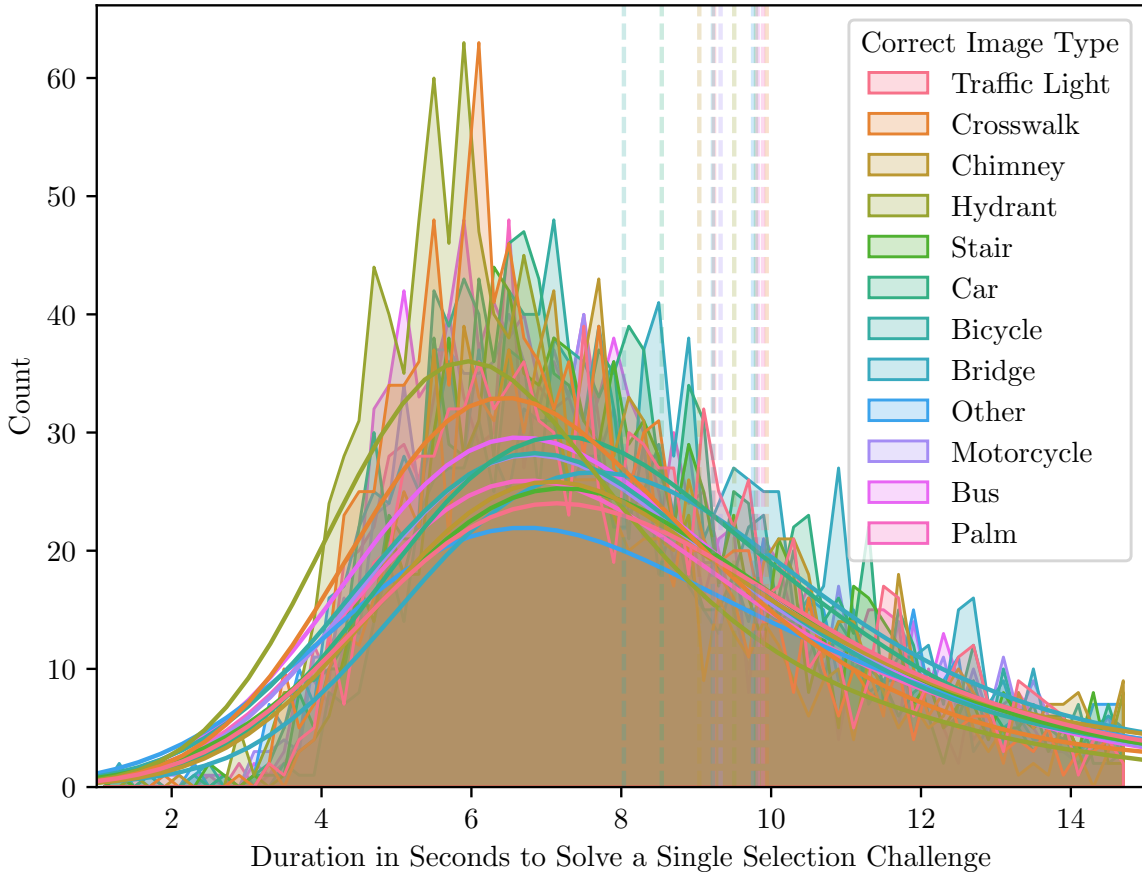


Figure 4.10: Solving time groups separated by true type.

Table 4.5: Image selection solution time by the false type

false type	count	mean	median	std	var	max	min
Bicycle	1191	9.48	7.96	5.85	34.19	59.16	1.43
Bridge	1156	9.59	7.99	6.40	40.96	58.59	1.25
Bus	1139	9.65	7.88	6.46	41.78	53.24	1.84
Car	1124	9.57	7.84	6.26	39.19	58.45	1.25
Chimney	1165	9.24	7.68	5.92	35.03	55.97	1.32
Crosswalk	1196	9.24	7.73	5.81	33.75	48.84	1.75
Hydrant	1176	8.67	7.29	5.24	27.46	58.92	2.40
Motorcycle	1145	9.53	7.77	5.86	34.35	47.53	2.66
Other	1168	9.23	7.69	5.93	35.13	59.18	2.06
Palm	1221	9.01	7.40	6.33	40.04	58.66	0.67
Stair	1148	9.54	7.77	6.32	39.93	58.47	1.50
Traffic Light	1212	9.36	7.89	5.73	32.87	57.20	2.42

Table 4.6: Image selection solution time separated by quitting

quit	count	mean	median	std	var	max	min
false	9279	9.14	7.63	5.77	33.28	59.18	1.27
true	4762	9.72	7.99	6.46	41.78	59.16	0.67

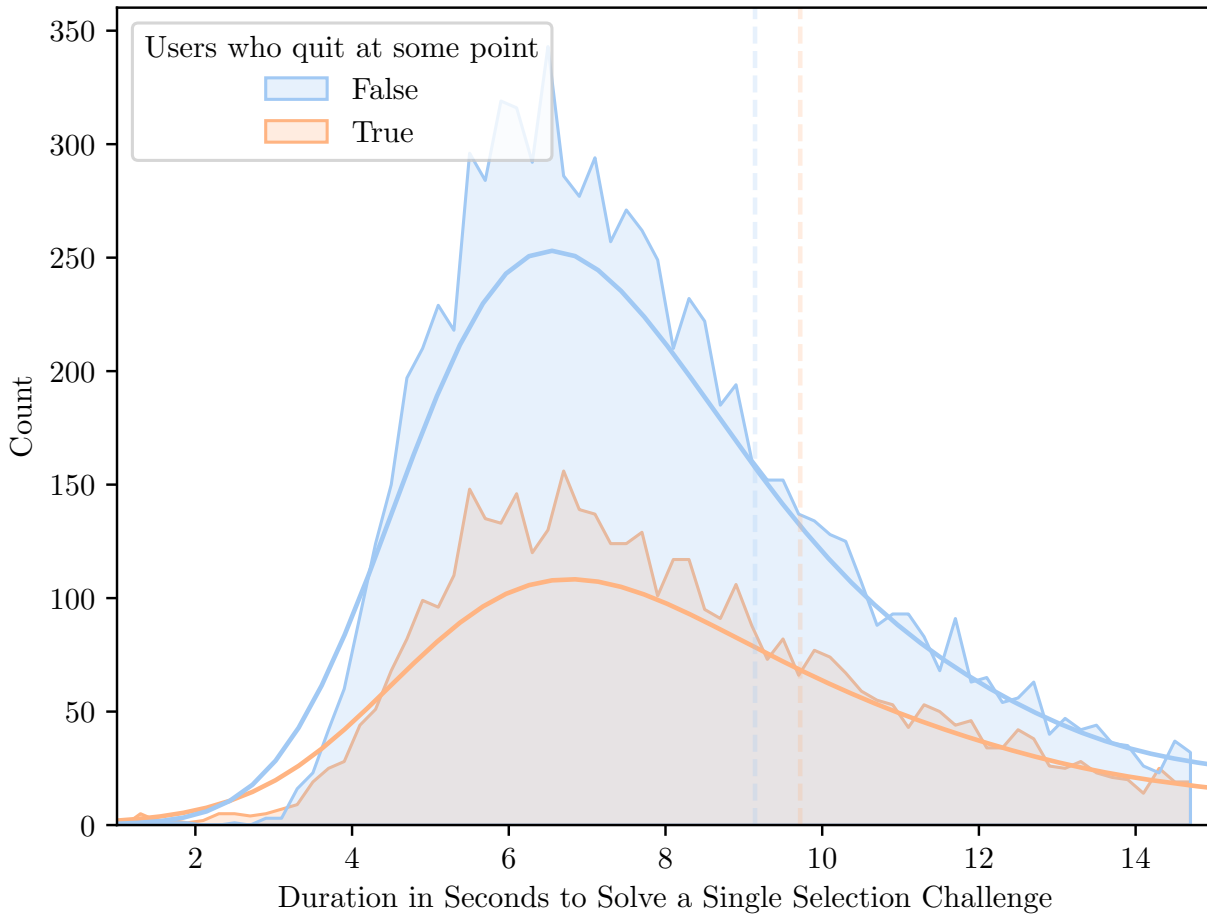


Figure 4.11: Solving time groups separated by quit

Table 4.7: Image selection solution time by the heuristic result

result	count	mean	median	std	var	max	min
false	4053	9.52	7.69	6.57	43.13	59.18	0.67
true	9988	9.26	7.74	5.78	33.42	59.16	2.42

Table 4.8: Image selection solution time by the challenge attempt number

attempts	count	mean	median	std	var	max	min
1	934	10.26	8.09	7.49	56.17	59.16	1.76
2	910	9.57	8.02	6.06	36.66	50.40	2.42
3	844	9.51	7.79	6.42	41.19	59.18	3.05
4	784	9.70	8.11	6.02	36.25	58.45	2.44
5	730	9.66	7.95	6.02	36.22	53.98	1.27
6	696	9.70	8.19	6.10	37.16	55.31	1.81
7	659	9.78	8.06	5.86	34.35	55.15	1.25
8	629	9.46	7.83	6.08	36.91	55.97	1.75
9	605	9.27	7.94	5.50	30.30	50.40	2.20
10	578	9.30	8.10	5.18	26.81	54.41	1.59
11	517	8.92	7.64	4.64	21.51	41.02	2.06
12	494	9.10	7.66	6.20	38.44	58.47	0.67
13	432	9.45	7.71	6.23	38.80	51.05	3.38
14	401	8.93	7.44	5.46	29.77	48.78	2.51
15	354	8.95	7.33	5.44	29.56	56.28	3.53
16	331	9.20	7.14	6.07	36.87	58.59	3.41
17	274	8.86	7.15	5.89	34.73	48.32	2.22
18	256	9.13	7.24	6.37	40.55	53.53	1.84
19	226	8.67	7.03	4.94	24.41	36.15	2.88
20	216	9.55	7.62	7.21	52.00	48.84	3.38
21	208	8.47	7.06	5.11	26.14	48.19	3.12
22	191	8.52	7.42	5.41	29.25	53.56	3.28
23	170	8.46	7.04	5.60	31.35	46.13	1.25
24	160	8.49	7.12	6.19	38.36	49.45	1.50
25	138	9.00	7.63	6.57	43.10	53.18	3.69
26	134	9.30	7.53	5.92	35.10	54.09	3.20
27	120	8.82	6.83	6.52	42.53	47.53	1.32
28	114	8.85	7.51	5.39	29.08	32.05	3.53
29	105	8.96	7.35	4.62	21.30	28.26	3.41
30	100	9.03	7.42	7.30	53.22	58.66	1.31

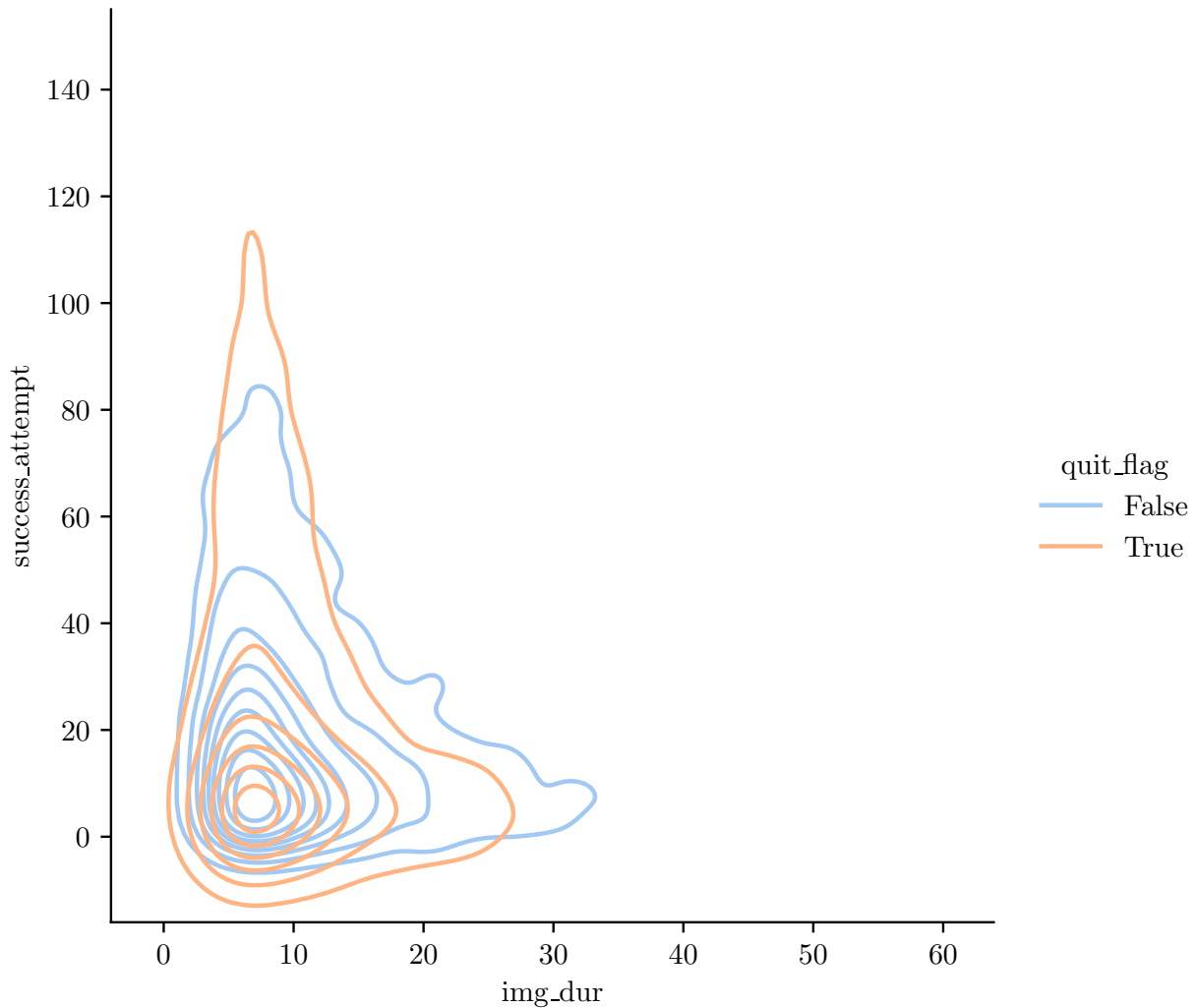


Figure 4.12: Solving time distribution across attempts separated by quitting (Image solution duration in seconds on x-axis)

Inter click and Reaction Timing

Capturing clicks and interactions with the QUITCHA offers two new dimensions of time: initial and average inter-selection. We use these to classify user solution behavior and trends. We measure every selection and un-selection event in QUITCHA. Using these timing events we compute the time between selections, which can be considered as the time to identify a single image. Inter-selection results presented in this section are presented on a average per challenge basis. We also considered the time between initial rendering and the initial

Table 4.9: Overview of the data from selection time

	inter average	initial selection
count	14205	14086
mean	1.86	3.66
median	1.27	2.36
std	2.91	5.08
var	8.50	25.79
max	59.08	59.85
min	0.40	0.01

selection event to be the reaction time. Table 4.9 shows an overview of the initial and inter-selection times. Notably, it takes 97% longer, on average, upon the initial selection than on subsequent ones. These results show that participants’ reaction time has a definite impact on solution time.

Furthermore, both inter-selection and reaction times show statistically significant trends across certain features of the study. A prominent feature that shows statistical significance is the true type for inter-selection time. Results in Table 4.10 as well as Figures 4.13 and 4.24 show that there are significant differences between most image True types and time for users to recognize them. “Other” image type took users the longest to select – 69% slower than that of hydrants. Also, as shown in Figure 4.14, 99.9% of users solved this challenge incorrectly, which likely impacted selection duration since both accuracy and timing exhibit significant trends based on the true type.

Moreover, results show statistically significant differences based on both heuristic accuracy and quit status. This significance holds for both reaction and inter-selection times. Figures 4.11 and 4.12 show results for the inter-selection time across accuracy and quit status. Participants were 24% slower on average with false heuristic result, as compared to those with true heuristic results. Participants who quit took an average 16% longer between selections. In general, results indicate a trend that the speed of performance in individual image selection impacts a participant’s quitting behavior.

Table 4.10: Inter selection time by the true type

true type	count	mean	median	std	var	max	min
Bicycle	1189	1.62	1.22	1.70	2.90	29.85	0.41
Bridge	1213	1.89	1.40	2.02	4.07	28.21	0.55
Bus	1212	1.71	1.21	2.21	4.87	26.20	0.40
Car	1254	1.69	1.30	1.64	2.70	21.30	0.60
Chimney	1165	1.81	1.34	1.88	3.55	28.92	0.55
Crosswalk	1205	1.49	1.09	1.74	3.02	24.14	0.42
Hydrant	1193	1.39	1.03	1.64	2.67	22.76	0.42
Motorcycle	1170	1.63	1.25	1.49	2.21	20.59	0.47
Other	1110	2.36	1.62	2.48	6.15	26.06	0.52
Palm	1154	1.87	1.29	2.51	6.32	26.94	0.47
Stair	1147	1.79	1.30	1.98	3.90	27.84	0.48
Traffic Light	1157	1.86	1.32	2.18	4.77	23.60	0.50

Table 4.11: Inter selection time by separated by heuristic result

accuracy	count	mean	median	std	var	max	min
false	4101	2.04	1.38	2.41	5.79	29.85	0.40
true	10068	1.64	1.23	1.78	3.16	28.92	0.41

Table 4.12: Inter selection time separated by quit

Quit Flag	count	mean	median	std	var	max	min
False	9345	1.67	1.23	1.76	3.09	27.84	0.42
True	4824	1.93	1.35	2.36	5.58	29.85	0.40

Comparison of Inter Click Averages For Single Image Challenges by the Correct Type

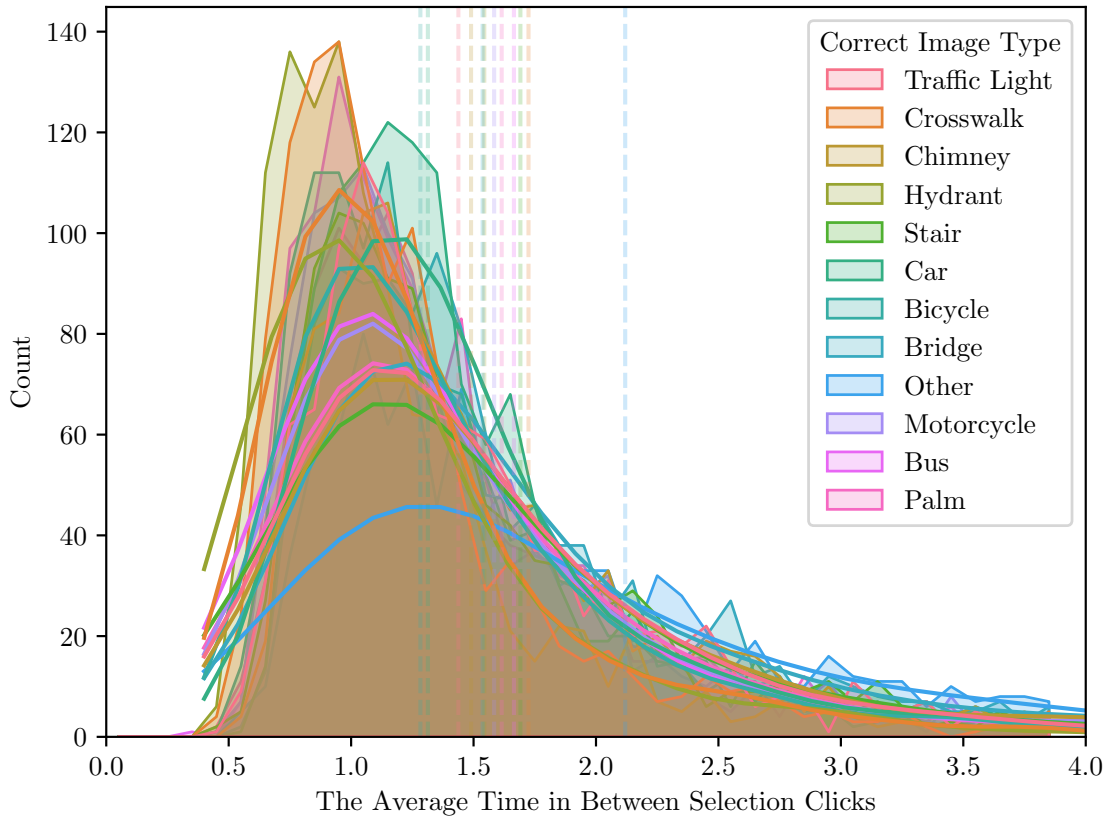


Figure 4.13: Inter-selection time groups by true image type

4.5.5 Accuracy of Image Selection

We now examine features against coarse- and fine-grained accuracy results. Based on 14,231 challenges, heuristic accuracy results show 10,096 true and 4,135 false. Therefore, overall heuristic accuracy is 70.8%. Tables 4.13 and 4.14 show fine-grained results for selection accuracy. There were 72,757 selection events, recorded with a true accuracy of 87.6%, and 2,864 un-selection events with a true accuracy of 43.2%.

Table 4.14 shows the mean and median percentage of true and false type. Results show that, for true challenge heuristic results 94.1% of the true type were correctly selected, while 3.7% of the false type were incorrectly selected. Alternatively for false challenge heuristic results, only 45.6% of the true type were correctly selected, while 40% of the false type

Table 4.13: Overview of selections per 14k challenges

	selections		unselections	
	true type	false type	true type	false type
sum	63712	9045	1238	1626
mean	4.48	0.64	0.09	0.11
median	4.00	0.00	0.00	0.00
std	2.11	1.28	0.37	0.44
var	4.43	1.65	0.14	0.19
max	14.00	11.00	8.00	6.00

Table 4.14: Percentage of true/false type selected

heuristic	true type		false type	
	mean	median	mean	median
false	45.58	57.14	40.23	25.00
true	94.07	100.0	3.63	0.00

were incorrectly selected. Therefore, most incorrect results include incomplete and incorrect selections of both types.

Figures 4.14, 4.15, 4.16, 4.21, and 4.22 show heuristic results separated by various study features. Figure 4.14 shows some of the most interesting results for true type accuracy with significant impact and trends for multiple types. We observe 96% true accuracy for hydrants, while for other types the accuracy rate is 0.8%. It is likely that participants do not take the time to read the full header text. Instead, they just see the word and try to click images surrounding it. When comparing with Figure 4.15, we notice no significant trends, meaning that using distinct image types as the wrong type does not impact heuristic accuracy.

Figure 4.16 shows that participants who quit had an overall heuristic accuracy of 62.5%, while those who completed had an overall accuracy of 75.4%. Since accuracy for the latter is lower, more challenges had to be solved for completion. Results indicated that having a lower accuracy significantly influences a user’s decision of whether to complete or quit a CAPTCHA-solving task.

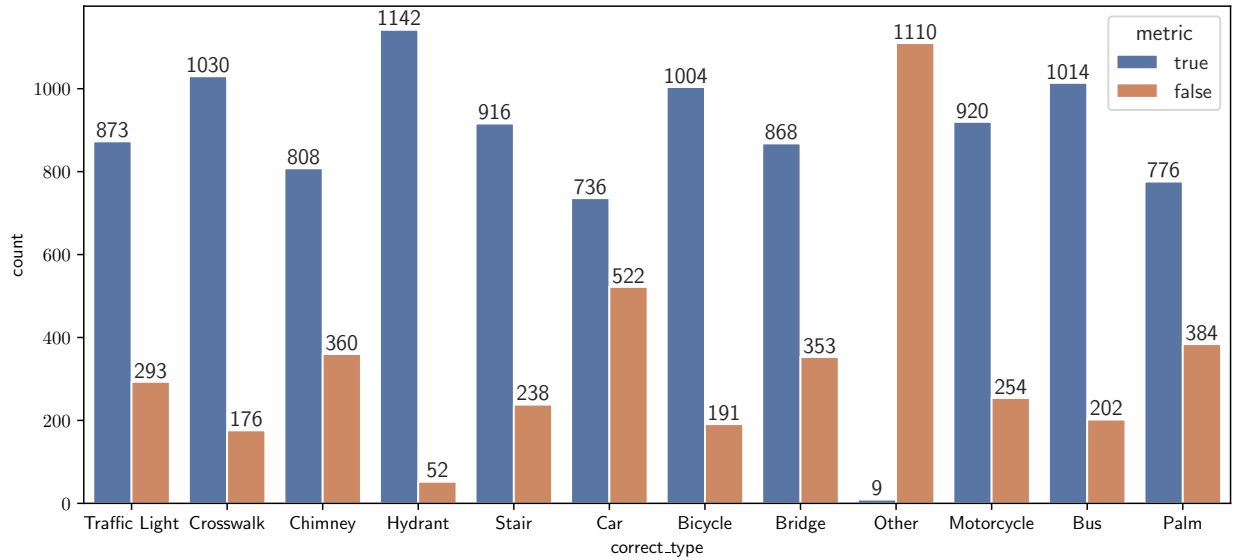


Figure 4.14: Accuracy across true types

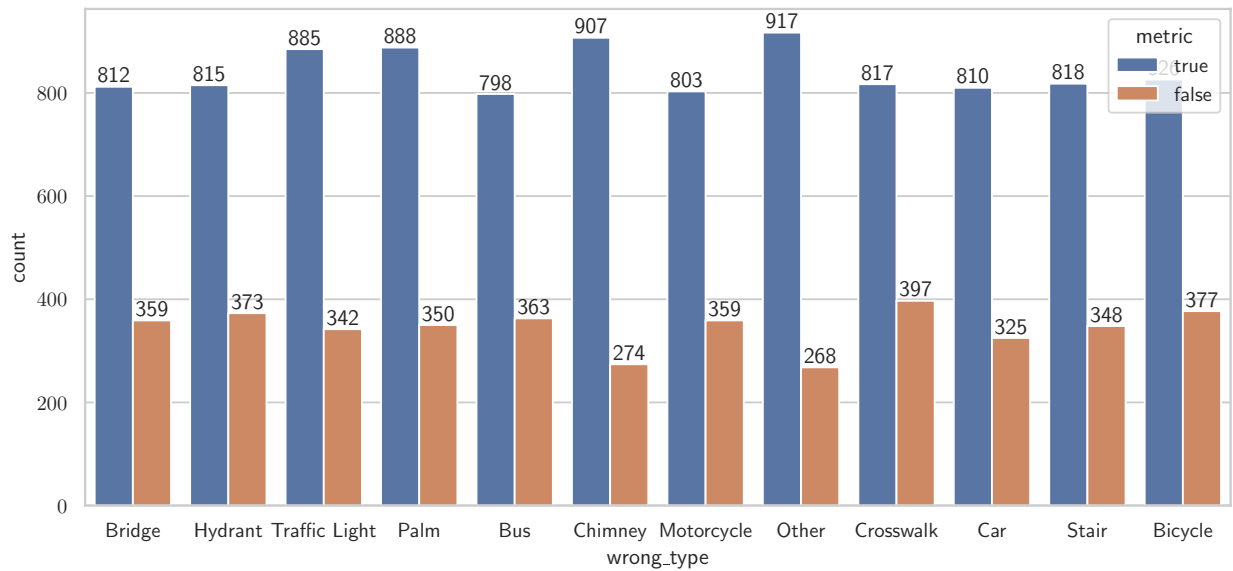


Figure 4.15: Accuracy across false types

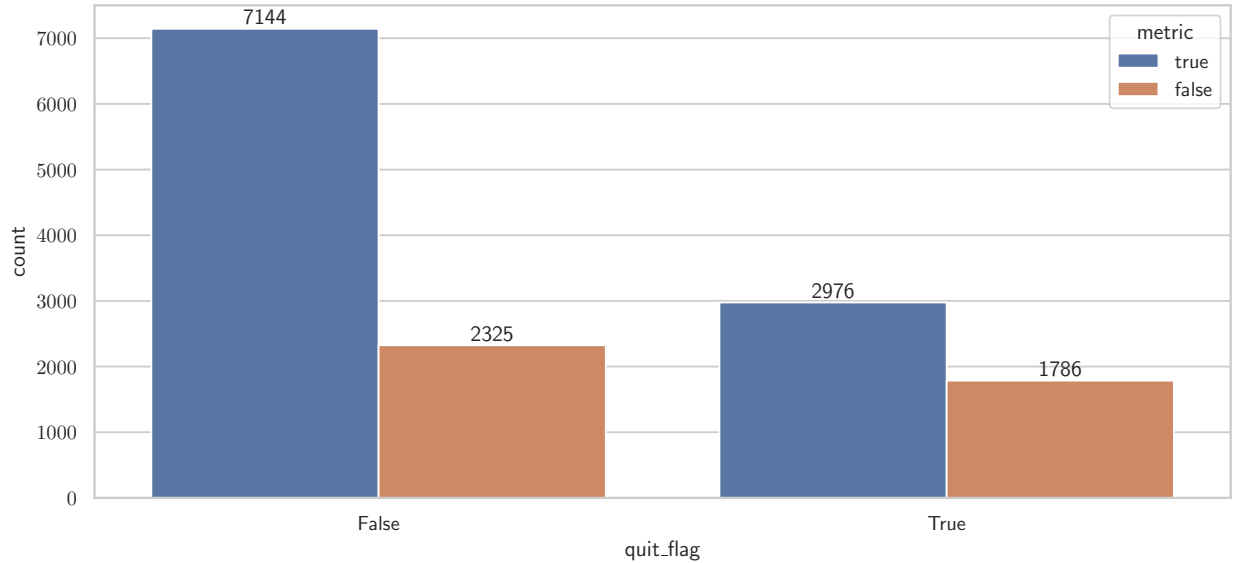


Figure 4.16: Accuracy across quitters

Figure 4.21 shows that participants in both direct and contextualized versions solve with a 71% accuracy. Therefore, there is no differences in accuracy based on the experimental context. Figure 4.22 shows that accuracy remains relatively consistent over number of attempts, i.e., and does not improve or worsen with multiple attempts.

4.5.6 Statistical Testing Methodology

All statistical testing and validation is done via ANOVA testing supported by python libraries of scikit-posthocs [39] and scipy [35]. Kruskal-Wallis one-way analysis of variance is performed followed by Conover’s test, which is adjusted via Bonferroni’s step-down method. Specifically, we use the `posthoc_conover` function, which does pairwise testing for multiple comparisons of mean rank sums. The results are grids containing p values for pairwise combinations. We compare all value columns against group columns and present some of the more complicated statistically significant results in Figures 4.23 and 4.24.

4.5.7 Quit Analysis

We now analyze the mechanics of CAPTCHAS abandonment. We define two main categories of participants who quit after interacting with QUITCHA.

Full Quit: Participants who quit completely and did not return.

Partial Quit: Participants who quit during the QUITCHA and returned to complete the HIT.

Recall that there are 740 valid completed HITs and 457 participants quit completely, 203 of whom quit upon seeing the welcome page. We believe that 254 quit due to CAPTCHAS, meaning that we observe a total dropout rate of 25.5%. Out of the 740 completed HITs, 301 are associated with partial quits. Thus, 40.7% of participants unloaded (via close, refresh, etc.) the page and came back at some later point.

Factors Influencing Quitting

The study shows that participants who quit took longer to select individual images inside an image challenge, with an average of 16% longer time between selections, compared to participants who completed the tasks. Also, participants who quit had a longer image solution time. Moreover, quit participants demonstrated lower solution accuracy of 62.5%, while those who completed had an overall accuracy of 75.3%.

The decision to quit (by participants who did so) was likely influenced by several factors, including the need to solve more image challenges at a slower pace. Interestingly, participants who quit made more CAPTCHA-solving attempts, on average. Specifically, the average number of attempts was 18 for those who quit, compared to 15 for those who completed. However, the median number of attempts for participants who quit is 6, which is quite a

bit lower than 8 – the median number of attempts for participants who completed. This indicates that many participants quit early, after solving only a few image challenges. However, a few participants solved a large number of challenges before giving up. In fact, we found one participant who solved 138 challenges before quitting. Also, there was a significant difference in both average and median attempt numbers for partial quit and full quit participants. Partial quit participants made 10 attempts on average, while full quit participants made 24. The median number of attempts for partial and full quit participants is 6 and 11, respectively.

4.5.8 Survey Analysis

In accordance with the information in Section 4.5.1, we received survey responses from 825 participants who completed at least one CAPTCHAS task. Of these, 439 completed the tasks, 301 partially quit, and 85 fully quit.

Age Distribution

We gathered self-reported age data from each survey participant. Majority of participants are in the 25 – 40 age range. The average age is 33.83, and the median is 32. The youngest participant is aged 18, while the oldest is 72. Age distribution is shown in Figure 4.17.

Opinion About captchas

Participants were asked an open-ended question regarding their opinion of CAPTCHAS. To assess the impact of priming on participants' views, we implemented three scenarios by altering the wording of the question: (1) Positive Priming: What do you think of CAPTCHAS, e.g., are they: fun, interesting, boring, annoying etc.? (2) Negative Priming: What do you

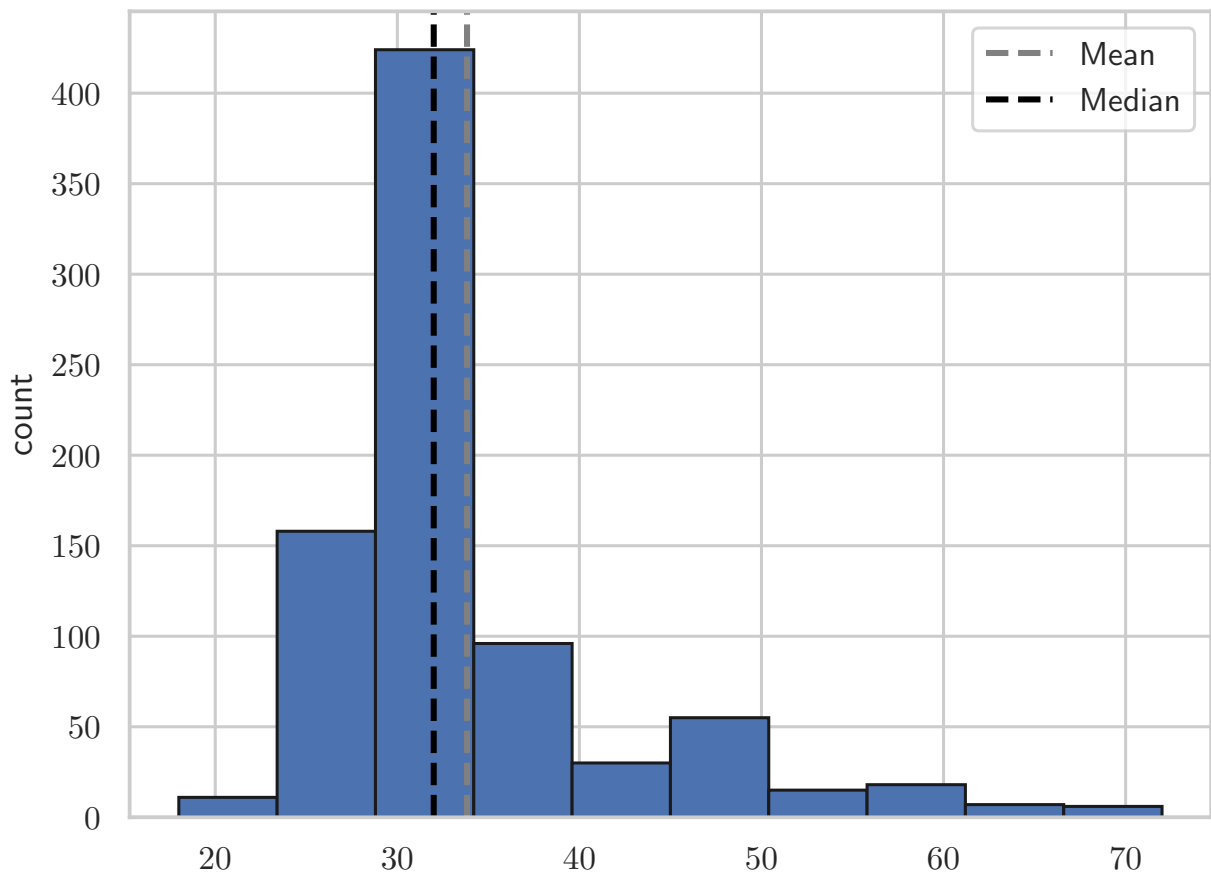


Figure 4.17: Age distribution of participants

think of CAPTCHAS, e.g., are they: annoying, boring, interesting, fun, etc.?’ (3) No priming: What do you think of CAPTCHAS? In positive and negative priming scenario, although the same words were given as an example their orderings were different. In the positive priming, positive words were listed the beginning, while the negative words came first in the negative priming.

Figure 4.18 illustrates the sentiments of participants’ opinions based on the priming situation.

In all three scenarios, the majority sentiment is positive: 78%, 74%, and 61% for positive, negative, and no priming, respectively. However, in the absence of priming, there is a higher negative sentiment at 39%, compared to the other two primings: 22% and 26%, respectively. This suggests that the ordering of example words does not have a significant impact. However, the presence of example words influences opinions. Participants are less likely to express negative opinions when example words include some positive ones, and they tend to use the positive words from the example. This becomes more evident when looking at the actual words used in the responses. In the positive and negative priming scenarios, the most prevalent word is “Interesting”, directly derived from the example words. In the no priming scenario, the most significant word is “Good”.

We also examined whether quit participants had different opinions about CAPTCHAS, compared to those who completed. However, we did not find any significant differences. The positive sentiment percentage is 68 – 69% and the negative is 31 – 32% across all quit situations, i.e., including participants who did not quit, partially quit, or fully quit.

Quit Reasons

To identify the reasons why participants quit, we utilized both open-ended and multiple-choice questions. The open-ended question was posed first to prevent potential bias from the options in the multiple-choice question. We conducted sentiment analysis of the open-

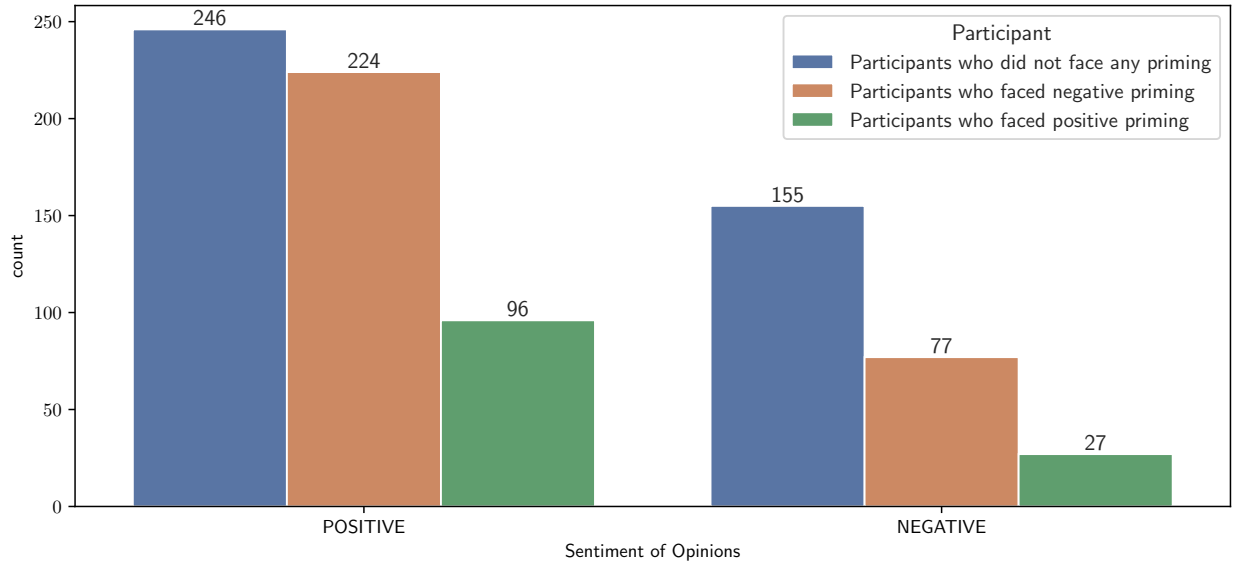


Figure 4.18: Effect of priming on opinion about CAPTCHA

ended responses and found that the majority conveyed negative sentiment, with 77% of responses being negative. We also generated a word cloud based on the responses, which highlighted key-words such as “long”, “task”, “complex”, and “exhausted.”

Results depicted in Figure 4.19 show the findings from a multiple-choice question. This question featured checkboxes, allowing participants to select multiple relevant factors. The primary reason for participants quitting (selected by 63% of quitters) was the perception that the task was too long or complex. This finding aligns with the significant words found in the open-ended responses. Selection percentage of aforementioned factor is even higher (67%) for participants who were presented with a contextualized version, i.e., those who believed they were only completing an account creation task. Other significant factors included unclear or confusing instructions (44%) and tasks differing from their initial descriptions (36%).

Motivation for Task Completion

Participants who completed the task were asked an open-ended question about their motivations. We analyzed the sentiment of their responses and found that the majority (72%) of

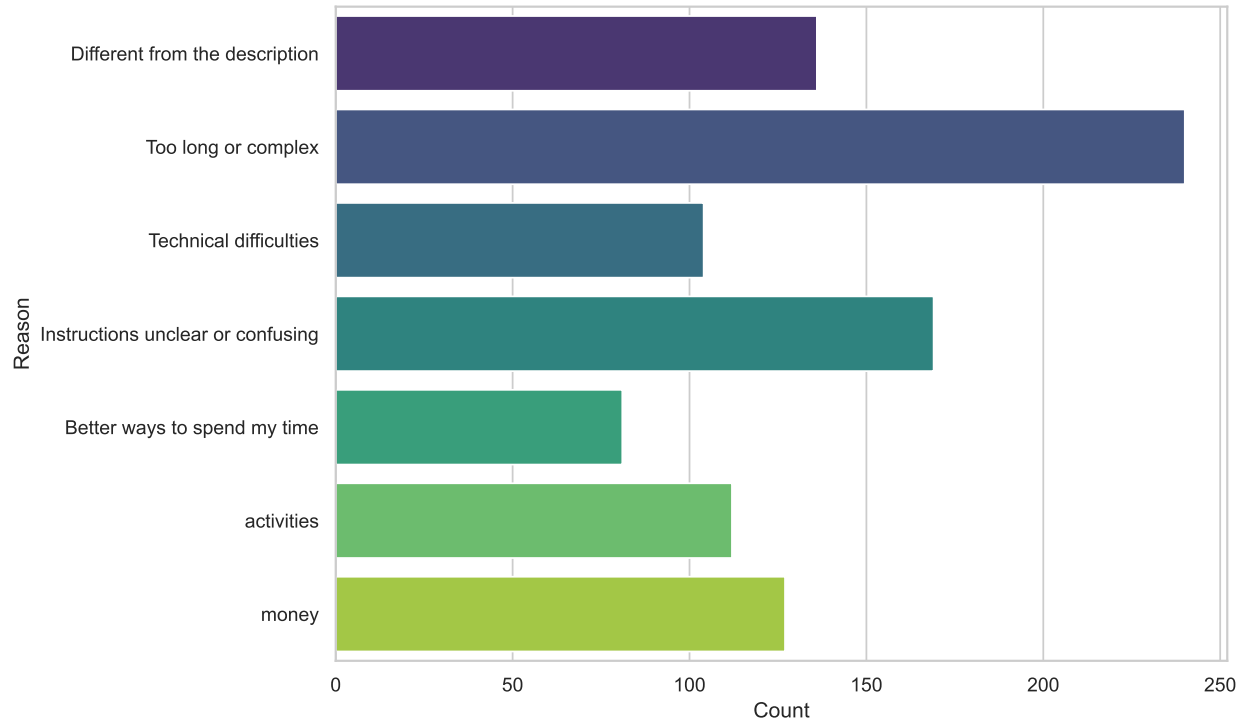


Figure 4.19: Factors behind CAPTCHAS task abandonment

the responses had positive sentiments. We also created a word cloud from the responses and found “interesting” to be the most prominent word. If interested, the word cloud is provided as Figure 4.26 in the Section 4.7.

Confidence in Solution

We attempted to ascertain participants’ confidence in their solutions. Our goal was to determine if confidence influenced their decision to quit, i.e., whether a more confident participant would attribute technical issues to the system and quit sooner than a less confident one. As shown in Figure 4.20, we do not find any such correlation. All participants expressed similar levels of confidence in their solutions, regardless of whether they ultimately quit or not.

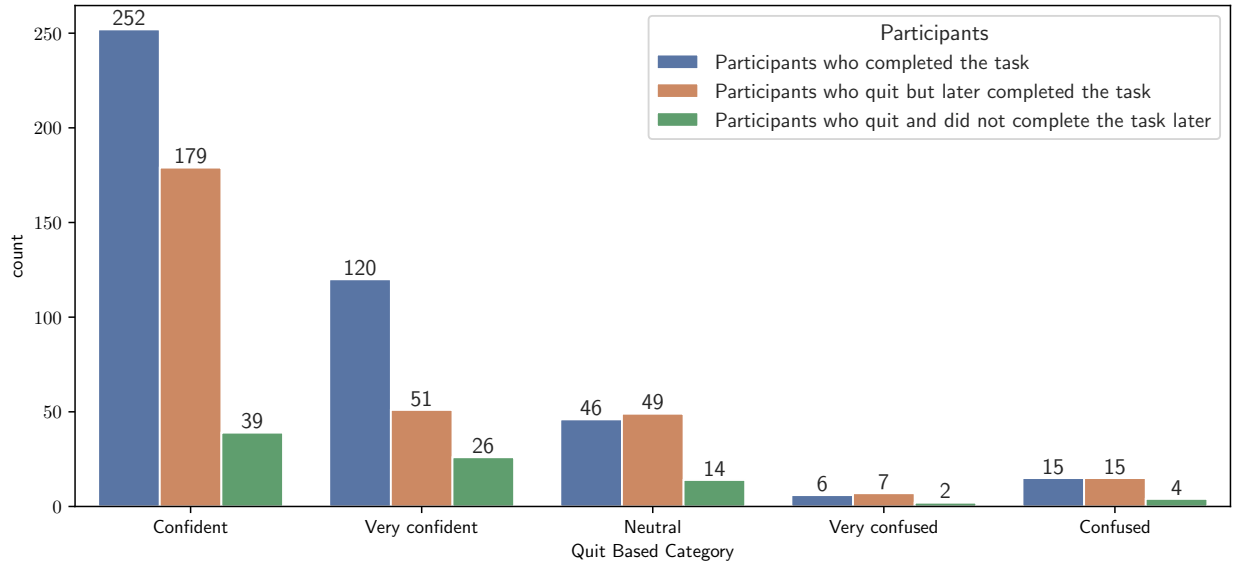


Figure 4.20: Effect of solution confidence in abandonment

Task Completion Time Satisfaction

We considered whether participants' perception of the task duration influenced their decision to quit. To this end, we asked if they were satisfied with the time spent on the task. Results show that most participants were content with the task duration, indicating that completion time did not play a significant role in their decision to quit.

Guaranteed Success Attempt Estimation

We surveyed participants to find out how many attempts they would be willing to make if they were guaranteed success after a specific number of attempts: the average number is 12 with the minimum of 1 and the maximum of 72. We also looked into whether these numbers differ for participants who quit versus those who completed and observed no significant effect.

SUS SCORE

Usability of QUITCHA was assessed by asking participants to respond to System Usability Scale (SUS) questions, and calculating the SUS score. The SUS score for QUITCHA is 51.44. According to the adjective scale of the SUS Score [47], this indicates an “OK” level of usability. We also examined the impact of context and abandonment on the SUS score and found no significant influence of these factors on the score. The SUS scores from participants who received direct versus contextualized versions are almost the same: 51.26 versus 51.65. Finally, SUS scores from the complete, partial quit, and full quit participants are 51.78, 50.34, and 53.5, respectively. All of these scores denote an “OK” level of usability.

4.6 Related Work

Prior research on CAPTCHAS include [94, 95, 53, 49, 65, 91, 104, 83, 77, 66, 78, 61, 60, 72, 102]. However only some of this research [94, 95, 102, 60] has studied image selection CAPTCHAS. In general the prior image selection CAPTCHA research results lack real granularity of timing interactions with the CAPTCHA services. The studied image selection CAPTCHA services are black boxes since they do not provide fine grained statistics surrounding CAPTCHA solutions. Thus prior timing results only consider the time from rendering the service until the service returns success. Furthermore, accuracy results from these studied CAPTCHA services are not reported in some cases leaving the data incomplete. We differentiate our work by capturing every user interaction with the QUITCHA and creating multiple new timing and accuracy metrics. Table 4.15 shows a direct comparison of our results and those of prior works that studied image selection CAPTCHAS. With a comparable number of participants, QUITCHA is able to gather much more data points surrounding user interaction with image selection CAPTCHAS. We are the only study to provide the time, network delay, count and accuracy for individual image selections.

4.6.1 Detailed Comparisons

Searles et al. [94] had image selection results with 1400 participants solving 4 separate CAPTCHA services and were only able to gain 3416 coarse grained image solution results. It is unclear from these results exactly how many image selection challenges were actually solved or interacted with since many CAPTCHA services may present multiple challenges to users. On the other hand our results from 1400 participant produced 14,041 image solutions, and 75,621 image selections. In total we produce at least 4-22x more data per participant using a single service. Comparing timing results our image solution results are 1.6-3.2x faster. Lastly, our findings differ from [94] regarding the impact of age on image solution time. [94] concluded that older participants were slower in solving image challenges, but we did not observe any such effect.

Searles et al. [95] had image selection results from 3625 participants solving reCAPTCHA v2 and were only able to approximate 1890 coarse grained image solution results. These results relied on approximating image solving time from daily reported statistics from Google's reCAPTCHA v2 dashboard [27], which is a considerable limitation. With 40% of the participants we are able to generate 7.4-40x more image selection data. This shows that although Google offers reCAPTCHA v2 as a free service, there is a massive cost in terms of data that website operators are missing out on. reCAPTCHA v2 reporting also lack statistics and features surrounding image selection solution such as: accuracy, image type, inter selection time, network time, attempts, and reaction time. [95] also investigated the usability of reCAPTCHA v2 using the SUS Scale. The SUS score of reCAPTCHA v2 image challenges was found to be 58.90, indicating an "OK" level of usability. This is similar to the usability level of QUITCHA as determined in our study. However, participants' opinions about CAPTCHAs differ between these two studies. While [95] did not perform sentiment analysis on users' opinions, the most significant word from the word cloud was "Annoying," which has a negative connotation. On the other hand, the majority of opinions in our study are

Table 4.15: Comparison of related work results. Time in average seconds.

author	us	[94]	[95]	[60]	[102]
participants	944	1400	3625	202	40
checkbox solutions	944	2800	7269	0	40
checkbox solution time	3.2	1.85	3.1-4.9	N/A	3.1
image solutions	14,041	3416	1890	202	N/A
image solution time	9.34	15-32	10.4	9.6	N/A
image solution accuracy	70%	71-81%	93%	N/A	N/A
image selections	75,621	N/A	N/A	N/A	N/A
inter selection time	1.86	N/A	N/A	N/A	N/A
inter selection accuracy	88%	N/A	N/A	N/A	N/A
reaction time	3.66	N/A	N/A	N/A	N/A
network delay time	1.1	N/A	N/A	N/A	N/A

positive.

Feng et al. [60] had minimal results with 202 participants solving 202 image selection challenges. However the focus of their work was in presenting a novel CAPTCHA type sen-CAPTCHA and comparing it to other versions of CAPTCHAS. The results from their work show that it took 9.6 seconds average to solve image selection challenges, which is similar to our result of 9.3 seconds. Nevertheless we present 69-374x more data points surrounding image selection solutions and a multitude of features to classify trends and behaviors surrounding solutions.

4.7 Accuracy, Statistical, Timing, and Survey Figures



Figure 4.21: Accuracy across context

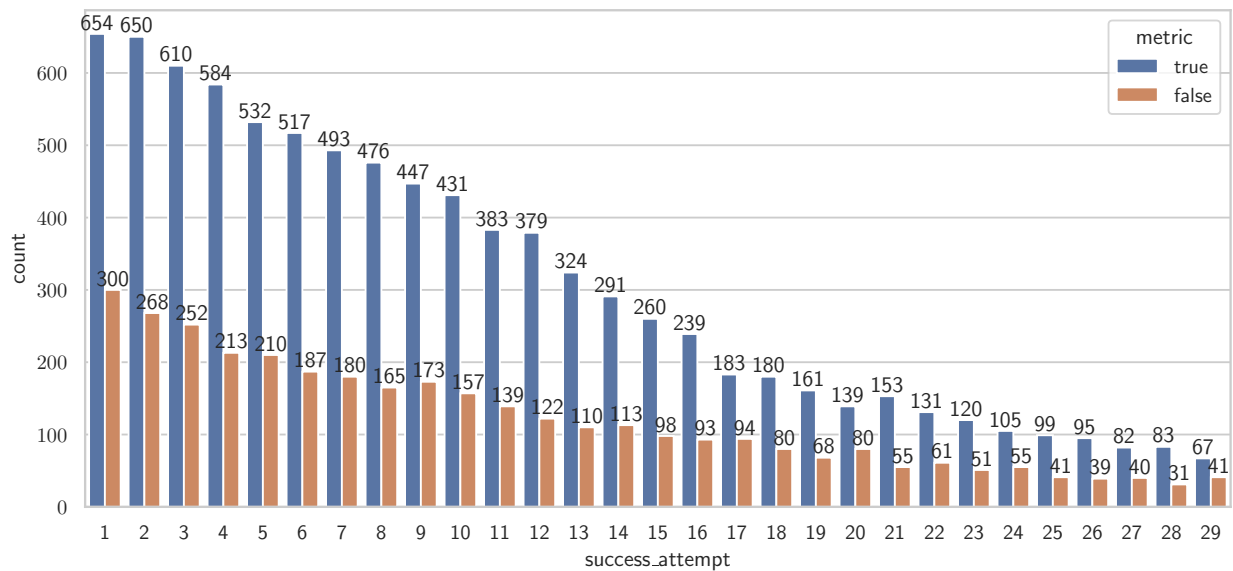


Figure 4.22: Accuracy across subsequent challenges

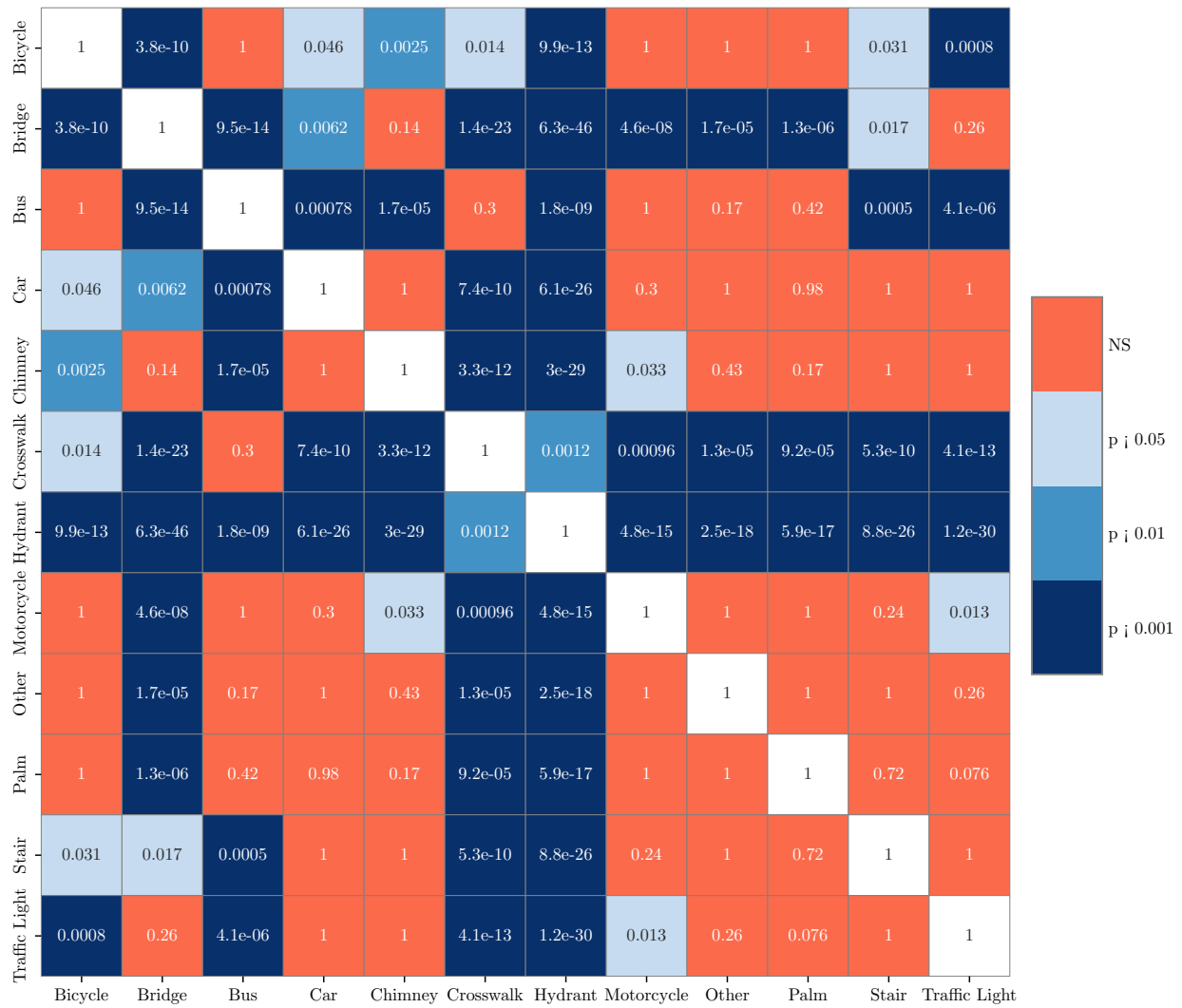


Figure 4.23: ANOVA results comparing true type

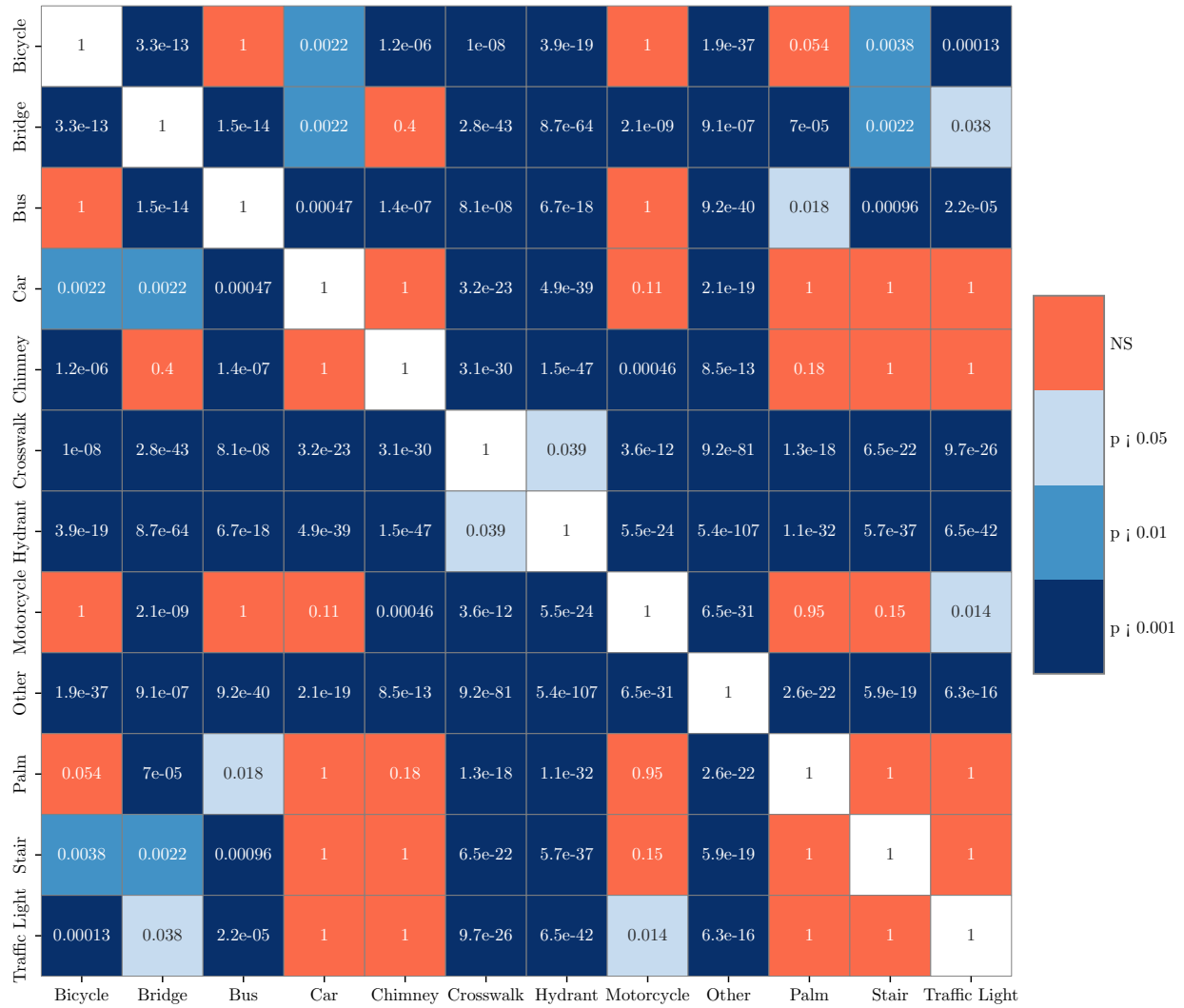


Figure 4.24: ANOVA results for inter selection averages across true types

4.8 Summary

This chapter introduced QUITCHA— an open-source system for image-based CAPTCHA challenges, providing website owners with a way to safeguard their users’ information by never sending it to a third party. QUITCHA also contributes a large-scale dataset, containing numerous features and measurements. Key research findings are:

- **RQ1:** QUITCHA produces fine-grained timing and accuracy features that exhibit statistically significant trends.
- **RQ2:** Repeated image selection challenges produce a significant amount of quitting, in the 26 – 41% range. Multiple features influence dropout rates.
- **RQ3:** Results show that exact accuracy, selections, clicks, timing, and function tracing allow for the recreation of user interaction and exhibit statistically significant results across certain features
- **RQ4:** We provide a large scale fine-grained image solution dataset and analysis tool-set that quantifies various solution-related timing events.
- **RQ5:** New features are discovered with statistically significant trends.

Chapter 5

Final Remarks

This dissertation presented three large-scale user studies on modern CAPTCHAS with over 6000 unique participants and over 100,000 CAPTCHA solutions.

Chapter 2 introduces the initial study titled “An Empirical Study & Evaluation of Modern CAPTCHAS ”. This study aimed to address five main research questions: 1. How long do human users take to solve different types of CAPTCHAS? 2. What CAPTCHA types do users prefer? 3. Does the experimental context affect solving time? 4. Do demographics affect solving time? 5. Does the experimental context influence abandonment?

To answer these questions, a study involving 1-400 participants was carried out on MTurk, with each participant solving 10 CAPTCHAS. The results revealed the following: 1. Humans take longer than bots to solve known attacks on CAPTCHA types. 2. User preference does not entirely align with CAPTCHA solving time. 3. The experimental context significantly impacts CAPTCHA solving times. 4. Age has an impact on solving time. 5. There is a high abandonment rate due to CAPTCHA-related tasks.

Chapter 3 discusses the second study titled “Understanding reCAPTCHA_{v2} via a Large-

Scale Live User Study”. This study investigates reCAPTCHA v2 in a real-world scenario, using a service for account creation and password recovery. The study involved over 3,600 unbiased participants, who solved over 9,000 reCAPTCHA v2 challenges. The study makes three main contributions:

1. A comprehensive quantitative analysis of reCAPTCHA v2 solving time.
2. An in-depth qualitative analysis of reCAPTCHA v2 usability.
3. A detailed discussion of the cost and security of reCAPTCHA v2.

The results of the study show the following:

1. Solving time is influenced by the number of attempts, the service or website settings, educational level, and field of study.
2. The usability of reCAPTCHA v2 is rated at 77/100 for checkbox-only challenges and 59/100 for checkbox and image challenges, using the SUS scale.
3. The estimated cost of over 13 years of deploying reCAPTCHA v2 is 819 million hours of human time. Additionally, Google may have potentially profited USD \$888 billion from cookies and USD \$8.75-32.3 billion per sale of their total labeled dataset.
4. Regarding security, reCAPTCHA v2 presents vulnerabilities such as click-jacking, trivial implementation of large-scale automation attacks, weakness of the security premise of fallback (image challenge), and usage of privacy-invasive tracking cookies.

The study concludes that reCAPTCHA v2 and similar technologies should be deprecated.

Chapter 4 discusses the third study, ”Exploring CAPTCHA-Induced Task Abandonment with QUITCHA.” The study addressed five main research questions: 1. What is inside the ”black box” of image labeling tasks? 2. What factors influence session abandonment induced by image labeling captchas? 3. What behavioral conclusions can be drawn from fine-grained event logging? 4. How long do users take to solve reCAPTCHA v2-like image labeling tasks? 5. What factors influence solving time and accuracy?

The results revealed the following: 1. QUITCHA captures fine-grained timing and accuracy features that exhibit statistically significant trends. 2. Repeated image selection challenges lead to significant quitting rates ranging from 26% to 41%. Multiple features influence dropout rates. 3. Exact accuracy, selections, clicks, timing, and function tracing allow for the recreation of user interaction and show statistically significant results across certain features. 4. A large-scale fine-grained image solution dataset and analysis toolset are provided to quantify various solution-related timing events. 5. New features were discovered with statistically significant trends.

Ultimately, the study presents an open-source alternative to reCAPTCHA_{v2} for website or service operators who wish to label their datasets while keeping their user’s data private.

We hope that the results presented in this work help improve the security standard on the web and unburden regular users from wasting time solving CAPTCHAS in the name of security.

Bibliography

- [1] 360.cn. <https://passport.360.cn/>.
- [2] Cisco Umbrella 1 Million. <https://umbrella.cisco.com/blog/cisco-umbrella-1-million>.
- [3] Cloudflare Radar - Domain Rankings. <https://radar.cloudflare.com/domains>.
- [4] Degree of reliance on third-party cookies in digital advertising in the United States as of July 2021. <https://www.statista.com/statistics/1222230/reliance-cookies-advertising-usa/>.
- [5] jrj.com. <https://sso.jrj.com/>.
- [6] NuData Security. <https://nudatasecurity.com/>.
- [7] Spending on third-party audience data supporting marketing related efforts in the United States from 2017 to 2021, by type. <https://www.statista.com/statistics/1202754/third-party-audience-data-spending-usa/>.
- [8] The Majestic Million. <https://majestic.com/reports/majestic-million>.
- [9] The Truth In User Privacy And Targeted Ads. <https://www.forbes.com/sites/forbestechcouncil/2022/02/24/the-truth-in-user-privacy-and-targeted-ads/>.
- [10] Xinhuanet. <https://mail.xinhuanet.com>.
- [11] Teaching computers to read: Google acquires recaptcha, Sep 2009.
- [12] recaptcha faq from 2010 archived, 2010.
- [13] Are you a robot? Introducing “No CAPTCHA reCAPTCHA”. <https://security.googleblog.com/2014/12/are-you-robot-introducing-no-captcha.html>, 2014.
- [14] reCAPTCHA v3. <https://developers.google.com/search/blog/2018/10/introducing-recaptcha-v3-new-way-to>, 2018.
- [15] Google will pay 22.5 million to settle ftc charges it misrepresented privacy assurances to users of apple’s safari internet browser, Feb 2019.
- [16] MongoDB. <https://www.mongodb.com/>, 2021.

- [17] Node.js. <https://nodejs.org/>, 2021.
- [18] 2023.
- [19] 2023.
- [20] 2023.
- [21] Ai platform data labeling service pricing, 2023.
- [22] Amazon Mechanical Turk. <https://www.mturk.com/>, 2023.
- [23] Arkose Labs. <https://www.arkoselabs.com/about-us/>, 2023.
- [24] CAPTCHA Usage Distribution on the Entire Internet. <https://trends.builtwith.com/widgets/captcha/traffic/Entire-Internet>, 2023.
- [25] Energy information administration faq, 2023.
- [26] GeeTest CAPTCHA. <https://www.geetest.com/en/Captcha>, 2023.
- [27] Google recaptcha admin dashboard, 2023.
- [28] Google recaptcha v2 image dataset with partially hand-marked images for yolo., 2023.
- [29] hCaptcha. <https://www.hcaptcha.com/>, 2023.
- [30] hCaptcha Is Now The Largest Independent CAPTCHA Service, Runs on 15% Of The Internet. <https://www.hcaptcha.com/post/hcaptcha-now-the-largest-independent-captcha-service>, 2023.
- [31] The post study survey via google forms (interactive version) do not submit personal info, 2023.
- [32] reCAPTCHA. <https://www.google.com/recaptcha/about/>, 2023.
- [33] reCAPTCHA v2. <https://developers.google.com/recaptcha/docs/display>, 2023.
- [34] reCAPTCHA v3. <https://developers.google.com/recaptcha/docs/v3>, 2023.
- [35] Scipy is an open-source software for mathematics, science, and engineering., 2023.
- [36] The top 500 sites on the web. <https://www.alexa.com/topsites>, 2023.
- [37] The Tor Project: Privacy & Freedom Online. <https://www.torproject.org/>, 2023.
- [38] Usa environmental protection agency greenhouse gas calculator, 2023.
- [39] scikit-posthocs is a python package which provides post hoc tests for pairwise multiple comparisons that are usually performed in statistical data analysis to assess the differences between group levels if a statistically significant result of anova test has been obtained., 2024.

- [40] U.s. department of labor federal minimum wage., 2024.
- [41] L. Ahn, M. Blum, and J. Langford. How lazy cryptographers do ai. *Communications of The ACM - CACM*, 01 2002.
- [42] W. Aiken and H. Kim. POSTER: DeepCRACK: Using Deep Learning to Automatically CRack Audio CAPTCHAs. In *Proceedings of the 2018 on Asia Conference on Computer and Communications Security, ASIACCS '18*, page 797–799, New York, NY, USA, 2018. ACM.
- [43] K. Akama, Y. Nakatsuka, M. Sato, and K. Uehara. Scrappy: Secure rate assuring protocol with privacy. In *Proceedings 2024 Network and Distributed System Security Symposium, NDSS 2024*. Internet Society, 2024.
- [44] I. Akrouf, A. Feriani, and M. Akrouf. Hacking google recaptcha v3 using reinforcement learning. *arXiv preprint arXiv:1903.01003*, 2019.
- [45] F. H. Alqahtani and F. A. Alsulaiman. Is image-based CAPTCHA secure against attacks based on machine learning? An experimental study. *Computers & Security*, 88:101635, 2020.
- [46] J. Aslan, K. Mayers, J. G. Koomey, and C. France. Electricity intensity of internet data transmission: Untangling the estimates. *Journal of Industrial Ecology*, 22(4):785–798, 2018.
- [47] A. Bangor, P. Kortum, and J. Miller. Determining what individual sus scores mean: Adding an adjective rating scale. *Journal of usability studies*, 4(3):114–123, 2009.
- [48] M. Belk, P. Germanakos, C. Fidas, A. Holzinger, and G. Samaras. Towards the Personalization of CAPTCHA Mechanisms Based on Individual Differences in Cognitive Processing. In A. Holzinger, M. Ziefle, M. Hitz, and M. Debevc, editors, *Human Factors in Computing and Informatics*, pages 409–426, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.
- [49] J. P. Bigham and A. Cavender. Evaluating Existing Audio CAPTCHAs and an Interface Optimized for Non-Visual Use. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '09*, page 1829–1838, New York, NY, USA, 2009. ACM.
- [50] K. Bock, D. Patel, G. Hughey, and D. Levin. unCaptcha: A Low-Resource Defeat of reCaptcha’s Audio Challenge. In *11th USENIX Workshop on Offensive Technologies (WOOT 17)*, Vancouver, BC, Aug. 2017. USENIX Association.
- [51] J. Brooke et al. Sus-a quick and dirty usability scale. *Usability evaluation in industry*, 189(194):4–7, 1996.
- [52] E. Bursztein, R. Beauxis, H. Paskov, D. Perito, C. Fabry, and J. Mitchell. The Failure of Noise-Based Non-continuous Audio Captchas. In *2011 IEEE Symposium on Security and Privacy*, pages 19–31, 2011.

- [53] E. Bursztein, S. Bethard, C. Fabry, J. C. Mitchell, and D. Jurafsky. How Good Are Humans at Solving CAPTCHAs? A Large Scale Evaluation. In *2010 IEEE Symposium on Security and Privacy*, pages 399–413, 2010.
- [54] V. G. Cerf. Guidelines for Internet Measurement Activities. RFC 1262, Oct. 1991.
- [55] J. Chen, X. Luo, Y. Guo, Y. Zhang, and D. Gong. A Survey on Breaking Technique of Text-Based CAPTCHA. *Security and Communication Networks*, 12 2017.
- [56] M. Chew and J. D. Tygar. Image recognition captchas. In K. Zhang and Y. Zheng, editors, *Information Security*, pages 268–279, Berlin, Heidelberg, 2004. Springer Berlin Heidelberg.
- [57] M. Chmielewski and S. C. Kucker. An MTurk crisis? Shifts in data quality and the impact on study results. *Social Psychological and Personality Science*, 11(4):464–473, 2020.
- [58] F. Consulting. State of online fraud and bot management. https://services.google.com/fh/files/misc/google_forrester_bot_management_tlp_post_production_final.pdf, 2021.
- [59] M. Darnstädt, H. Meutzner, and D. Kolossa. Reducing the Cost of Breaking Audio CAPTCHAs by Active and Semi-supervised Learning. In *2014 13th International Conference on Machine Learning and Applications*, pages 67–73, 2014.
- [60] Y. Feng, Q. Cao, H. Qi, and S. Ruoti. Sencaptcha: A mobile-first captcha using orientation sensors. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 4(2), jun 2020.
- [61] C. A. Fidas, A. G. Voyiatzis, and N. M. Avouris. On the Necessity of User-Friendly CAPTCHA. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '11, page 2623–2626, New York, NY, USA, 2011. ACM.
- [62] C. Fritsch, M. Netter, A. Reisser, and G. Pernul. Attacking image recognition captchas. In S. Katsikas, J. Lopez, and M. Soriano, editors, *Trust, Privacy and Security in Digital Business*, pages 13–25, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg.
- [63] H. Gao, W. Wang, and Y. Fan. Divide and conquer: an efficient attack on Yahoo! CAPTCHA. In *2012 IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications*, pages 9–16. IEEE, 2012.
- [64] H. Gao, J. Yan, F. Cao, Z. Zhang, L. Lei, M. Tang, P. Zhang, X. Zhou, X. Wang, and J. Li. A Simple Generic Attack on Text Captchas. In *Network and Distributed System Security Symposium (NDSS)*, San Diego, California, United States, 2016.
- [65] H. Gao, D. Yao, H. Liu, X. Liu, and L. Wang. A Novel Image Based CAPTCHA Using Jigsaw Puzzle. In *2010 13th IEEE International Conference on Computational Science and Engineering*, pages 351–356, 2010.

- [66] S. Gao, M. Mohamed, N. Saxena, and C. Zhang. Emerging-Image Motion CAPTCHAs: Vulnerabilities of Existing Designs, and Countermeasures. *IEEE Transactions on Dependable and Secure Computing*, 16(6):1040–1053, 2019.
- [67] I. J. Goodfellow, Y. Bulatov, J. Ibarz, S. Arnoud, and V. Shet. Multi-digit number recognition from street view imagery using deep convolutional neural networks. *arXiv preprint arXiv:1312.6082*, 2014.
- [68] M. Guerar, L. Verderame, M. Migliardi, F. Palmieri, and A. Merlo. Gotta CAPTCHA 'Em All: A Survey of Twenty years of the Human-or-Computer Dilemma. *CoRR*, abs/2103.01748, 2021.
- [69] G. Gugliotta. Deciphering old texts, one woozy, curvy word at a time, 2011.
- [70] C. J. Hernandez-Castro and A. Ribagorda. Pitfalls in CAPTCHA design and implementation: The Math CAPTCHA, a case study. *Computers & Security*, 29(1):141–157, 2010.
- [71] C. J. Hernandez-Castro and A. Ribagorda. Pitfalls in captcha design and implementation: The math captcha, a case study. *Computers & Security*, 29(1):141–157, 2010.
- [72] C.-J. Ho, C.-C. Wu, K.-T. Chen, and C.-L. Lei. DevilTyper: A Game for CAPTCHA Usability Evaluation. *Comput. Entertain.*, 9(1), apr 2011.
- [73] E. Homakov. The no captcha problem, 2014.
- [74] M. I. Hossen and X. Hei. A Low-Cost Attack against the hCaptcha System. *CoRR*, abs/2104.04683, 2021.
- [75] M. I. Hossen, Y. Tu, M. F. Rabby, M. N. Islam, H. Cao, and X. Hei. An Object Detection based Solver for Google's Image reCAPTCHA v2. In *23rd International Symposium on Research in Attacks, Intrusions and Defenses (RAID 2020)*, pages 269–284, San Sebastian, Oct. 2020. USENIX Association.
- [76] Imperva. Imperva bad bot report. <https://www.imperva.com/resources/resource-library/reports/bad-bot-report/>, 2022.
- [77] M. Jain, R. Tripathi, I. Bhansali, and P. Kumar. Automatic Generation and Evaluation of Usable and Secure Audio ReCAPTCHA. In *The 21st International ACM SIGACCESS Conference on Computers and Accessibility, ASSETS '19*, page 355–366, New York, NY, USA, 2019. Association for Computing Machinery.
- [78] K. Krol, S. Parkin, and M. A. Sasse. Better the Devil You Know: A User Study of Two CAPTCHAs and a Possible Replacement Technology. In *2016 NDSS Workshop on Usable Security*, pages 1–10, 2016.
- [79] C. Li, X. Chen, H. Wang, P. Wang, Y. Zhang, and W. Wang. End-to-end attack on text-based CAPTCHAs based on cycle-consistent generative adversarial network. *Neurocomputing*, 433:223–236, 2021.

- [80] D. Lorenzi, J. Vaidya, E. Uzun, S. Sural, and V. Atluri. Attacking Image Based CAPTCHAs Using Image Recognition Techniques. In V. Venkatakrisnan and D. Goswami, editors, *Information Systems Security*, pages 327–342, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg.
- [81] K. M. Miller and B. Skiera. Economic consequences of online tracking restrictions: Evidence from cookies. *International Journal of Research in Marketing*, 2023.
- [82] N. Mitra, H. Chu, T. Lee, L. Wolf, H. Yeshurun, and D. Cohen-Or. Emerging images. In *Proceedings of ACM SIGGRAPH Asia 2009, SIGGRAPH Asia '09*, volume 28, pages 163:1–163:8, 2009. ACM SIGGRAPH Asia 2009, SIGGRAPH Asia '09 ; Conference date: 16-12-2009 Through 19-12-2009.
- [83] M. Mohamed, S. Gao, N. Saxena, and C. Zhang. Dynamic Cognitive Game CAPTCHA Usability and Detection of Streaming-Based Farming. In *2014 NDSS Workshop on Usable Security*, pages 1–10, 2014.
- [84] A. Moss. After the bot scare: Understanding what’s been happening with data collection on MTurk and how to stop it. <https://www.cloudresearch.com/resources/blog/after-the-bot-scare-understanding-whats-been-happening-with-data-collection-on-mturk-and-how-to-stop-it/>, Aug 2020.
- [85] M. Motoyama, K. Levchenko, C. Kanich, D. McCoy, G. M. Voelker, and S. Savage. Re: CAPTCHAs—Understanding CAPTCHA-Solving Services in an Economic Context. In *19th USENIX Security Symposium (USENIX Security 10)*, Washington, DC, aug 2010. USENIX Association.
- [86] Y. Nakatsuka, E. Ozturk, A. Paverd, and G. Tsudik. CACTI: Captcha Avoidance via Client-side TEE Integration. In *30th USENIX Security Symposium (USENIX Security 21)*, pages 2561–2578. USENIX Association, aug 2021.
- [87] S. Perez. Google now using recaptcha to decode street view addresses, Mar 2012.
- [88] D. Phillips. Secureimage: PHP CAPTCHA script. <https://www.phpcaptcha.org/>, 2023.
- [89] V. L. Pochat, T. van Goethem, S. Tajalizadehkhoob, M. Korczynski, and W. Joosen. Tranco: A research-oriented top sites ranking hardened against manipulation. In *26th Annual Network and Distributed System Security Symposium, NDSS 2019, San Diego, California, USA, February 24-27, 2019*. The Internet Society, 2019.
- [90] M. Prince and S. Isasi. Moving from reCAPTCHA to hCaptcha. <https://blog.cloudflare.com/moving-from-recaptcha-to-hcaptcha/>.
- [91] S. A. Ross, J. A. Halderman, and A. Finkelstein. Sketcha: A Captcha Based on Line Drawings of 3D Models. In *Proceedings of the 19th International Conference on World Wide Web*, page 821–830, New York, NY, USA, 2010. ACM.

- [92] S. Sano, T. Otsuka, and H. G. Okuno. Solving Google’s Continuous Audio CAPTCHA with HMM-Based Automatic Speech Recognition. In K. Sakiyama and M. Terada, editors, *Advances in Information and Computer Security*, pages 36–52, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.
- [93] Q. Scheitle, O. Hohlfeld, J. Gamba, J. Jelten, T. Zimmermann, S. D. Strowes, and N. Vallina-Rodriguez. A long way to the top: Significance, structure, and stability of internet top lists. In *Proceedings of the Internet Measurement Conference 2018*, IMC ’18, page 478–493, New York, NY, USA, 2018. ACM.
- [94] A. Searles, Y. Nakatsuka, E. Ozturk, A. Paverd, G. Tsudik, and A. Enkoji. An empirical study & evaluation of modern CAPTCHAs. In *32nd USENIX Security Symposium (USENIX Security 23)*, pages 3081–3097, Anaheim, CA, Aug. 2023. USENIX Association.
- [95] A. Searles, R. T. Prapty, and G. Tsudik. Dazed & confused: A large-scale real-world user study of recaptchav2, 2023.
- [96] H. Shekhar. Breaking Audio Captcha using Machine Learning/Deep Learning and Related Defense Mechanism. *San Jose State University Master’s Projects*, 2019.
- [97] V. Shet. Street View and reCAPTCHA technology just got smarter. <https://security.googleblog.com/2014/04/street-view-and-recaptcha-technology.html>, 2014.
- [98] S. Sivakorn. I’m not a human: Breaking the google recaptcha. 2016.
- [99] S. Sivakorn, I. Polakis, and A. D. Keromytis. I am Robot: (Deep) Learning to Break Semantic Image CAPTCHAs. In *2016 IEEE European Symposium on Security and Privacy (EuroS P)*, pages 388–403, 2016.
- [100] S. Solanki, G. Krishnan, V. Sampath, and J. Polakis. In *(Cyber)Space Bots Can Hear You Speak: Breaking Audio CAPTCHAs Using OTS Speech Recognition*, page 69–80. ACM, New York, NY, USA, 2017.
- [101] M. Tang, H. Gao, Y. Zhang, Y. Liu, P. Zhang, and P. Wang. Research on Deep Learning Techniques in Breaking Text-Based Captchas and Designing Image-Based Captcha. *IEEE Transactions on Information Forensics and Security*, 13(10):2522–2537, 2018.
- [102] N. Tanthavech and A. Nimkoompai. Captcha: Impact of website security on user experience. *ICIIT ’19: Proceedings of the 2019 4th International Conference on Intelligent Information Technology*, pages 37–41, 02 2019.
- [103] A. M. Turing. Computing machinery and intelligence. *Mind*, 59(October):433–60, 1950.

- [104] E. Uzun, S. Chung, I. Essa, and W. Lee. rtCaptcha: A Real-Time Captcha Based Liveness Detection System. In *Network and Distributed System Security Symposium (NDSS)*, San Diego, California, United States, 02 2018.
- [105] L. von Ahn, M. Blum, N. J. Hopper, and J. Langford. CAPTCHA: Using Hard AI Problems for Security. In E. Biham, editor, *Advances in Cryptology — EUROCRYPT 2003*, pages 294–311, Berlin, Heidelberg, 2003. Springer Berlin Heidelberg.
- [106] L. von Ahn, B. Maurer, C. McMillen, D. Abraham, and M. Blum. recaptcha: Human-based character recognition via web security measures. *Science*, 321(5895):1465–1468, 2008.
- [107] M. A. Webb and J. P. Tangney. Too good to be true: Bots and bad data from mechanical turk. *Perspectives on Psychological Science*, 2022.
- [108] H. Weng, B. Zhao, S. Ji, J. Chen, T. Wang, Q. He, and R. Beyah. Towards understanding the security of modern image captchas and underground captcha-solving services. *Big Data Mining and Analytics*, 2(2):118–144, 2019.
- [109] Q. Xie, S. Tang, X. Zheng, Q. Lin, B. Liu, H. Duan, and F. Li. Building an open, robust, and stable voting-based domain top list. In K. R. B. Butler and K. Thomas, editors, *31st USENIX Security Symposium, USENIX Security 2022, Boston, MA, USA, August 10-12, 2022*, pages 625–642. USENIX Association, 2022.
- [110] Y. Xu, G. Reynaga, S. Chiasson, J.-M. Frahm, F. Monrose, and P. Van Oorschot. Security and usability challenges of moving-object captchas: Decoding codewords in motion. In *Proceedings of the 21st USENIX Conference on Security Symposium, Security’12*, page 4, USA, 2012. USENIX Association.
- [111] Y. Xu, G. Reynaga, S. Chiasson, J.-M. Frahm, F. Monrose, and P. C. van Oorschot. Security analysis and related usability of motion-based captchas: Decoding codewords in motion. *IEEE Transactions on Dependable and Secure Computing*, 11(5):480–493, 2014.
- [112] J. Yan and A. S. El Ahmad. A Low-cost Attack on a Microsoft CAPTCHA. In *Proceedings of the 15th ACM conference on Computer and communications security*, pages 543–554, 2008.
- [113] J. Yan and A. S. El Ahmad. Usability of captchas or usability issues in captcha design. In *Proceedings of the 4th Symposium on Usable Privacy and Security, SOUPS ’08*, page 44–52, New York, NY, USA, 2008. ACM.
- [114] H. Yu and M. O. Riedl. Automatic generation of game-based captchas. In *Proceedings of the FDG workshop on Procedural Content Generation*, 2015.
- [115] Y. Zi, H. Gao, Z. Cheng, and Y. Liu. An End-to-End Attack on Text CAPTCHAs. *IEEE Transactions on Information Forensics and Security*, 15:753–766, 2020.