

Can a High Color Contrast Touch Interface Increase User Reaction Time when Using a Smart Phone Web Based Application.

Albin Hübsch

Department of Computing Science
Umeå University, Sweden
`id11ahh@cs.umu.se`

Abstract. This study aims to investigate if it is possible to receive better user performance when using a high color contrast compared to a low color contrast touch screen interface in a mobile web application. The performance was measured by combining the time each user took to respond to instructions (reaction time) and the number of input errors from the user. Our result shows that the low color contrast interface produced 475% more user input errors compared to the high color contrast interface. No difference in reaction time could be found between the two versions.

1 Introduction

The main goal with this paper is to evaluate if color contrast has a significant impact on usability in web based smart-phone applications. Consumer touch screen devices such as smart-phones have rapidly increased in amount and availability recent years. The touch screen technology have made great advances [1] and it is frequently used as a way of receiving user input. When this technology moves to a broader audience, higher demands on usability needs to be set [2]. To further explore how usability can be improved using touch screen technology we have in this paper investigated if reaction time (the time it takes for a user to react on instructions, further referred to as RT in this paper) and user input errors (the number of user input errors when interacting with a system, further referred to as UIE in this paper) can reach better results by increasing color contrast within the interface. Better results are assumed as faster RTs and less number of UIEs.

Earlier studies completed in this field has explored how we perceive different color contrast and how it can affect our reading performance [3]. It has also been proved that chromaticity, contrast, and cone opponency in color space can affect RTs [4]. Many best practices for Mobile development have also shown that "requirements for sufficient color contrast" [5] must be met to best exaggerate the content. Still the question remains, can a high color contrast interface increase the the RT and decrease UIE?

High RTs and low user IE are especially desirable when designing user interfaces for situations with high demands on quick user input and low error

tolerance such as in emergency situations. The results of this paper can be used by any designer or developer.

2 Method

In order to be able to test if color contrast has a substantial impact on the RT and UIE of a touch screen user interface we have designed a simple web application for an Android smart-phone. The application exists in two versions, one with low color contrast (referred to as LCC in this paper) and one with high color contrast (referred to as HCC in this paper) as can be seen in Fig 1. We conducted an A/B test to measure the performance between the versions.

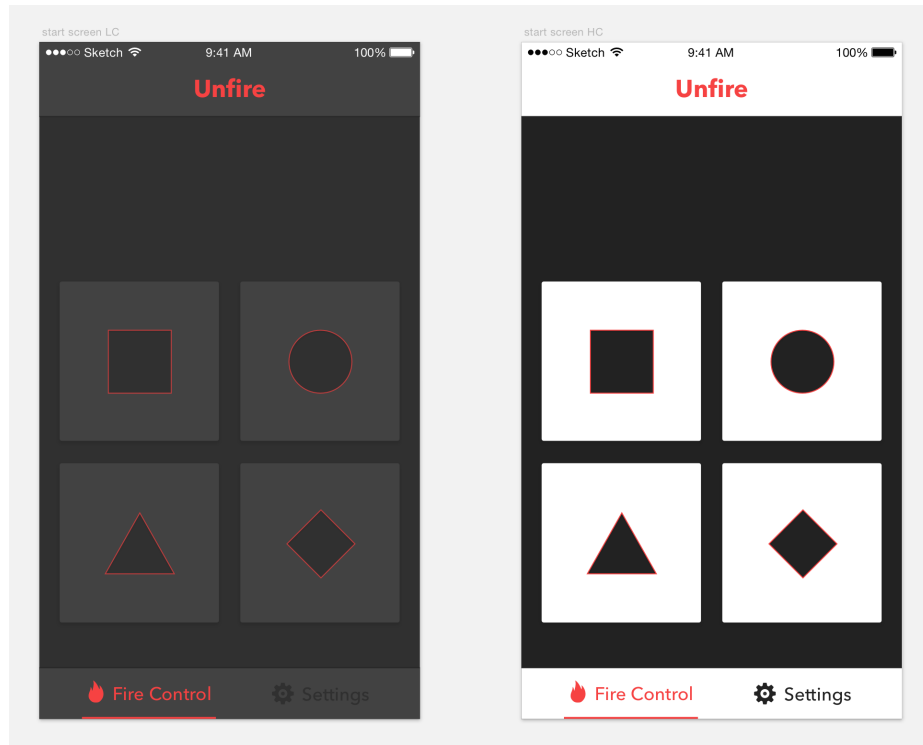


Fig. 1. Our two versions of the application designed with different color contrast. To the left the low color contrast interface and to the right the high color contrast interface. The application and its interface was designed specifically for this test.

We designed the application by following design guidelines given here [6] [7] [2] [8]. These guidelines can be seen as a set of rules to follow when designing any kind of user interface. We used guidelines to get proper sized buttons, good spacing between content and best positioning of navigation.

2.1 Designing the A/B test

We conducted an A/B test to measure the differences in RT and UIE between the two versions. An A/B test is a commonly used method by developers and designers to test differences in performance between applications when the differentiated factors are known [7]. Where in this case our known factor is the contrast difference.

The application consists of 4 buttons¹, each button representing a function. To differentiate each button from the others we put a unique shape in each button. The shapes we used were a, square, circle, triangle and a rhomb (Fig 1).

To be able to give the test persons consistent instructions through the complete test we designed a program that presented the instructions on a secondary screen. The instructions were a series of 20 instructions with shapes identical

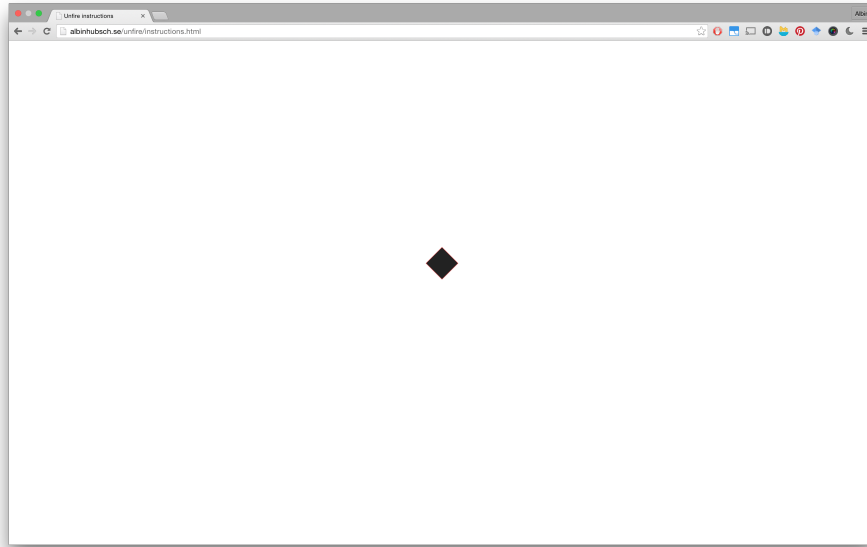


Fig. 2. Our program designed to present user instructions. Instructions were given in a series of 20. Between each instruction there was a delay set to be between 2.5 - 4 seconds. Each time a new symbol where showed a timestamp were logged in a unique user instruction file.

to the ones in the application buttons. The test persons were told to press the representative shape in the application as fast as they could in order to measure the reaction time. If they pressed the wrong button they were told to continue the test by pressing the right button, the application registered this as an UIE.

¹ On screen button

Each test was performed indoors with varying surroundings such as different lightning conditions and noise levels.

We tested each version of the application on 5 persons, all within the same age group (20-30) and with a variety of backgrounds. In total we got 10 test results with a gender split of 50% women and 50% men that we later grouped into LCC and HCC. We analyzed them against each other to measure if any differences in RT or UIE between the two versions could be statistically confirmed.

2.2 Evaluation of the A/B test

Our data received from the tests consisted of UNIX timestamps² measured in milliseconds. Each time the user interacted with the application it registered which button that was pressed and saved it together with a UNIX timestamp. Equally, every time the instruction program showed a new symbol it logged the symbol together with a UNIX timestamp. As a result we received two data sets of symbols with corresponding timestamps for each test person, resulting in a total of 200 records. Although the data was hard to interpret due to its compact look, a python script was written to structure the resulting timestamps into a more readable format. We calculated each RT by subtracting the timestamp when the instruction was given from the registered user input timestamp. Out of this we could calculate the mean RT for both groups LCC and HCC. The calculation of RT did not take in consideration if user input was a no match. It was still considered as an reaction.

UIEs were counted every time the user pressed a non matching object compared to the one given in the instructions. All IEs in each group, LCC and HCC, were counted to see if one of the groups produced more UIEs.

3 Result

From our results we can not draw any statistically significant differences in reaction time between the two groups high color contrast and low color contrast. This is most probably due to corrupt test data, a result from limitations in web application technology, see section 4. With user input errors we found that the low color contrast group produced significantly more user input errors than the high color contrast group.

3.1 Reaction Times (RT)

Table 1 shows our average RT and standard deviation for each group. According to the results people in HCC had insignificant slower RTs compared to the ones in LCC.

To validate that the groups RTs are insignificantly different we assumed the two data sets were normal distributed based on the $n > 30$ rule, and performed

² More info at: <http://www.unixtimestamp.com/>

	Average RT	Standard Deviation
HC	783 ms	969 ms
LC	550 ms	1393 ms

Table 1. Average reaction time and standard deviation for each group, low color contrast and high color contrast. A two tailed t-test was performed on the two groups and showed that no significant difference could be determined between them.

a two tailed T-Test. The T-Test produced a p-value of 0.17 which is bigger than a desired error margin (α) of 5%. The T-Test is therefore proving that no differences in RT exists within our data. Regardless what the t-test says our standard deviations are indicating that something might be wrong with our data (see section 4).

3.2 Input Errors (IE)

None of the groups were completely free from user input errors. As can be seen in table 2, users that were given the LCC interface produced 475% more UIE than the HCC group.

	Errors	Standard Deviation	Errors/User
HC	4	1.10	0.8
LC	19	3.49	3.8

Table 2. Number of user input errors registered in each group and a calculation of IEs per user in each group. The low color contrast interface produced 475% more UIE compared to the high color contrast version. A two tailed t-test also showed that there is a significant difference between the two versions. A high color contrast interface can therefore be a preferred design method if a low amount of user input errors is desired.

Table 2 shows us that the users in the HCC group generally had a more stable and consistent performance compared to the LCC group. We also performed a two tailed t-test on the UIE data set. The outcome of this test was a p-value of 0.001 which is significantly smaller than a α of 5% telling us that the HCC group performed better.

4 Discussion

Our test results have shown that we can not increase the user reaction time by using a high color contrast interface when using a smart phone web based application. However we could see that the high color contrast interface was more

reliable in performance when it comes to the amount of user input errors, according to our results. What we have seen in our collected test data can strongly negotiate the legibility of the results. The calculated standard deviations gave us hints on that something was wrong. We dug deeper into each test persons results and found that the data must be corrupt. We found that we had randomly received negative RTs which in theory should be completely impossible. We were prepared that the timestamps could be off sync due to individual system clocks and no boot pairing. This would have resulted in a consistent off sync. Instead we got inconsistent hick ups in our timestamps. We suspect this is a result from either hardware limitations or limitations in the JavaScript³ engine for web applications on Android.

Our conclusion is that HCC interfaces are the preferred method to design interfaces as they deliver a more stable user interaction pattern. Apart from this, smart phone web based applications should only be used if sometimes slow technology, user input errors and long reaction times from the user can be accepted.

4.1 Drawbacks and Limitations

The main drawback of this study is the corrupt data set. Something to have in mind when we draw any conclusions out of the results. This problem could be solved by simply rerunning the whole test using a native smart phone application instead of a web based solution.

Each test person performed only one test with a supervisory selected interface, the LCC or HCC version. To eliminate possible affecting surrounding factors each person should have done the test several times and with both the LCC and HCC interfaces. This would also have made it possible to recognize if the test persons memorized the patterns in any way.

Small test group. To be able to get a more statistical valid result the test group has to be bigger.

Many affecting parameters. The prototype is simple in its appearance but there are still many parameters that can affect the users interaction and their RTs. Icons, icon size, optimal number of on screen buttons, size of hand-held device etc. All this, and more, should have been researched and taken into account if done again.

Limited test environment. Our tests were made using an Android based smart-phone⁴ and the application were made using HTML5⁵, CSS3⁶ and JavaScript. We can not surely imply, without testing, that the same results would appear on an iPhone or any other smart-phone or even with a native application.

³ A prototype based dynamic scripting language

⁴ OnePlus Two <https://oneplus.net/2>

⁵ HyperText Markup Language, markup standard for WWW

⁶ Cascading Style Sheet, styling standard for WWW

4.2 Future Work

Due to the failed data set that we received in this paper there are things that could be improved or continued to be worked on. We do have some concrete suggestions that could be case for future work.

- Perform tests using a native application.
- Extend the test group with more test subjects. It is possible to think that the patterns that slightly appeared in our results will appear even more significant with a larger group of test subjects.
- Our collected test data can be downloaded and used freely to investigate other aspects not mentioned in this paper. One proposal is to look at the different shapes and try to detect possible error patterns between them. Is the rhomb more frequently a case for IE compared to the other shapes?
- Do our results apply on other systems and techniques? Our tests are limited to the Android system and a web based application solution. It remains to answer if our results applies on all other systems. This is also a way of isolating some of the affecting parameters mentioned in 4.1.

5 Acknowledgments

The author would like to thank the peer reviewers who provided valuable comments on both the content and structure of this paper.

References

1. Jennings, A., Ryser, S., Drews, F.: Touch screen devices and the effectiveness of user interface methods. In: Proceedings of the Human Factors and Ergonomics Society Annual Meeting. Volume 57., SAGE Publications (2013) 1648–1652
2. Gong, J., Tarasewich, P.: Guidelines for handheld mobile device interface design. In: Proceedings of DSI 2004 Annual Meeting, Citeseer (2004) 3751–3756
3. Wu, J.H., Yuan, Y.: Improving searching and reading performance: the effect of highlighting and text color coding. *Information & Management* **40**(7) (2003) 617–637
4. McKeefry, D.J., Parry, N.R., Murray, I.J.: Simple reaction times in color space: the influence of chromaticity, contrast, and cone opponency. *Investigative Ophthalmology & Visual Science* **44**(5) (2003) 2267–2276
5. Marcus, A.: Design, User Experience, and Usability: Health, Learning, Playing, Cultural, and Cross-cultural User Experience: Second International Conference, DUXU 2013, Held as Part of HCI International 2013, Las Vegas, NV, USA, July 21–26, 2013, Proceedings. Volume 8013. Springer (2013)
6. Hooper, S., Berkman, E.: Designing mobile interfaces. " O'Reilly Media, Inc." (2011)
7. Johnson, J.: Designing with the Mind in Mind: Simple Guide to Understanding User Interface Design Guidelines. Elsevier (2013)
8. Norman, D.A.: The design of everyday things: Revised and expanded edition. Basic books (2013)