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. Here goes paper abstract

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 has rapidly increased in amount and availability recent years. The touch screen technology have made great advances [1] and it is more 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 are in this paper investigating if reaction time (RT) and user input errors (IE) can reach better results by increasing color contrast within the interface. Earlier work done in this field are exploring 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 also shows that "requirements for sufficient color contrast" [5] must be meet to best exaggerate the content. But still the question remains, if a high color contrast interface can increase the the RT[1] and decrease user IE[2].

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 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 an substantial impact on the RT and user IE of a touch user interface we have designed a simple web application

[1] Reaction time: the time it takes for a user to react on instructions

[2] Input errors: the number of user input errors when interacting with a system

for an Android smart-phone. The application exists in two versions, one with low color (LC) contrast and one with high color contrast (HC) as can be seen in Fig 1.

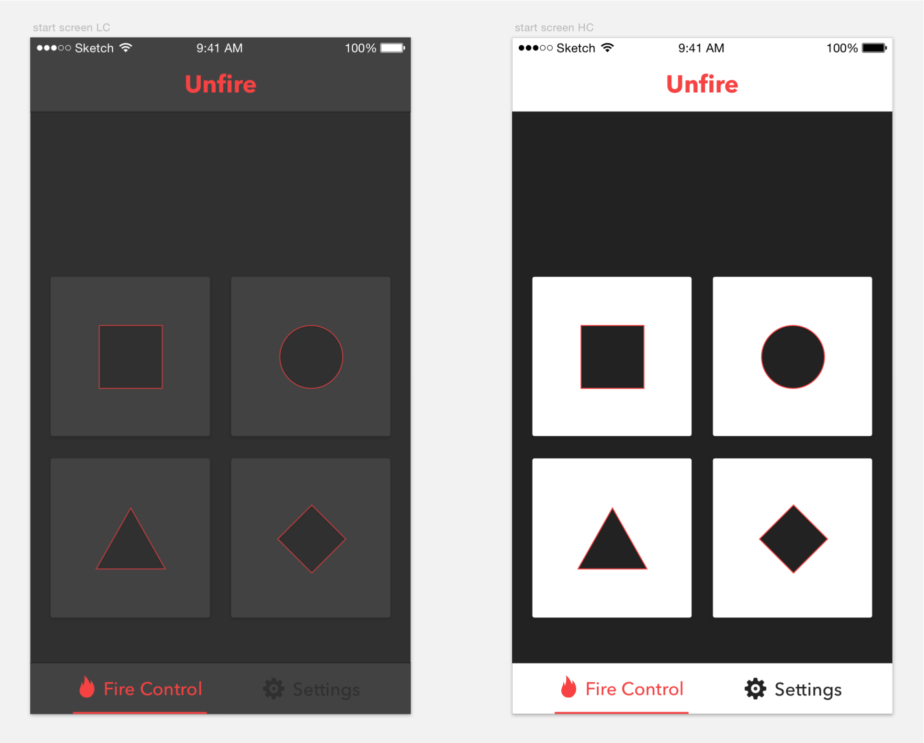

Fig. 1. Our two color contrast versions of our application designed.

We designed the application by following design guidelines given here [6] and [7].

2.1 Designing the A/B test

We conducted an A/B test to measure the differences in RT and IE 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]. As in this case our known factor is the contrast difference.

The application consists of 4 buttons [3], each button representing a function. To fraction each button from the others we put a unique shape in each button. The shapes we used are a, square, circle, triangle and a rhomb 1.

[3] On screen button

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 shapes equally to the shapes in the application buttons and the test persons where told to press the representative shape in the application as fast as they could to measure the reaction time. If they pressed the wrong button they where told to continue the test by pressing the right button, the application registered this as an IE. Each test were 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. We had a gender split of 50% women and 50% men. In total we got 10 test results that we later analyzed against each other to measure if any differences in RTs or IEs between the two versions could be statistically confirmed.

2.2 Evaluation of the A/B test

Our test data received from the tests consisted of UNIX timestamps [4] 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 the instruction program did the same. Every time it showed a new symbol it logged that together with a UNIX timestamp. As a result we received two sets of symbols with corresponding timestamps for each test person, resulting in a total of approximately 400 records. All this data is very hard to interpret due to its compact look produced by the loggers. We designed a python script to parse all records and put it into more of a readable dataset based on groupings of HC and LC.

3 Result

All results produced from the tests are here presented with clear numbers, conclusions and figures/graphics. All tests are done but we are still analyzing the data.

4 Discussion

In this section we will discuss the results from the tests. We will discuss what the results mean and how they should and could be interpreted. We are currently in the stage of analyzing our data.

4.1 Drawbacks and Limitations

One of the main drawbacks of this study is the limited number of test subjects. In order to get more credible result a larger group of test persons should be

[4] `http://www.unixtimestamp.com/`

included. Preferable we should also had let the test persons do the test several of times to eliminate possible surrounding factors that could have affect the results.

Below is a asd

Our group of test subjects is to small 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 soft-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 with using an Android smart-phone [5] system and the application were done using HTML5 and CSS3. We can not surely imply, without testing, that the same results would appear on an iPhone or any other smart-phone.

4.2 Future Work

Due to limited time given for this paper there are things that could be improved or just continued to be worked on. We have some concrete suggestions that could be a case for future work.

- 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 aspect 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 pressed wrongly compared to the other shapes?
- Does our results apply on other systems and techniques? Our tests are limited to the Android system and a web application solution. It remains to answer if our results applies on all other systems. This is also a way of isolate some of all the affecting parameters mentioned in **??**.

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

[5] OnePlus two `https://oneplus.net/2`

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. Hoober, 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. Paredes, H., Fonseca, B., Cabo, M., Pereira, T., Fernandes, F.: Sosphone: a mobile application for emergency calls. Universal Access in the Information Society **13**(3) (2014) 277–290
9. Sax, C., Lawrence, E.: Point-of-treatment: Touchable e-nursing user interface for medical emergencies. In: Mobile Ubiquitous Computing, Systems, Services and Technologies, 2009. UBICOMM'09. Third International Conference on, IEEE (2009) 89–95
10. Jørgensen, M.: Software quality measurement. Advances in engineering software **30**(12) (1999) 907–912
11. Norman, D.A.: The design of everyday things: Revised and expanded edition. Basic books (2013)