Volume 4 • Issue 1 • April 2013

INTERNATIONAL JOURNAL OF HUMAN COMPUTER INTERACTION (IJHCI)

Publication Frequency: 6 Issues / Year

CSC PUBLISHERS
http://www.cscjournals.org

INTERNATIONAL JOURNAL OF HUMAN COMPUTER INTERACTION (IJHCI)

VOLUME 4, ISSUE 1, 2013

EDITED BY
DR. NABEEL TAHIR

ISSN (Online): 2180-1347

International Journal of Human Computer Interaction (IJHCI) is published both in traditional paper form and in Internet. This journal is published at the website http://www.cscjournals.org, maintained by Computer Science Journals (CSC Journals), Malaysia.

IJHCI Journal is a part of CSC Publishers

Computer Science Journals

http://www.cscjournals.org

INTERNATIONAL JOURNAL OF HUMAN COMPUTER INTERACTION (IJHCI)

Book: Volume 4, Issue 1, April 2013
Publishing Date: 30-04-2013
ISSN (Online): 2180-1347

IJHCI Journal is a part of CSC Publishers
http://www.cscjournals.org

Published in Malaysia

Typesetting: Camera-ready by author, data conversation by CSC Publishing Services – CSC Journals, Malaysia

CSC Publishers, 2013

EDITORIAL PREFACE

This is *first* Issue of Volume *four* of International Journal of Human Computer Interaction (IJHCI). IJHCI is an International refereed journal for publication of current research in Human Computer Interaction. Publications of IJHCI are beneficial for researchers, academics, scholars, advanced students, practitioners, and those seeking an update on current experience, state of the art research theories and future prospects in relation to applied science. Some important topics covers by IJHCI are affective computing, agent models co-ordination and communication, computer mediated communication, innovative interaction techniques and user interface prototyping for interactive systems etc.

The initial efforts helped to shape the editorial policy and to sharpen the focus of the journal. Started with Volume 4, 2013, IJHCI appears with more focused issues related to human computer interaction studies. Besides normal publications, IJHCI intend to organized special issues on more focused topics. Each special issue will have a designated editor (editors) – either member of the editorial board or another recognized specialist in the respective field.

This journal publishes new dissertations and state of the art research to target its readership that not only includes researchers, industrialists and scientist but also advanced students and practitioners. IJHCI seeks to promote and disseminate knowledge in the applied sciences, natural and social sciences industrial research materials science and technology, energy technology and society including impacts on the environment, climate, security, and economy, environmental sciences, physics of the games, creativity and new product development, professional ethics, hydrology and water resources, wind energy.

IJHCI editors understand that how much it is important for authors and researchers to have their work published with a minimum delay after submission of their papers. They also strongly believe that the direct communication between the editors and authors are important for the welfare, quality and wellbeing of the Journal and its readers. Therefore, all activities from paper submission to paper publication are controlled through electronic systems that include electronic submission, editorial panel and review system that ensures rapid decision with least delays in the publication processes.

To build its international reputation, we are disseminating the publication information through Google Books, Google Scholar, Directory of Open Access Journals (DOAJ), Open J Gate, ScientificCommons, Docstoc, Scribd, CiteSeerX and many more. Our International Editors are working on establishing ISI listing and a good impact factor for IJHCI. We would like to remind you that the success of our journal depends directly on the number of quality articles submitted for review. Accordingly, we would like to request your participation by submitting quality manuscripts for review and encouraging your colleagues to submit quality manuscripts for review. One of the great benefits we can provide to our prospective authors is the mentoring nature of our review process. IJHCI provides authors with high quality, helpful reviews that are shaped to assist authors in improving their manuscripts.

Editorial Board Members
International Journal of Human Computer Interaction (IJHCI)

EDITORIAL BOARD

TABLE OF CONTENTS

Volume 4, Issue 1, April 2013

Pages

Identifying the Factors Affecting Users' Adoption of Social Networking

Duygu Fındık Coşkunçay *fduygu@metu.edu.tr*
Informatics Institute/Information Systems
Middle East Technical University
Ankara, 06800, Turkey

Abstract

Through the rapid expansion of information and communication technologies, social networking sites have received much more attention in the scope of internet communication. Success of a social web primarily depends on users' satisfaction. In this context, this study aims to identify the influencing factors that affect users' satisfaction towards social networking site use. A multidimensional model has been proposed based on the Information Quality, System Quality, Environmental and Affective dimensions to assess the effects of key variables – Semantic Intention, Usability, Web-Page Aesthetics, Subjective Norm and Trust- on users' satisfaction. Facebook was chosen as a focused social networking site, because of its popularity. A comprehensive survey instrument was applied to 203 Facebook users. Also, Structural Equation Modeling, particularly Partial Least Square, was conducted to analyze the proposed research model. As a result, proposed multidimensional research model predicts the factors influencing users' satisfaction towards social networking site use and relationships among these factors. The findings of this research will be valuable for literature by analyzing the influencing factors that have not been previously researched in the context of social networking satisfaction area.

Keywords: Social Networking Satisfaction, Structural Equation Modeling, Partial Least Squares

1. INTRODUCTION

Social networking concept has emerged out of the growing social needs such as establishing new social relations, finding friends with similar interests, sharing knowledge and content with other people [1]. Social networking with the expanding popularity have become among the most famous sites on the Web. The number of users might be shown as evidence for their popularity: LinkedIn has 41 million users, MySpace has 67 million users, Twitter has 98 million users and Facebook has 540 million users [2]. Although the number of users is high enough to easily accept popularity of social networking sites, the reasons behind their success are unclear [3]. According to researchers [4], users' satisfaction plays the major role for the success of web sites. In this regard, the literature needs research to identify the factors influencing users' satisfaction towards social networking sites.

In this study, Facebook is chosen as social web to identify users' satisfaction toward social networking. The major reason underlying this selection is that Facebook is the most popular social networking site in TURKEY with 22 million users; also TURKEY is among the first five countries with Facebook use [5]. Social Networking Satisfaction Model (SNSM) has been proposed to identify the factors affecting users' satisfaction towards social networking. Multidimensional perspective is considered when developing the research model; **Information Quality** – *Semantic Intention*, **System Quality** – *Usability* and *Web Page Aesthetics,* **Environmental Issue** - *Subjective Norm* and **Affective Issues –** *Trust.* The present study makes contribution *to the literature for several reasons.* Firstly, this study is the first attempt to examine users' satisfaction from the constructed multidimensional perspectives in the context of social networking. Secondly, effects of semantic information have never been examined before by researchers to evaluate users' satisfaction. Thirdly, usability and web-page aesthetic as a system

quality have not been employed to measure users' satisfaction in the context of social networking. In addition, there has been no other study examining the effects of other people on users' satisfaction towards social web. On the other hand, SNSM can greatly benefit system developers to better understand how users' satisfaction can be increased and how the social networking sites can be improved.

As a result, in order to increase users' satisfaction when using social networking sites, it is essential to understand the reasons behind users' rejection and identify the critical factors affecting their satisfaction. Therefore the aims of this study are as follows:

- Identifying the key factors affecting users' satisfaction of social networking site use.
- Developing a multidimensional model to reveal the main reasons behind the users' satisfaction of social networking site use.

The research question of this research is as follow;

- To what extend Semantic Intention, Usability, Web-Page Aesthetics, Subjective Norm and Trust affect Facebook users' satisfaction towards social web use.

2. LITERATURE REVIEW

2.1 Social Networking

Social networking sites enable individuals to "(1) construct a public or semi-public profile within a bounded system, (2) articulate a list of other users with whom they share a connection, and (3) view and traverse their list of connections and those made by others within the system" [6, pp.211]. Most of these sites share the basic features like allowing individuals to represent themselves with creation of profiles, meeting with others, finding new jobs, receiving or providing recommendation or much more [7].

According to the Facebook press room, Facebook is the most popular and fastest growing social networking site and it has been using since 2004 and today it has more than 500 million active users. This social networking site initially considered college and universities as a target group; however, it extended this group with everyone including high schools and other organizations [8]. As a study states that [9], researchers evaluated Facebook in different aspects like trust and privacy issues [6], [7], [8], [10], [11], motivations to use [9], [12], [13], [14], [15], usage effects [16], [17], [18] and identity and self-presentation of Facebook [19], [20], [21].

Although researchers were examined Facebook from different angles, little research investigated the factors that influence social networking adoption [12]. Users' adoption and decision to continue social networking use primarily depends on users' satisfaction [9].

2.2 Satisfaction of Information System

"…satisfaction in a given situation is the sum of one's feelings or attitudes towards a variety of factors affecting that situation" [22, p. 531]. Satisfaction is suggested as a success measure in the information systems area [23], [24]. The studies show that there is a strong relation between users' satisfaction and their intention to information system use [25], [26].

Satisfaction has been considered in different theoretical frameworks in the area of information systems [27]. However, any single study has not considered information quality, system quality, environmental issues and affective factors as a theoretical framework to evaluate users' satisfaction towards social networking. Success of information systems depends on quality of information and systems [24]. Also environmental factors and affective conditions are important determinants of users' satisfaction. In the scope of this study, a research model is proposed to assess users' satisfaction in the context of social networking as taking information quality, system quality, environmental and affective issues base.

3. RESEARCH METHOD

3.1 Research Model

In this study, a multidimensional model has been proposed to reveal the critical factors affecting users' satisfaction towards use of social networking. The model is named as Social Networking Satisfaction Model (SNSM) (shown in Figure 1). This simulation increase understandability and visuality between social networking satisfaction factor and the dimensions and the related variables examined under this dimensions. When developing SNSM, it is avoided to examine users' satisfaction from one perspective. Because it is recommended that a researcher should avoid using single linear methodology when assessing the attitudes of users towards a technology [28]. The critical measurement constructs are categorized by taking the IS Success Model [24], Social Cognitive Theory [29] and affective issue as base. *Information Quality, System Quality, Environmental Issue and Affective Issue* dimensions are considered when selecting the factors influencing users' satisfaction towards social networking. Initially, the effects of critical factors on satisfaction and relations between the constructs under the same dimension are hypothesized. During the structural model evaluation the meaningful relations among the constructs are tried to be revealed.

Information Quality: This concept is dealt with the relevance, timeliness, reliability, completeness, precision and accuracy of information produced by an information system [24]. User satisfaction is affected from "Informativeness" including relevance, comprehensiveness, recentness, accuracy and credibility, "Accessibility" consisting of convenience, timeliness and interpretability and "Adaptability" [30], [31]. Information quality is critical if it is expected to produce information for decision-making from an application of information technology [24]. For example, word-processor does not actually produce information therefore information quality is not a critical factor such a system. However, it is expected from the social networking sites to produce meaningful information for users. For this reason, Semantic Intention is explored under the Information Quality dimension to examine the importance of semantic information in social networking and how users' satisfaction is affected with the availability of semantic information.

Semantic Intention (SI): With the emergence of limitations in accessing the vast information on the Web [32], semantic web concept has become popular as a solution to overcome this dilemma. Semantic web helps to create an information infrastructure by accessing data from variety of sources to achieve a task [33]. Semantic web is defined as "... an extension of the current web in which information is given well-defined meaning, better enabling computers and people to work in cooperation" [34, p.3].

Support of social networks with semantic web brings innovations to attract the intention of users. The closer interaction between semantic web and social networking will be beneficial [32]. He stated that "the creation of mashups or combination of diverse sources of data and services could greatly benefit from the shared representations and protocols proposed by the Semantic Web community". Success of the Web depends on social adaption as much as technological issues [35]. With the help of semantic web, interoperability and social adoption have become critical factors for the global scale of the Web [32]. Therefore, with the help of Semantic Intention construct, it is possible to understand how users' satisfaction is affected if social networking sites that are supported with semantic information. The hypothesis related with Semantic Intention is as below;

H1: Support of social networking with semantic information positively influences users' satisfaction towards social networking.

System quality is concerned with the consistency of user interface, ease of use, quality of documentation and system flexibility [24]. A study shows that system quality indirectly influences users' perception towards system use [36]. According to [14], system quality is a critical point should be taken into consideration, because users are reluctant to use a web site if they experience problems on access, difficulty in navigation, constantly lived delays in response and

disconnection [37]. Usability and Web Page Aesthetics factors are examined under the System Quality dimension to reveal their effects on satisfaction of the user.

Usability (U): Usability is summarized as "the capability in human functional terms to be used easily and effectively by the specified range of users, given specified training and user support, to fulfill the specified range of tasks, within the specified range of environmental scenarios" [38, p.340]. With the help of Usability construct, it is tried to measure that how users' success rate meeting in specified aims, ease of use perception and perceived efficiency of system use affect their gratification towards system. The hypotheses related with Usability are as follows;

H2: Usability positively influences users' satisfaction towards social networking.

Web Page Aesthetics (WPA): Aesthetic concept was neglected until the first quarter of 21th century, because of the different focus points between computer industry and design criteria of human computer interaction [39]. However, as Norman's prescience [40], [41], the appropriation of modern design based on setting aesthetics ahead of usability. As it is seen aesthetics and usability present two orthogonal dimensions of human computer interaction; however there is a strong relation between usability and aesthetics as seen in the definition "measurement of usability defines the success or lack thereof in a GUI design" [42, p.60]. Also, recent research revealed that there is a strong relation between visual aesthetics of computer interfaces and users' satisfaction and pleasure [43]. Therefore, it is aimed to experimentally explore users' perception of aesthetics on their usability and satisfaction perception.

H3: Web Page Aesthetics positively influence users' satisfaction towards social networking.

H4: Web Page Aesthetics positively influence users' usability perception towards social networking.

Environmental Issues: Social Cognitive Theory (SCT) [29] is widely accepted and empirically validated model to evaluate an individual's behavior [44]. The theory examined the reciprocally determined factors that are environmental influences, cognitive and other personal factors and behavior. Individuals select the environment in which they exist and influenced by those environments. In addition behavior is affected by environmental factors or situational characteristics and cognitive and other personal factors [44]. Therefore, Subjective Norm is added into the research model under Environmental Issue dimension in order to analyze the effects of social pressure over users' satisfaction of social networking.

Subjective Norm (SN): Subjective Norm reflects the effects of social influence to perform or not to perform a behavior [45]. In other words, subjective norm concerns with normative beliefs about the expectation from other people [46]. Subjective norm is defined as "person's perception that most people who are important to him think he should or should not perform the behavior in question" [47, p.302]. With the help of Subjective Norm construct it is aimed to measure how the others' opinions affect users' gratification. Thus the following hypotheses are formulated;

H5: Subjective Norm positively influences users' satisfaction towards social networking.

Affective Issue: Affective concept states that systems have emotion and ethical dilemma, as well as a number of social and philosophical questions [48]. According to Picard [49], emotion plays a vital role between users and systems; because it underlines the process of perception, decision-making, creativity, empathic understanding, memory, as well as in social interaction [50], [51], [52], [53]. According to Lee and See [54] states that trust which is an effective response and it affects information selection, interpretation and intention to rely. In the scope of this study, the relation between trust and satisfaction is examined to reveal how users' affective basis effects their system interaction.

Trust (T): Trust is defined as "the extent to which one is willing to ascribe good intentions to and have confidence in the words and actions of other people" [55, p.39]. Trust is explained as a belief that affects our attitude towards the system use [56]. Activities performed in social networking sites like creating profile (name, surname, birth date, ethnicity, etc.), sharing documents (images, text, audio and video), manifesting the status information, connecting with friends, and etc. expose privacy concerns. In this point, it can be said that trust is an important factors for successful online interaction for sharing information and building new relationships [57]. In this sense, some questions come into prominence. What role does trust play in users' satisfaction from using the system and how does it play role when users intent to use social networking sites? In order to answer these questions, the following hypothesis is presented below;

H6: Trust perception positively influences users' gratification towards social networking.

In addition to the direct relation between the proposed critical factors and satisfaction, the relations among the influencing factors will be examined during the structural model evaluation.

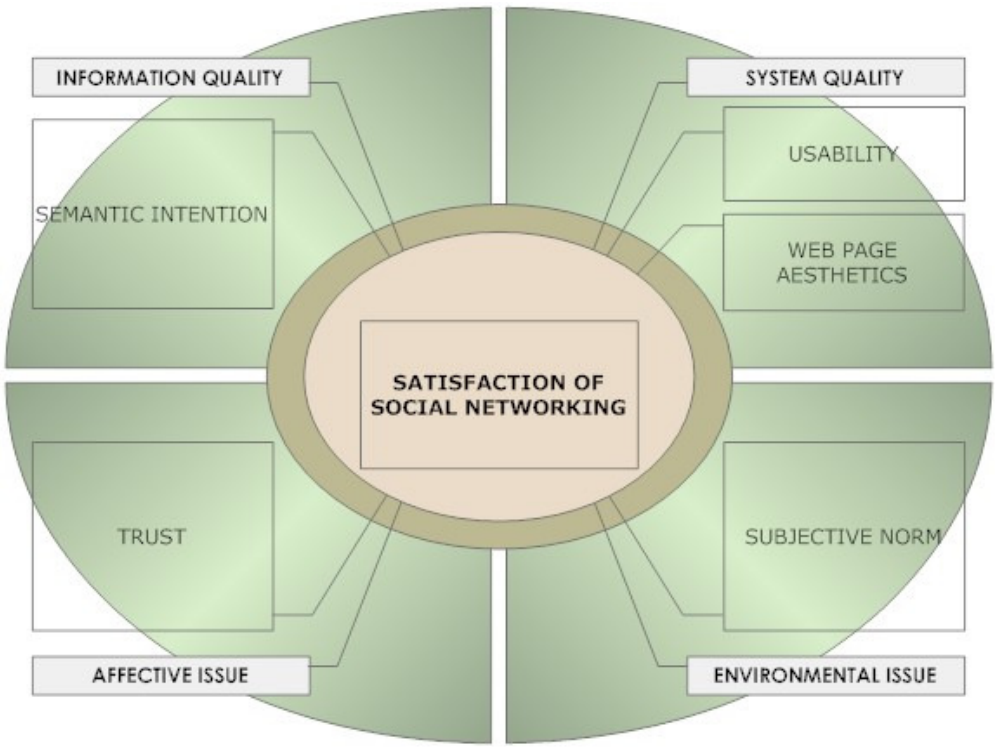

FIGURE 1: Social Networking Satisfaction Model (SNSM).

3.2 Instrument Development

After a detailed literature review, a comprehensive survey instrument was developed to empirically validate the research model. The survey instrument consisted of two parts. The first part included demographic questions and the second part included measurement items to assess Semantic Intention, Usability, Web-Page Aesthetics, Trust, Subjective Norm and Satisfaction factors. The references of the measurement items are as follows; Semantic Intention was self-developed, Usability was adopted from Nielsen's Ten Usability Heuristics [58], [59], Web-Page Aesthetics was adopted from Mullet & Sano [60], Trust was adopted from the researches Pan & Zinkhan [61] and Dwyer, Hiltz &Passerini [8], Satisfaction was adopted from Bhattacherjee [62],

[63]. Five point likert scale type was used for measurement items in which 1 stated strongly disagreement and 5 stated strongly agreement.

Content validity helps to assess whether the measurement items represent the constructs tried to measured, and cover full range of the constructs [64]. Before survey instrument applied in pilot study, content validity was checked with group including ten PhD students and expert judgments. According to their feedbacks some re-wording was done to make measurement items more understandable and comprehensive. After content validity was checked, reliability of the survey instrument assessed with pilot study, 47 measurement items were used in main research. The results of the pilot study are shown below.

3.3 Pilot Study Data Collection and Participants

The aim of the pilot sampling was to find people who use Facebook social networking site most frequently. In the pilot study, convenience sampling method was used. The sample size of the pilot study was 31 (52% female and 48% male). The participants were in 20 to 50 age range and the average age was 28.5. Most of the participants were graduate and undergraduate students. 65% percent of the participants reported more than once Facebook usage in a day. This result shows that the aimed target sample was reached. Also the actual survey instrument was prepared according to the results of reliability statistics (based on inter item consistency - Cronbach's Alpha was 0.890) and feedbacks of the participants.

3.4 Main Study Data Collection and Participants

The final survey instrument was distributed 500 Facebook users in Turkey. Snowball sampling technique was considered in data collection. A total of 269 responses were obtained in a month. The total response rate was 53.8%. The demographic profile of the participants was presented with frequency of Facebook use, gender, age, education level, occupation and used social networking sites. 67% of the participants were using Facebook more than once in a day. 49% of the participants were female and 51% of them were male. This demographic result shows that there was no big diversity in gender in terms of sample sizes. Participants were in 18 to 53 age range and the average age was 26.7. 81% of the participants were including graduate and undergraduate students, 15% of them had a PhD level and the remaining had high school graduate. 46% of the participants were working in public sectors and then private sector and students followed with 26% and 21% respectively. Also, 25%, 13%, 6%, 5% of the participants were using Twitter, LinkedIn, Xing and MySpace social networking sites respectively in addition to Facebook.

4. DATA ANALYSES AND RESULTS

4.1 Preliminary Analysis

In order to prepare the data set for further analyses, a set of preliminary data analyses that are missing value analysis, outlier detection and normality need to be conducted [65]. Firstly, missing values need to be handled by considering the missing value analyses. The cases should be removed from the data set, if the extent of missing values is greater upper than 50% [65]. Therefore the sample size dropped from 269 to 203 after the cases including missing values higher than 50%. If the missing data level is under 10% any imputation method can be used to solve the missing value problems [65]. In the data set, the missing value level was under 10% for both in all the cases and variables. Therefore, mean substitution was used for missing values. Secondly, outliers that are the cases with values well above or well below the majority of other cases should be detected and handled [66]. In order to detect problematic values as an outlier in the data set the comparison between mean and trimmed mean values were considered [67]. However there were not huge differences were detected between these values to consider outlier problem. Lastly, normality assumption is evaluated to determine the required type of structural equation modeling. If the data set is normally distributed covariance based structural equation will be performed and if the data set is not normally distributed component based structural equation modeling will be performed to determine the relation between constructs of the proposed research model [68]. Normality assumption was checked with skewness and kurtosis values and

Kolmogorov-Simirnow test [64] and it was assumed that the distribution of the data set was not normal.

4.2 Identifying the Factor Structure

The aim of exploratory factor analysis is to identify the factor structure of data set for a set of variables by determining how many factors exist [69]. A maximum likelihood was conducted on the 47 items with direct oblimin rotation because the factors were related with each other [64]. The Kaiser-Meyer-Olkin (KMO) measure equaled to .85 confirmed the sampling adequacy for the explanatory factor analysis [64]. Bartlett's test of sphericity x2 (1081) = 5866.45, $p < .001$, indicated that correlations between items were sufficiently large for maximum likelihood. Seven components had eigenvalues over Kaiser's criterion of 1 and combination explained 52.70% of the variance. According to Hair and his colleagues [65] factor loadings should be 0.3-0.4 to meet the minimal level for explanation of structure. Table 1 shows the factor loadings after rotation. The items had a factor loading value above 0.40 cluster on the same factor suggest that factor 1 represents Usability, factor 2 represents Aesthetics, factor 3 represents Semantic Intention, factor 4 represents Subjective Norm, factor 5 represents Trust, factor 6 represents Satisfaction. The problems with measurement items (written italic in Table 1) were as follows;

The eighth and ninth items of Usability did not cluster under the first factor as expected. The sixth item of Semantic Intention did not cluster under the Semantic Intention factor. The third, fourth and ninth items of Satisfaction did not strongly load under the same factor.

4.3 Reliability Assessment

Reliability analysis is required to measure the consistency of the survey instrument [64]. Reliability of the survey instrument can be evaluated with the inter-item consistency and that is assessed with Cronbach's Alpha [70]. Cronbach's Alpha value around 0.8 indicates a good reliability [64]. Total scale Cronbach's Alpha value was .920 and this result showed that the total scale had a high reliability. Also, reliability analyses were performed for all sub-scales and the results showed that (in Table 1) all had high reliabilities.

Construct / Item	**Factor Loading**	**Cronbach's Alpha Coefficient**	**Reliability Result**
Usability (U)			
U1: While navigating the Facebook, I know exactly where I am.	.543		
U2: After visiting a page on Facebook, I generally know where I go next.	.599		
U3: The words, phrases, and concepts presented on the Facebook are familiar. In other words, the Facebook speak my language.	.443		
U4 : I am able to easily recover when I went to go wrong page	.705		
U5: When encountering errors, Facebook provides good error messages.	.523	.784	Good
U6: Facebook does not require an extensive use of memory. I am able to recognize and do not need recall.	.530		
U7: Facebook is flexible to use.	.464		
U8: Any extraneous information is not given on Facebook	-		
U9: Facebook is an efficient social network to use.	-		
Web Page Aesthetic (WPA)			
WPA1: The same visual language elements are used throughout the Facebook.	.552		
WPA2: There are not any extraneous details either functionally or aesthetically on Facebook.	.442		
WPA3: Design elements are related with main purpose of the Facebook and the expected visitors.	.573	.826	Good
WPA4: Interactive elements are clearly distinguishable in both form and function.	.686		
WPA5: I am able to see clearly how individual parts of the	.443		

Facebook relate to each other. WPA6: Contrasting colors are used to make it easy for users to distinguish between different areas of focus on Facebook. WPA7: Similar elements are grouped together contextually on Facebook. WPA8: Elements are ordered in a hierarchy on Facebook. WPA9: Design elements are balanced in either a symmetrical or asymmetrical layout, throughout the Facebook.	.465 .600 .634 .642		
Semantic Intention (SI) SI1: I would be glad if Facebook had an option to give me current information about my occupation SI2: I would be glad if Facebook provides personalized advertisements through taking account of my previous online shopping history SI3: I would be glad if Facebook had an option to provide me newsbreaks from my daily followed newspapers. SI4: I would be glad if Facebook had provided special offer information from my favorite shopping websites. SI5: I would be glad if Facebook had brought me information from other social networking sites about my friends of whom I am searching about. *SI6: I would be glad if Facebook had an option to remove the photos in my profile tagged to friends who are deleted from my friend list.* SI7: I would be glad if Facebook informs me of social activities in the city I live (i.e cinema, theatre, concert)by considering my areas of interest. SI8: I would be glad if Facebook became a much more semantically (personalized) social network.	.632 .696 .732 .742 .502 - .556 .630	.832	Good
Subjective Norm (SN) SN1: My friends think that I should use Facebook SN2: My family think that I should use Facebook SN3: People important to me support my use of Facebook SN4: People who influence me think that I should use Facebook SN5: People whose opinions I value prefer that I should use Facebook	-.620 -.459 -.771 -.979 -.886	.885	Good
Trust (T) T1: Facebook is a trustworthy social network T2: I can count on Facebook to protect my privacy T3: Facebook can be relied on to keep its promises T4: I can count on Facebook to protect customers' personal information from unauthorized use T5: I feel that the privacy of my personal information is protected by Facebook T6: I trust that Facebook will not use my personal information for any other purposes	-.607 -.897 -.739 -.889 -.938 -.947	.922	Good
Satisfaction (STS) STS1: I am satisfied with the performance of Facebook. STS2: I am pleased with the experience of using Facebook. *STS3: My decision to use Facebook was a wise one* *STS4: Overall, the information I get from Facebook is very satisfying.* STS5: All things considered, I am very satisfied with Facebook as a social network. STS 6: Using Facebook is pleasant. STS7: I have fun using Facebook. STS8: I find using Facebook to be enjoyable. *STS9: I find using Facebook to be interesting.* STS10 - STS6: Overall, my interaction with Facebook is very satisfying	.482 .833 - - .467 .865 .729 .868 - .476	.900	Good

TABLE 1: Factor Structure and Reliability of Data Set.

4.4 Assessment of the Proposed Research Model

The validity of the proposed research model was evaluated with component based structural equation modeling because of the non-normal data structure. SMART PLS was used to assess structural model of proposed research model. Before analyzing the structural validity of the model, the measurement model was assessed.

4.4.1 Assessment of the measurement model

Confirmatory factor analysis was used to evaluate measurement model. This analysis helped to examine which variable was load on which factor as well as the correlation among the factors. Measurement model evaluation was performed with convergent validity and discriminant validity that are the most important components of construct validity.

Convergent validity is defined as the degree to which two variables share variance due to a given concept and correlation [71]. Convergent validity is evaluated with Factor Loadings (FL), Composite Reliability (CR), and Avarage Variance Extracted (AVE) [65].

Factor loadings validate the correlation between each measurement items and their constructs. Factor loadings ideally should be 0.7 or higher, however 0.5 or higher is acceptable. As shown in Table 2 factor loadings of all measurement items were change between 0.420 and 0.927. U1, U2, WPA2, WPA3, SI5 measurement items had factor loadings below 0.6 and they were removed from the data set.

Composite reliability represents internal consistency that means all measurement items represent its latent constructs and this value should be minimum 0.7. In this study, composite reliability values were between 0.845 and 0.956.

Average variance extracted (AVE) value should be calculated for each latent constructs of the model. AVE value should be over 0.5 for adequate convergent validity. The AVE values of the factors in the proposed research model ranged between 0.501 and 0.788.

These results show that the measurement model provided an adequate convergent validity.

Item	Factor Loadings	Composite Reliability (CR)	Average Variance Extracted (AVE)
U1	0.494	0.845	52%
U2	0.590		
U3	0.716		
U4	0.767		
U5	0.672		
U6	0.692		
U7	0.691		
WPA1	0.665	0.868	50%
WPA2	0.420		
WPA3	0.603		
WPA4	0.744		
WPA5	0.661		
WPA6	0.640		
WPA7	0.761		
WPA8	0.680		
WPA9	0.649		
SI1	0.663	0.874	54%
SI2	0.750		
SI3	0.815		
SI4	0.785		
SI5	0.549		
SI7	0.723		
SI8	0.624		

SN1 SN2 SN3 SN4 SN5	0.701 0.720 0.892 0.912 0.916	0.918	69%
T1 T2 T3 T4 T5 T6	0.781 0.922 0.855 0.913 0.919 0.927	0.956	78%
STS1 STS2 STS5 STS6 STS7 STS8 STS10	0.691 0.877 0.773 0.814 0.868 0.867 0.802	0.845	66%

TABLE 2: Convergent Validity for Measurement Model.

Discriminant Validity was proved to show each constructs of the model was adequately different from each other. In order to show differences the square root of the average variance calculated for each constructs should be greater than the correlation between a given constructs and all other constructs [72]. As shown in Table 3, the diagonal shows the square root of average variance calculated for each constructs and these values are greater than the other correlation values. Therefore, discriminant validity was also met the construct validity.

Construct	WPA	SI	SN	STS	T	U
WPA	**0.697**					
SI	0.095	**0.735**				
SN	0.196	0.237	**0.833**			
STS	0.360	0.252	0.528	**0.815**		
T	0.227	0.219	0.481	0.433	**0.887**	
U	0.506	0.116	0.116	0.421	0.119	**0.722**

TABLE 3: Discriminant Validity for Measurement Model.

4.4.2 Assessment of Structural model

Statistical significance of hypotheses was assessed by considering the path coefficient values (standardized betas). A bootstrapping procedure was applied on the data set including 203 samples to evaluate significance level of the relations between constructs. The result of analyses showed that the SNSM accounted for 45% variances of Satisfaction. Figure 2 shows the estimated path coefficients between constructs of the structural model.

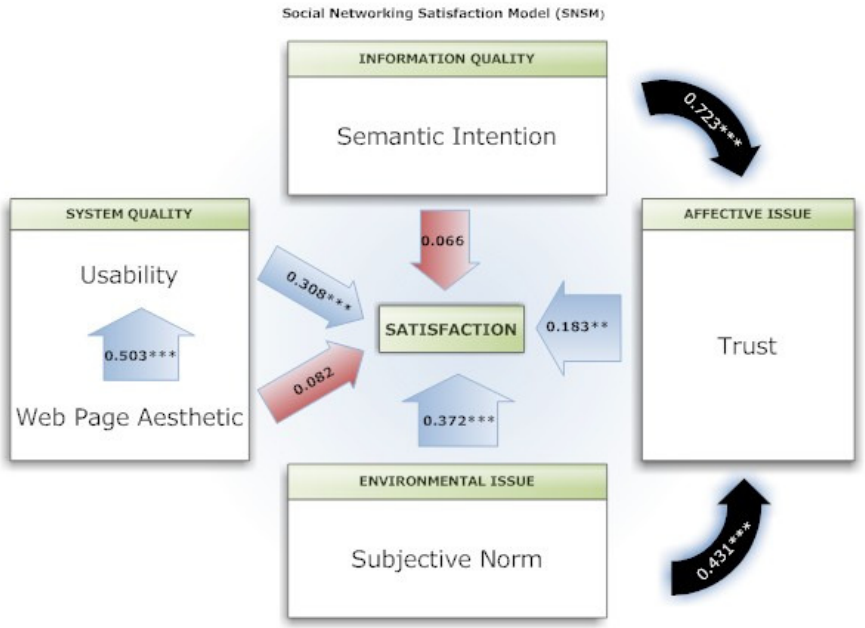

FIGURE 2: Structural Model of SNSM.

* Blue arrows show significant hypotheses, black arrows show additional relations, red arrows show rejected hypotheses

The structural evaluation of SNSM showed that Semantic Intention and Web Page Aesthetics factors did not significantly affect Satisfaction, so H1 and H3 were rejected. On the contrary, three strong positive relations were found between Satisfaction and Usability, Trust and Subjective Norm at the $p<0.001$ level. Therefore H2, H5 and H6 were accepted. Also, Web Page Aesthetics positively affected Usability at $p<0.001$ level, so H4 was accepted. In addition to initially hypothesized relations, additional significant relations were found during the structural model evaluation. For example, there was a strong positive relation between Semantic Intention and Trust at $p<0.001$ level. Moreover, SN had a positive and significant relations with Trust at $p<0.001$ level. The significance value and standardized path coefficients of the SNSM are shown in Table 4.

Relationships	Hi	T-Values	B	Decision
SI->STS	H1	1.235	0.066	Rejected
U->STS	H2	5.407	0.308***	Accepted
WPA->STS	H3	1.240	0.082	Rejected
WPA->U	H4	9.641	0.503***	Accepted
T->STS	H5	3.132	0.183**	Accepted
SN->STS	H6	6.286	0.372***	Accepted
SI->T	Additional Relation	6.723	0.202***	Accepted
SN->T	Additional Relation	6.723	0.431***	Accepted

**p < 0.01; ***p<0.001

TABLE 4: Summary of Hypotheses Tests.

5. DISCUSSION

In this empirical research, a number of relations were examined to reveal the factors affecting users' satisfaction towards social networking. SNSM showed the relations among the factors examined under the Information Quality, System Quality, Environmental and Affective Issues dimensions.

The first dimension examined the effects of information quality on users' satisfaction towards social networking use. With this dimension it was expected to measure effects of information content on users' satisfaction. Diversification and usefulness of the information was examined in

the scope of semantic concept. In the proposed research model the relation between Semantic Intention and Satisfaction was analyzed. The main aim of the semantic information in social networking was to give beneficial information from multiple resources. However, statistical result showed that there was not a significant relation between Semantic Intention and Satisfaction. This result can be interpreted as semantic information gathered from different source of the web does not as effective as expected on users' satisfaction of social networking use. The reason of this insignificant relation may be reliability of data obtained from the web [32]. Also there are most prominent positive and negative aspects of social networking sites [3]. As positive experiences, enjoyment, fun/playfulness, excitement, self-expression and curiosity come into prominence. This means that social networking navigation is strongly related with his or her enjoyment. Also, as a curiosity consideration of social networking, users only wonder information about their friends [3]. Also, this insignificant relation is parallel with the study of [14]. The researcher found insignificant relation between information quality and users satisfaction. He interpreted this insignificant relation as the users do not care good information, the social web mainly focus on interaction and communication among the members. In addition, vast amount of data available in web platform like, text, audio-video, image and etc. Trust and privacy issues come into prominence with the facing this huge amount of data. While the availability of this vast amount of data helps to complement social networking with semantic information, there is privacy and trust consideration appears. With the help of SNSM, the relation between Semantic Intention and Trust was examined. This relation was an additional relation was not previously hypothesized. The statistical result showed that semantic information provided by social networking significantly related with trust. This means that people want to trust the information gathered from different sources in social web.

The second dimension tried to identify the effects of system quality on users' satisfaction towards semantic web use. Under the system quality dimension the effects of usability and web-page aesthetics on users' satisfaction were examined. The statistical analysis showed that Usability had a strong effect on users' satisfaction. This means that users found the site effective and easy to learn to accomplish tasks, adapted easily to variation in tasks and satisfied with the system use [38]. On the contrary, there was a non-significant relation was found between web page aesthetics and satisfaction of users. This result shows that users did not care the aesthetics of the social networking site. Dramatically, the users' expectation of interface quality towards Facebook fairly poor and satisfaction of users can be achieved without good interface quality because their main expectations are interaction and communication [14]. According to the study of Butler [42], "aesthetics is referred as non-quantifiable, subjective, and affect based experience of system use; however usability is commonly measured by relatively objective means and sets efficiency as its foremost criterion" [39]. One may well assert that the different effects that usability and Web-Page aesthetics have on satisfaction may be explained by the fact that aesthetic is a subjective concept, while usability is objective. In addition there was a significant relation was found between web page aesthetics and usability factors. The finding refers that the users who interacted with an aesthetic social networking site perceived the system as more usable than users who interacted with less aesthetic social networking site. This result is parallel with the other studies existing in the literature [39], [73], [74]. The previous researchers considered different applications while assessing the relation between aesthetics and usability. For example, the effects of aesthetics on usefulness by considering ATM machines were examined [39], [73] and DeAngeli, Sutcliffe & Hartmann used web-sites to evaluate the relation between these two factors and the researcher found a strong relation between aesthetics and usability perception of users [75].

The third dimension identified the effect of environmental issue examining effects of subjective norm on users' satisfaction. There was a positive and significant relation was observed between subjective norm and satisfaction of users. This significant relation refers that users' gratification increase when the other people around the users give concrete support towards their use of social networking site. In other words, people pleased with other people's encouragement of using information systems. According to the information obtained so far, several studies prove the effects of subjective norm on users' continuance intention and system use in information system

contexts [46], [76], [77]; however any study has not examined the relation between subjective norm and satisfaction. Moreover, a significant relation was found between subjective norm and trust during the analyses. This relation shows that the trust perception of users strongly affected from opinion of other peoples. The strong relation between subjective norm and trust is supported with the statement of Wu and Chen [78]. They stress that "whatever types of trust are with direct and indirect influences on subjective norm, they are all the important antecedents of subjective norm in on-line service".

The last dimension affective issue identified the effects of trust over satisfaction. The structural model showed that there was a positive and direct relation between trust and satisfaction. This relation emphasize that the users who feel more confidence towards social networking site have more pleasure with social web interaction.

6. CONCLUSION

This study was conducted to propose social networking satisfaction model titled as SNSM from the perspective of Facebook users. The concept of satisfaction is not adequate to consider the human factors for usability [79]. Therefore the model composed of four dimensions – Information Quality, System Quality, Environmental and Affective Issues. The proposed multidimensional model was evaluated structurally to examine the effects of following variables of corresponding dimensions on satisfaction of social networking site users; (1) Information Quality – Semantic Intention, (2) System Quality – Usability and Web-Page Aesthetics, (3) Environmental Issue – Subjective Norm, (4) Affective Issue – Trust. The structural model explained a significant amount of variance of satisfaction towards social networking site use. In addition, the findings contribute the literature by revealing the factors that influence users' satisfaction of social web. The findings guide the system developers for continuous improvement of the system.

When we look at the literature, it is seen that the number of studies examining the user satisfaction on Facebook usage is not satisfactory. In the literature, the researchers examined the relation between Facebook use and students' life satisfaction, social trust, civic engagement, and political participation [80]. However, any single study in the literature has not considered effects of variables examined in the scope of this research on users' satisfaction.

As a guide for further researchers, this multidimensional model can be extended with external dimensions in order to provide different perspectives. There may be different influencing factors affecting users' satisfaction like privacy, perceived ease of use, perceived usefulness, organizational support and so on. Web-applications are different from traditional software systems and maintenance process becomes a cumbersome process [81]. How the maintenance of the systems will affect the users' satisfaction can also be examined by the future researchers. Another future research would be performed to validate SNSM on different social networking sites. In addition, the effects of Semantic Intention and Web-Page Aesthetics should be examined in new research in the scope of social networking site satisfaction.

7. REFERENCES

[1] A. Mislove, M. Marcon, K. P. Gummadi, P. Druschel and B. Bhattacharjee. "Measurement and analysis of online social networks." *7th ACM SIGCOMM conference on Internet measurement*. pp. 29-42. 2007. Newyork.

[2] P. Kiser. "Social media 3Q update: Who uses Facebook, Twitter, Linkedln, & Myspace?" http://socialmediatoday.com/index.php?q=paulkiser/199133/social-media-3q-update-who-uses-facebook-twitter-linkedln-myspace, 2010 [Jan. 18, 2011]

[3] J. Hart, C. Ridl ey, F. Taher, C. Sas and A. Dix. "Exploring the Facebook experience: A new approach to usability." *In NordiCHI '08: Proceedings of the 5th Nordic conference on Human-computer interaction.* 2008. pp. 471-474. New York, NY, USA. ACM.

[4] S. Muylle, R. Moenaert, M. Despontin. “The conceptualization and empirical validation of web site user satisfaction.” *Information & Management*. 41(5), 543-560, 2003.

[5] L. Bianchi. “Infographic: Facebook 500 million members.” http://webcache.googleusercontent.com/search?q=cache:WFnDcGv4yR8J:www.viralblog.com/social-media/infographic-facebook-500-million-members/, 2010 [May. 18, 2010]

[6] D.M. Body and N.B. Ellison. “Social network sites: definition, history, and scholarship.” *Journal of Computer-Mediated Communication*. 13(1), 210-230, 2007.

[7] R. Gross and A. Acquisti. (2005). “Information revelation and privacy in online social networks.” *ACM Workshop on Privacy in the Electronic Society*; Alexandria, VA. http://www.heinz.cmu.edu/~acquisti/papers/privacy-Facebook-gross-acquisti.pdf., [Jan. 2011]

[8] C. Dwyer, S. R. Hiltz and K. Passerini. (2007). “Trust and privacy concern within social networking sites: A comparison of Facebook and Myspace.” *30th Americas Conference on Information Systems.* Keystone, Colorado. http://csis.pace.edu/~dwyer/research/DwyerAMCIS2007.pdf, [Dec. 2010]

[9] N. Shi, M. K. O. Lee, C. Cheung and H. Chen. “The continuance of online social networks: How to keep people using Facebook*?” The 43rd Hawaii International Conference on System Sciences* pp. 1-10. 2010. Koloa, Kauai, Hawaii

[10] S. Barnes. “A privacy paradox: Social networking in the United States.” *First Monday,* 11(9). http://www.firstmonday.org/issues/issue11_9/barnes/index.html, 2006 [Jan. 8, 2011]

[11] A. Acquisti and R. Gross. “Imagined communities: awareness, information sharing, and privacy on the Facebook.” *Privacy Enhancing Technologies*. pp. 36-58. 2006. Lecture Notes in Computer Science 4258.

[12] H. Gangadharbatla. “Facebook me: collective self-esteem, need to belong, and internet self-efficacy as predictors of the igeneration's attitudes toward social networking sites.” *Journal of Interactive Advertising*, 8(2), 5-15, 2008.

[13] C.M.K. Cheung, P. Y. Chiu and M.K.O. Lee. “Online social networks: Why do “we” use Facebook?” *Computers in Human Behavior*, 19, 67-74, 2008.

[14] W. P. Yan. (2008). “Factors affecting individual’s intention to continue participating in a social networkings site”. Honours Project , Hong Kong Baptist University, 2008

[15] O. Kwon and Y. Wen. “An empirical study of the factors affecting social network service use.” *Computers in Human Behavior*, 26(2), 254-263, 2010.

[16] S. P. Andon. “Evaluating computer-mediated communication on the university campus: The impact of Facebook.com on the development of romantic relationships, in college of communication.” Master’s Thesis, The Florida State University, Florida, USA, 2007.

[17] N. B. Ellison, C. Steinfield and C. Lampe. “Spatially bounded online social networks and social capital: The role of Facebook.” Annual *Conference of the International Communication Association.* Dresden, Germany, 2006.

[18] N. B. Ellison, C. Steinfield and C. Lampe. “The Benefits of Facebook "Friends:" Social Capital and College Students’ Use of Online Social Network Sites.” *Journal of Computer-Mediated Communication*, 12(4), 1143-1168, 2007.

[19] J.M. DiMicco and D. R. Millen. (2007). “Identity management: Multiple presentations of self in Facebook.” *International Conference on Supporting Group Work.* Sanibel Island, Florida, USA. [Online]. Avaliable: http://portal.acm.org/citation.cfm?id=1316682 [Jan. 2011]

[20] S. D. Gosling, S. Gaddis and S. Vazire. (2007). “Personality impressions based on Facebook profiles.” *International Conference on Weblogs and Social Media*, Boulder, Colorado, USA. [Online]. Avaliable: http://www.icwsm.org/papers/3--Gosling-Gaddis-Vazire.pdf [Dec. 2010]

[21] S. Zhao, S. Grasmuck and J. Martin. “Identity construction on Facebook: Digital empowerment in anchored relationships.” *Computers in Human Behavior,* 24(5), 1816-1836, 2008.

[22] J. E. Bailey and S. W. Pearson. “Development of a tool for measuring and analyzing computer user satisfaction.” *Management Science*, 29(5), 530–545, 1983.

[23] P. Eindor and E. Segev. “Organizational context and the success of management information systems.” *Management Science*, 24(10), 1064-1077, 1978.

[24] W.H. DeLone and E.R. McLean. “Information systems success: The quest for the dependent variable.” *Information Systems Research*, 3(1), 60-95, 1992.

[25] C. S. Lin, S. Wu and R. J. Tsai. “Integrating perceived playfulness into expectation-confirmation model for web portal context.” *Information & Management,* 42(5), 683-693, 2005.

[26] C.M.K. Cheung and M. K. O. Lee. “Understanding the sustainability of a virtual community: Model development and empirical test.” *Journal of Information Science*, 35(3), 279-298, 2009.

[27] C.M.K. Cheung, M. K. O. Lee. “The structure of web-based information systems satisfaction: Testing of competing models.” *Journal of the American Society for Information Science and Technology*, 59(10), 1617-1630, 2008.

[28] S. S. Liaw, H. M. Huang, and G. D. Chen. “Surveying instructor and learner attitudes toward e-Learning.” *Computers and Educatio*n, 49(4), 1066-1080, 2007.

[29] A. Bandura. “Social Learning Theory”. New York, US: General Learning Press, 1977.

[30] J. Iivari. “A planning theory perspective on information system implementation.” In: Gallegos, L. Welke, R., Wetherbe, J. (eds.) *Sixth International Conference on Information Systems,* 1985, pp. 196-211, Indianapolis, Indiana.

[31] E. Koskela. “The PIOCO model for information systems design.” *MIS Quarterly*, 11(3), 401-419, 1987.

[32] P. Mika. “Social Networks and the Semantic Web.” Doctoral Dissertation, Virije University, 2006.

[33] B. McBride. “Four Steps Towards the Widespread Adoption of a Semantic Web.” *ISWC 2002*, pp. 419–422, Sardinia, Italy, LNCS 2342.

[34] T. Berners-Lee, J. Hendler and O. Lassila. (2001). "The semantic web." *Scientific American*. [Online]. Avaliable: http://www.ipgems.com/present/swui_chi_20070502.pdf [Jan. 9, 2011]

[35] T. Berners-Lee, M. Fischetti and M. L. Dertouzos. (1999). "Weaving the Web: The original design and ultimate destiny of the World Wide Web by its inventor." *1rd ed*. Harper San Francisco.

[36] P. B. Seddon. "A respectation and extension of the DeLone and McLean model of IS success." *Information Systems Research* 8(3), 240-253, 1997.

[37] W.S. Yoo, K.S. Suh and M.B. Lee. "Exploring the factors enhancing member participation in online communities." *Journal of Global Information Management,* 10(3), 55-71, 2002.

[38] B. Shackel. "Usability—context, framework, design and evaluation." *Interacting with Computers,* 21(5-6), 339-346, 1991.

[39] N. Tractinsky, A. S. Katz and D. Ikar. "What is beautiful is usable." *Interacting with Computers*, 13(2), 127-145, 2000.

[40] D.A. Norman. "The psychology of everyday things." Basic Books, New York, 1988.

[41] D.A. Norman. "Turn signals are the facial expressions of automobiles." Addison-Wesley, Reading, MA, 1992.

[42] K.A. Butler. "Usability engineering turns 10." *Interactions*. 3 (1), 59-75, 1996.

[43] T. Lavie and N. Tractinsky. "Assessing dimensions of perceived visual aesthetics of web sites." *Int. J. Human-Computer Studies*. 60(3), 269–298, 2004.

[44] D. R. Compeau and C. A. Higgins. "Computer self-efficacy: Development of a measure and initial test." *MIS Quarterly*, 19(2), 189-211, 1995.

[45] I. Ajzen. "The theory of planned behavior." *Organizational Behavior and Human Decision Processes*, 50(2), 179–211, 1991.

[46] M. Lee. "Explaining and predicting users' continuance intention toward e-learning: An extension of the expectation–confirmation model." *Computers & Education*, 54(2), 506-516, 2009.

[47] M. Fishbein and I. Ajzen. "Belief, attitude, intention, and behavior: An introduction to theory and research." Reading, MA: Addison-Wesley, 1975.

[48] R. W. Picard. "Affective computing." Cambridge, MA: MIT Press, 1997.

[49] R. W. Picard. "Affective computing: challenges." *Int. J. Human-Computer Studies*, 59(1-2), 55-64, 2003.

[50] A. Damasio. "Descartes' error: Emotion, reason, and the human brain." New York, NY: Gosset/Putnam Press, 1994.

[51] A.M. Isen. "Positive affect and decision making." In: Lewis, M., Haviland, J. (Eds.), Handbook of Emotions. Guilford, New York, 2000.

[52] J.E. LeDoux. "The emotional brain." Simon & Schuster, New York, 1996.

[53] P. Salovey, J.D. Mayer. “Emotional intelligence.” *Imagination, Cognition and Personality,* 9 (3), 185–211, 1990

[54] J. D. Lee and K. A. See. “Trust in Automation: Designing for appropriate reliance.” *Human Factors and Ergonomics Society*, 46(1), 50-80, 2004.

[55] J. Cook and T. Wall. “New work attitude measures of trust, organizational commitment, and personal need nonfulfillment.” *Journal of Occupational Psychology*, 53, 39-52, 1980.

[56] A. Kini and J. Choobineh (1998). “Trust in electronic commerce: Definition and theoretical considerations.” In Blanning, R. and King, D., editors, *Proceedings of the 31 Annual Hawaii Conference on System Sciences*, volume IV. IEEE Computer Society

[57] N.W. Coppola, S.R. Hiltz and N.G. Rotter. “Building trust in virtual teams.” *IEEE Transactions on Professional Communication*, 47(2), 95-105, 2004.

[58] J. Nielsen. “Guerrilla HCI: Using discount usability engineering to penetrate the intimidation barrier.” In R. Bias & D. Mayhew (Eds.), Cost-Justifying usability. San Diego, California: Academic Press, 1994a.

[59] J. Nielsen. “Heuristic evaluation.” In J. Nielsen & R. Mack (Eds.), Usability inspection methods . New York, New York: John Wiley & Sons, 1994b.

[60] K. Mullet, D. Sano. “Designing visual interfaces.” Prentice Hall, 1995.

[61] Y. Pan and G. M. Zinkhan. “Exploring the impact of online privacy disclosures on consumer trust.” *Journal of Retailing*, 82(4), 331–338, 2006.

[62] A. Bhattacherjee. “An empirical analysis of the antecedents of electronic commerce service continuance.” *Decision Support Systems*, 32(2), 201–214, 2001a.

[63] A. Bhattacherjee. “Understanding information systems continuance. An expectation–confirmation model*.” MIS Quarterly*, 25(3), 351–370, 2001b.

[64] A. Field. “Discovering Statistics Using SPSS.” S. Sage, London, 2009.

[65] J. F. Hair, W. C. Black, B. J. Babin, R. E. Anderson and R. L. Tatham. “Multivariate data analysis.” (6rd. ed). Pearson Prentice Hall, 2006.

[66] J. Pallant. “SPSS survival manual: A step by step guide to data analysis using SPSS for windows.” Version 10. Open University Press, 2001.

[67] S. Walfish. “A Review of Statistical outliers Methods.” *Pharmaceutical Technology*, 30(11), 82-88, 2006.

[68] I. Qureshi and D. Compeau. “Assessing between-group differences in information systems research: A comparison of covariance- and component-based SEM.” *MIS Quarterly: Management Information*, 33(1), 197-214, 2009.

[69] J. Stevens. “Applied multivariate statistics for the social sciences.” 4rd ed. Mahwah, NJ: Lawrence Erlbaum, 2002.

[70] G.A. Morgan, N.L. Leech, G.W. Gloeckner and K.C. Barrett. “SPSS for introductory statistics use and interpretation.” 2rd. ed. Lawrence Erlbaum associates, publishers, 2004.

[71] C. S. Reichardt and S. C. Coleman. “The criteria for convergent and discriminant validity in a multitrait-multimethod Matrix.” *Mirlfivariafe Behavioral*, 30, 513-538, 1995.

[72] C. Fornell and D. Larcker. “Evaluating structural equation models with unobservable variable and measurement error.” *Journal of marketing research*, 18(1), 39-50, 1981.

[73] M. Kurosu and K. Kashimura. “Apparent usability vs. inherent usability: experimental analysis on the determinants of the apparent usability.” *In CHI '95: Conference companion on Human factors in computing systems.* 1995, pp. 292-293, New York, NY, USA. ACM.

[74] N. Tractinsky. “Aesthetic and apparent usability: Empirically assessing Cultural and methodological Issues.” *CHI 97 the SIGCHI conference on Human factors in computing systems,* 1997*, pp.*115-122. Newyork, NY, USA.

[75] A. DeAngeli, A. Sutcliffe and J. Hartmann. “Interaction, usability and aesthetics: What influences users’ preferences?” *6th ACM conference on Designing Interactive systems*, 2006, pp.271-280. New York, N.Y.: University Park, PA.

[76] W. T. Wang and C. C. Wang. “An emprical study of instructor adoption of web-based learning systems.” *Computers & Education*, 53(3), 761-774, 2009.

[77] S. Y. Park. “An analysis of the technology acceptance model in understanding university.” *Educational Technology & Society*,12(3), 150-162, 2009.

[78] I. L. Wu and J. L. Chen. “An extension of trust and TAM model with TPB in the initial adoption of on-line tax: An empirical study.” *International Journal of Human Computer Studies*, 62 (6), 784-808, 2005.

[79] J. Sung, Y. Yun. “Toward a More Robust Usability concept with Perceived Enjoyment in the context of mobile multimedia service.” *International Journal of Human Computer Interaction.* 1(3), 12-32, 2010.

[80] V. Sebastián, N. Park and K. F. Kee. "Is There Social Capital in a Social Network Site?: Facebook Use and College Students' Life Satisfaction, Trust, and Participation1." *Journal of Computer-Mediated Communication*. 14(4), 875-901, 2009.

[81] P. Tripathi, M. Pandey and D. Bharti. “Cognitive Approach Towards the Maintenance of Web-sites Through Quality Evaluation in Operative Phase.” *International Journal of Human Computer Interaction*. 2(2), 31-37, 2011.

User Centered Design Patterns and Related Issues – A Review

Sara Imtiaz *saraimtiazcheema@gmail.com*
Department of Computer Software Engineering
National University of Sciences and Technology,
Islamabad, Pakistan.

Arif Raza *arif_raza@mcs.edu.pk*
Department of Computer Software Engineering
National University of Sciences and Technology,
Islamabad, Pakistan.

Abstract

A design pattern describes possible good solutions to common problems within certain context. This is done by describing the invariant qualities of all those solutions where good patterns improve with time and widespread use. In this research paper some existing user centered design patterns and their issues are discussed. We have studied many user centered design patterns; however most of them do not provide diagrammatic solutions which can be implementable. It is observed that there is a need of a design pattern which can address issues specifically related to Open Source Software (OSS) users.

Keywords: Open Source Software (OSS), Human Computer Interaction (HCI), Users, User Centered Design, Design Patterns.

1. INTRODUCTION

A pattern describes possible good solutions to a common design problem within a certain context, by describing the invariant qualities of solutions [1]. A design pattern language is a collection of design patterns which are structured so that patterns cross-reference other patterns in the set [2]. Christopher Alexander posits that good patterns improve with time and widespread use [1]. A good design pattern consists of several different components. There are the content elements, which encapsulate the description of the problem, context and solution. Relationships between patterns are essential part to be benefitted from the full reuse potential promised [2]. User involvement for the user satisfaction in designing interface is a necessary part. Like all languages, a pattern language has vocabulary, syntax, and grammar. The difference is that the language is applied to some complex activity other than communication. [3]

In OSS projects usability is one of major issues besides software functionality to make it a good usable product [4]. There are architectures and patterns including HCI patterns, which are used in software development.

Following sections consist of a detailed review of HCI oriented design patterns, their applications and relevant issues. A general discussion is presented in Section 2 about HCI oriented design patterns. In Section 3, issues related to some user centred design patterns are discussed.

2. HCI ORIENTED DESIGN PATTERNS – IN GENERAL

According to [5], diagrams included in software design patterns help in understanding the patterns. In HCI oriented design patterns, the focus of diagrams is on user point of view. However the pattern language is not specific to the single user interface, style or guidelines. The authors state that designers should communicate with users to understand their requirements. They propose MoLIC (Modeling Language for Interaction as Conversation) for representing

interaction. MoLIC is the language which is used for interaction between users and the designer. MoLIC takes views of users in order to assist designer to incorporate them in the design.

Wilde et al. [6] state that there are many interesting questions related to HCI design patterns of pervasive computing systems. In this new area it is difficult to create a HCI design pattern which is also universally accepted. The important question is how to test the interaction patterns to validate time and expenses; and how to assess the process of using them. The HCI patterns are very helpful because they motivate designers of pervasive systems to improve the quality of interaction among the systems. As a result the authors consider that pattern collection is in need and will be more widespread in the future. So the developers of pervasive systems will pass on HCI pattern collection and continue effort to classify the up comings interaction patterns. [6]

A survey [7] says that HCI is an important factor in software development. Using the concept of work reusability, two researchers, Ward Cunningham and Kent Beck use the work of Christopher Alexander based on the concepts of object-oriented-programming (OOP) and user interface (UI) implementation. In their work they present five patterns for designing window-based user interfaces in Smalltalk. These design patterns become beneficial to the software engineering community when a book “Design Patterns: Elements of Reusable Object-Oriented Software” by Gamma et al. was published. This book becomes one of the bestselling books in software engineering. These researchers with their innovative thoughts have established a series of design patterns and pattern languages in software engineering. A first design pattern of user-centered interface design is published by Coram and its group fellows. Basic intention was to provide high level patterns with which user interface designers could build graphical user interfaces which are pleasurable and productive to use. First Computer Human Interaction (CHI) workshop on pattern languages in user interface design was organized in 1997. The participants focused on the usage of pattern languages in HCI/user interfaces to.

Kruschit and Hitz [7] describe that the relationships between content elements and sharing design knowledge are the most important concepts in reusing the design patterns. Mostly HCI patterns describe a brief relationship of connections to other patterns. A proper relationship is needed to be described among patterns so that design problems are reduced and the patterns communicate with other patterns successfully. The tools used in the patterns are the major assets because they describe the reuse of HCI design patterns. There are many tools which are used such as *“online libraries/catalogs, pattern management tools, and pattern-based UI design tools.”* The interface tools are helpful for user interface designers to understand user requirements instead of gathering it again and again. Design patterns exist at a higher level of concept than UI toolkits and are not dependent on source code for specific implementation.

At a Human Computer Interaction workshop in 2003 the standardization approach was presented [7]. PLML (pronounced “pell mell”) is the standard pattern form for HCI patterns proposed by Fincher et al. Its purpose is to provide standard patterns for authors and users to use the patterns. Later on, an improved version of (PLML v 1.2.) is developed by Deng et al. to overcome many problems. Moreover the patterns are written in such a way that they give users a possibility to change the specific widgets according to their choice of solution. [7]

3. USER CENTERED DESIGN PATTERNS AND THEIR ISSUSES

3.1 Alexander’s Notion of Pattern

Alexander et al. originates their work from an observation *"Most of the wonderful places of the world were not made by architects but by the people [8-a]".* Alexander uses patterns as a guide to construction at human levels of scale, from towns and neighborhoods down to houses and rooms. Patterns go hand-in-hand with an underlying process. Alexander relates three properties of patterns that have become staples of software development. First, the process of building with patterns is incremental. User cannot achieve in a single leap. Second, it is grounded deep in human sensibilities, more so than in just engineering practices; and third, the process is iterative. [8-b]

The patterns are like elements of a language, and the constraints define the language. Indeed, the total system of patterns is called a pattern language. The notion of process is fundamental to patterns' place in system evolution. We grow a system one pattern at a time. While patterns are elements of form, it is also useful to think of them as functional transformations on a system that map the system from one state to another. [8-b]

Alexander et al. prefer users rather than technical persons as their consultants. It might be complex to develop an enriched user friendly product, but it would be easy in usage. User will not need any technical skills to work and handle a product. [8]

According to [9] every project has some details and description to construct. By observing that description one can easily set out what user wants and what is the possibility to construct in it.

3.2 Jan Borchers

Description of Jan Borchers' pattern is considered a standard. The pattern describes the solution of a problem by analyzing its context of the problem using examples and diagrams [10].

Pattern in Urban Architecture (e.g. Alexander's pattern for construction homes, buildings etc), patterns in Software Engineering (e.g. small talk UI, etc), patterns in HCI (e.g. apple's human interface guidelines etc) and patterns in Application domain (e.g. pattern mapping etc) are different pattern dimensions. [10]

Jan Borchers proposes a pattern language. Each pattern language is acyclic directed graph and it has nodes and edges. Each node represents a pattern and every two nodes form an edge. Each edge is a reference and every reference has some context. So each set has its name, ranking, illustration, forces, problem, solution and diagram [11].

3.3 Hybrid

The hybrid pattern is based on Alexander's notions. It has the basic elements of the Alexandrian pattern in an augmented form. Major change is involvement of the latest technologies based on user input. Thus hybrid pattern is a mixture of concepts of Alexander's and the working approach of technical and the user requirements. [12]

Example of this pattern is Toolbox in HTML. Further sub examples are:

a. Shape palettes in PowerPoint, Illustrator, MacDraw, etc.

b. Toolbar buttons for font styles and text alignment in Word, Netscape Composer, etc [12].

There are some disadvantages of this pattern too. It provides a single item solution. It is based on rush over technology and is mainly dependent on the developers experience. The pattern describes a single entity description/ solution. The problem addressed is theoretical, and is not supported by diagrammatical construction or through any development language. Practical implementation emergencies are not discussed in the pattern as well. [12]

3.4 User Interface Design Patterns

Sari's collection of User Interface Design Patterns explicitly states that *"we have not tried to apply the format of Alexander's design patterns."* The collection *"tries to outline the recurring design problems faced when trying to create good design."* In an opening section each pattern has a description of some characterizing features and an indication of use. This is then followed by several examples, which are screenshots. Finally, there is a reference section.

It is based on Model Driven Architecture (MDA), Unified Modeling Language (UML) and Platform Specific Model (PSM). The concept of UI design pattern can be enhanced by incorporating user requirements. The major innovative idea of this pattern is reusability for upcoming projects [9].

3.5 Pattern Language

Although *Hughes* [13] describes his work as a pattern language, he only gives a single example of a pattern form with no larger structure, organizing principle or "grammar". His form has some familiar components such as Trigger, Context, Forces and the unusual "Claims Analysis" which can contain negative consequences for deploying the pattern. This goes against the more common assumption that a pattern is a positive way of resolving a design problem. [13].

3.6 A pattern Language for Web Usability

Graham [14] has proposed a pattern language for web usability. He states that *"each pattern is presented using the same layout, semantic structure and typographical conventions. These are very closely based on the structure pioneered by Alexander et al."* An interesting feature of WU (web usability) however, is that it distinguishes different type of patterns depending on their placement within the collection. The types are Abstract, Concrete, and Terminal. *"Patterns being terminal does not mean that design thinking stops with them - merely that the language considers the further design issues as beyond its scope or ambitions."* It is an interesting concept which hints towards the possibility of separate but inter-locking collections. [14].

The main weakness is that the concept of reusability has not been considered. The areas which are not addressed in this pattern are software development**,** reusability and distributed development environment. [14]

3.7 30 card deck / card sorting

30-card deck has been constructed to assist long-term (5-, 10- or 20- year) design incorporating environmental sustainability [15]. Each card has:

- A sensitizing example on the face (they call this an "evocative image"),
- A name (they call this "the title of a concept"),
- A brief description,
- An activity "to assist a design team in considering that particular concept."

They are collected into four suits of "four critical envisioning dimensions": Stakeholders, Time, Values and Pervasiveness. [15].

There are some disadvantages of this pattern as well. The pattern does not mention any criteria of selecting the priority of cards. It does not mention what happens if the cards exceed the limit of deck as well. There is no interlinking between the decks. The main weakness is that the pattern has theoretical resemblance with card sorting and less implementation approach for large products. The pattern does not address interlink between decks as well [15].

4. DISCUSSION & CONCLUSION

It is evident from the above discussion that no design pattern provides a comprehensive solution. Alexander did marvelous work in theory; however it does not propose a pattern in concrete form. The 30 card deck pattern lacks interlinking of the decks. The architecture of hybrid pattern can be used to merge multiple patterns.

In OSS development, the main concern is design. Lack of expert designers or almost absence of expert designer results a low standard HCI conformance while developing an OSS product. Nielsen's usability engineering life cycle model is one of the comprehensive solutions to conform the usability in a project. World beat project is one of the implementation of the pattern language.

Various issues exist in HCI oriented design pattern that need to be addressed. Less user's involvement in design, no style guidelines, ignoring experience designers, less focus on essential values and design invariants that can be encoded in software, are some of the examples. The patterns mentioned in Section 3 highlight different issues. For example, Alexander et al. do not propose any fixed solution to project type. The hybrid pattern does not provide a fixed solution of development. And the 30 card deck pattern does not mention any criteria of selecting the priority of cards.

To overcome these issues a new design pattern is needed that should incorporate user requirements. We need a comprehensive practical oriented design pattern which focus on users' needs and expectations. A user centered design pattern is the need of the hour. Some practical and diagrammatical solution need to be presented in the proposed pattern so that with the help of the diagrammatical solution developers follow these steps and develop the project accordingly. End users should be fully involved. Similarly involvement of HCI experts at every phase, from requirement to design phase, is also recommended. Based on the recommendations and observations of HCI experts, developers could follow guidelines to address usability issues.

5. REFERENCES

[1] T.Jenifer., " A Pattern Language for Human-Computer Interface Design.," 17 May 1999. USA: MIT PRESS REALEASE.

[2] D. D. Lisa Battle. Lisa.B, *Usability Body of Knowledge*, 2010. Publisher: The Usability Professionals' Association (UPA)

[3] "http://c2.com/ppr/ui.html#1. ", accessed on 02 September 2012

[4] "http://www.cs.kent.ac.uk/people/staff/saf/patterns/gallery.html. ",

accessed on 02 September 2012

[5] M. Paula and S. Barbosa, " Bringing interaction specifications to HCI design patterns.," in Workshop Perspectives on HCI Patterns: Concepts and Tools, workshop CHI 2003.

[6] A. Wilde, P. Bruegger and B. Hirsbrunner, "An Overview of Human-Computer Interaction Patterns in Pervasive Systems.," in IEEE CONFERENCE PUBLICATIONS, International Conference on User Science Engineering (i-USEr) ., University of Fribourg, Switzerland, 2010.

[7] K.Christian, H.Martin. "The Anatomy of HCI Design Patterns" ,IEEE Computer Society Washington, DC, USA ©2009, Vols. 978-0-7695-3862-4 , no. Proceeding COMPUTATIONWORLD '09 Proceedings of the 2009 Computation World: Future Computing, Service Computation, Cognitive, Adaptive, Content, Patterns. Pages 202-207, November 15 2009.

[8] J.Borchers, "Teaching HCI Patterns: Experience from Two University Courses, position paper for "Patterns in Practice: A Workshop for UI Designers",," in International Conference

on Human Factors of Computing Systems , Apr. 21–25, 2002.

[9] B. B. Russell Beale, " Using modelling to put HCI design patterns to work & quot," in presented at (2005) HCI International. 11th International Conference on HumanComputer Interaction,, Las Vegas, 2005.

[10] M. W. M. Z. ,. B. Christian Remy, "A Pattern Language for Interactive Tabletops in Collaborative Workspaces.," in Proceeding EuroPLoP '10 Proceedings of the 15th European Conference on Pattern Languages of Programs, RWTH Aachen University, 52056 Aachen, Germany., 2010.

[11] J. Borchers, "A pattern approach to interaction design.," in Proceeding: DIS '00 Proceedings of the 3rd conference on Designing interactive systems: processes, practices, methods, and techniques, Department of Computer Science Darmstadt, University of Technology Alexanderstr. 6, 64283 Darmstadt, Germany..2000.

[12] M. H. Christian Kruschitz, " Analyzing the HCI Design Pattern Variety.," in 1st Asian Conference on Pattern Languages of Programs (AsianPlop'2010), Department of Informatics-Systems, University of Klagenfurt,Klagenfurt, Austria. , 17th March 2010.

[13] M. Hughes, "A Pattern Language for User Assistance.," Published by ACM., vol. Volume 14 , no. Issue 1, Pages 27 - 29 , January + February 2007,.

[14] I. Graham, " A pattern language for web usability. SBC Warburg," Addison-Wesley Professional Copyright: 2003, no. ISBN-10: 0201788888 ISBN-13: 9780201788884 , 2003.

[15] E. Hutchins., "The Cognitive Consequences of Patterns of Information Flow", Department of Cognitive Science, University of California, San Diego," Intellectica., vol. 1:30, pp. 53-74.

AudiNect: An Aid for the Autonomous Navigation of Visually Impaired People, Based On Virtual Interface

Mario Salerno *salerno@uniroma2.it*
Faculty of Engineering, Department of Electronics
University of Rome,"Tor Vergata"
Rome, 00133, ITALY

Marco Re *marco.re@uniroma2.it*
Faculty of Engineering, Department of Electronics
University of Rome,"Tor Vergata"
Rome, 00133, ITALY

Alessandro Cristini *alessandro.cristini84@gmail.com*
Faculty of Engineering, Department of Electronics
University of Rome,"Tor Vergata"
Rome, 00133, ITALY

Gianluca Susi *gianluca.susi@uniroma2.it*
Faculty of Engineering, Department of Electronics
University of Rome,"Tor Vergata"
Rome, 00133, ITALY

Marco Bertola *mrc.bertola@gmail.com*
Faculty of Engineering, Master in Sound Engineering
University of Rome,"Tor Vergata"
Rome, 00133, ITALY

Emiliano Daddario *emiliano182@yahoo.it*
Faculty of Engineering, Master in Sound Engineering
University of Rome,"Tor Vergata"
Rome, 00133, ITALY

Francesca Capobianco *francesca.capobianco@yahoo.it*
Faculty of Engineering, Department of Electronics
University of Rome,"Tor Vergata"
Rome, 00133, ITALY

Abstract

In this paper, the realization of a new kind of autonomous navigation aid is presented. The prototype, called *AudiNect*, is mainly developed as an aid for visually impaired people, though a larger range of applications is also possible. The *AudiNect* prototype is based on the *Kinect* device for Xbox 360. On the basis of the *Kinect* output data, proper acoustic feedback is generated, so that useful depth information from 3D frontal scene can be easily developed and acquired. To this purpose, a number of basic problems have been analyzed, in relation to visually impaired people orientation and movement, through both actual experimentations and a careful literature research in the field. Quite satisfactory results have been reached and discussed, on the basis of proper tests on blindfolded sighted individuals.

Keywords: Visually Impaired, Blind, Autonomous Navigation, Virtual Interfaces, Kinect, AudiNect.

1. INTRODUCTION

It is well known that the traditional aids for visually impaired people are canes and guide dogs, though they can present a number of limitations. Indeed, they do not allow natural movements or extended mobility, so that external helps and additional costs can often be required. In recent years, Virtual and Augmented Reality has been applied to rehabilitation of children and adults in presence of visual impairment, so that increased autonomy and significant life quality improvement can be reached [1], [2], [3], [4].

Various localization and identification techniques have been developed and discussed in the technical literature. In particular, systems based on radio waves [5], infrared [6], ultrasound [7, 8], GPS [9, 10], RFID [9] [11] are of interest in the field. However, some actual limitations can often be pointed out. For instance, systems based on radio waves seem not very accurate in position, while GPS systems appear not quite reliable in the case of indoor applications. It has been pointed out that RFID (Radio Frequency Identification) systems can overcome some limitations of many other technologies. However, in general, all the actual needs of visually impaired people seem not always fully solved by many technical approaches [12].

The use of *Kinect* for Xbox has been recently considered in the literature [13], as a new promising approach. Besides its very low cost, the main advantage is its robustness to light, being able to work both in light and in dark environments. A significant contribution in this field is the Navigational Aids for Visually Impaired (NAVI), developed at the University of Konstanz, Germany [14]. This device can improve the navigation in indoor environments, exploiting several *Kinect* capabilities. NAVI works through a vibrotactile belt that can mimic the room layout. The *Kinect* is placed on a helmet worn by the user. In addition, a number of Augmented Reality (AR) markers is placed along the walls, in order to identify a significant path from room to room. Depth information is provided by *Kinect* and processed by a C++ software, so that some vibration motors can properly be activated (on the right, the center and the left sides of the patient waist). In this way, according to the markers location, the system can guide the user through the correct path. To this purpose, a synthetic voice is also generated as an auxiliary navigation aid. A similar project is Kinecthesia, that uses a *Kinect* placed at the waist [15, 16]. Vibrating motors are placed on a belt and activated when obstacles along the path are detected. Proper audio alarms are also generated. In particular, different vibration intensities stand for different obstacle distances. On the basis of the new technologies and solutions, the studies on optimal autonomous navigation tools seem still an open research field. In particular, new or better solutions should be very important to satisfy the typical blind issues with regard to both real time and robust interaction with the environment.

In this paper, the development of a useful tool for the autonomous navigation is presented. This tool, called *AudiNect*, is based on the synthesis of proper acoustic feedback, designed as an aid for visually impaired people. It seems able to overcome some traditional tool limitations, and it is efficacy since the acoustic feedback encode the information in a simple way to be learned by a user. Moreover, the use of the *Microsoft Kinect* provides both a low cost design (e.g., compared with GPS or RFID technologies) and the possibility to encode more information from the environment than the classical tools. Also, since a system based on this new kind of technology is robustness to any light conditions (due to the use of both IR and RGB cameras, and also thanks to a proper software, introduced in the next section), can represent a valid alternative for the autonomous navigation. Finally, the proposed system does not need to be supported by other external devices to obtain information from the environment. For example, RFID systems need AR markers located along the particular considered paths; whereas GPS systems need signals sent by GPS satellites, but urban canyon and indoor site are no suitable environment due to technological limits in GPS signals capturing [12].
The paper is organized as follows: first, we will provide the basic characteristics of the proposed model. Second, we will illustrate the operation mode of the system. Third, we will test the latter on blindfolded sighted people in order to prove its validity in real cases, that is in presence of real scenarios. Results and future developments will be discussed in the last section.

2. AUDINECT

2.1. Basic Characteristics

AudiNect is based on the use of the sense of hearing, usually very well developed in people with visual impairment. To this purpose, proper auditory feedbacks are generated to detect and identify the presence of obstacles on the path in front of the user. The auditory feedbacks consist of proper sound signals, both frequency and intensity modulated. In this way the information on the most useful path can easily be recognized. The basic visual data are acquired by the *Kinect* device. By using the infrared technology, it generates proper output stream information, whose features depend on the used Software Library. A number of 30 frames per second can be produced, representing the frontal scene depth map. A matrix of up to 640 x 480 pixels can be generated, in which the distances of identified obstacles are coded and memorized. The angular ranges, covered by the depth map, are about 60 degrees (horizontal) and 40 degrees (vertical) in front of the device. Distances between 0.8 and 4.0 meters can be detected with good accuracy. In the case of the use of OpenNI Library, information of distances out of this range are provided with less accuracy. The distance information is coded in the gray scale for the usable detectable distance in the depth map. On the contrary, no-information pixels are generated in all other cases. On the basis of the *Kinect* data, the main purpose of *AudiNect* is to synthesize the proper acoustic feedbacks by means of depth map careful analysis and interpretation. This is made by proper Digital Signal Processing software [17], based on PureData for the sound engine [18], SimpleOpenNI and Open Sound Control OSCp5 Libraries [19]. In this way, the proper

acoustic information can be generated, so that they can quite be useful as an aid to the autonomous navigation. *AudiNect* can work in the range from 0.5 m to 4 m.

"Auditory display" is the term used in literature to name the use of the sound output to convey information to the user. The homonymous block in FIGURE 1 is the software component that receives numerical values computed only from visual data, and properly translates them in sound amplitudes, frequencies and timing messages sent to PureData, which is simply used as a tone generator.

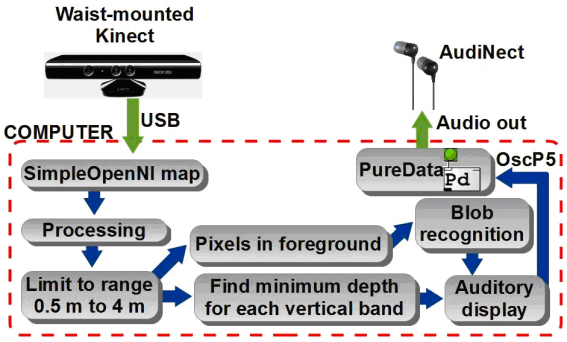

FIGURE 1: Scheme for the Proposed Navigation Prototype.

2.2. Operating Mode

The first step for the depth map processing consists of the generation of two synthesized "images". Using Simple OpenNI library, the first image is defined on the basis of a static maximum threshold of 4 metres, the second one is derived from the first image, applying to it a dynamic maximum threshold placed at 0.5 metres beyond the pixel with the minimum depth. This pixel will be denoted as the *foreground point*. In addition, at every scan and for each of the two images, three "vertical bands" (left, middle, and right) are defined as a partition of the whole image. In correspondence to every vertical band, the first image is analyzed to identify the minimum depth of all the pixels. Furthermore, the second image is used to identify the significant obstacles, with a process called *blob recognition*. This is the only processing done working on the whole image and not in correspondence to a single band. The blobs are used to evaluate the width occupied by the obstacles. This last global information is further processed and separated in three pieces of information each relative to each band. To this purpose, all the blobs can be used, or else only some of them, as for instance the closest *N*. The blob identification and the distance are analyzed to generate the proper acoustic feedbacks, consisting of sequences of stereophonic pulses and modulated continuous tones. In correspondence to the left and right bands, using only amplitude modulation, proper impulsive sounds are generated, according to the relative obstacle presence in each band. In particular, a single impulsive sound is generated if the obstacle width is less than 33.3% of the band width, two sounds if it is in the range (33.3% - 66.7%), three if it is greater than 66.7%. Pulses of 0.1 seconds are used, adding proper stereophonic effects for the user convenience. The complete pulse train repetition occurs every 1.5 seconds.

Each block in FIGURE 2 has a well determined high-level, object-oriented meaning, which can be implemented using different low-level types. For example, the term "range" here is meant in the mathematical sense; its data structure is just a pair of integers (xMin, xMax).

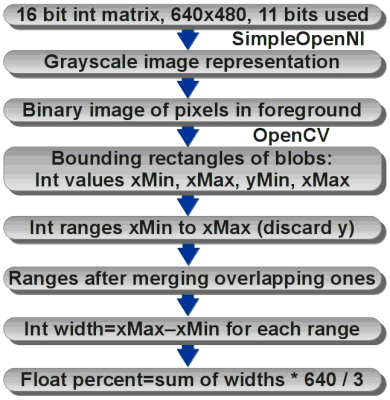

FIGURE 2: All the Intermediate Data Structures Used in Computing of the Width Percentage Occupied By Obstacles

With regard to the central band, a continuous signal is generated, using both frequency and amplitude modulation. The tone frequency is proportional to the width percentage occupied by the obstacles. In particular, when a "clear path" is identified, a high-pitched signal is produced. On the contrary, a low-pitched signal indicates that the free available space becomes lower and, if a significant obstacle is faced, the continuous signal fully stops. Thus, the total signal absence clearly denotes the presence of an obstacle in the frontal direction. In addition, the continuous signal intensity is used to indicate the minimum distance of the closest obstacle in front of the user. The impulsive sound sequence and the continuous signal frequency and amplitude modulation can easily be interpreted by the user to identify the different and usable free paths. As a consequence, it is quite easy to identify the correct path with fewer obstacles. Such a spatialization allows the user to correlate the sounds sequence with each of the three bands.

In summary, the information about the occupied band width and the minimum depth for each band are coded in the acoustic feedback, and helps the user to find the correct path.

It is important to note that, if a wall, or a large obstacle, is located less than 50 cm from the *Kinect*, the number of no-information pixels becomes quite large, within the whole depth map. If this number exceeds a fixed threshold, the acoustic feedback is temporarily stopped and a special tone is produced to indicate the presence of a significant obstacle in front of the user.

In the two graphs, shown in the following figure, the two sound layers (A: lateral; B: central), in the particular case in which the device is stopped and completely hampered, are represented. In order to avoid that central sound pulse could mask lateral sound pulses, the condition $a_2 << a_1$ is required.

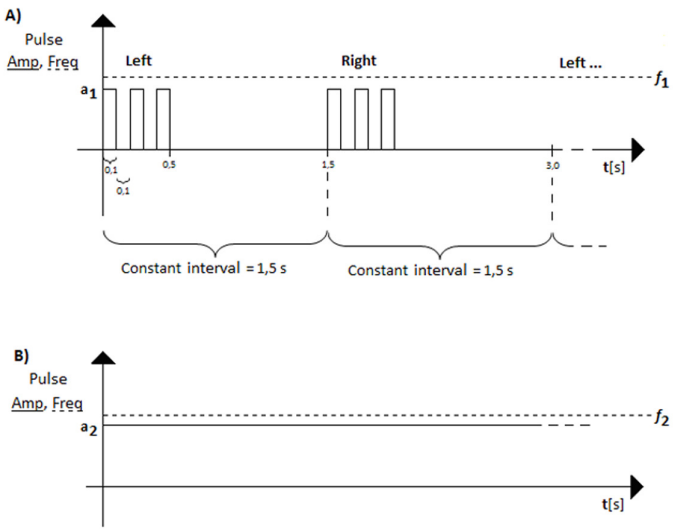

FIGURE 3: Explanation of the Acoustic Feedback Coding With Respect To the Bands.

The proposed approach has been tested on actual scenarios, as it will be shown in the next section. The test analysis appears quite satisfactory, even though some particular critical situations can be observed. The first one concerns the presence of glass surfaces or mirrors; indeed, the *Kinect* source depth map could not be quite correct for the scene representation. This problem can be overcome introducing a proper noise detector software. The second critical issue is that the *Kinect* field of view is supposed not to contain the ground plane. Indeed, in certain cases, this plane could be erroneously detected as an obstacle on the ground. This phenomenon is due to the fact that a proper algorithm for the ground plane identification is not present in the Simple OpenNI software. The use of the Microsoft SDK easily overcomes this limitation.

2.3. Actual Tests For The AudiNect Evaluation

Actual analysis of the *AudiNect* operation mode is presented in this section on the basis of a number of experimental tests, in order to discuss its actual usefulness. To this purpose, two different sets of experiments have been carried out at the University of Rome, "Tor Vergata" (*Master in Sound Engineering* Laboratories). The aim is to compare some actual walking tests on the device, so that the validity of the proposed acoustic feedback can be evaluated. In particular, the learning curves are measured by proper comparison among untrained and trained test people. The walking paths are realized by means of proper obstacle sets, placed in different ways, in a closed room. In the first set, initially untrained people are employed, and the walking tests are analyzed to evaluate the learning behaviour. In the second set, new untrained people are compared with the people already trained in the previous set, in order to make a comparison between these two groups.

a) *First set*. Five properly blindfolded people are equipped with battery-powered *AudiNect* and wireless headphones. Each person is asked to walk in a number of walking paths of similar difficulty (*walking set*). The aim was to measure the walking time behaviour (*learning curve*) in a number of similar difficulty paths, so that the validity of the acoustic guide can be evaluated. The paths have been realized in a room (5.70 m x 4.80 m), in which 13 main obstacles have been placed in random way, as shown in FIGURE 4. In addition, some other smaller obstacles are used. In order to avoid memorization, different paths have been proposed to each testing person. In addition, in order to ensure similar difficulties, the new paths were obtained by flipping the first one with respect to x and y axes. Random path sequences are used to measure the learning curve of each person.

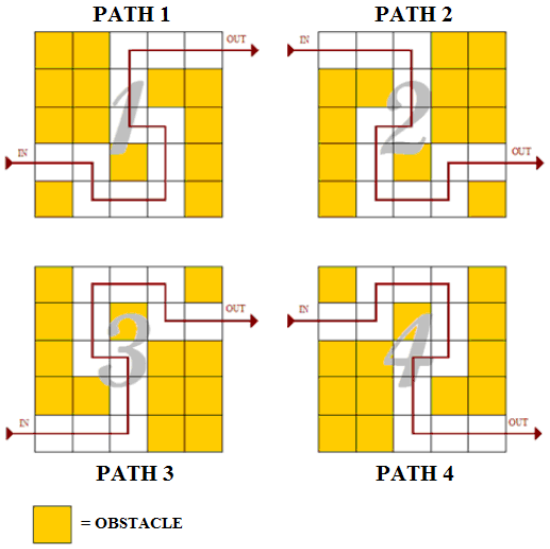

FIGURE 4: The Four Different Paths Proposed.

As the total number of trials increases, the travelling time spent to complete the path decreases, according to the behaviour of FIGURE 5. In particular, on average, a person shows the best improvements in his first trials, as shown in TABLE 1.

Trial Number	Path	Travelling Time (s)
1	3	156
2	2	117
3	1	56
4	4	42
5	2	50
6	3	41

TABLE 1: The Proposed Paths sequence To an Untrained Individual, and the Relatively Exhibited Travelling Times.

These results show the validity of the *AudiNect* approach. Indeed, the acoustic code seems easy to learn.

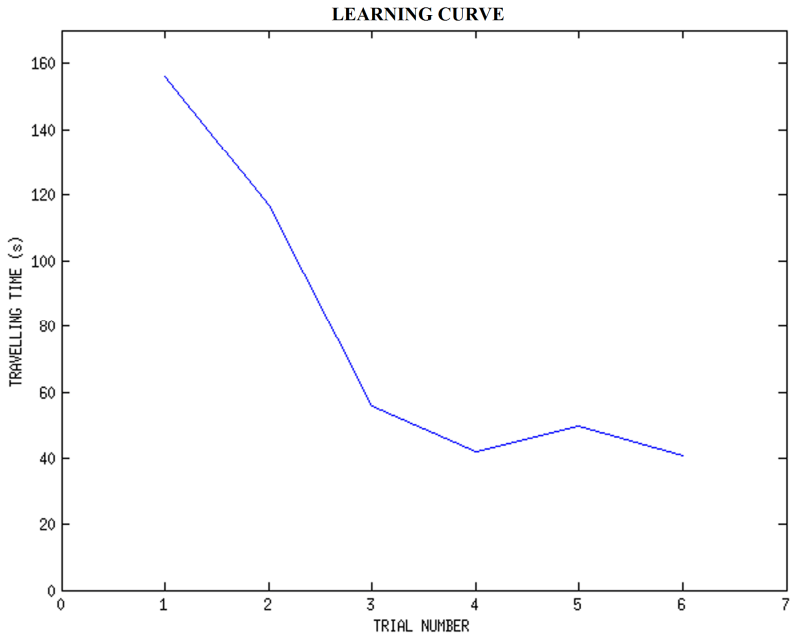

FIGURE 5: Learning Curve Determined By the Trials Performed By the Same Individual

The personal learning times and curves are shown and compared in TABLE 2 and FIGURE 6. They appear quite similar for all the involved test people.

Individual	Travelling Time (s)					
1	156	117	56	42	50	41
2	141	134	76	40	42	47
3	175	132	80	74	48	43
4	161	124	75	74	53	40
5	164	182	70	40	51	68
Trial Number	1	2	3	4	5	6

TABLE 2: Exhibited Travelling Times By the Individuals For Each Trial.

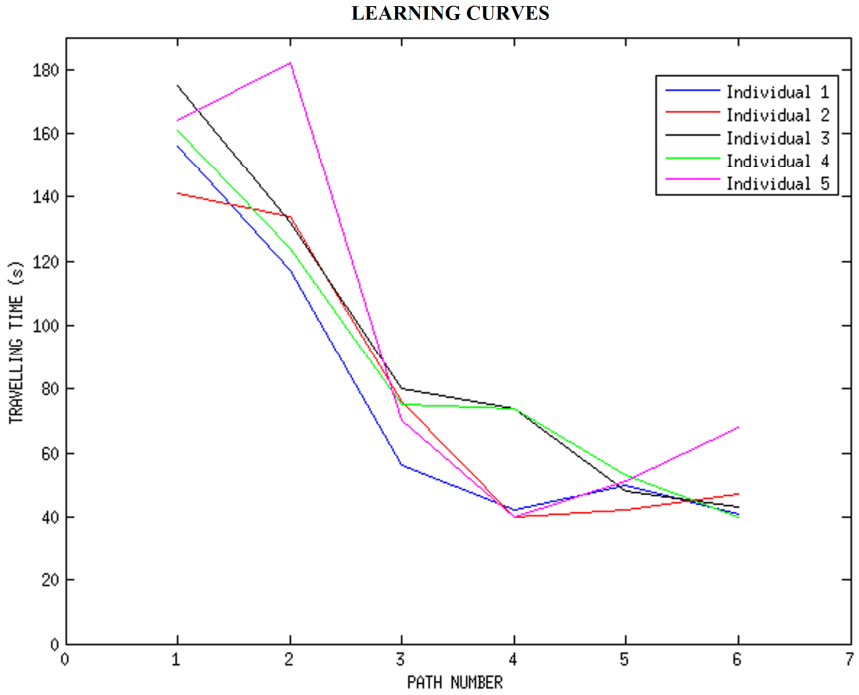

FIGURE 6: Learning Curves Determined By the Trials Performed By Five Individuals.

On the basis of the learning behaviours, the *AudiNect* approach appears quite easy to correctly be added to the human cognitive process. Thus, we can assume that the system can be used, after a little training time, as an assistance to autonomous navigation. It may be applied in the case of visually impaired people, as well as in other applications in which the direct human vision is not possible.

b) *Second set*. The same paths proposed in the case a) are still applied, proposed in a random way again. The main difference is that the second set is devoted to the comparison between untrained and trained people. In particular, people involved in the first set of experiments are now compared with new people. The following results are now obtained.

- Person “A” (already trained) travelling times (TABLE 3).

Trial Number	Path	Travelling Time (s)
1	4	71
2	3	65
3	2	56
4	1	68

TABLE 3: Travelling Times Exhibited By A Trained Person.

- Person “B” (untrained) travelling times (TABLE 4).

Trial Number	Path	Travelling Time (s)
1	1	160
2	2	121
3	3	155
4	4	100

TABLE 4: Travelling Times Exhibited By An Untrained Person.

The data comparison is shown in FIGURE 7. Note that the order of the walks is not the same for both the individuals.

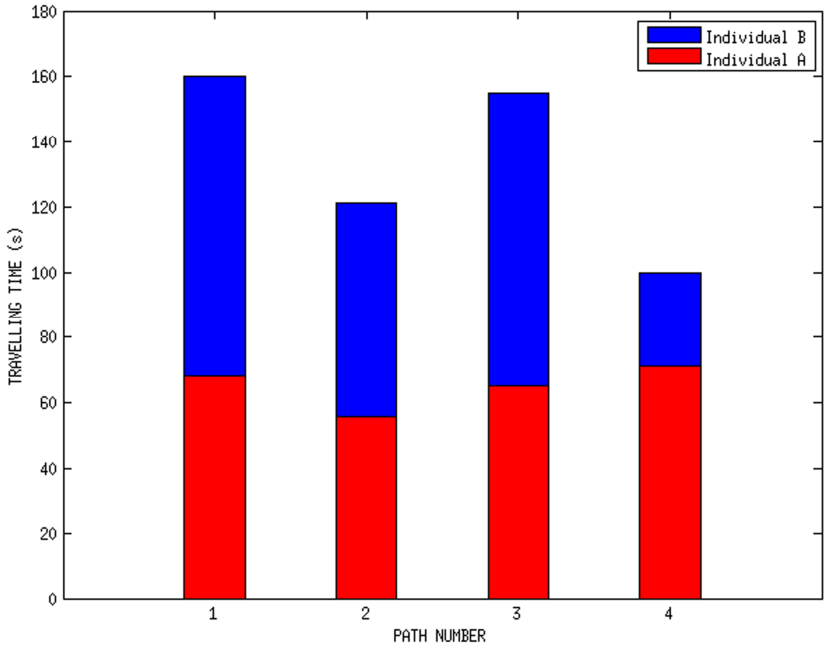

FIGURE 7: Difference Between the Travelling Times Shown By Both Trained and Untrained Individuals.

In particular, it can be noted that the travelling times of trained person "A" are quite smaller than those of the untrained person "B". On the contrary, this last person shows the most remarkable average improvements.

3. CONCLUSIONS

In this paper, a prototype for blind autonomous navigation is presented. As discussed in the introduction, the prototype is able to overcome some limitations of the systems proposed in the technical literature, both in terms of costs and in the terms of quantity of information extractable from the environment. The *AudiNect* project makes use of a recent low cost device, the *Kinect*, in order to obtain useful data related to the surrounding environment. As illustrated in the section 2.2, the data are processed by proper DSP techniques. Through PureData, it is possible to synthesize the acoustic feedback related to the obstacle detection for a given path. The system was tested on blindfolded sighted individuals. The related tests show satisfactory results on the basis of learning curves, showing a rapid adaptation of the individuals to the proposed method. This suggest that the technology is well integrated with the human cognitive processes. Indeed, the *AudiNect* let the user to easily identify the best free path.

In further works we will improve the portability of the system, by making a miniaturization of the latter using a tablet device. This can reduce the overall power consumption, increasing the autonomy of the system. In order to further improve the autonomous navigation capabilities, another future development would be to integrate a device with tactile feedback in the system. Indeed, this could be helpful to reduce the use of acoustic feedback only in critical cases of danger, thus allowing to avoid potential sound masking of the natural sound information from the environment.

4. REFERENCES

[1] K.P. Beier. "Virtual reality: A short introduction." Internet: http://www-vrl.umich.edu/intro/, Nov. 25, 2008 [Nov. 16, 2012].

[2] D. Talaba and A. Amditis. Product engineering, tools and methods based on virtual reality. Dordrecht, Springer, 2008.

[3] A.B. Craig, W. Sherman, J. Will. Developing Virtual Reality Applications: Foundations of Effective Design. Burlington, The Morgan Kaufmann Series in Computer Graphics, 2009.

[4] N.I. Ghali, O. Soluiman, N. El-Bendary, T.M. Nassef, S.A. Ahmed, Y.M. Elbarawy, A.E. Hassanien. "Virtual Reality Technology for Blind and Visual Impaired People: Reviews and Recent Advances," in Advances in Robotics and Virtual Reality, 1st ed., vol. 26, T. Gulrez and A.E. Hassanien, Ed.: Springer, 2012, pp. 363-385.

[5] L.W. Alonzi, D.C. Smith, G.J. Burlak, M. Mirowski. "Radio frequency message apparatus for aiding ambulatory travel of visually impaired persons," U.S. Patent 5 144 294, Sep. 1, 1992.

[6] D. Ross and A. Lightman. "Talking braille: a wireless ubiquitous computing network for orientation and wayfinding," in Proceedings of the seventh international ACM SIGACCESS conference on Computers and accessibility, 2005, pp. 98-105.

[7] J. Souquet, P. Defranould and J. Desbois. "Design of Low-Loss Wide-Band Ultrasonic Transducers for Noninvasive Medical Application." IEEE Tran. Sonics and Ultrasonics, vol. SU-26, no. 2, pp. 75-81, 1979.

[8] A.J. Ali and A.H. Sankar. "Artificial Guidance System for Visually Impaired People." Bonfring International Journal of Man Machine Interface, vol. 2, Special Issue 1, Feb. 2012.

[9] K. Yelamarthi, D. Haas, D. Nielsen, S. Mothersell. "RFID and GPS Integrated Navigation System for the Visually Impaired," in 53rd IEEE International Midwest Symposium on Circuits and Systems, 2010, pp. 1149-1152.

[10] N. Márkus, A. Arató, Z. Juhász, G. Bognár and L. Késmárki. "MOST-NNG: An Accessible GPS Navigation Application Integrated into the MObile Slate Talker (MOST) for the Blind," in Computers Helping People with Special Needs, Lecture Notes in Computer Science, K. Miesenberger, J. Klaus, W. Zagler and A. Karshmer, Ed.: Springer, 2010, pp. 247-254.

[11] M. Bessho, S. Kobayashi, N. Koshizuka, K. Sakamura. "Assisting mobility of the disabled using space-identifying ubiquitous infrastructure," in Proceedings of the tenth international ACM SIGACCESS conference on Computer and accessibility, 2008, pp. 283-284.

[12] U. Biader Ceipidor, E. D'Atri, C.M. Medaglia, A. Serbanati, G. Azzalin, F. Rizzo, M. Sironi, M. Contenti, A. D'Atri. "A RFID System to Help Visually Impaired People in Mobility," presented at the EU RFID Forum, Brussels, Belgium, 2007.

[13] S. Keane, J. Hall, P. Perry. Meet the Kinect: An Introduction to Programming Natural User Interfaces. New York, Apress, 2011.

[14] M. Zöllner, S. Huber, H.C. Jetter, H. Reiterer. "NAVI – A Proof-of-Concept of a Mobile Navigational Aid for Visually Impaired Based on the Microsoft Kinect," in Proceedings of thirteenth IFIP TC13 conference on Human-Computer Interaction, 2011, pp. 584-587.

[15] E. Berdinis and J. Kiske. "Kinecthesia: Using Video Game Technology to Assist the Visually Impaired." Internet: http://www.noblevisiongroup.com/ophthalmic/kinecthesia-using-video-game-technology-to-assist-the-visually-impaired, Nov. 17, 2011 [Nov. 16, 2012].

[16] E. Berdinis and J. Kiske. "Students Hack Kinect to Make the Kinecthesia Personal Radar for the Vision Impaired." Internet: http://au.ibtimes.com/articles/251028/20111116/students-hack-kinect-make-kinecthesia-personal-radar.htm, Nov. 17, 2011 [Nov. 16, 2012].

[17] C. Reas and B. Fry. Processing: a programming handbook for visual designers and artists. Cambridge, The MIT Press, 2007.

[18] Audio Programming Languages: Mathematica, Csound, Max, Pure Data, Supercollider, Comparison of Audio Synthesis Environments. Books LLC, 2010.

[19] R.T. Dean. The Oxford handbook of computer music. New York, Oxford University Press, 2009.

Usability of User Interface Styles for Learning Graphical Software Applications

Andisheh Feizi *feiziandisheh@yahoo.com*
Interface Design Department, Faculty of Creative Multimedia
Multimedia University
Cyberjaya, 63100, Malaysia

Chui Yin Wong *cywong@mmu.edu.my*
Interface Design Department, Faculty of Creative Multimedia
Multimedia University
Cyberjaya, 63100, Malaysia

Abstract

This paper examines usability of different user interface styles for learning graphical software applications, namely Adobe Flash CS4 and Microsoft Expression Blend 4. An empirical study was performed to investigate the usability attributes of effectiveness, efficiency and satisfaction scores for learning the graphical software applications. There were 32 participants recruited whom consist of interface designers and software developers. A set of 7 tasks was designed to compare the different effects of user interface styles including graphical user interface (GUI) and command line interface (CLI). User Performance variables (effectiveness, efficiency, duration, number of errors and number of helps) were measured for tasks performed by all the participants in the test. Satisfaction score was measured using QUIS (Questionnaire for User Interface Satisfaction) tool. The result revealed that the average effectiveness scores are higher than 75% for both software applications. Although Adobe Flash CS4 gained slightly higher on effectiveness, Microsoft Expression Blend 4 obtained better results in terms of efficiency, duration, errors and helps. The user satisfaction rates also showed Microsoft Expression Blend 4 gained higher satisfaction comparing Adobe Flash CS4. Generally, both software applications gained scores above average (>3.5) for majority of the user interface satisfaction attributes of software regardless of users' background.

Keywords: Usability, User Interface Styles, Graphical User Interface (GUI), Command Line Interface (CLI), Graphical Software Application.

1. INTRODUCTION

User Interfaces (UIs) have been around since the invention of computers, even before the field of Human-Computer Interaction (HCI) was initiated [1]. Users carry out information communication efficiently with computers through User Interface (UI) to complete their tasks [2]. Since UI design is an important component of HCI system [2], great burden has been on software designers to create interfaces that effectively predict and interpret the operator's needs besides allowing the user to perform tasks in natural ways [3].

Throughout the last four decades, programming has evolved from platforms with great amount of difficulties and constrains to enter, read and debug a programme into command language strategies, and eventually into the approach of Graphical User Interface (GUI) [4]. In the context of programming, user interface design plays an important role [5].

The development of GUI software applications has been one of the noteworthy improvements in programming field that reduce the difficulty of remembering syntax and semantics with the guidance of menu-based interactive properties it delivers [6]. However, there are still circumstances that require the users to use command-line interface (CLI) since CLIs often

afford more options than their equivalent GUIs, leading to greater flexibility available for users or one can perform a task by using command that its function is not supported by its GUI counterpart [7]. Command Line Interfaces (CLI) are considered quite inconvenient environment for new generation of users since they are substantially used to GUIs [8]. Hence, it is important to examine usability level of different user interface styles of using GUI and/or CLI in learning graphical software applications.

The International Organization for Standardization and the International Electro Technical Commission ISO/IEC 9126-1 classify software quality attributes into six categories: functionality, usability, reliability, efficiency, maintainability and portability [9]. Usability is increasingly accepted as a significant quality factor for interactive software systems like GUI style applications, Web sites, and variety of mobile interactive services [10]. Benson et al. [11] consider usable software *"a win–win situation for developers, corporations, and the users"*.

In regards of the issue of users learning a software application to produce new knowledge, usability is considered as an essential attribute for quality of software design. We are concerned with the usability of software packages because nowadays, large numbers of people use applications at work and for personal tasks as well. These users desire to learn software to meet their professional needs. Some may use a software application frequently or occasionally, but they do not use it intensively, as clerical workers do [12]. So, they rarely become experts in the use of software [12].

Having analyzed the above-mentioned issues, this paper aims to examine the usability of different user interface design styles including GUI and CLI for learning a graphical software application by interface designers and software developers. The followings are the objectives of this research:

- To evaluate usability attributes in terms of user performance measure (effectiveness, efficiency, time duration, number of errors, and number of helps) for two competing graphical software applications;
- To evaluate user satisfaction for user interfaces of two competing graphical software applications;
- To examine usability of different user interface styles (i.e. GUI and CLI) for learning a graphical software application.

2. LITERATURE REVIEW

Users "have contact with an information system only with the help of an interface that defines information flow rules between a human and a machine" [13]. Lauesen [14] defines the user interface (UI) as organizing and designing screens in a way that user can easily understand and efficiently utilize the system. User Interface Design (UID) refers to the "overall process of designing how a user will be able to interact with a system" [15]. UID concerns about "facilitating clear and accurate information exchanges, efficient transactions, and high-quality collaborative work" [16].

Software user interface is an essential medium for information transmission between users and computers for successfully performing various tasks, besides designing new software products [17]. Software user interface features will dramatically influence the user's efficiency and attitude towards it [18]. It is the user interface of a computer program, which provides users with the perception of what a user interface can do and how to do it [19]. User interface design is a fundamental concern for the usability of a software product [20] and is also one of the significant concerns in HCI field [17].

In Human-Computer Interaction (HCI) discipline, researchers in the field mainly focus on five 'E' of usability, which propose an interactive system must be 'effective, efficient, engaging, error tolerance, and easy to learn'. HCI is a branch of human factors field, which involves user interface design, human-computer communications, and user engagement [21]. The goal is to provide users with information systems (i.e. software interface) and work environments in which they can do their tasks efficiently [22]. Despite the importance of

usability in software development, it is still insufficient in majority of software applications [23] [24]. The IFIP Working Group comments that 'there are major gaps of communication between the fields of HCI and software engineering (SE) [25]. Since the most prevalent perspective in the field of SE is that usability is mainly related to the UI [23] [26]. Some mentioned usability primarily concerns the UI rather than the system's core. However, Juristo et al. [27] demonstrate that usability is not confined to the interface and can affect the core functionality of a system. They believe usability is associated with the entire user–system interaction, not just the UI [27].

2.1 Usability of Software Design

Based on research from theoretical and practical perspectives in software field, some guideline standards established for clarifying the usability of software products [28] [29] [30]. Usability is defined as "the ease with which a user can learn to operate, prepare inputs for, and interpret outputs of a system or component" [31]. Gould [32] categorizes usability into system performance, system functions, and user interface. McCall et al. [33] outlines usability as operability, training and communicativeness. Booth [34] explains that usability has four attributes as usefulness, effectiveness, learnability, and attitude. Hix et al. [35] classify usability into performance, learnability, retainability, first impression, and long-term user satisfaction. Software Usability Measurement Inventory (SUMI) determines usability in terms of efficiency, effectiveness, helpfulness, control and learnability [36]. Donyaee et al. [37] established quality in use integrated measurement (QUIM) model including attributes as effectiveness, efficiency, satisfaction, productivity, safety, accessibility, and internationality. Battleson et al. [38] discusses that to enhance usability, an interface must be easy to learn, use, and remember with few errors for its intended users. Sauro et al. [39] planned a 'single and summated' usability metric for each task by averaging four values for task time, errors, completion, and satisfaction. Shneiderman [40] claims that 'a clever design for one community of users may be inappropriate for another community' and 'an efficient design for one class of tasks may be inefficient for another class'.

Bevan [29] proposed a detailed description for the term of usability, which considers effectiveness, efficiency, and satisfaction as quality factors for usability. Usability is defined, in ISO 9241-11, as the extent to which a software product can be employed by particular users to achieve specific objectives with satisfaction, efficiency and effectiveness within a certain context of use [41]. Subsequently, ISO 9126-1 explains usability in terms of learnability, operability, understandability, and attractiveness [42].

Nielsen [43] proposed one of the popular definitions for usability involving the learnability and memorability of a software program, its capacity to avoid and control user errors, its efficiency of use and user satisfaction. Technically speaking, "efficient use of the computer is intrinsic to usability" [44]. To the sense that in assessing the usability of a software product, it is required to examine user performance in addition to considering the amount of effort a user puts in applying the software. Therefore, a system is not usable if it requires high amount of effort in order to complete a task with a high performance [44]. The most prevalent perspective in the field of software engineering (SE) is that usability is intertwined with the user interface [23]. Shneiderman et al. [45] determine features of user interface design based on evaluation of several human factors such as length of time to learn, learner's rate of errors, pace of performance, user's satisfaction, and retention over time. However, despite the significance of usability in software development, it is still unsatisfactory in majority of software programs [23].

Measuring software usability is a significant indicator of the deficiency level of software application, and software testing is the foundation for software usability enhancement [2]. Software usability is not directly measurable; it can be simply evaluated indirectly through observing measures, such as effectiveness, user's satisfaction and performance assessment [46].

Software tools used for usability evaluation have been available since 1980s. They consisted of two groups, questionnaire tools measuring user's perception and satisfaction (e.g. QUIS) and behavioral data collection software to capture and record user's performance (e.g. Camtasia) [47]. The procedure usability practitioners establish includes (1) observing subjects individually in real-time session to collect instant physical and verbal behaviors, (2) obtaining

performance measures such as number of errors, and time on task (3) comparing two or more systems, designs, or product features (usually from competitors (4) performing statistical analysis of the collected data to justify product design [47].

2.2 Categories of User Interface Styles

The style of human–computer information flow within a single-user interface is determined by the application of interactions [48]. Generally, basic interaction styles include command-line languages, filling forms, menus, direct manipulation, and natural language [48]. According to International Business Machines [49] user interfaces can be categorized into three fundamental groups:

(i) Command line user interface that is a full-text display mode on a computer screen controlled by a keyboard, in which users type in data, commands or instructions notifying the computer to do a task. A common example of a Command Line Interface (CLI) is UNIX-based that text is only shown on the entire screen [50].

(ii) Menu-Driven user interfaces, "in which a user is provided with a hierarchically organized set of choices" [6]. Robertson et al. [51] mention that users fail to correctly perform a task on a menu when structure of the menu is complex. However, Gray [52] believes that such a result can be regarded to the psychological issues with user interfaces and the limitation of learner's short-term memory. In a menu-based environment, a user clicks on a command from pre-defined array of commands exhibited in menus. If command names on the menus are understandable and well organized, users can easily perform their tasks since discovering a command in a menu is equivalent to recognition instead of recall [53]. This type of interface is ideal for novice learners as they support error handling; however, they can appeal to expert users if arrangement and selection processes are quick enough as well as convenient shortcuts are provided. On the other hand, possibly menus are slow for regular users besides the fact that numerous numbers of menus may result in overload and too much complexity.

(iii) Graphical User Interfaces (GUIs), which is an interactive human-computer interface that makes use of widgets including windows, icons, menus, buttons, dialog boxes and etc. It is often directly manipulated by a computer mouse, and to a limited extent by a keyboard [7]. The widgets are basic visual blocks combined in an application that hold all the data processed by the application and the available interactions required to achieve goals of the user. Users can interact with information by manipulating visual widgets provided; according to the kind of data they hold [54]. GUIs are direct manipulation systems currently familiar to users in the Windows environment [55].

Nowadays, we have become so accustomed to interact with a Graphical User Interface (GUI) since it makes it easier for people to work with computer software regardless of their computer skills [56].

Researchers working with students regarding learning programming languages came to conclusion that interface of software applications play an important role in quality of learning and efficiency, and the learning process should be underpinned by a rich programming environment [5]. According to Shneiderman [57], employment of Direct Manipulation Interfaces (DMI in which GUIs are included) has reinforced the accuracy and diminished errors, besides facilitating learning. Another investigation on the influence of interface styles on perceived ease of use and usefulness came to the conclusion that menu-based interface was more beneficial rather than command-based interface [58]. Davis et al. [59] compared DMI and CLI styles. Their results indicated no significant distinction on perceived ease of use. Davis [60] looked at user perception in using a text editor and electronic mail applications finding out that system features had considerable effect on ease of use. Wiedenbeck et al. [61] examined DMI, menu-driven and CLI. Their outcome showed that interface style did not affect participants' perception towards the usefulness of the system, however, DMI style was considered easier to use by participants. Moreover, Gururajan et al. [62] investigated on icon-based and menu-based interfaces claiming that interface style has no considerable influences on ease of use. Shneiderman [6] states users can track down information more quickly with GUIs compared to CLIs. Besides that, a user's understanding and satisfaction is higher for

GUI applications. Additionally, Faulkner [55] expresses that there is an evidence to confirm humans recall pictures better than words [22].

However, the most popular UI for software today is the GUI [63]; McGraw [64] points out, even GUIs can bring difficulties to navigate and use. Virvou et al. [65] describes that users of current GUIs may repeatedly find themselves brought into problematic situations even without realizing it. Testing GUIs produces many challenges, due to the enormous number of possible combinations of commands that can be executed on the GUI. Testing all possible orderings of events is not practical. Alternatively, testers of GUI applications attempt to limit the number of test cases that need to be executed [63].

3. RESEARCH METHODOLOGY

We conducted a usability testing to evaluate the usability attributes of comparing different user interface design (i.e. GUI and CLI) of learning graphical software applications using Adobe Flash CS4 and Microsoft Expression Blend 4. The research method is an experimental study with a mixture of observation, user interface satisfaction questionnaire (QUIS) and user testing for data collection. The QUIS questionnaire was adapted for this research purpose since it is a validated instrument for conducting comparative evaluations for software applications [66].

3.1 Rationale of Graphical Software Applications

There have been several popular graphical software applications in the market that integrate graphical user interface and programming scripting functions all-in-one for interface designer and software developer to work seamlessly for software development work. Adobe Flash CS4 and Microsoft Expression Blend 4 are selected for the study because they are competitor software specifically designed by Adobe™ and Microsoft to bridge the development platform for interface designers and developers/programmers. They combine GUI and CLI in the software applications and enable the developers and designers to work apart more effectively without losing each other's work in a software development process.

Generally, Adobe Flash and Microsoft Expression Blend are authoring tools that can be utilized to design and create presentations, software applications that act in response to user's interactions. Projects created by them can contain animation, video content and complex user interfaces. On one hand, they allow software developer to use Command Languages (a scripting language) to create functions and determine how the elements in the application act, and the code also allows adding interactivity and logic in a project.

3.2 Apparatus and Testing Facility

This study was conducted at User Interface Lab. The apparatus use for the user testing is a laptop (with a 14 inch monitor, 4 GB RAM, 2.20 GHZ CPU having 1280 * 800 display resolution) as a workstation, Windows Vista Home version as operating system, and Adobe Flash CS4 and Microsoft Expression Blend 4 as graphical software tools to carry out the task sets, and Camtasia Studio 3 was employed to record the screen capture for data analysis.

3.3 Participants and Tasks

32 participants were recruited with the background of interface design or programming for the study. The participants were tested individually. The user testing took around an average of 1 to 1.5 hours. Upon arrival, all participants were given a consent form before the experiment commenced. They were then asked to complete a demographic and software products experience questionnaire. Then, the participants were randomly given 10-minute trainings to learn the basic conventions of Adobe Flash CS4 and Microsoft Expression Blend 4. The participants are also given an average of 7 minutes to practice on their own and gain confidence before taking the tests. Subsequently, all participants were given a set of 7 tasks (Table 1) to perform in the software applications. If the participants were unable to complete a task, they were free to proceed without task completion. Figure 1 and 2 show an example of screen shot for Task 4 and Figure 3 and 4 indicate an example of screen shot for Task 6.

TABLE 1: Tasks During User Testing.

Task number	Adobe Flash CS4		Microsoft Expression Blend 4	
	Task type	Task explanation	Task type	Task explanation
1	GUI	Set background color	GUI	Set background color
2	GUI	Create text	GUI	Create text
3	GUI + CLI	Create animation for text	GUI	Create animation for text
4	GUI	Import image to the file	GUI	Import image to the project
5	GUI	Place image on screen, resize it and make it symbol	GUI	Place image on screen and resize it
6	CLI	Create Mouse Over event for image	GUI	Create Mouse Over event for image
7	CLI	Change image transparency	CLI (with suggestion)	Change image transparency

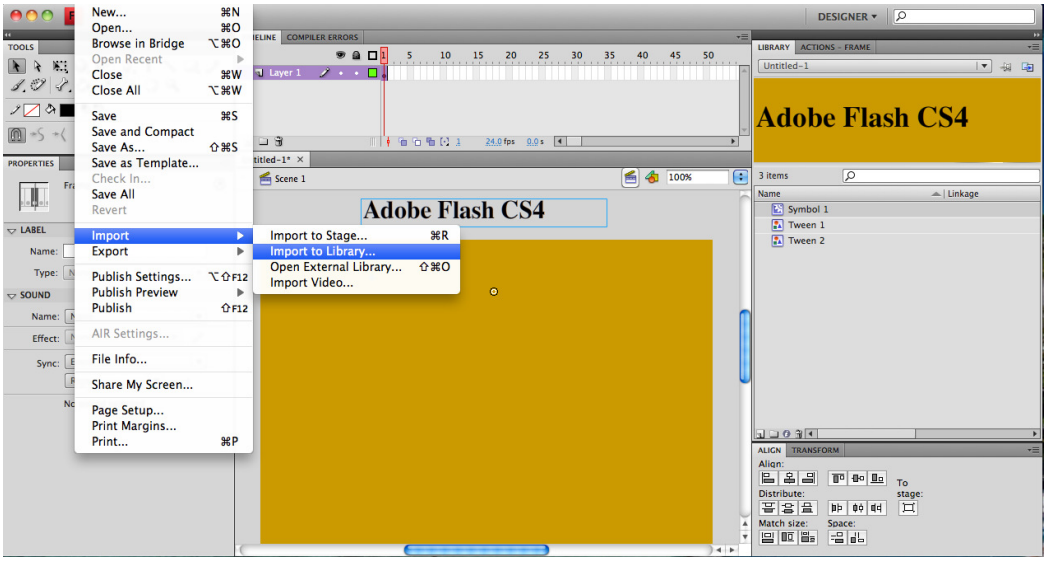

FIGURE 1: Task 4 (import image to the file) screenshot in Adobe Flash CS4.

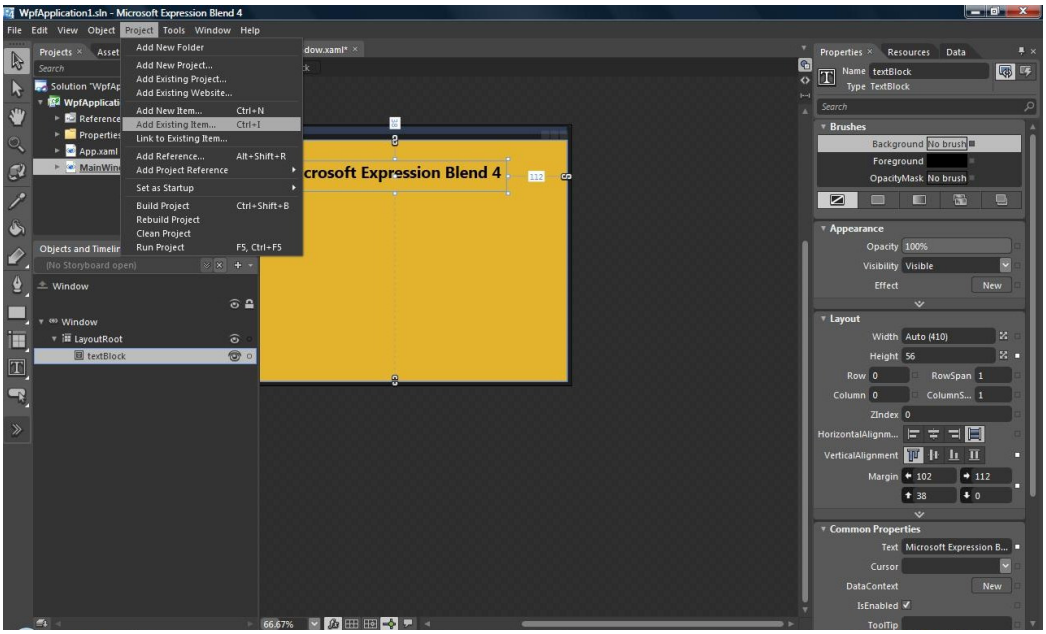

FIGURE 2: Task 4 (import image to the file) screenshot in Microsoft Expression Blend 4.

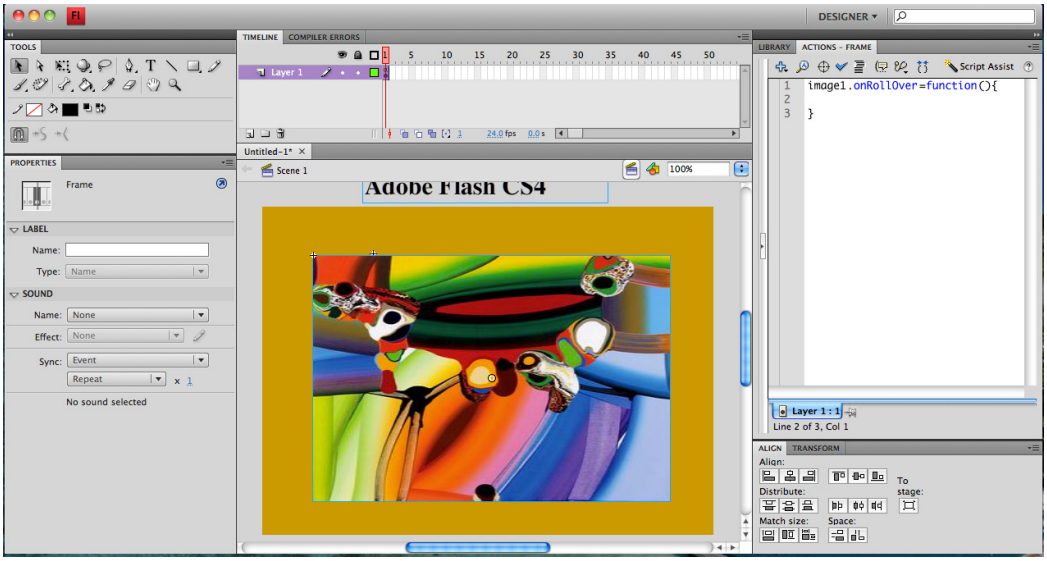

FIGURE 3: Task 6 (create mouse over event for image) screenshot in Adobe Flash CS4.

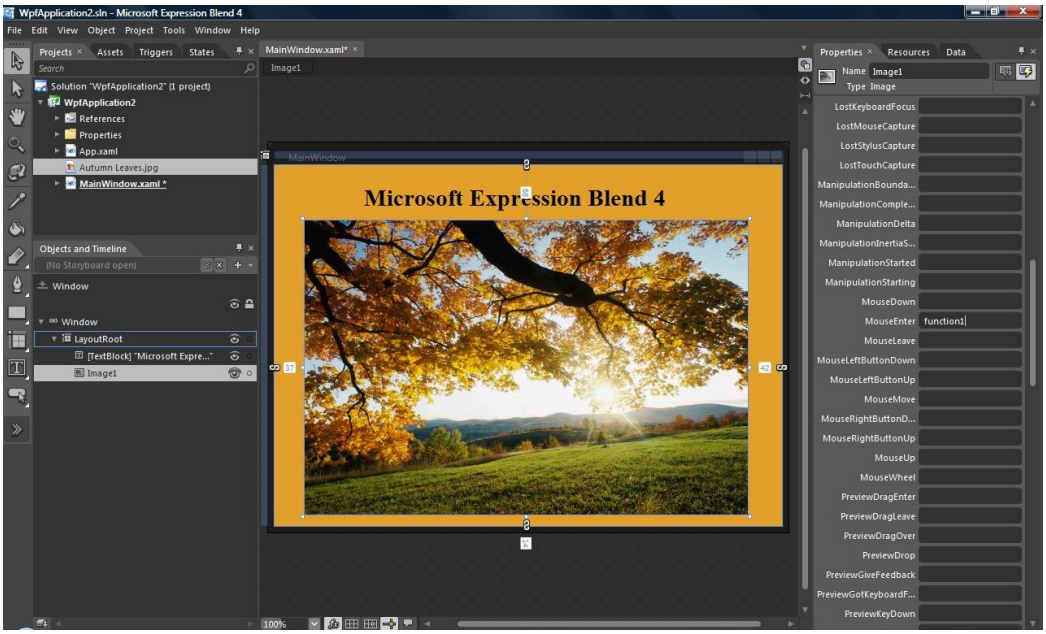

FIGURE 4: Task 6 (create mouse over event for image) screenshot in Microsoft Expression Blend 4.

3.4 Usability Metrics

The User Performance Variables for usability as follows:

- Effectiveness score: Percentage of successful completion for each task.
- Time taken: The total time spent to complete each task.
- Efficiency Rate: It is calculated by dividing effectiveness score by the time taken to do the task.
- Error: Any error made during performing each task.
- Help: Any help received during performing each task.

For subjective satisfaction, QUIS questionnaires measure users' subjective satisfaction using a 7-point semantic differential scale on the interfaces of both software applications. Each questionnaire covered items such as overall reaction, screen, terminology and software feedback, learning, and software capabilities.

4. RESEARCH RESULTS

The result is analyzed using a statistical analysis software, SPSS 16. The data gathered from the user testing were analyzed using descriptive analysis with mean and standard deviation. The rationale was to compare effects of two different GUI styles and CLI styles on learning Adobe Flash CS4 and Microsoft Expression Blend 4. The performance measure of each task is calculated for all users in the test.

4.1 User Performance Analysis

Among all 32 users, 17 (53.13%) participants were designers with interface design background while 15 (46.88%) participants were programmers from IT background. Users were from both genders (22 male and 10 female). Users were asked about their knowledge and usage of Adobe Flash and Microsoft Expression Blend, 28 (87.5%) individuals already had training for Adobe Flash while 4 individuals did not have (12.50%), and 5 (15.63%) individuals already had training for Microsoft Expression Blend while 27 (84.38%) individuals did not have any training.

One-Sample Kolmogorov-Smirnov test was conducted to examine normal distribution of effectiveness scores, duration, efficiency, number of errors and helps for all 7 tasks. Table 2 shows the results of user performance measure for Adobe Flash CS4 test. Table 3 indicates the result of user performance measure for Microsoft Expression Blend 4.

TABLE 2: A summary of usability testing results for Adobe flash CS4.

Task no.	Effectiveness (%)	Time Duration (Sec)	Efficiency	Error	Help
1	100	10.84	14.08	0.19	0.13
2	100	15.13	7.86	0.0	0.0
3	92.97	54.38	2.23	0.44	0.31
4	100	12.34	9.12	0.0	0.06
5	98.44	39.94	2.99	0.16	0.06
6	85.94	47.47	2.74	0.25	0.63
7	91.41	21.63	5.85	0.22	0.34
Mean*	95.54	28.81	6.41	0.18	0.22
SD**	5.54	18.06	4.3	0.15	0.22

Mean* indicates average for total 7 tasks.

SD*=standard deviation

The result in Table 2 shows that the effectiveness scores higher than 75% for all the 7 tasks (Mean for total task=95.54%, SD=5.54). However, the minimal effectiveness score is associated with CLI (Task 6 of creating Mouse Over event) and all GUI tasks are linked with maximum effectiveness score. On one hand, the minimum time duration taken to complete the tasks is 10.84 min (Task 1 of setting background color), which also means the task is fairly simple to achieve. On another hand, the maximum time duration is 54.38 min (for Task 3), which indicates the task is more complex and challenging to complete in terms of creating animation using GUI and CLI technique. The average time for 7 tasks is 28.81 minutes (SD=18.06).

In terms of efficiency, the total tasks score for 6.41 in average. Task 1 again achieves the highest efficiency rate of 14.08; however, the least efficient task to complete is Task 3 (2.23). This indicates that a task that combines GUI and CLI will take longer time for task completion. Apart from this, the average error rate for an overall task accounts for 0.18, which is considered minimal, and Task 3 shows the highest score of making mistakes. The number of help accounts for 0.22 for the total task, which is almost acceptable for task performance. The highest number of seeking help is Task 6, which shows the users are not familiar with creating a mouse over event for an image using CLI approach.

TABLE 3: A summary of usability testing results for Microsoft Expression Blend 4.

Task no.	Effectiveness (%)	Time Duration (Sec)	Efficiency	Error	Help
1	92.67	27	5.74	0.25	0.34
2	100	20.71	5.5	0.06	0
3	96.09	42.46	2.88	0.15	0.15
4	92.96	18.25	7.07	0.18	0.12
5	98.43	18.12	6.59	0.03	0.03
6	84.37	12.25	14.61	0.18	0.12
7	94.53	19.78	7.75	0.15	0.09
Mean*	94.15	22.65	7.16	0.14	0.12
SD**	5.09	9.76	3.63	0.07	0.11

Mean* indicates average for total 7 tasks.
SD*=standard deviation

The result in Table 3 demonstrates that the effectiveness score is higher than 75% for all the 7 tasks again (Mean for total task=94.15%, SD=5.09). However, the minimal effectiveness score is associated with GUI (Task 6 of creating Mouse Over event) and CLI task is nearly linked with average effectiveness score. The minimum time duration taken to complete the tasks is 12.25 min (Task 6 of creating Mouse Over event), meaning the task is rather effortless to achieve. But then again, the maximum time duration is 42.46 min (for Task 3), which indicates the task is intricate and challenging to complete in terms of creating animation via GUI. The average time for 7 tasks is 22.65 minutes (SD=9.76). Regarding efficiency, the total tasks score for 7.16 in average. Task 6 achieves the highest efficiency rate of 14.61; however, the least efficient task to complete is Task 3 (2.88). This shows that even a GUI task can be so much complex taking longer time for task completion. Apart from this, the average error rate for an overall task accounts for 0.15, which is considered minimal, and Task 1 shows the highest score of making mistakes (Mean=0.25). The number of help accounts for 0.12 for the total task, which is absolutely acceptable for task performance. The highest number of seeking help is Task 1, which shows the users are not familiar with setting background color using GUI approach in Microsoft Expression Blend 4.
Performance variables gathered from user testing was analyzed by parametric (Independent-Samples T-Test) and non-parametric (Mann-Whitney) tests to compare usability of the two applications. Below are the results:

4.1.1 Effectiveness

The effectiveness score of each task is calculated for every user in tests. Tasks 2, 3, 5, 6 and 7 did not show any significant difference ($p>0.05$) comparing two applications. Regarding Task 1: setting background color, Mann-Whitney test shows effectiveness score was significantly higher for Adobe Flash CS4 (Mean = 100) rather Microsoft Expression Blend 4 (Mean = 92.67), $U = 416$, $Z = -2.55$, $p = 0.011$. Regarding Task 4: importing image to the project, Mann-Whitney test indicates effectiveness score was significantly higher for Adobe Flash CS4 (Mean = 100) rather than Microsoft Expression Blend 4 (Mean = 92.96), $U = 432$, $Z = -2.3$, $p = 0.021$.

The total numbers of effectiveness for the two software applications were compared by Mann-Whitney test. The result showed no significant difference for Adobe Flash CS4 and Microsoft Expression Blend 4 ($U = 469$, $Z = -0.6$, $p = 0.54$).

4.1.2 Duration

Each task duration is calculated for every user for both tests. Tasks 3 and 7 did not show any significant difference ($p>0.05$) comparing two applications. Regarding Task 1: setting background color, Independent-Samples T-Test shows less time duration for Adobe Flash CS4 ($M = 10.84$, $SD = 11.65$) rather than Microsoft Expression Blend 4 ($M = 27$, $SD = 16.56$) condition; $t(55.65) = 4.51$, $p = 0.000$. Regarding Task 2: creating text, Independent-Samples T-Test shows less time duration for Adobe Flash CS4 ($M = 15.13$, $SD = 7.27$) comparing to Microsoft Expression Blend 4 ($M = 20.71$, $SD = 7.64$) condition; $t(62) = -2.99$, $p = 0.004$. Regarding Task 4: importing image to the project, Independent-Samples T-Test shows less time duration for Adobe Flash CS4 ($M = 12.34$, $SD = 4.62$) comparing to Microsoft Expression Blend 4 ($M = 18.25$, $SD = 10.58$) condition; $t(42.42) = -2.89$, $p = 0.006$. Regarding Task 5: placing image on screen, Independent-Samples T-Test shows less time duration for Microsoft Expression Blend 4 ($M = 18.12$, $SD = 8.94$) rather than Adobe Flash CS4 ($M = 39.94$, $SD = 18.25$) condition; $t(45.05) = 6.07$, $p = 0.000$. Regarding Task 6: creating Mouse Over event for image, Independent-Samples T-Test shows less time duration for Microsoft Expression Blend 4 ($M = 12.25$, $SD = 10.06$) comparing to Adobe Flash CS4 ($M = 47.47$, $SD = 30.11$) condition; $t(38.92) = 6.21$, $p = 0.000$.

The total time duration for the two software applications were compared by Independent-Samples T test. The result showed that time duration for Adobe Flash CS4 ($M = 201.72$, $SD = 68.44$) is higher than Microsoft Expression Blend 4 ($M = 156.50$, $SD = 49.71$); $t(56.58) = 3.02$, $p = 0.004$.

4.1.3 Efficiency Rate

Each task efficiency rate was calculated for every user for both tests. Tasks 3 and 7 did not show any significant difference ($p>0.05$) comparing two applications. Regarding Task 1:

setting background color, Independent-Samples T-Test shows higher efficiency for Adobe Flash CS4 (M = 14.08, SD = 6.89) comparing Microsoft Expression Blend 4 (M = 5.74, SD = 4.39) condition; t (62) = 5.76, p = 0.000. Regarding Task 2: creating text, Independent-Samples T-Test shows higher efficiency for Adobe Flash CS4 (M = 7.86, SD = 3.11) comparing Microsoft Expression Blend (M = 5.5, SD = 2.19) condition; t (62) = 3.5, p = 0.001. Regarding Task 4: importing image to the project, Independent-Samples T-Test shows higher efficiency for Adobe Flash CS4 (M = 9.12, SD = 3.11) comparing Microsoft Expression Blend 4 (M = 7.07, SD = 4.16) condition; t (62) = 2.23, p = 0.029. Regarding Task 5: placing image on screen, Independent-Samples T-Test shows less efficiency for Adobe Flash CS4 (M = 3.01, SD = 1.36) comparing Microsoft Expression Blend 4 (M = 6.59, SD = 2.85) condition; t (44.46) = - 6.39, p = 0.000. Regarding Task 6: creating Mouse Over event for image, Independent-Samples T-Test shows less efficiency for Adobe Flash CS4 (M = 2.74, SD = 1.96) comparing Microsoft Expression Blend (M = 14.61, SD = 11.02) condition; t (32.96) = -4.83, p = 0.000.

The total efficiency rates for the two software applications were compared by Independent-Samples T test. The result showed that efficiency rate for Microsoft Expression Blend 4 (M = 0.67, SD = 0.25) is higher than Adobe Flash CS4 (M = 0.53, SD = 0.17); t (62) = -2.63, p = 0.011.

4.1.4 Errors

Number of errors is counted for every user related to do every single task for both tests. Tasks 1, 2, 3, 4, 5, 6 and 7 did not show any significant difference (p>0.05) comparing two applications.

The total number of errors during tests for the two software applications were calculated and compared by Mann-Whitney test. The result showed no significant difference for Adobe Flash CS4 and Microsoft Expression Blend 4 (U = 473, Z = -0.55, p = 0.57).

4.1.5 Helps

Number of helps is calculated for every user seeking for assistance when performing every single task for both tests. Tasks 1, 2, 3, 4 and 5 did not show any significant difference (p>0.05) comparing two applications. Regarding Task 6: creating Mouse Over event for image, Mann-Whitney test showed the number of helps was significantly higher for Adobe Flash CS4 (Mean = 0.63) rather than Microsoft Expression Blend 4 (Mean = 0.12), U = 250, Z = -4.02, p = 0.000. Regarding Task 7: changing image transparency, the number of helps was significantly higher for Adobe Flash CS4 (Mean = 0.34) rather than Microsoft Expression Blend (Mean = 0.09), U = 398.5, Z = -2.18, p = 0.029.

The total numbers of helps for the two software applications were compared by Mann-Whitney test. The result showed the number of helps was higher for Adobe Flash CS4 (Mean = 0.22) rather than Microsoft Expression Blend 4 (Mean = 0.12); U = 347, Z = -2.31, p = 0.021.

4.2 QUIS Questionnaire Analysis

For subjective user satisfaction, data collected through QUIS questionnaire at the end of the test is summarized below (see Table 4). Users' satisfaction for Adobe Flash CS4 and Microsoft Expression Blend 4 software were measured on a 7-point semantic differential scale. Software applications were ranked by users from different aspects (i.e. overall software performance, screen, terminology and software feedback, learning and software capabilities).

TABLE 4: QUIS questionnaire result analysis

	Adobe Flash CS4		**Microsoft Expression Blend 4**	
Items	**Mean**	**Standard deviation**	**Mean**	**Standard deviation**
Category: Overall software performance				

TERRIBLE-WONDERFUL	5.16	1.19	5.94	0.88
DIFFICULT-EASY	4.63	1.34	5.34	1.21
INEFFICIENT-EFFICIENT	5.34	1.12	5.53	0.94
UNFRIENDLY-FRIENDLY	4.69	1.47	5.69	1.02
FRUSTRATING-SATISFYING	4.56	1.41	5.78	0.94
INEFFECTIVE-EFFECTIVE	5.59	1.07	5.72	0.89
RIGID-FLEXIBLE	4.87	1.43	5.94	1.22
Category: Screen				
ONSCREEN INFORMATION (Inadequate-Adequate)	4.91	1.63	5.16	1.27
USER INTERFACE ARRANGEMENT (Not organized-Organized)	5.34	1.28	5.47	1.48
EASY TO FIND FUNCTIONS (Never-Always)	4.38	1.48	4.56	1.46
READING CHARACTERS (Difficult-Easy)	5.38	1.10	5.34	1.49
SCREEN BACKTRACK (Difficult-Easy)	5.81	1.35	6.16	1.30
CREATING NEW PROJECT (Confusing-Very clear)	6.19	1.09	5.94	1.16
TOOLBAR ACCESS (Difficult-Easy)	5.91	1.28	5.75	1.32
Category: Terminology and software feedback				
SIMPLE AND NATURAL DIALOGUE (Never-Always)	4.74	1.29	4.94	1.32
TERMS USED IN THE SOFTWARE (Inconsistent-Consistent)	5.5	1.28	5.69	1.35
POSITION OF WINDOWS DIALOG BOX (Inconsistent-Consistent)	5.31	1.28	5.69	1.12
INFORMS ABOUT WORK PROGRESS (Never-Always)	4.48	1.71	5.41	1.36
ERROR MESSAGES (Unhelpful-Helpful)	3.97	1.97	5.62	2.07
PROMPT FOR DOING SCRIPTING INPUT (Confusing-Clear)	3.35**	1.64	5.66	1.54
Category: Learning				
SOFTWARE LEARNING (Difficult-Easy)	4.47	1.70	5.75	1.05
EXPLORING BY TRIAL AND ERROR (Difficult-Easy)	3.65*	1.70	5.19	1.78
REMEMBERING	4.13	1.56	5.5	1.46

COMMANDS (Difficult-Easy)				
PERFORMING TASKS IS SIMPLE (Never-Always)	4.50	1.19	5.5	1.11
HELP ACCESS OR DOCUMENT (Difficult-Easy)	4.25	1.8	6.28	1.61
HELP MESSAGES ON SCREEN (Unhelpful-Helpful)	4.31	1.97	5.94	1.13
Category: Software capabilities				
CORRECTING MISTAKES (Difficult-Easy)	4.13	2.06	5.31	1.60
DESIGNED FOR ALL LEVELS OF USERS (Never-Always)	3.19**	1.55	4.62	1.86
IMPORT AND EXPORT PROJECT IN AND OUT OF SOFTWARE (Difficult-Easy)	5.38	1.41	5.87	11.38
SOFTWARE RELIABILITY (Unreliable-Reliable)	5.37	1.27	6.03	1.15

**<3.5; *just above 3.5

For Adobe Flash CS4, the average user satisfaction for the overall software performance, screen, terminology and software feedback, learning and software capabilities are 4.99, 5.41, 4.72, 4.51 and 4.62 respectively. In general, the overall subjective user satisfaction for Adobe Flash CS4 scores (Mean=4.88) above average of >3.5 for all the above-mentioned 5 categories. However, under the category of 'Software capabilities', 'Designed for all levels of users (Never-Always)', it scored only 3.19, which is lower than the average level (3.5). Again, under the category of 'terminology and software feedback', 'prompt for doing scripting input (Confusing-Clear)' rates only 3.35 score, which is also lower than the average acceptance level (3.5). This user feedback is consistent and proven by the CLI performance result of having more difficulties of performing Task 6 and 7 that requires prompts for doing scripting input in Adobe Flash. The application reaches its highest convenience in terms of screen due to the familiar and clear menu labels, ease of toolbar access, organized interface arrangement and ease of reading characters. Furthermore, it obtains its lowest satisfaction in terms of software learning due to the difficulty of exploring by trial and error, remembering commands and help material.

Regarding Microsoft Expression Blend 4, the average user satisfaction for the overall software performance, screen, terminology and software feedback, learning and software capabilities are 5.71, 5.48, 5.5, 5.69 and 5.33 respectively. Overall, the subjective user satisfaction for Microsoft Expression Blend 4 scores (Mean=5.55) above average of >3.5 for all the categories. The application reaches its highest convenience in terms of overall software performance due to its flexibility, effectiveness, satisfactory and wonderful user experience. Furthermore, it obtains its lowest satisfaction scores in terms of software capabilities due to the difficulty of correcting mistakes, and lack of design for all levels of users.

5. DISCUSSIONS

This study evaluates the usability aspect in terms of user performance and satisfaction towards two graphical software applications, Adobe Flash CS4 and Microsoft Expression Blend 4, among user interface designers and software programmers. Users' performance in terms of effectiveness, efficiency, task duration, errors and number of help on working with GUI and CLI of Adobe Flash CS4 and Microsoft Expression Blend 4 were investigated in the usability testing. It showed that users could easily pick up the interfaces' functionalities when some training was given.

5.1. User Satisfaction

To examine user satisfaction in usability tests, QUIS questionnaires were used on Adobe Flash CS4 and Microsoft Expression Blend 4. The finding showed that Microsoft Expression Blend 4 and Adobe Flash CS4, both scored higher than average acceptance level. Although Microsoft Expression Blend 4 gained higher satisfaction rates in terms of overall software performance, terminology and software feedback, learning and software capabilities, they score almost the same in terms of screen. In regards of sub-categories, Adobe Flash CS4 gained higher rank than its counterpart for some 'screen' sub-categories (i.e. Reading Characters, Creating new project and Toolbar access).

5.2. User Performance

In terms of user performance variables for the usability test of both applications, CLI task in Adobe Flash CS4 was associated with higher duration, errors and helps and less efficiency rate comparing to its equivalent GUI in Microsoft Expression Blend 4. Another CLI task in Adobe Flash was linked with less effectiveness, efficiency and higher duration, errors and helps comparing to its equivalent CLI with suggestion task in Microsoft Expression Blend. A combination of GUI and CLI task in Adobe Flash again scored less effectiveness, efficiency and higher duration, errors and helps comparing to its equivalent GUI task on Microsoft Expression Blend. It is consistent with Wisher et al. [67] claiming that failing to remember just one of the essential facts leaves some tasks unachievable. This result is consistent with Wiedenbeck et al. [61] considering Direct Manipulation Interface easier to use as compared to menu-driven and CLIs. In addition, Shneiderman [6] claiming that users could track down information more quickly with GUI as compared to CLI. The result is also consistent with Gunderloy [68] stating that learning and using GUI software is easy and effortless, Schneiderman's [57] study on DMI interfaces and another investigation on benefits of menu-based interface rather than command-based interface [58].

Comparing GUI tasks in two applications, GUI tasks including same step of completion scored the same in terms of effectiveness and number of helps despite the differences in interface design for both software applications. However, finding the right icon to perform the task in Microsoft Blend took more time and is considered less efficient with higher number of errors due to the less efficacy icon design on Microsoft Expression Blend comparing to Adobe Flash. A GUI task on Microsoft Expression Blend having intricacy in completion scored less effectiveness and efficiency, but higher duration, errors and help comparing to the same GUI task with a standard design on Adobe Flash. Another GUI task on Microsoft Expression Blend performed by clicking on an unusual label of a menu, gained less effectiveness and efficiency, and higher duration, errors and helps compared to the same GUI task in Adobe Flash with standard and common menu label. Last but not least, a GUI task in Microsoft Blend consisting of fewer steps rather than its equivalent in Adobe Flash is associated with less duration, errors and helps and higher efficiency rate.

5.3. Comparison of User Task Performance for Both Graphical Software

Concerning Task 1 (Set background color) users simply need to change the color of background via the properties menu available to them in Adobe Flash CS4 workspace, While for Microsoft Expression Blend 4, they first have to select the background item in the object menu in order to activate the properties menu and complete the task. Comparing performance measures for Task 1 for both applications indicates that Adobe Flash is more successful since the completion of task requires less steps besides the fact that it made use of the ordinary method of performing such a task, therefore, individuals could learn to carry out the task more efficiently. However, Microsoft Expression Blend implementing two distinct menus in properties menu and switching between them using a small icon designed in the menu is confusing and baffling for the users.

Concerning Task 2 (Create text) users need to click on the text icon from the tools menu and start typing a text. Comparing performance measures for Task 2 for both applications shows that Microsoft Expression Blend was associated with lower efficiency due to the weak design of the icon. It took more time for users to find the proper icon on the related menu in Microsoft Expression Blend. Additionally, the rate of errors was higher in finding the text icon in the before-mentioned application.

Concerning Task 3 (Create animation for text) users are supposed to perform 5 steps in Adobe Flash (consisting of both CLI and GUI) and 3 steps in Microsoft Expression Blend to achieve a same result. Comparing performance measures for Task 3 for both applications reveals that Microsoft Expression Blend owns more usable design for this task. First of all, it calls for fewer steps. Secondly, it is done only via GUI. Thirdly, it made use of familiar labels comparing to confusing procedure, similar labels and options with different actions in Adobe Flash.

Concerning Task 4 (Import image) users need to find the proper label from the applications' menu. Comparing performance measures for Task 4 for both applications indicates that Adobe Flash again scored better due to the use of common labels while Microsoft Expression Blend is less usable as it offers variety of options through very similar labels.

Regarding Task 5 (Place image on screen, resize it) users gained better scores using Microsoft Expression Blend, as it requires fewer number of steps to achieve the result, while in Adobe Flash users made more mistakes due to the complex basic design feature of the application (i.e. necessity of converting every object to a specific kind of symbol in order to create events or animation). Therefore it is difficult for a user to get the basic idea of working with the software.

Concerning Task 6 (Create Mouse Over event for image) users need to write a line of command using Adobe Flash whilst they are just required to find the correct event label from list of events in properties menu using Microsoft Expression Blend. Comparing performance measures for Task 6 for both applications shows that Microsoft Expression Blend is more usable and efficient. Higher number of errors and helps for Adobe Flash arise from the fact that Individuals mostly forget pieces of script. Besides, writing a command line from memory takes much more time comparing to recognizing a label.

Concerning Task 7 (Change image transparency) that requires typing a command, users completed the task with fewer errors and help using Microsoft Expression Blend since it offers a list of suggested properties while typing a code. Adobe Flash was associated with less efficiency due to the unfamiliar property's name, syntax and scripting language.

Having examined the effects of participants' background study on Adobe Flash CS4 and Microsoft Expression Blend 4 tests, results showed no significant difference for interface designers and programmers on user performance variables; in other words, performing tasks in both GUI and CLI did not have significant difference on user performance among programmers and interface designers. More precisely, programmers carried out CLI tasks with the same performance as interface designers despite their background of study when the programming language is a new one for them. Moreover, investigating effects of prior knowledge of Adobe Flash and Microsoft Expression Blend did not show any significant difference on usability performance measures as well. This fact is associated with the skills decay after a long period of time and weak usability of both software applications as far as knowledge retention is concerned.

6. CONCLUSIONS AND FUTURE WORK

The goal of this research was to compare the impacts of different interface styles (GUI and CLI) on software applications for interface designers and developers. CLI were found to be more difficult to learn and less ease of use, even for software developers as well as designers. However, GUI was perceived to be simpler to learn for both groups; however, when it comes down to unfamiliar menu labels or icons difficulty to find, users can easily make the mistake of selecting wrong menu items. Moreover, for procedural tasks with higher number of levels, users more likely forget the series of steps due to the dependency to recall issues. For the software application to be usable for end-users, employment of familiar menu labels and toolbars, less number of steps necessary for accomplishment of tasks, and providing GUI equivalent for CLI tasks are highly recommended. Besides, experience of software will be more satisfying if it is designed for larger group of users not only professional ones. Utilizing commands without input prompting is where most of difficulties arise in usability of software. Nevertheless, we still witness development and implementation of many

new software applications by reputable companies undergoing the same flaws prevalent in the past.

Making use of standard icon designs or metaphors is of a great help for users to learn new software effortless, designing new metaphors for an application creates more memory load as opposed to the necessity of the reduction of memory burden to gain new knowledge. Arranging a group of related properties in one menu is highly recommended, whilst representing a great number of options in one menu creates a perplexing workspace for the user. Software interface designers are highly advised to plan a well-organized workspace without redundancy of options having distinct functioning. Users easily get mixed up with superfluity of information. They are also suggested to avoid similar labels providing various actions. Software designers are encouraged to employ the preceding logics implemented in earlier versions of applications instead of planning the whole new structure and forcing users to acquire entire new skills. Additionally, software developers are advocated to take advantages of GUI to the most possible extent as an alternative to CLI. In a situation that CLI cannot be evaded; employing CLI with suggestions is far more usable rather than CLI per se. However, in the context of GUI per se, efficient organization of options, using common labels and icons, procedure of performing tasks similar to popular software applications, and minimizing the complexity of performing tasks by reducing the number of required steps; are essential factors to achieve higher level of software usability.

All in all, software applications are basic tools for developers and interface designers to create new software. Therefore, it is important to ensure usability of software applications in terms of meeting the needs of their users. Since user interface is the basis for all interactions between users and applications. Thus, it is important that software applications have to be usable and provide more enjoyable experience that put users in control of interface and reduce their memory load.

Future studies will investigate various CLI or GUI applications by themselves. Looking at comparable CLI/GUI applications marketed by independent companies and studying the effects of their design styles on performance and retention of skill would be an essential affair for the next generation of software applications.

7. REFERENCES

1. A.H. Jorgensen and B.A. Myers. "User Interface History." Proceeding CHI '08 Extended Abstracts on Human Factors in Computing Systems, pp. 2415-2418, 2008.

2. P. Zhou and X. Fang. "Analysis of Cognitive Behavior in Software Interactive Interface." Computer-Aided Industrial Design and Conceptual Design, CAID/CD 9th International Conference, pp. 113-116, 2008.

3. C. Frankish, P. Morgan and R. Hull. "Recognition Accuracy and Usability of Pen-Based Interfaces." IEE Savoy Place. Vol. 126, pp. 7/1- 7/6, 1996.

4. J.A. Jacko and A. Sears. The Human-Computer Interaction Handbook: Fundamentals, Evolving Technologies and Emerging Applications. New York: Taylor & Francis Group, 2002.

5. M. Fetaji, S. Loskovska and B. Fetaji. "Software Engineering Interactive Virtual Learning Environment," in Proceedings of World Conference on Educational Multimedia, Hypermedia and Telecommunications. C. Montgomerie and J. Seale, Ed. Chesapeake., VA: AACE , 2007, pp. 939-944.

6. B. Shneiderman. Designing the User Interface: Strategies for Effective Human-Computer Interaction. Reading, MA: Addison Wesley, 1998.

7. The Linux Information Project. "GUI Definition". Internet: http://www.linfo.org/gui.html, [September 6, 2010].

8. A.O. Ajayi, E.A. Olajubu, D.F. Ninan, S.A. Akinboro and H.A. Soriyan. "Development and

Testing of a Graphical FORTRAN Learning Tool for Novice Programmers." Interdisciplinary Journal of Information, Knowledge, and Management, vol. 5, pp. 277-289, 2010.

9. International Standardization Organisation. "International Standard ISO/IEC 9126-1." Software Engineering—Product Quality—Part 1: Quality Model, 1st ed. pp. 9–10, 2001.

10. A. Seffah, M. Donyaee, R.B. Kline and H.K. Padda. "Usability Measurement and Metrics: A Consolidated Model." Software Qual J., vol. 14, pp. 159–178, 2006.

11. C. Benson, M. Muller-Prove and J. Mzourek. "Professional Usability in Open Source Projects: GNOME, OpenOffice.org, NetBeans." Proceedings of the CHI '04 Extended Abstracts on Human Factors in Computing Systems, pp. 1083-1084, 2004.

12. S. Davis and S. Wiedenbeck. "The Effect of Interaction Style and Training Method on End User Learning of Software Packages." Interacting with Computer, vol. 11, pp. 147–172, 1998.

13. R. Michalski, J. Grobelny and W. Karwowsk. "The Effects of Graphical Interface Design Characteristics on Human–Computer Interaction Task Efficiency." International Journal of Industrial Ergonomics, vol. 36. pp. 959–977, 2006.

14. S. Lauesen. User Interface Design: A Software Engineering Perspective. Addison Wesley: Reading MA, 2005.

15. Foviance. "Glossary" Internet: http://www.foviance.com/glossary/u/user-interface design, [December 20, 2010].

16. J.K. Burgoon, J.A. Bonito, B. Bengtsson, C. Cederberg, M. Lundeberg and L. Allspach. "Interactivity in Human–Computer Interaction: a Study of Credibility, Understanding, and Influence." Elsevier Science Publishers, vol. 16, pp. 553-574, 2000.

17. U. Jamil, T. Mustafa, A.R. Sattar, Shafia and F. Shahzad. "Cognitive Analysis of Software Interfaces." European Journal of Scientific Research, vol. 1, pp. 99-108, 2010.

18. X. Pan and Y. Lu. "Study of CAID Software User Interface Design Based on Usability." Computer-Aided Industrial Design and Conceptual Design, 9th International Conference, 2008, pp. 209-213.

19. K. Horvath and M. Lombard. "Social and Spatial Presence: An Application to Optimize Human-Computer Interaction." Psychology Journal, vol. 8, pp. 87–114, 2010.

20. R. Oppermann. "User-Interface Design." Internet: http://fit.fraunhofer.de/~oppi/publications/UserInterfaceLearningSystems.pdf, 2002 [September 6, 2010].

21. J. M. Carey. Human Factors in Information Systems: An Organizational Perspective. NJ: Ablex, Norwood, 1991.

22. G.A. Berg. "Human-Computer Interaction (HCI) in Educational Environments: Implications of Understanding Computers as Media." Journal of Educational Multimedia and Hypermedia. vol. 9. pp. 347-368, 2000.

23. A. Seffah and E. Metzker. "The Obstacles and Myths of Usability and Software Engineering." Communications of the ACM, vol. 47, pp. 71–76, 2004.

24. R. G. Bias and D.J. Mayhew. Cost-Justifying Usability: An Update for the Internet Age. San Francisco: Morgan Kaufmann, 2005.

25. IFIP Working Group 2.7/13.4. "On User Interface Engineering: Bridging the SE & HCI Communities" Internet: http://www.se-hci.org/bridging/index.html, [August 12, 2008].

26. E. Folmer, J. Group and J. Bosch. "Architecting for usability: A Survey." Journal of Systems and Software, vol. 70, pp. 61–78, 2004.

27. N. Juristo, A.M. Moreno and M. Sanchez-Segura. "Analysing the Impact of Usability on Software Design." The Journal of Systems and Software. vol. 80, pp. 1506–1516, 2007.

28. W. Dzida. "International User-Interface Standardization," in The Computer Science Engineering Handbook. J.A.B. Tucker, Ed. Florida: Boca Raton, CRC Press, 1997, pp. 1474–1493.

29. N. Bevan. "Human-Computer Interaction Standards." Elsevier Science B., vol. 20, pp. 349-354, 1995.

30. M. Macleod. "Usability in Context: Improving Quality of Use." Internet: http://www.nigelbevan.com/papers/mm- con94.pdf, [May 12, 2011].

31. IEEE Std. 1061. IEEE Standard for a Software Quality Metrics Methodology. New York: IEEE Computer Society Press, 1992.

32. J.D. Gould. "How to design usable systems," In Handbook of Human Computer Interaction. M. Helander, Ed. New York: Elsevier, 1988, pp. 757-789.

33. J.A. McCall, P.K. Richards and G.F. Walters. Factors in Software Quality. Springfield, VA: National Technical Information Service, 1977.

34. P. Booth. An Introduction to Human-Computer Interaction. USA: Lawrence Erlbaum Associates Publishers, Hillsdale, 1989.

35. D. Hix and H.R. Hartson. Developing User Interfaces: Ensuring Usability Through Product & Process. New York: John Wiley, 1993.

36. M. Porteous, J. Kirakowsky and M. Corbett. SUMI User Handbook. University College Cork: Human Factors Research Group, 1993.

37. M. Donyaee and A. Seffah. "QUIM: An Integrated Model for Specifying and Measuring Quality in Use," Eighth IFIP Conference on Human Computer Interaction, Tokyo, Japan, 2001.

38. B. Battleson, A. Booth and J. Weintrop, "Usability Testing of an Academic Library Web Site: A Case Study." The Journal of Academic Librarianship, vol. 27, pp. 188-198, 2001.

39. J. Sauro and E. Kindlund. "A Method to Standardize Usability Metrics Into a Single Score." CHI '05, pp. 401–409, 2005.

40. B. Shneiderman, Designing the User Interface, Strategies for Effective Human–Computer Interaction. USA: Addison Wesley, 2004.

41. ISO/DIS 9241-11. Ergonomic Requirements for Office Work with Visual Display Terminals (VDTs) Part 11: Guidance on Usability. 1998.

42. International Standard ISO/IEC 9126-1. Software Engineering—Product Quality—Part 1: Quality Model. 2001, pp. 9–10.

43. J. Nielsen. Usability Engineering. Boston, London: Academic Press, 1993.

44. N. Bevan and I. Curson. "Methods for Measuring Usability." Proceedings of the IFIP TC13 International Conference on Human-Computer Interaction, 1997, pp. 672-673.

45. B. Shneiderman and C. Plaisant. Designing the User Interface, Strategies for Effective Human–Computer Interaction. Reading, MA: Addison Wesley, 2004.

46. A. Raza, L.F. Capretz and F. Ahmed. "Users' Perception of Open Source Usability: An Empirical Study." Engineering with Computers, vol. 28, pp. 109-121, 2001.

47. S. Rosenbaum. "The Future of Usability Evaluation: Increasing Impact on Value." Maturing Usability, pp. 344-378, 2008.

48. R. Michalski, J. Grobelny and W. Karwowsk. "The Effects of Graphical Interface Design Characteristics on Human–Computer Interaction Task Efficiency." International Journal of Industrial Ergonomics, vol. 36, pp. 959–977, 2006.

49. International Business Machines. "The library for system solutions End User Interface reference." Internet: http://www.redbooks.ibm.com/redbooks/pdfs/gg244107.pdf, [May 13, 2011].

50. The Linux Information Project. "GUI Definition." Internet: http://www.linfo.org/gui.html, 2004 [September 6, 2010]

51. O. Robertson, D. McCracken and A. Newell. "The ZOG Approach to Man- Machine Communication." International Journal of Human-Computer Studies, vol. 51, pp. 279-306, 1999.

52. J. Gray. "The Role of Menu Titles as a Navigational Aid in Hierarchical Menus." SIGCHI Bulletin, vol. 17, pp. 33-40, 1986.

53. M. Soegaard. "Interaction Styles." Internet: http://www.interaction design.org/encyclopedia/interactionstyles.html, [November 18, 2010].

54. D. Higgins. "Widget." Internet: http://whatis.techtarget.com/definition/0,,sid9_gci213364,00.html, [September 6, 2010].

55. C. Faulkner. The Essence of Human-Computer Interaction. New York: Prentice Hall, 1998.

56. S. Passini, F. Strazzari and A. Borghi. "Icon-Function Relationship in Toolbar Icons." Elsevier, vol. 29, pp. 521-525, 2008.

57. B. Shneiderman. "Direct Manipulation: A Step Beyond Programming Languages." Computer, vol. 16,pp. 57-69, 1983.

58. B. Hasan and M.U. Ahmed. "Effects of Interface Style on User Perceptions and Behavioral Intention to Use Computer Systems." Computers in Human Behavior, vol. 23, pp. 3025–3037, 2007.

59. S. A. Davis and R.P. Bostrom. "An Experimental Investigation of the Roles of the Computer Interface and Individual Characteristics in Learning of Computer Systems." International Journal of Human–Computer Interaction, vol. 4, pp. 143–172, 1992.

60. F.D. Davis. "User Acceptance of Information Technology: System Characteristics, User Perceptions and Behavioral Impacts." International Journal of Man–Machine Studies, vol. 38, pp.457–487, 1993.

61. S. Wiedenbeck and S. Davis. "The Influence of Interaction Style and Experience on User Perceptions of Software Packages." International Journal of Human–Computer Studies, vol. 46, pp. 563–588,1997.

62. R. Gururajan and D. Fink. "A Study of Influences of Application Interfaces on End User Training Outcomes." Internet: http://www.informingscience.org/proceedings/IS2002Proceedings/pa pers/Guruj098Study.pdf, [May 17, 2011].

63. P.A. Brooks and A.M. Memon. "Automated GUI Testing Guided By Usage Profiles." ACM, pp. 333-342, 2007.

64. K.L. McGraw. "Performance Support Systems: Integrating AI, Hypermedia and CBT to Enhance User Performance." Journal of Artificial Intelligence in Education, vol. 5, pp. 3–26, 1994.

65. M. Virvou and K. Kabassi. "An Empirical Study Concerning Graphical User Interfaces that Manipulate Files," Proceedings of ED-MEDIA 2000, World Conference on Educational Multimedia, Hypermedia and Telecommunications, AACE, Charlottesville, VA., 2000, pp. 1724–1726.

66. J.P. Chin, V.A. Diehl and K.L. Norman. "Development of an Instrument Measuring User Satisfaction of the Human-Computer Interface," in CHI '88 Conference Proceedings: Human Factors in Computing Systems, Association for Computing Machinery. J. J. O'hare, Ed. New York, 1988, pp. 213-218.

67. R.A. Wisher, M.A. Sabol and J.A. Ellis. Staying Sharp: Retention of Military Knowledge and Skills(ARI Special Report 39). Alexandria, VA: U.S. Army Research Institute for the Social and Behavioral Sciences, 1999.

68. M. Gunderloy. Developer to Designer: GUI Design for the Busy Developer. Alameda, CA: Joel Fugazzotto, 2005.

A Method to Provide Accessibility for Visual Components to Vision Impaired

Azadeh Nazemi *azadeh.nazemi@postgrad.curtin.edu.au*
Electrical and Computer Engineering Department
Curtin University
Perth, WA, Australia

Iain Murray *i.murray@curtin.edu.au*
Electrical and Computer Engineering Department
Curtin University
Perth, WA, Australia

Abstract

Non-textual graphical information (line graphs, bar charts, pie charts, etc.) are increasingly pervasive in digital scientific literatures and business reports which enabling readers to easily acquire the nature of the underlying information [1]. These graphical components are commonly used to present data in an easy-to interpret way. Graphs are frequently used in economics, mathematics and other scientific subjects. In general term data visualization techniques are useless for blind people. Being unable to access graphical information easily is a major obstacle to blind people in pursuing a scientific study and careers [2].This paper suggests a method to extract implicit information of Bar chart, Pie chart, Line chart and math's graph components of an electronic document and present them to vision impaired users in audio format. The goal is to provide simple to use, efficient, and available presentation schemes for non textual which can help vision impaired users in comprehending form without needing any further devices or equipments. A software application has been developed based on this research. The output of application is a textual summary of the graphic including the core content of the hypothesized intended message of the graphic designer. The textual summary of the graphic is then conveyed to the user by Text to Speech software .The benefit of this approach is automatic providing the user with the message and knowledge that one would gain from viewing the chart.

Keywords: K**.** Vision Impaired, Bar Chart, Pie Chart, Line Chart, Function Graph, Text to Speech, GNUPLOT, GraphicReader.

.

1. INTRODUCTION

This Graphics are widely used in newspapers, text books, web pages, metro maps, instruction manuals etc. They provide significant cognitive benefits over text. These graphical components have an important role in conveying, clarifying, and simplifying information [3]. The majority of information graphics that appear in formal reports, newspapers, and magazines are intended to convey a message or communicative intention [4].Unfortunately, graphics are not easily accessed by vision impaired people. In visual form, they are not accessible to vision impaired. Vision impaired people unable to learn about the processes involved in reading, analysing, and interpreting information presented in data visual graphs, tables and charts which are frequently used in math and science materials to present and summarize data. It is fair to say that lack of access to diagrams and other graphical content significantly limits educational and workplace opportunities for people with vision impairment as well as access to popular media. This is in contrast with textual content in which assistive technology have improved access [5]. Traditionally charts are used to display trends and relationship, and also can be used to communicate processes or display complicated data simply. These charts may be designed for the experts

trained users for data visualization or in popular media without complicated scientific reasoning [6]. Charts are typically intended to convey a message that is an important part of the document and this information generally not repeated in the article [7]. Thus people who are unable to see and understand this chart will lose important part of information.

Several approaches have been addressed the accessibility of chart using alternative methods. A number of projects have attempted to make graphic components accessible to vision impaired users by reproducing the image in an alternative medium, such as sound [8], touch [9] or a combination of the two [10][11].Viable alternatives include generation of a tactile graph, delivering the information in text, interaction with an audio graph, or a combination tactile/audio approach [12]. Non-textual components can be made accessible to the vision impaired in verbal description or audio format and Haptic(tactual) feedback. These techniques help to express the data being graphed in non visual ways. Synthesized speech and Braille are two common examples of conveying information through the auditory and tactile senses [2]. Current approaches have limitations such as the cost of translating into an accessible graphics format, use of expensive tactile graphics or expensive peripheral devices, or lack of congruence with the original visual graphic[5].Traditionally, graphs and diagrams are presented in Braille, or raised dots and lines on the swell-paper [13]. Tactile graphics are images that use raised surfaces and vision impaired user can feel them. They are used to convey non-textual information such as maps, paintings, graphs and diagrams. Tactile graphics can be seen as a subset of accessible images [14].

Several problems are associated with Braille and Tactile technique:

- Only a small proportion of blind people can use Braille, because reading it requires sufficient tactile sensitivity which all vision impaired do not have [15].
- Blind people can only get a rough idea about the content [13].
- Tactile diagrams are not durable. It is not easy to make changes to tactile diagrams.
- Reheating the swell paper could overcook the already raised parts and reduce the quality [16].
- In the method ,making graphs tactile Picture in Flash (PIAF)[21], the general shape of the graph can be understood by touching it carefully, but hardware is needed to generate tactile charts. Two main effective restriction factors in this method which must be considered for tactile symbols in charts are :Discriminating ability and Searching ability [22]. Tactile symbols without these properties could not help to explore concepts of charts by vision impaired people.

In addition, Haptic feedback which is more useful for guidance and assisting users' navigation on the graphs is not efficient to present exact data values to the user. Moreover, if too much information is presented via Haptics, the narrow bandwidth can be easily overloaded [2]. Also it may take users some time to familiarize themselves with the new interface. The limitations of the force feedback devices also hinder users exploration on the graphs[18]. Vocalizing the form of the graph and individual data is useful as long as a screen reader is provided to interpret trends in the data, however, the shape of the chart has to be imagined in the user's mind[20].

In addition other methods have some limitation such as:

- Conveying data via sound pitch and 2-dimensional acoustics[19].It is difficult to convey data accurately with the acoustic method(non speech sound), and moreover, since acoustics are volatile, information can easily be misheard.

- The Interactive SIGHT (Summarizing Information GrapHics Textually) system provides visually impaired individuals with the high-level knowledge that one would gain from viewing. SIGHT uses image processing techniques to extract communicative signals from a chart, but it is still limited to present information of the bar chart within the WebPages [4].

2. METHODOLOGY

There are two categories for methods to present visual component to vision impaired:

- Passive (description tool):The user is presented with a representation of the entire visual part at one time, with limited user input.

- Active (exploration tool):The user can explore non textual part [23].

The offered method in this paper has been attempted to combine two mentioned categories. It means this method presents a completed description of illustration (passive) and provides navigation ability through various field of data to give user opportunity to explore and build a mental map of the visual components (active).

On the other hand, providing an alternative equivalent for inaccessible content is one of the primary ways authors can make their documents accessible to people with disabilities [18], providing a hierarchy of text equivalents can also convey the hierarchical structure of the graphic components [3]. The alternative content fulfills essentially the same function or purpose for users with disabilities as the primary content does for users without any disability. Text equivalents are always required for graphic information [3]. Therefore this research has focused to extract these text equivalents from graphic components where they are not provided by authors.

For implementation this method, accurate understanding of the ways in which non-textual parts benefit sighted people, together with the problems encountered when exploring them non-visually, are required [24]. Also understanding of nature, meaning and concepts and obtaining comprehensive information from graphics must be considered to present detailed descriptions. The factors which appear to affect graphical comprehension are the purposes for using graphs, task characteristics, discipline characteristics, the differences between presenting information visually and aurally, and how and what graphical parts are represented in the mind[25]?

Most significant benefits for readers of visual parts are reducing demands on short-term memory and gaining a quick overview [26] which have been considered in aural representation in this method.

In general term each chart structure contains two parts:

1. Text part including :the number and amount of each axis ticks, axis labels ,legends and title.

2. Image part including: graph body.[22].

To provide non textual graphical components accessibility to vision impaired users it is essential to obtain comprehensive information from image .In addition advanced techniques for the graphic understanding are needed [27].Since large quantities of data are presented in a graphic, thus efficient access strategies are required according to the tasks that users are trying to accomplish during navigation through information.

2.1 GraphicReader Software Application

GraphicReader is a software application which has been developed in order to provide real-time and independent access to the vast repository of information in the graphics (bar chart, line chart,

pie chart and math's graph), with automated image processing, math processing, decoding, and extracting information which these graphical components convey and emphasis. After extracting data as text, the system analyze them to generate XML format which is convertible to audio format and used to provide easy navigation ability to vision impaired people .The application integrates appropriate representation for information accessibility.

This main core is Visual Extraction Module (VEM) and regarding the charts structure contains two Sub-Modules:

1. Text data extraction by performing accurate Optical Character Recognition (OCR) .Before OCR some preparation steps must be done to clean the text background and enhance the text .Theses processing steps are:

- 90 degree rotate if aspect does not match layout
- Crop
- convert to grayscale
- enhance filter to clean background and remove noise
- text smoothing
- Sharpening
- saturation change
- alternate text smoothing via adaptive blur

After OCR , running Text Analyzing Module or Data Interpretation leads to gain access to chart title, axes labels, the number and value of ticks ,and number of slices.

2. Image data extraction contains graph digitization to provide data table .Data table is a table with two columns to shows amount of each dependent variable related to its independent variable.

After data extraction and data collection, XML generating Module generates XML code based on all collected data and considering the result of Data Interpretation Module. The availability of markup format for the description of a chart helps to concentrate on the development of presentation strategies which comprehensively expresses the original chart in audio format without needing further peripherals.

By completing XML code user is passively presented key statistical information about the presented chart. For users who do not need detailed description, an static description should be generated for a quick access.

Since any technique that aims to provide non-visual accessibility of charts needs to support the processes of mental model formation and manipulation, at this stage user can easily navigate the chart and has the benefit of both *passive* and *active* presentation .It means the user is guided by the system in navigating the content of the chart or can control the presentation by deciding what parts of the charts to focus on. The main important features of navigation ability in GraphicReader, are reducing the demands on memory and reassuring user to explore the chart meaning.

Thus depending on user navigation request which is send to system through user interface different classified data could be extracted from XML representation, converted to plain text and sent to TTS module to make audio representation.

In addition XML representation is converted to a .dat file and .gnu configuration file and sent to GNUPLOT module to generate accessible graphical component using classified data in XML representation. In fact this module is a feedback module and by comparing graphical component produced by GNUPLOT and original image, accuracy of system can be estimated and errors are found and corrected. Figure 1 illustrates the overall architecture of the GraphicReader access system.

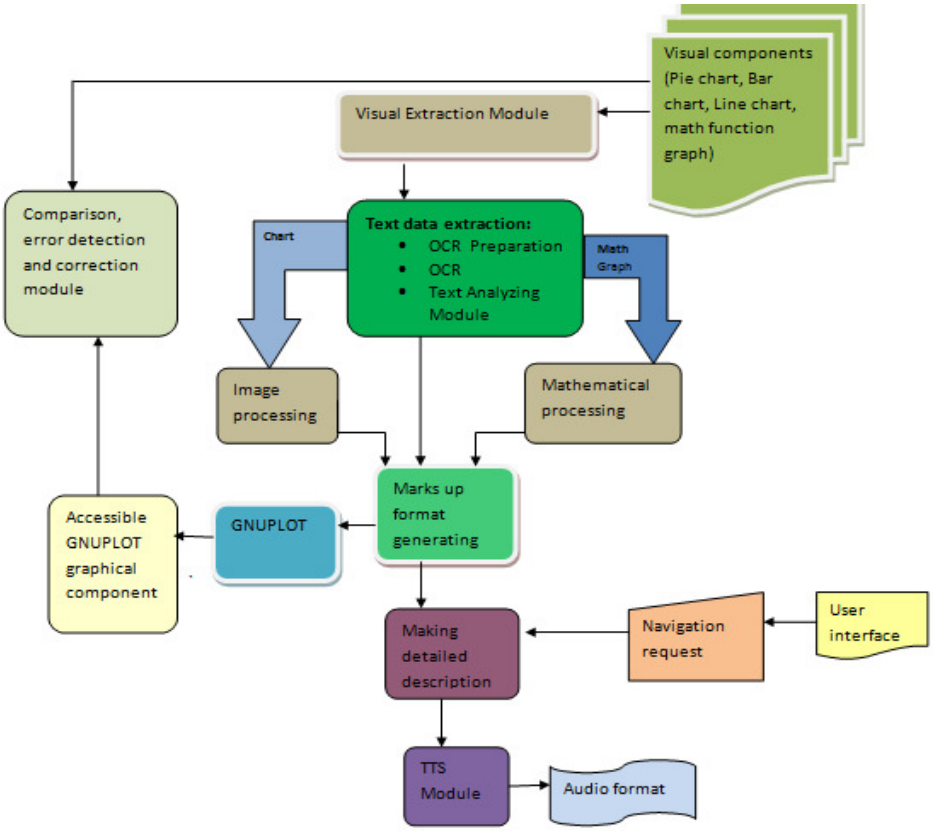

FIGURE 1: GraphicReader access system flowchart.

2.2 GNUPLOT

GNUPLOT is an open source command-line program that can generate two- and three-dimensional plots of functions, data, and data fits. It is frequently used for publication-quality graphics as well as education. One major accessibility benefit derived from GNUPLOT is encoded as plain text. Authors can create and edit it with a text-processor authoring tool. Plain text encoding also means that people may use relatively simple, text-based. This can help users with visual impairments, and can be used to supplement graphical rendering. GNUPLOT presents prepared, simple, text based and easy to use pattern to generate math's function graph, bar char and pie chart.

3. EXAMINED VISUAL COMPONENTS

3.1 Bar Chart

The Bar charts are one of the most commonly used visualization techniques and are often encountered daily, e.g. on newspapers, journals and magazines. Bar chart contains rectangles

which the height indicates the quantities of the corresponding variables. They usually show discrete and independent variables.

For a bar chart, the representation includes the number of bars in the graph, the labels of the axis, and information for each bar such as its label, the height of the bar. Some emphasized information implicit in bar chart are:

- The highest bar on the graph.
- The lowest bar on the graph
- Two bars which have the closest values.[16].

GraphicReader currently handles electronic images contain Bar chart produced with no overlapping characters. Also it hypothesizes standard placement of labels and axis headings. VEM for Bar chart contains several image processing steps such as OCR.

In order to extract vertical and horizontal axis labels and number of bars image must be sent to OCR engine.

Figure 2 illustrates the sample of a bar chart which indicates the "Average Price of Advertisements of five different channels Grey scale mono graph ,image of vertical lines and ,extracted text from OCR.

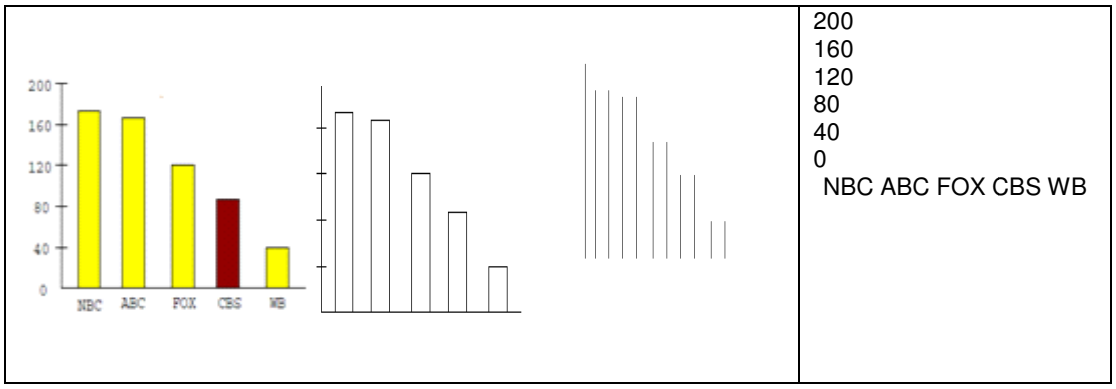

FIGURE 2: Sample of Bar chart, mono color image, vertical lines image, , OCR output.

Output Text file generated by OCR helps to recognize number of bars (in this example is five), labels of bars (NBC, ABC, FOX, CBS, WB) and scaling value on vertical axis(between 0 to 200 by step value 40). By converting the chart to grey scale mono graph, removing horizontal lines, splitting image of vertical lines to tiles and using histogram of each tile, number of black pixels in each bar has been obtained, the height of each bar can be calculated by comparison with vertical axis maximum value and its number of pixels. For example vertical axis has value 200 by 177 black pixels, thus the value of NBC bar by having 153 black pixels is 172 .Table1 shows relation between value of each bar and its number of black pixels.

Tile no	No of black pixels	Height of Bar
0	330	Vertical axis +NBC
1	153	NBC=200*153/177=172
2	147	ABC=2000*147/177=166
3	147	
4	106	FOX=200*106/177=120
5	106	
6	76	CBS=200*76/177=86
7	76	
8	34	WB=200*34/17=39
9	34	

TABLE 1: Height of bars in the sample bar chart

The following is XML codes generated by XML generating Module using data extraction from VEM:

<bar-chart>
<horizontal-axis-label>”Channel” </horizontal-axis-label>
<vertical-axis-label>”Millions of Dollars” </vertical-axis-label>
< vertical-axis-range >”0:200” < /vertical -axis-range >
<vertical-axis-tic>”40” </vertical-axis-tic>
<x,y>NBC,172</x,y>
<x,y>ABC,166</x,y>
<x,y>FOX,120</x,y>
<x,y>CBS,86</x,y>
<x,y><WB,39</x,y>
</bar-chart>

After XML generation, the system produces .gnu configuration file and .dat file to provide GNUPLOT package required files such as the following:

```
#Bar.gnu
reset
set output "bar.png"
set ylabel " Dollars"
set ytics "40"
set style fill solid
set term png truecolor
set xlabel "Channel"
set yrange [0:200 ]
set boxwidth 0.5
 plot "data.dat" using 1:3:xtic(2) with boxes
#Bar.dat
0    NBC   172
1   ABC    166
2   FOX    120
3   CBS     86
4   WB      39
```

Figure 3 shows an accessible image which is output result of GNUPLOT package. This image can be replaced by original image for more usage and getting more efficient and quicker results in terms of accessibility. Comparison this output image and inaccessible original input image has shown a little difference in value of bars height.

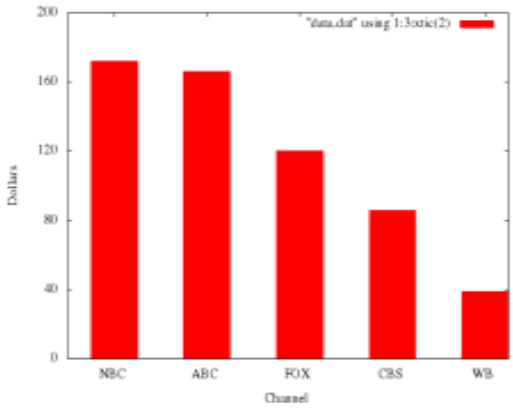

FIGURE 3:GNUPLOT output Bar chart.

3.2 Pie Chart

A pie chart is a circular chart divided into sectors, illustrating proportion. In a pie chart, the arc length of each sector (and consequently its central angle and area), is proportional to the quantity it represents [28] The symbols are the pie sectors. Each pie sector has to be matched with the variable via the legends. By the matching algorithm, the underlying data can be interpreted. VEM for pie chart supposes which legend information are in descending order.

Figure 4 illustrates the sample of a pie chart which indicates "Annual Market Share Analysis for three different products", extracted text from OCR and color histogram information extracted by image processing.

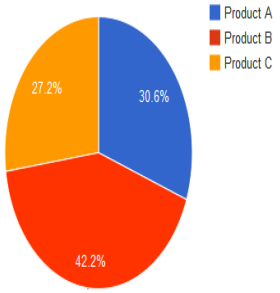	Product A
	Product B
	Product C
		17029: (255, 52, 1,255) #FF3401
		rgba(255,52,1,1)
		12654: (53,101,201,255) #3565C9
		rgba(53,101,201,1)
		10898: (255,154, 2,255) #FF9A02
		rgba(255,154,2,1)

FIGURE4: Sample of Pie chart, OCR output, Histogram output.

As it is observed from histogram information the total number of colored pixels is 17029+12654+**1**0898=40581.Table 2 has shown the percentage of each product.

Slot	Percentage	Accuracy level
Product A	17029*100/40581=41.96%	99.4%
Product B	12654*100/40581=31%6.	98.7%
Product C	10898*100/40582=26.85%	98.7%

TABLE 2: Slots percentage of sample pie chart.

XML codes generated by Marks up Format Generating Module are as the following:

<pie-chart>
<title>" Annual Market Share Analysis for three different products"" </title>
<no of slots l>"3" </no of slots>
< slot-1 >"Product A,41.96%" < /slot-1 >

< slot-2 >"Product B,31%" < /slot-2 >
< slot-3 >"Product C,26.85%" < /slot-3 >
</pie-chart>

3.3 Line Chart

Line chart displays information as a series of data points connected by straight line segments [29]. It is a basic type of chart common in many fields. It is created by connecting a series of points that represent individual measurements with line segments. A line chart is often used to visualize a trend in data over intervals of time [30]. A line chart is typically drawn by two crossed lines, called axes. The horizontal axis is called the x-axis which represents the independent variable and the vertical axis is called the y-axis which *represents* the dependent variable. The line chart may contain additional lines drawn parallel to both axes which is called grid. Each axis indicates significant values with small marks, called ticks. A short description of the axis is label The chart overall description is title.

For line chart, Graph digitization is a process involves converting the pixels in the image of line chart to original (x, y) data values. Extracted original (x, y) data values from the image helps to generate digitizing data table [31].

Therefore graph digitization has two responsibility, First finding the position of pixels of the main graph in the image and then running mathematical function for all found pixels to assign a coordinate system to the pixels in the graph based upon the extracted scaling values, line chart image size ,and line chart origin point pixel position. Although the basic concept of converting the image pixel values to scaled values is straightforward, practical consideration such as specifying accurate origin point must be addressed.

- Finding graph pixels position process includes several image processing steps as the following :
  - Resizing image to 640x480
  - Removing all gridlines it is essential to discriminate grid lines from the axes.
  - Obtaining graph image without axes and text by using color separation and color histogram information
  - Conversion graph only image to grayscale
  - Flipping image to move origin point from top left corner to bottom left corner
  - Negation graph image
  - Converting image to a text file including information of each pixel location
  - Removing all black pixels information from text file .Remaining pixels are body graph pixels
  - Sorting data in text file based on first column(x values)

- Conversion graph pixels position to original (x, y) data value depends on finding origin point pixel position, obtaining X-ratio and Y-ratio. To approach this goal, following step must be done:
  - Obtaining axes image without graph using color separation and color histogram information
  - Removing axes ticks and axes labels and generating an image which contains only horizontal axe.
  - Converting horizontal axe image to a text file including information of each pixel location in horizontal line
  - Removing all white pixels information from text file related to horizontal axe
  - Obtaining position of origin by finding Minimum X value and Maximum Y value.(which Xo, Yo are origin coordinate values, and W,H are weight and height pixel sizes of image)
    Xo= Minimum_X_ value
    Yo=480- Maximum_ Y_ value
    W= Maximum_X_ value- Minimum_X_ value+1

H= Maximum_Y_ value- Minimum_ value+1
X-ratio=amount of X_ticks*number of X_ticks/W= Max-x-Scale /W
Y-ratio=amount of Y_ticks*number of Y_ticks/H= Max-y-Scale /H

- Conversion pixel to data has been done using following formulae:
  (which Xp,Yp are pixel position coordinate values, and X,Y are original value in chart)
  X= (Xp –X`o)*X_ratio
  Y= (Yp-Yo)*Y_ratio
- Converting X floating values to integer using function "floor(x)" as the following :
  If x<0: then return int(x)-1
  If x>0: then return int(x)
  Function floor(x) is used except for last X value which is used "ceil(x)" function to convert to integer.
  If x<0: then return int(x)
  If x>0: then return int(x) +1
  Converting Y floating values to integer using function round(y)

After performing all these processes , digitizing data table of different value for X ,Y has been generated and with presenting numerical data in the form of a line chart relative values ,relative ratios and interpretation of trends , can be understood instantly. Also it should be useful in addition X, Y, slope of the individual line between two adjacent points has been calculated and added to data table. It should be calculate from this formula:

$$\text{Slope\_i} = ((Y_i - Y_{i-1}) / (X_i - X_{i-1}))$$

Calculating slope between two points helps to provide better summary description for line chart trends using the following rules:

- If a line segment slopes upward from left to right, the slope is positive.
- If a line segment slopes downward from left to right, the slope is negative.
- If a line segment is horizontal, the slope is 0 (zero).
- If a line segment is vertical, the slope is undefined.

Most considerable factors which must be described in a line chart are as the following:

- Counting the number of lines.
- Identifying the line friction.
- Counting the number of bends on the line
- Counting the number of intersections

Table 3, indicates a table of original values which generate the sample line chart.

X	Y
0	5
1	7
2	3
3	11
4	15
5	13
6	9
7	14

TABLE 3: Original values of the sample line chart.

Figure 5 illustrates the sample line chart generated by above

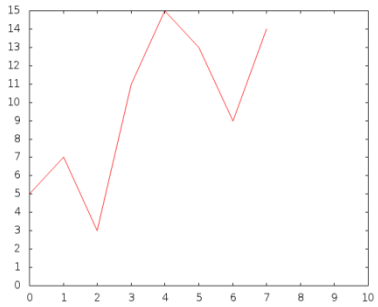

FIGURE 5: The sample line chart .

The results of first VEM sub module are :
Data extracted Module 1:Max-x-Scale=10 ,Max-y-Scale=15
The result of second VEM sub module contain origin point pixel position and graph dimension are: Xo=60 Yo=22 W=547 H=420

And finally table 4 shows the result of Graph Digitization before converting to integer values:

X	Y
0.18	5
1.05	6.92
2.10	3.035
3.16	10.96
4.03	14.85
5.09	12.89
6.14	9
6.98	13.98

TABLE 4: Data extracted from line chart by GraphicReader.

The generated XML codes regarding extracted data table from sample line chart comes below:

<line-chart>
<title>” “</title>
<horizontal-axe-label>”X” </horizontal-axe-label>
<vertical-axe-label>”Y” </vertical-axe-label>
< horizontal-axe-range >” 0: 10” < /horizontal-axe-range >
< vertical-axe-range >” 0 : 15” < /vertical -axe-range >
< x_tick>”1”</x_tick>
< y_tick>”1”</y_tick>
<x0,y0>”0,5”</x0,y0>
<x1,y1><”1,7”></x1,y1>
……
<x7,y7><”7,14”></x7,y7>
<slope0>”2”<.slope0>
<slope1>”-4”<.slope1>
….
</line-chart>

The application provides users the opportunity to select several presentation options such as chart title, average point, maximum point, minimum point, start point, endpoint, or trend of the chart points. The user provides input to the navigation process through keyboard controls.

3.4 Graph of math’s function

Graphs are visual presentation for math information and useful to illustrate qualitative information, which are sometimes even impossible to describe them .A graph is a picture that shows how sets

of data are related to each other. Generally structure framework of graph of the function contains two crossed axes which meet on origin point. The axes divide the plane into four quadrants. Scaled values are increased from left to right on horizontal axe and respectively from down to up on vertical axe. For describing graph of the function labels , scaled value and the range of each axes, where the function grows or decreases, where the maximums, the minimums and the flexes are and where the points of discontinuity are must be considered. Also following factors should be included in graph description [14][16].

- Meet points of the graph with axes: These points are obtained by assigning zero to "x "and finding "y "value and assigning zero to" y" and finding "x". This step requires solving the equation .For example one method for this purpose is using :
  Quadratic Formula : ax2 + bx + c = 0 value of x is given by:

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

- Inflection point: The function is continuous at the point and the concavity of the graph changes at that point. This point is found by make the y", zero.

- Starting and ending points: These points are obtained by assigning minimum and maximum values indicated on vertical axe to y and finding x.

- Minimum and maximum points: These points are obtained by assigning zero to y' and finding x at this point :

  - If y ">0, then the point is maximum
  - If y "<0, then the point is minimum
  - If y"=0 ,then the point can be both or neither

- Shape: Regardless details, shape of graph function depends on function, so in most cases general shape of graph can be guessed by having function. Some rules to find shape of graph are as the following:

  - If y=ax+b ,then the shape is a line
  - If y=ax2 + bx + c or quadratic functions ,then shape of graph is parabola
  - If 'a' is positive, the graph will be 'U' shaped. If 'a' is negative, the graph will be 'n' shaped. The graph will always cross the y-axis at the point c (so c is the y intercept point).
  - If for an specific interval x2>x1 and y2>y1 ,or y'>0 ,then in this interval graph is increasing
  - If for an specific interval x2>x1 and y2<y1 ,or y'<0 ,then in this interval graph is decreasing
  - If for an specific interval y'=0,then in this interval graph is constant
  - If f(x)=f(-x) ,then graph has symmetry regarding y or even
  - If f(x)=-f(-x) ,then graph has symmetry regarding origin or odd
  - If y">0 then graph is concave up
  - If y"<0 then graph is concave down

Figure 6 illustrates a sample for graph function using y=x^2+2*x as function formula.

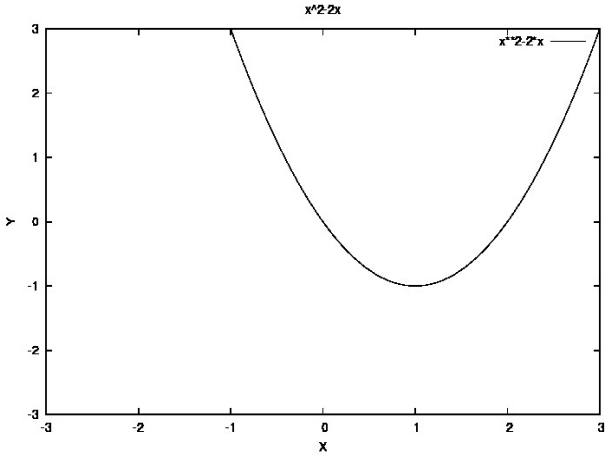

FIGURE 6. Sample of math's graph function sample.

Table 5 is the data table of above graph obtained from performing Data Digitization using mathematic processing.

TABLE : Extracted data from math's graph function.

X	Y
-3	15
-2	8
-1	3
0	0
1	-1
2	0
3	3

XML codes are generated by GraphicReader using information extracted from mathematical analyzing.:

</graph>
<shape>"parabola"</shape>
<horizontal-axis-label>"X" </horizontal-axis-label>
<vertical-axis-label>"Y" </vertical-axis-label>
< horizontal-axis-range >"[-3:3]" < /horizontal-axis-range >
< vertical-axis-range >"[-3:3]" < /vertical -axis-range >
<horizontal-axis-tic>"1" </horizontal-axis-tic>
<vertical-axis-tic>"1" </vertical-axis-tic>
<equation>"x**2-2*x"</equation>
<vertical -intercept>"0,0"</vertical-intercept>
<horizontal-intercept>"0,0"</horizontal-intercept>
<horizontal-intercept>"2,0"</horizontal-intercept>
<starting -point>"-1,3"</starting -point>
<ending-point >"3,3"</ending-point>
/<minimum-point>"1,-1"</minimum-point>
</graph>

Description results of GraphicReader application using mathematical analyzing and data s

- Horizontal axis is labeled X and its range is between -3 to 3
- Vertical axis is labeled Y and its range is between -3 to 3
- Horizontal and vertical step value is 1
- This graph is plotted from the equation: y= x^2 -2x
- The graph is started from point (-1, 3) from top left quarter of plane until (3,3) located on top right quarter of plane :

$$x^2 - 2x = 3 \rightarrow x = 3 \quad x = -1$$

x^2-2x=-3 → no answer

- Horizontal intercept occurs at points 0,0 and 2,0(y=0)
- Vertical intercept occurs at point 0, 0 (x=0)
- Shape of graph is parabola and has minimum (a=1>0) value at (if y'=0→ 2x-2=0, then x=1, y=-1) point (1,-1)

4. COMAPRITIVE EVALUATION

Comparison between this research method (using synthesis speech) and tactile representation for non textual components to vision impaired users shows that tactile representation requires manual preparation, a peripheral device and user training . Additionally, these original visual components must have some properties that may be transposed to a tactile representation thus limiting the possible attributes that may be included. For example;

- Pie charts must be enlarged to display the divisions of the circle. If numbers are shown, they should be preceded by the numeric indicator (number sign) regardless of whether the text is transcribed in literary or Nemeth (mathematics and science) code. The outline of the circle is tactually difficult to discern from the lines separating the divisions. A few of these lines may be stopped just short of the centre to avoid clutter. [34].

- Tactile representations for bar charts provide relative comparison of magnitude, not detailed, accurate values.

- Using tactile methods to access information in line charts provides presentation without measurement or scale. It may be difficult to differentiate between division lines that are close together.

- Tactile method for Math's graph only represents the shape of graph without any detail. Non-speech methods, when compared with synthesis speech, provides an overview of the data, as it can be delivered in a shorter time than synthetic speech. Encoding the data value in the pitch of a MIDI note is a common strategy used to represent relative values of numerical data. A user can scan the pie chart or math's graph to get an idea of the overall distribution of the data, without accessing to the exact values [33].

- Synthetic speech which is used in this research, delivers detailed information such as actual values and proportion to vision impaired users. It can be used to represent text labels and precise data values without cluttering the tactile image with Braille or raised letter components. However listening and comparing many values passively through speech alone can be laborious and time consuming, considering navigation through information and active representation solves the issue regarding sequential passive representation.

- Previous researches show accuracy for non speech audio method to transfer information of line chart is approximately 80%[12] and for virtual touch (Haptic) is between 85.56% and 94.07% [32]. This method utilized in this research showed an accuracy in conveying implicit data in line chart at approximately 98% and for pie charts, 98.7%.

5. CONCLUSION AND FUTURE WORKS

This research, has designed a system to provide non-visual aural access to charts. The proposed approach has advantage of the ability to extract substantial information of charts from the image. Since the Visual Extraction Module assumes that it is dealing with a simple chart created by "GNUPLOT" and title position, axes ticks and, axes label locations must be in "GNUPLOT" forms, further work is required to extend this work to other charts layouts and digital formats. Issues remain with the user interaction and making the most appropriate and efficient description for effective communication of graphical information in documents to vision impaired

6. REFERENCES

1. J. Gao , Y. Zhou, K.E. Barner," Making Information Graphics Accessible for Visually Impaired Individuals" ,University of Delaware Newark ,DE19716 USA, March 2012

2. W. Yui, S. Brewster, "Evaluation of Multimodal Graphs for Blind People" , Glasgow,2003

3. C. McCathie Nevile, M.R.Koivunen, Accessibility Features of SVG, Available at : www.w3.org/TR/SVG-access, Aug 2000,[Nov.1, 2012].

4. S Elzer, E Schwartz, S Carberry, D Chester, Demir, P Wu,"A browser extension for providing visually impaired users access to the content of bar charts on the web", Millersville University, POB 1002, Millersville, PA 17551, USA,2007

5. Accessible Graphics: Graphics for Vision Impaired People,Diagram 2012 7th International Conference on the theory and application ,2012

6. C. F. Greenbacker, P.Wu S. Carberry, K.E. McCoy,S. Elzer ,D. D. McDonald ,D. Chester, S. Demir. "Improving the Accessibility of Line Graphs in Multimodal Documents" Published in: Proceeding SLPAT '11 Proceedings of the Second Workshop on Speech and Language Processing for Assistive Technologies,2011

7. S Carberry, S. Elzer, S. Demir." Information graphics: an untapped resource for digital libraries" Proceedings of the 29th Annual International ACM SIGIR Conference on Research, 2006

8. P.B.Meijer, *An experimental system for auditory image representations,* IEEE Transactions, on Biomedical Engineering, 39(2):112–121. 1992

9. S. Ina,"Computer graphics for the blind" ACM SIGCAPH Computers and the Physically Handicapped,55:16–23,1996

10. R.Ramloll,W. Yu, S. Brewster,B. Riedel, M. Murton,G. Dimigen, " Constructing sonified haptic line graphs for the blind student:First steps", Proc. of ASSETS 2000, p. 17–25,Arlington, VA, 2000

11. P.Roth, D Richoz, L. Petrucci, T. Pun,"An audio-haptic tool for non-visual image representation" ,Computer Science Department CUI, University of Geneva, XCH – 1211 Geneva 4, Switzerland,2001

12. National Centre for Accessible Media (NCAM),Provide access to graphs for users who are blind or visually impaired, Available at : http://ncam.wgbh.org/invent_build/web_multimedia/accessible-digital-media-guide/guideline-f-graphs, [2 Dec 2012]

13. W. Yu, S. Brewster, "Multimodal Virtual Reality Versus Printed Medium in Visualization for Blind People", Glasgow,2002

14. D.McCallum,*"The design and manufacture of tactile maps using an ink-jet process"*, Journal of Engineering Design, ,2005

15. I.Murray," Instructional eLearning technologies for the vision impaired ".published doctoral thesis, 2008. Curtin University, Perth,2008

16. W.Yu,R. Ramloll,S.Brewster," Haptic Graphs for Blind Computer Users", Glasgow,2001

17. A. R. Kenne",Audiograf: A diagram-reader for the blind". In *Second Annual ACM Conference*

on Assistive Technologies, p. 51–56.1996

18. University of South Australia ,2004,"Online Accessibility Action Plan" ,Available at: http://w3.unisa.edu.au/footer/accessibility/actionplan.pdf,[20 Nov 2012]

19. L. Brown, S. Brewster, R. Ramboll, M. Burton and B. Riedel ,"Design guidelines for audio presentation of graphs and tables" ,Proc. of ICAD 2003, 2003.

20. L. Ferres, P. Verkhogliad, G. Lindgaard, L. Boucher, A. Chretien, and M. Lachance, "Improving accessibility to statistical graphs: The iGraph-Lite system", ASSETS'07, Proc. of 9th Int. ACM SIGACCESS Conf,2007

21. C. Goncu , K. Marriott, "Tactile chart generation tool", ASSETS'08, Proc. of 10th Int. ACM SIGACCESS Conf. Computers and Accessibility, pp.255-256, Halifax, Nova Scotia, Canada, October 2008.

22. T.Watanabe, T. Yamaguchi, M. Nakagawa, "Development of Tactile Graph Automated Creation Software", University of Niigata, Japan,2012

23. R Cohen, V. Haven, J.A. Lanzoni, A..Meacham,"Using an Audio Interface to Assist Users who are Visually Impaired with Steering Tasks",Boston, 2006

24. A. Brown, R. Stevens ,S Pettifer, "Making graph-based diagrams work in sound: the role of annotation", Manchester,2012

25. S. Friel, F.R. Curcio, G.W. Bright," Making Sense of Graphs:Critical Factors Influencing Comprehension and Instructional Implications", Journal for Research in Mathematics Education 2001, Vol. 32, No. 2, 124–158

26. A.J. Brown," Non-Visual interaction with graphs", Manchester, 2007

27. W. Huang,C.L. Tan," A system for understanding imaged info graphics and its applications" , Proceedings of the 2007 ACM symposium on Document engineering,2007

28. W.S.Cleveland. *The Elements of Graphing Data*. Pacific Grove, CA: Wadsworth & Advanced Book Program,1985

29. Burton G. Andreas . *Experimental psychology*,1965

30. N J. Salkind *Statistics for People who (think They) Hate Statistics*, The Excel Edition,2006

31. D.L.Ryan," Modern Graphic Communications" ,Prentice-Hall, Englewood Cliffs, NJ, 1986

32. W,Yu, R Ramloll, S Brewster, B Ridel,"Exploring computer-generated line graphs through virtual touch"Department of Computing Science, *Department of Psychology University of Glasgow, G12 8QQ, UK,2012

33. S. A. Wall , S. A. Brewster "Tac-tiles: Multimodal Pie Charts for Visually Impaired Users" Glasgow Interactive Systems Group, Department of Computing Science University of Glasgow, G12 8QQ, UK,2012

34. Guidelines and Standards for tactile Graphics .available at http://brailleauthority.org/tg/web-manual/, 2010,[3 Dec 2012]

INSTRUCTIONS TO CONTRIBUTORS

The International Journal of Human Computer Interaction (IJHCI) publishes original research over the whole spectrum of work relevant to the theory and practice of innovative and interactive systems. The journal is inherently interdisciplinary, covering research in computing, artificial intelligence, psychology, linguistics, communication, design, engineering, and social organization, which is relevant to the design, analysis, evaluation and application of human computer interactive systems.

IJHCI addresses the cognitive, social, health, and ergonomic and other aspects of interactive computing including cognitive ergonomics, collaborative design, ergonomics & human factors, legal, ethical & social aspects of IT, human computer interaction, ergonomics, human computer interaction and user interface. IJHCI emphasizes the human element in relation to the systems in which humans function, operate, network, and communicate, including software, hardware, and their various contexts of use.

To build its International reputation, we are disseminating the publication information through Google Books, Google Scholar, Directory of Open Access Journals (DOAJ), Open J Gate, ScientificCommons, Docstoc and many more. Our International Editors are working on establishing ISI listing and a good impact factor for IJHCI.

The initial efforts helped to shape the editorial policy and to sharpen the focus of the journal. Started with Volume 4, 2013, IJHCI appears with more focused issues. Besides normal publications, IJHCI intend to organized special issues on more focused topics. Each special issue will have a designated editor (editors) – either member of the editorial board or another recognized specialist in the respective field.

We are open to contributions, proposals for any topic as well as for editors and reviewers. We understand that it is through the effort of volunteers that CSC Journals continues to grow and flourish.

IJHCI LIST OF TOPICS

The realm of International Journal of Human Computer Interaction (IJHCI) extends, but not limited, to the following:

- Agent models Co-ordination and communication
- Agent-based Computing
- Computer Mediated Communication
- Computer Supported Cooperative Work
- Design and Evaluation of Innovative Interactive Sy
- Empirical Studies of User Behaviour
- Graphic Interaction
- Human language Technologies and Machine Learning i
- Human-Computer Interaction Theory
- Innovative Interaction Techniques
- Intelligent Tutoring, Coaching and Debugging Syste
- Intelligent user Interfaces
- Interaction in Mobile and Embedded Systems
- Interactive Decision Support Systems
- Interface Design and Evaluation Methodologies
- Knowledge Acquisition, Discovery, Modeling and Man
- Mixed and Augmented Reality
- Multimodal Interaction
- Natural language interaction
- Peer to Peer Communication Between Intelligent Sys
- Presence
- Speech Interaction

- User Interface Prototyping for Interactive Systems
- Virtual Reality

CALL FOR PAPERS

Volume: 4 - **Issue:** 3

i. Paper Submission: July 30, 2013 **ii. Author Notification:** September 15, 2013

iii. Issue Publication: October 2013

CONTACT INFORMATION

Computer Science Journals Sdn BhD

B-5-8 Plaza Mont Kiara, Mont Kiara

50480, Kuala Lumpur, MALAYSIA

Phone: 006 03 6207 1607

006 03 2782 6991

Fax: 006 03 6207 1697

Email: cscpress@cscjournals.org

CSC PUBLISHERS © 2013
COMPUTER SCIENCE JOURNALS SDN BHD
B-5-8 PLAZA MONT KIARA
MONT KIARA
50480, KUALA LUMPUR
MALAYSIA

PHONE: 006 03 6207 1607
006 03 2782 6991

FAX: 006 03 6207 1697
EMAIL: cscpress@cscjournals.org