CoUIM: Crossover User Interface Model for Inclusive Computing

Abdullah Almurayh
University of Colorado
Colorado Springs
aalmuray@uccs.edu

Sudhanshu Kumar Semwal
University of Colorado
Colorado Springs
ssemwal@uccs.edu

ABSTRACT

Persons with disabilities can face considerable challenges accessing many computing systems, such as cloud computing. We created six low-cost user interfaces using: keyboard-based, touchable, speech-based, touch-less gesture, tactile, and then combined them all in one user interface termed Crossover User Interface Model (CoUIM). We measured inclusiveness, error occurrence, user performance, and user satisfaction though an IRB approved study of twenty-nine participants. We chose Xen cloud platform to evaluate our research. We focused on three groups of users: persons with no disability, persons with blind and visually impairment (B/VI), and persons with motor-impairment. When we combined several interactions in one user interface, results improved for persons with disability. Using CoUIM improved inclusiveness, error rate, user performance and even user satisfaction. Persons with motor disability needed a little more time to complete the same tasks in our study. In particular, we show that persons with blind and visually impairment (B/VI) can compete on equal footing with their sighted peers based on error rate and time to complete the tasks using CoUIM.

Keywords: User Interfaces, Disability Applications, Usability Study.

1 INTRODUCTION

A disability is defined as a physical or mental impairment that significantly confines a person to minor life activities and limits them from major desirable activities [2, 6, 7, 11, 24]. Human computer interaction (HCI) can play a pivotal role in enabling users with various disabilities to manage computer systems. Our goal is to enhance this interaction with a computing system through new interactive interfaces that are geared towards users with disabilities, yet can be used by anyone. This research investigates the effectiveness of creating several crossover [26] user interfaces for users with different (dis)abilities to manage clouds, our chosen application. Our idea is that a user interface can have different interaction methods, and method chosen could be selected by the user based on their (dis)abilities. Additionally, our ideas could be extended to manage other applications, other than clouds, thus creating user interfaces on demand [3] by using the same familiar set of interactions. The cloud management supported by our interfaces are the following commands: start, reboot, shutdown, suspend, resume, reset.

Our research seeks to provide the same level of accessibility and usability for everyone. Our hypothesis is that anyone can manage the cloud if we provide variety of suitable user interfaces for both a person with disability or not. We were statistically able to show that our hypothesis is true. We evaluate our interface based on inclusiveness, accuracy (error occurrence), user performance, and user satisfaction. Our interface was tested by people with visual and motor impairment.

2 PREVIOUS WORK

Any impairment does not have to restrict the person with disablity from performing a major life activity [10, 26]. The U.S. Department of Labor (DOL) states, "*Having a disability can also affect your employment status–such as a job loss or reduction in hours–in a way that could affect your group health benefits, depending on the plan rules*" [11]. People with impairments can perform well in jobs. For example, in Winston-Salem, NC, a factory allows persons who are blind or with visual impairment to make clothing for colleges and universities [16].

Cowan [8] and Mann define assistive technology as a general term that comprises technologies, tools, products, facilities, systems, and mechanisms used by people who have impairments such as the persons with disablity, persons with blindness and visual impairment, and other disabilities [18]. Assistive technology can also be considered as tools to accommodate functional limitations [5]. IDEA describes assistive technology as any equipment or customized product that is meant to improve people's capabilities [9]. Assistive technology varies from simple inexpensive tools to expensive advanced electronic devices, which are classified as backend tools (i.e. optical character recognition tools), input (i.e. speech recognition), and output (e.g. screen readers) [20]. We define assistive technology as any shape and form of interactivity between a user with a disability, and a computer application.

Assistive technology can be categorized into access and spatial controls, localization, moving, listening, communication, computer-based instruction, computer applications, vision, recreation, and special care [2, 4, 22]. Assistive technology may be stand-alone such as Mobility Smart Better-Grip Reacher, or embedded hardware such as haptic feedback glove for the blind Another form of assistive technology is computer software, which could be embedded (e.g. Speech recognition) or stand-alone like Braille Translator [14]. Braille Plus 18 is an example of Assistive technology product invented by APH[1]. The Braille Plus 18 is essentially aimed to assist persons with visual impairment. It allows a person with visual impairment read books, write assignments, scan documents, search the web, keep track of appointments, find directions, record lectures, listen to podcasts, use GPS Navigation, run Android apps, use camera, support SD card slot and USB port, connect through wireless, send text messages, make phone calls, and have many other features. However, it can be extremely expensive, approximately $3,599.00. Everyone agrees that assistive tools should not be pricy and should be affordable [20].

Unfortunately, there are different concerns and issues when using assistive technology in real world. Users face social, cultural, design, privacy, security, compatibility, and other limitations. Incorporating social acceptance into the design of assistive technology is necessary; assistive technology should have appealing design [23]. Similarly, the proliferation and wide usage of assistive technology can be affected by cultural differences including life style, language, economy, and diversity [2]. Cultural aspects must be considered in the design of assistive technology for crossover applications. Moreover, as cultural changes may occur in some communities, it is important to occasionally update the existing assistive products and tools to convey these changes. Some assistive tools depend on the structure and the design of existing sources. Having poor design affects the functionally of assistive technology as changes are not easily incorporated. Screen readers, for example, do not read by looking at web pages; however, they indicate text phrases through the HTML code and announce whatever is found. If the screen readers misinterpreted the HTML code, a meaningless sound is played. Another related issue is privacy as the users concern about their private information being read [19]. Assistive technology may disclose personal and confidential information without permission or detection; putting users at tremendous risk.

In term of security, there are two opposite perspectives of security challenges against assistive technologies: vulnerability and those related to settings. Vulnerability affects the underlying system if the assistive tool does not comply with high security level. Setting related issues (i.e. rules, restrictions, permissions, privileges, policies, or firewalls) are set to the underlying system or resource. Such settings can restrict assistive tools to not perform efficiently or prevent them from accessing the resources permanently. Adobe Acrobat software, for example, allows the user to forbid some parts of a PDF file from being copied, printed, extracted, commented on, or edited. Screen readers, on the other hand, will not be able to extract the documents text in order to transform it into a spoken format [1]. Hence, the user who creates the PDF file must be aware of this situation as well as users who depend on screen readers for accessibility.

The compatibility challenge is demonstrated by whether content works well with different assistive tools (e.g. various screen readers) on different platforms (i.e. software and operating systems). Google, for instance, lists supported assistive technology with Chrome web browsers [13]. The considerations mentioned above should be taken into account as we design assistive technology.

JAWS (Job Access with Speech) is one of the most widely used screen readers which is developed for persons with visual impairment and blindness. JAWS is a Windows based screen-reading application that reads text-based content on programs and the Internet (www.freedomscientific.com). Many schools and universities provide screen readers, such as JAWS, to accommodate the standards of accessibility. When we checkes, JAWS is priced at $1,095 for the professional edition and $895 for the standard edition. It could be considered expensive and lacks multilingualism [2]. Screen reader software such as JAWS still lacks some other features (i.e. a screen zoom function, only windows-based version, and reading when mouse is over an item) because it is aimed for those who have no vision at all [25].

Duxbury Systems has produced a Windows-based Braille translation application called Duxbury Braille Translator [12]. This is an example of Braille translators that translate some text or entire file into Braille cells, and sends it to a Braille embosser or printer to produce a hard copy in Braille script of the original text or any display. A drawback that may limit the

---

[1] Website: www.aph.org.

distribution and the wide usage of Duxbury Braille Translator is the high-priced license that starts from \$595 [2]. On the other hand, there are free Braille translators such as Braille Translator service provided by BrailleTranslator.org. Although it is easy to access, free to use, and simple to convert text to braille, much work remains to be done to support more functions, languages, and applications. Still more work can be done as screen readers are not completely fulfilling the users' expectations, specially they do not provide facilities for Xen-management, which is our application.

3 VALIDATION STUDY AND FRAMERWORK

Our focus is on application programming interfaces (API) for building cloud-based applications [21]. A computing system cannot be effectively utilized without a proper user interface (UI) because some of these system are not usable [15] by everyone. Meeting the needs and preferences of all users is difficult. Our interface provides several modalities at the same time. Our API allows developers, or even users, to manipulate the interactive user interface so that various inputs in different forms and shapes can be transformed into a unified form that the API accepts. The Crossover User Interface Model (CoUIM) is a two-part framework that implements collaborative user interface development models for inclusive computing. First part is to use multiple interactions to map user intention to a command, and second part is to execute that command. The beneft of our model is that we can easily replace the application and use the *same framewrok* of commands to map a new application. This helps to avoid redesigning existing computing systems.

Our validation study is based on following characteristics of our interface:

- Mouse, is used for selecting and clicking on entities. This works for sighted-users and blind users since the type of interactions allows visual and audible feedback.
- Remote Controller, is used as a mobile mouse for selecting and clicking on entities from distance. It works remotely just like the method (A) where any clicked or hovered entity is seen and heard.
- Leap Motion, is used for pointing at entities and taping on them from distance. So far, this method requires vision as audible feedback is not available.
- Mouse Pad. is used for moving mouse cursor around the screen and hover entities and clicking on them. It supports users' preferences. It also, allows audible feedback and can be seen.
- Tangible dots, are stuck on a specific key that has a function. It is a helpful tangible tool that blind and visually impaired users can utilizes especially those who are not spatial learners and are unfamiliar with the keyboard layouts.
- Keyboard, is used as an alternative emitter for the mouse. Users can use Tab and Enter buttons to interact with the interface.
- Speaker, produces audible feedback.
- Large Obvious Cursor, helps visually impaired users and those who interact with the interface from distance to locate the cursor easily.
- Screen, displays the components of the user interface visually to the user to receive information. In addition, it can be touchable screen; the user can use it as emitter (or display) device.

A total of 29 participants participated in our University IRB (Internal Review Board) approved study. There were eleven participants with no disability. Nine participants were either with some form of visual impairment or were blind. Nine had some form of physical impairment. Xen Cloud Platform was used for all testing.

4 MULTIPLE INTERACTIONS

We implemented a total of six approaches and they were all tested. Five of these approaches are: (a) Typing based, (b) Touchable Mobile interface, (c) Speech Conversation, (d) Touchless interaction, and (e) Straight forward input board. The sixth approach, which includes all the previous five types of interactions in one single interface, is an example of the Crossover User Interface model (CoUIM) which allows people with visual and motor impairment to complete the tasks asked by people with no impairment.

4.1 Typing-based Cloud Management

This contribution is conventionally used in many computing systems where keyboard is used. It is also known as the command line interface. The user needs eyesight to find the keyboard keys and needs fingers to type, making it highly inconvenient for people with visual impairment. The Computer requires a monitor to display output and speakers to voice results (Figure 1).

4.2 Touchable Mobile Interface for managing the Cloud

This contribution is unique in that it uses a smart phone for input and visual and auditory feedback for output. The user touches the screen to scroll the information. Touch includes mouth stick, head stick and other similar devices for people with motor disability. The user needs vision and/or hearing and the ability to touch the screen of the device. The user interface is the touch screen for input and voice for input and output.

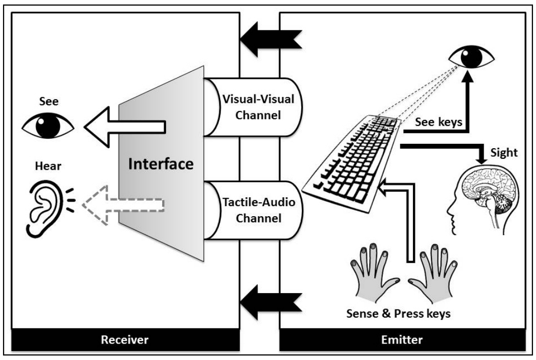

Figure 1: High level Design of typing-based cloud management.

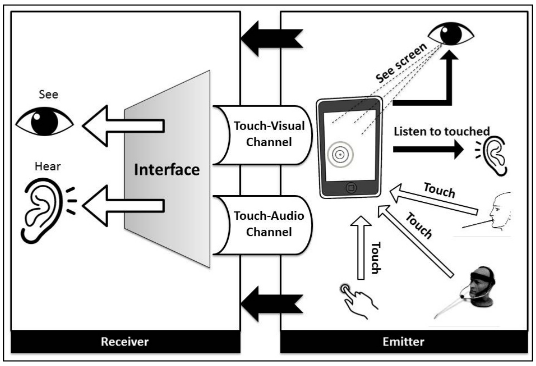

Figure 2: High level Design of Touchable Mobile Interface for managing the Cloud.

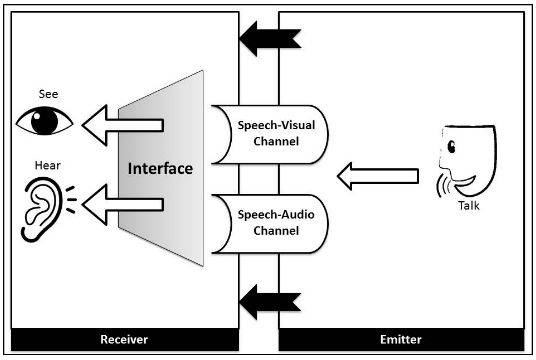

Figure 3: High level Design of Managing the Cloud via Human-computer Conservation..

4.3 Managing the Cloud via Human-computer Speech-conversation

This work depends completely on speech-based interaction to the point that human and machine can converse. The user interface is a computer equipped with the ability to receive and output speech. The computer requires a microphone to interface with the speech recognition. The user receives spoken and/or visual information. The user interface is a screen with text display and a microphone or a headset.

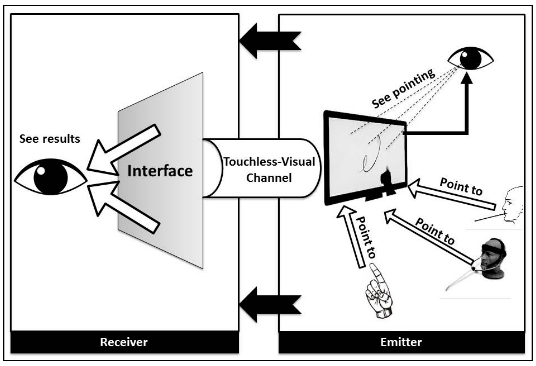

Figure 4: High level Design of Touchless Interaction for Managing the Cloud.

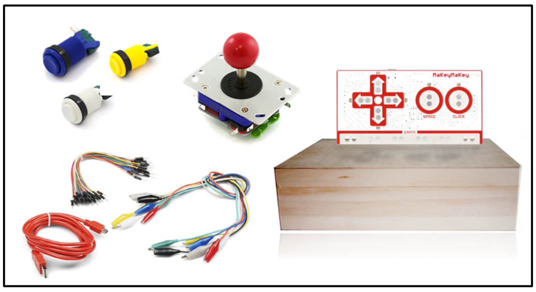

Figure 5: The tools and materials needed for building the MaKey MaKey-based Custom Broad.

4.4 Touchless Interaction for Managing the Cloud

The User Interface is a USB device attached to the machine to track user's movement. The user points to areas on the screen with a finger or with a mouth stick, head stick or similar device. The Computer requires a monitor and a USB port. The user needs vision to see where he/she is pointing and to view the results. The User Interface is a colored circle on the screen that will change colors as the user points and moves the pointer. The Computer requires a monitor to display input and output.

4.5 Managing the Cloud via straight Forward Input Board

MaKey MaKey[2] is an invention kit which refers to the combination of the words Make and Key. The machine assumes inputs come from MaKey MaKey is a regular keyboard and mouse. Figure 5 indicates the collection of objects and tools that we needed to implement to custmize for our application. Collection of objects include: a wooden box, a Joystick, leads, tactile buttons, a USB cable, and a MaKey MaKey tool kit underneath.

Figure 6 shows the architectural design of the MaKey MaKey-based custom board. An earth lead is connected to all buttons and the joystick device. As the user

---

[2] Available on: http://www.makeymakey.com.

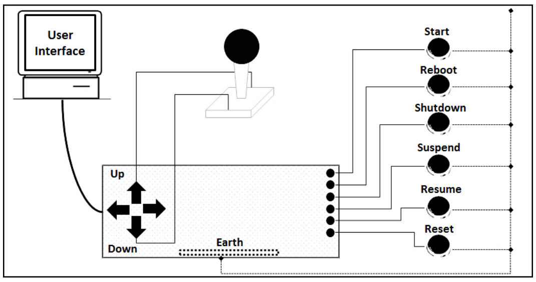

Figure 6: The architecture of the MaKey MaKey-based custom broad.

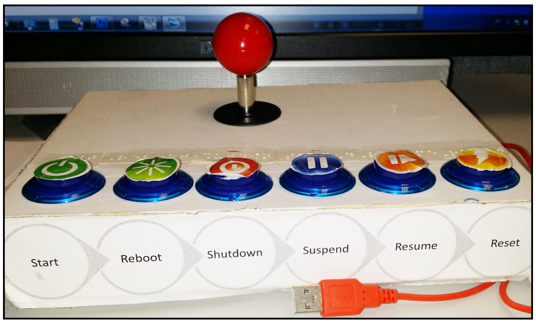

Figure 7: The final look of the built MaKey MaKey-based custom board.

presses a button, it sends a unique signal that the user interface illustrates as a function. Therefore, when the user pushes the joystick up/down the user interface navigates through the virtual machines. This custom board is connected to the computer via a USB cable. Figure 7 exhibits the final look of the built MaKey MaKey-based custom board. It has labels, signs, and Braille labels so that the users can determine the functions based on their (dis)abilities.

4.6 CoUIM: Crossover User Interface Model for Managing the Cloud

This contribution combines different interaction methods to improve the inclusiveness (Figure 8). The user can now interact with the machine using either vision and/or hearing, and tactile sense. The user can touch, gestures, tact, press keys, and click mouse to interact with the computing system based on what they feel comfortable with. The computer requires a display to reveal the interface and audio device to inform the user verbally about the occurrence. Monitor, speakers or headset provide feedback to the user. Results can be simultaneously presented using visual and audio cues.

5 RESULTS BASED ON QUANTITATIVE DATA ANALYSIS

The CoUIM approach that allows multiple interaction channels has excellent inclusiveness as our results show (see Figure 9). Table 1 indicates that all participants were 100% able to utilize this approach using different interaction methods to accomplish the required tasks.

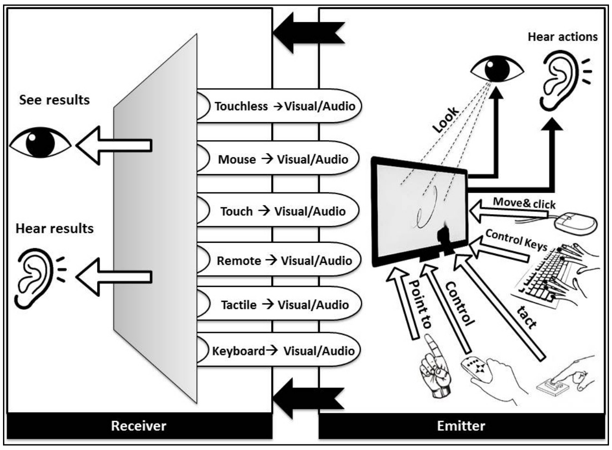

Figure 8: High level Design of Crossover User Interface for Managing the Cloud.

Feasibility: Figure 10 shows how obviously all groups had the same opportunity of utilizing this approach – each group reached 100% effectiveness. While participants with no disability, and participants with motor disabilities were able to use touchless interaction and vision to receive information, most of the participants with blindness and visual impairment (B/VI) used tangibles, mouse, and keyboard to emit and speakers to receive the verbalized results.

Error Occurrence: In terms of error rates, excellent results were obtained as indicated in Figure 11. Users with B/VI were more accurate in completing the tasks precisely– this group of users had a *zero* error rate. Dependence on verbalized feedback helped the B/VI users to be more accurate compared to the other two groups. Hand and finger gestures provides natural interactions with user interface [17]. Nevertheless, finger identification and hand gesture recognition accuracy still needs more improvements[17] in touchless interface. This problem influenced users' gesture-based interaction accuracy so that non-disabled users made 0.4 error per users whereas users with motor disabilities made 0.9 error per users on the average. Although the interactions of both groups were affected by the unperfected gesture recognition accuracy, users with motor disabilities struggled because of their motor impairments.

User performance: Although the users with B/VI were more accurate, they had some latency where each user with B/VI could take as an average of 176.7 seconds to accomplish the required tasks. While non-disabled users used gesture-based interaction, they needed at an average of 153.5 seconds; so they performed more efficiently compared to the B/VI users. The main reason of this difference is that a B/VI participant needed additional time to navigate through the interface depending on tactile and audible interaction. Users with motor disabilities needed an average of 234 seconds to accomplish the tasks because of their motor difficulties. Fig-

ID	*Disability*	*Inclusiveness*	*Error*	*Time*
I	B/VI	1	0	229
K	B/VI	1	0	172
L	B/VI	1	0	339
M	B/VI	1	0	128
O	B/VI	1	0	131
P	B/VI	1	0	158
Q	B/VI	1	0	139
R	B/VI	1	0	137
T	B/VI	1	0	157
A	M	1	2	289
AA	M	1	1	155
C	M	1	3	190
S	M	1	0	180
V	M	1	0	321
W	M	1	0	279
X	M	1	0	126
U	M/VI	1	1	329
Y	M/VI	1	0	153
AB	Non	1	0	115
D	Non	1	0	163
E	Non	1	1	150
F	Non	1	0	132
J	Non	1	0	174
B	Non	1	1	113
G	Non	1	1	160
N	Non	1	0	142
AC	Non	1	1	127
H	Non	1	0	223
Z	Non	1	0	190

Table 1: The overall results of testing the crossover user interface for cloud management. Disabilities: M, motor disabilities; B/VI, blind and visually impaired; Non, non-disabled.

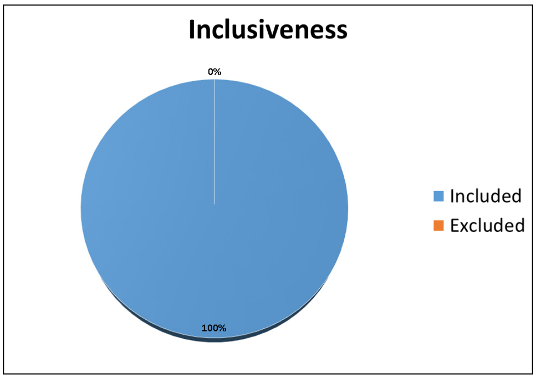

Figure 9: The percentage of included users in the crossover user interface for cloud management.

ure 12 indicates the user performance averages of each group.

5.1 Results based on Qualitative Data Analysis

The results were as shown in Table 3, each criterion has different feedback. However, since not all groups utilized the same interaction method, the qualitative results were categorized based on the groups. Most of the persons with blindness and with Visual Impairment (B/VI) utilized tactile interaction methods for requests. Some participants with visual impairment, who were

Group Of users	*Inclusiveness*	*Error*	*Time (sec)*
B/VI	100%	0.0	176.7
M	100%	0.8	224.7
Non	100%	0.4	153.5

Table 2: The averages of the results of testing the crossover user interface for cloud management. Disabilities: M, motor disabilities; B/VI, blind and visually impaired; Non, non-disabled.

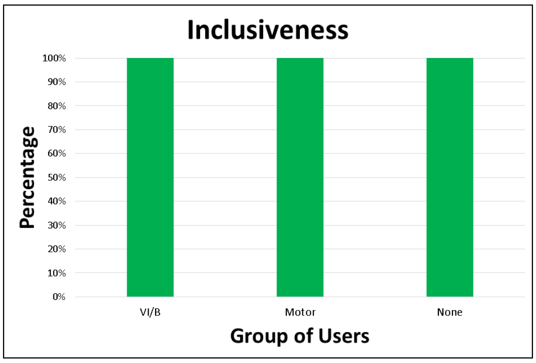

Figure 10: The percentage of included per group users in the crossover user interface for cloud management.

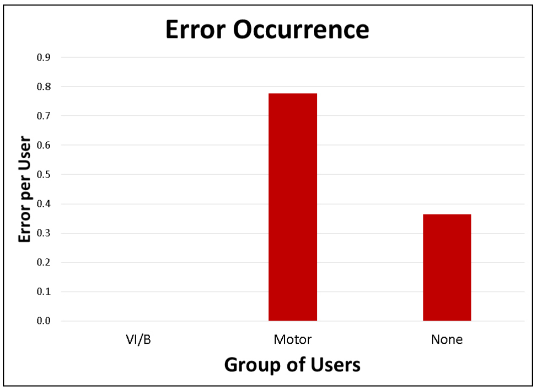

Figure 11: The error occurrence per user for each group using the crossover user interface for cloud management.

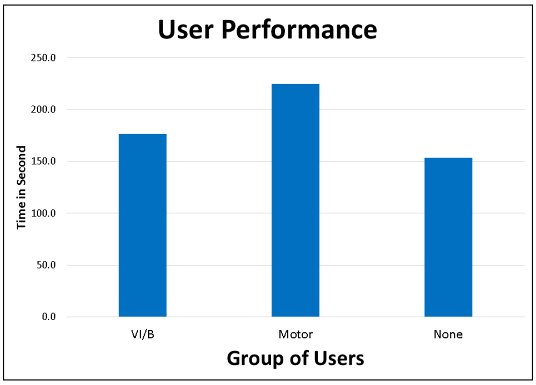

Figure 12: The averages of time per group needed to accomplish the tasks using the crossover user interface for cloud management.

not completely blind, used the mouse with audible feedback for navigating and emitting requests and receiving spoken information. Therefore, not only did the crossover user interface enable them in computing, but also it gave them various options to utilize in this group of users based on differences in degree of blindness. This means that participants with blindness and visual impairment conveyed their preferences. For example, a person with visual impairment who could utilize tangibles could also use mouse as an input. As shown in Table 3, participant have different opinions and experiences based on what they prefer to use: Strongly Disagree (SD), Disagree (D), Neutral (N) , Agree (A), Strongly Disagree (SD).

5.2 Statistical Significance

In terms of ***error occurrence***, the inclusion of group of participants with blindness and visual impairment boosted the statistical significant difference from ($p = 0.19, P > 0.1$) to ($p = 0.07, 0.05 < P < 0.1$). Two reasons influenced the results, they were:

- The sample size increased from 18 → 29 participants.
- The error rate was obviously *decremented* from 0.6 → 0.38.

The t-Test results showed that including a set of users who performed the experimental test with extreme preciseness enhanced the statistical significance of error occurrence. We were pleased that a previously excluded motor-impaired participant was included in the present approach (see Participant "*Y*" in Table 3). The inclusion of this user enlarged the size of the sample by one person. The main point here is that this user was able to perfrom the tasks with zero error occurrence. Inclusion of this participant enhanced the level of accuracy differences between two groups.

The t-Test results showed that B/VI group accuracy is significantly better than the other groups because:

- The statistical significance difference between the non-disabled group vs the B/VI group is ($p = 0.04, P < 0.5$) with the error occurrence means $M_{non} = 0.36$ and $M_{bvi} = 0.0$.
- The statistical significance difference between the motor-impaired group vs B/VI the group is ($p = 0.06, P \leq 0.5$) with the error occurrence means $M_m = 0.77$ and $M_{bvi} = 0.0$.

The variety of interactions allows each participant to select the best suitable interaction that overcomes the participant's difficulty and enables the participant to perform tasks effectively and precisely.

Similarly, the CoUIM approach surprisingly demonstrated a vast improvement in terms of ***user performance*** significant difference. Accordingly, the p-value was increased–compared to the user performance *p-value* from ($p = 0.007, P < 0.01$) to ($p = 0.05, 0.05 < P < 0.1$). The reasons that influenced the results were:

- The size of the sample was enlarged from 18 → 29 participants.
- The means of user performance were obviously decreased from the average of 193.5 → 184.9.

The t-Test results showed that the statistically significant difference occurred when we compared the non-disabled and motor-impaired groups so that ($p = 0.03,\ 0.01 < P < 0.05$). The reason of such difference is that the participants with motor-impairment performed much slower compared to users with no disability, which is naturally expected as most of participants with motor-impairment used touchless motion based interaction. Allowing such crossover user interface showed that there was no statistically significant difference between group with no disability and the group with blindness and visual impairment, which was ($p = 0.37,\ P > 0.1$) with means of $M_{non} = 154$, $M_{vib} = 177$. This means that the group with blindness and visual impairment was similar in performance to the group with no disability. Likewise, there was a weak statistical difference between groupd with motor-impairment and the group with blindness and visual impairment, which was ($p = 0.18, P > 0.1$) with means of $M_m = 225, Mean_{bvi} = 177$. The similarity of the two groups' sizes and the reasonable difference of their user performance means decreased statistical significant differences. The variety of interactions in CoUIM allows every participant to select the best suitable interaction and increase efficiency. Our statistical and qualitative analysis shows that increasing interaction options for users would allow more accommodation, precision, efficient completion of tasks.

6 CONCLUSIONS AND FUTURE WORK

The Crossover User Interface (CoUIM) model makes the task of managing a Xen-cloud more inclusive so that persons with no disability, persons with blindness and visual impairment, and persons with motor disability can share the same user interface by utilizing a variety of interactions. We were able to deliver a very effective solution for a minimal cost ($200). Our results show that people with disability, especially those with blindness or visual impairment, can compete on equal footing with the participants with no disability. Specifically the research showed that user inclusiveness was improved by using our CoUIM that contain a variety of

Participant	*Disability*	*Easy*	*Engaging*	*Quick*	*Interesting*	*Precise*	*Useful*
A	Motor	SA	SA	SA	A	SA	SA
B	—	SA	N	SA	SA	A	A
C	Motor	SA	SA	D	D	SA	D
D	—	SD	SD	SD	SD	SD	SD
E	Visual	SA	A	A	A	A	A
F	Motor , Hearing	N	A	A	A	A	A
G	—	N	A	N	N	N	N
H	Visual	N	N	A	N	A	N
I	—	A	A	A	SA		SA
J	—	SA	SA	A	SA	A	SA
K	Motor	D	A	N	N	D	SD
L	Motor	SA	SA	A	SA	A	SA
M	—	SD	SD	SD	SD	SD	SD
N	—	SA	SA	SA	SA	SA	SA
O	—	SA	SA	SA	SA	A	SA
P	—	SA	SA	A	A	A	A
Q	Visual Motor	A	A	A	SA	A	SA
R	Motor , Visual, Brain injury	SD	D	D	SD	SD	SD
S	Motor	A	A	A	A	A	A
T	Motor ,Brain injury	SA	A	A	A	SA	A
U	Visual, Motor	SA	SA	SA	SA	A	SA
V	Visual	SA	SA	SA	SA	SA	SA
W	Visual	N	D			D	D
X	Visual Motor	SA		D	SA	SA	SA
Y	Visual	A	A	A	A	N	N
Z	Visual	SD	SD	SA	SD	SD	SD
AA	Motor	A	SA	N	N	D	A
AB	—	SA	SA	A	SA	SA	SA
AC	—	A	A	SA	N	SA	N

Table 3: The qualitative data of experiencing the crossover user interface.

Groups	*Count*	*Sum*	*Average*	*Variance*
B/VI	9	0	0	0
Motor	9	7	0.78	1.19
None	11	4	0.36	0.25

Table 4: Statistical summary of error occurrence using Anova single factor method.

Groups	*Count*	*Sum*	*Average*	*Variance*
B/VI	9	1590	176.67	4661.75
Motor	9	2022	224.67	6272.25
None	11	1689	153.55	1114.67

Table 5: Statistical summary of user performance using Anova single factor method.

interactions. The results also showed that user performance can be imrproved by using a more inclusive and better design. CoUIM provides improved user satisfaction as each user utilized the method that best fit their abilities.

Based on our experiences and observations, we would like to provide the following in future: (a) Our inclusive design framework could be extended to provide novel solutions towards other types of disabilities which we were not able to include in our study; (b) Complex disabilities or multiple disabilities (e.g. a person with blindness, hearing, speaking, and motor impairments) pose new forms of challenges and need to be considered in future; and (c) Disabilities often change dynamically over time, and the changes may require different or additional accommodations.

7 ACKNOWLEDGMENTS

Ms. Bonnie Snyder who provided inspiration for our work. We are thankful to our friends at the Independence Center in Colorado Springs. We are also grateful to all the participants who participated in our study. This paper is based on Dr. Abdullah Almurayh's Ph.D. Thesis, entitled Corssover User Interface for Inclusive Computing, Dissertation Advisor Dr. S.K. Semwal at the University of Colorado, Colorado Springs, pp. 1-341 (2014).

8 REFERENCES

[1] Adobe. Prevent security settings from interfering with screen readers.

[2] Abdullah Almurayh and Sudhanshu Semwal. Cultural considerations for designing crossover applications for the visually impaired. In *Information*

Reuse and Integration (IRI), 2013 IEEE 14th International Conference on, pages 668–675. IEEE, IEEE, Aug 2013.

[3] Pradipta Biswas, Peter Robinson, and Patrick Langdon. Designing inclusive interfaces through user modeling and simulation. *International Journal of Human-Computer Interaction*, 28(1):1–33, 2012.

[4] P. Blenkhorn and D.G. Evans. Using speech and touch to enable blind people to access schematic diagrams. *Journal of Network and Computer Applications*, 21:17–29, 1998.

[5] Michael W Boyce, Drea K Fekety, and Janan A Smither. Resource consumption and simulator driving performance using adaptive controls. *Assistive technology*, 25(3):158–165, 2013.

[6] Dirk Burkhardt, Kawa Nazemi, Nadeem Bhatti, and Christoph Hornung. Technology support for analyzing user interactions to create user-centered interactions. In *Universal Access in Human-Computer Interaction. Addressing Diversity*, volume 5614 of *Lecture Notes in Computer Science*, pages 3–12. Springer Berlin Heidelberg, 2009.

[7] Christian BÃ¼hler. Design for all-from idea to practise. In *Computers Helping People with Special Needs*, volume 5105 of *Lecture Notes in Computer Science*, pages 106–113. Springer Berlin Heidelberg, 2008.

[8] Donna M Cowan and Yasmin Khan. Assistive technology for children with complex disabilities. *Current Paediatrics*, 15(3):207–212, 2005.

[9] D.E. Individuals with disabilities education improvement act of 2004, 2004.

[10] D.O.L. What are "mitigating measures"?, 2008.

[11] DOL. Disability, health benefits advisor, 2014.

[12] Duxbury. Braille translation software from duxbury systems, 2012.

[13] Google. Accessibility: Assistive technology support, 2012.

[14] Zhang Hua and Wei Lieh Ng. Speech recognition interface design for in-vehicle system. In *Proceedings of the 2Nd International Conference on Automotive User Interfaces and Interactive Vehicular Applications*, AutomotiveUI '10, pages 29–33, New York, NY, USA, 2010. ACM.

[15] Arthur M. Langer. *The User Interface*, pages 21–35. Springer London, 2008.

[16] Nicole Laporte. Hiring the blind, while making a green statement, 2012.

[17] Unseok Lee and Jiro Tanaka. Finger identification and hand gesture recognition techniques for natural user interface. In *Proceedings of the 11th Asia Pacific Conference on Computer Human Interaction*, APCHI '13, pages 274–279, New York, NY, USA, 2013. ACM.

[18] W.C. Mann and J.P. Lane. *Assistive technology for persons with disabilities*. American Occupational Therapy Association, 1995.

[19] Suzanne Martin, Johan E Bengtsson, and Rose-Marie Dröes. Assistive technologies and issues relating to privacy, ethics and security. In *Supporting People with Dementia Using Pervasive Health Technologies*, pages 63–76. Springer, 2010.

[20] Joyojeet Pal, Manas Pradhan, Mihir Shah, and Rakesh Babu. Assistive technology for vision-impairments: Anagenda for the ictd community. In *Proceedings of the 20th International Conference Companion on World Wide Web*, pages 513–522, New York, NY, USA, 2011. ACM.

[21] Dana Petcu, Ciprian Craciun, Marian Neagul, Iñigo Lazcanotegui, and Massimiliano Rak. Building an interoperability api for sky computing. In *High Performance Computing and Simulation (HPCS), 2011 International Conference on*, pages 405–411. IEEE, 2011.

[22] Joanne Pompano. *Electronics in the 20th Century: Nature, Technology, People, Companies, and the Marketplace*, volume VII, chapter Designing Accessible Websites for Blind and Visually Impaired. Yale-New Haven Teachers Institute, 1999.

[23] Kristen Shinohara. A new approach for the design of assistive technologies: Design for social acceptance. *SIGACCESS Access. Comput.*, (102):45–48, January 2012.

[24] B. Shneiderman. *Designing the User Interface: Strategies for Effective Human-Computer Interaction*. Addison Wesley, Menlo Park, CA, 3rd edition, 1998.

[25] TopTenREVIEWS. Jaws 12, 2012.

[26] Brian Wilke, Jonathan Metzgar, Keith Johnson, Sudhanshu Semwal, Bonnie Snyder, KaChun Yu, and Dan Neafus. Crossover applications. In *Proceedings of the 2009 IEEE Virtual Reality Conference*, pages 305–306, Washington, DC, USA, 2009. IEEE Computer Society.