Interface Modeling for Digital Device Control According to Disability Type in Web

Joo Hyun Park[1], Jongwoo Lee[1], Soon-Bum Lim[1*]

Abstract

Learning methods using various assistive and smart devices have been developed to enable independent learning of the disabled. Pointer control is the most important consideration for the disabled when controlling a device and the contents of an existing graphical user interface (GUI) environment; however, difficulties can be encountered when using a pointer, depending on the disability type; Although there are individual differences depending on the blind, low vision, and upper limb disability, problems arise in the accuracy of object selection and execution in common. A multimodal interface pilot solution is presented that enables people with various disability types to control web interactions more easily. First, we classify web interaction types using digital devices and derive essential web interactions among them. Second, to solve problems that occur when performing web interactions considering the disability type, the necessary technology according to the characteristics of each disability type is presented. Finally, a pilot solution for the multimodal interface for each disability type is proposed. We identified three disability types and developed solutions for each type. We developed a remote-control operation voice interface for blind people and a voice output interface applying the selective focusing technique for low-vision people. Finally, we developed a gaze-tracking and voice-command interface for GUI operations for people with upper-limb disability.

Key Words: Web Accessibility, Device Accessibility, Type of Disability, Blind, Low Vision, Upper Limb.

I. INTRODUCTION

To improve the quality of life of people with disabilities, achieving self-sustaining economic stability is the most important issue 0. The rate of participation of persons with disabilities in economic activities is the highest for those with at least a university degree, and the rate of employment exhibits a similar trend. Hence, the education and employment rates of the disabled are closely related 0. Therefore, the university attendance of the disabled is an important process for increasing the possibility of their participation in future economic activities. This requires an environment where people with disabilities can learn by themselves at school and at home.

Although learning methods using various assistive and smart devices to enable independent learning by the disabled exist, they are inconvenient for the disabled to use on their own 0. In the existing smart device environment, general users use a graphical user interface (GUI) environment with the windows, icons, menus, pointers metaphor to record, retrieve, and reuse information. To control the device and contents in the existing GUI environment, it is critical for the disabled to control a pointer using a mouse. However, this can be difficult, depending on the disability type 0. Hence, technology is being developed to enable the disabled to use a GUI.

Most applications for all blind people are the only means of manipulation by touching the screen. Touching the screen to operate is a practically impossible task for all blind people without visual ability. To compensate for this, a voice recognition function was added, but applications using the voice recognition function only provide text input function, so there is a limit to controlling the device [5, 6].

People with low vision can browse the web by enlarging the screen with the help of special-purpose auxiliary devices such as web screen readers [7, 8]. However, the biggest problem is that accuracy is degraded in selecting or manipulating small menus and click objects of web browsers and web contents. To compensate for this, the current smartphone device provides voice guidance services that combine touch and voice, such as VoiceOver [9] and Talkback [10], but there are limitations in expanding and selecting content, including GUI and TTS control problems. Since unnecessary content is also outputted by voice, it causes fatigue to be used by people with low vision.

The gaze tracking technology is receiving the most attention as a technology for web browsing of upper limb disabilities. In the existing research, when controlling an object through gaze tracking, an execution command is performed on a pointed object such as a mouse click through the dwell time [11] or eye blink [12]. However, the dwell time and eye blink method for pointer execution cause a problem of repeated pointer execution malfunction when the user wants to see only the screen without giving a gaze command.

However, the provision of consistent technology is limited for various types of disabled people. For example, a blind person cannot select and execute contents and menus because a screen is not visible to him/her. People with poor vision experience difficulty in selecting and executing contents and menus in a GUI, which are typically small, whereas people with upper-limb disability experience a malfunction problem when using eye-tracking technology to manipulate a pointer.

Therefore, it is essential to solve problems that occur when people with various types of disabilities control a web interaction using a GUI in a web and smart device environment. Web interaction means the interaction that occurs when the screen is recognized, manipulated, and navigated in a web environment.

In this study, a multimodal interface pilot solution is presented to enable people with various disability types to control web interactions more easily [13]. The multimodal interface is an interface that combines two or more NUI technologies.

First, we classify web interaction types using digital devices and derive essential web interactions among them. Second, to solve the problems that occur when performing web interactions considering the disability type, the necessary technology according to the characteristics of each disability type is presented. Finally, a pilot solution to the multimodal interface for each disability type is proposed.

II. MODELING OF WEB INTERACTIONS

To utilize the interface design for each disability type, the essential interactions were derived by classifying web interaction types for content and menu manipulation. The analysis was conducted in three stages to classify the interaction type and operation details for access control and web content for devices systematically. First, we analyzed the Windows GUI interaction types and web browser interaction function, and then, the interaction types of the two environments were matched and categorized. Finally, we used the web browser interaction function frequency

and the result of the previous step to extract the final interaction type and function.

Windows GUI interaction types can be classified into six categories 0, 0: the menu, move–grow, text, trace, new point, and angle interactors. These include all types of interaction methods used in mouse-based GUIs including text inputs.

MODELING OF WEB INTERACTIONS
① Analysis of general interaction function of PC and web browser
② Web browser interaction type mapping according to Windows GUI interaction type
③ Extract web interaction function list

CLASSIFICATION OF DISABILITY TYPES
① Selection of disability types in the special education law
② Classification of disability types and suggestion of alternative technologies
③ Feasibility review
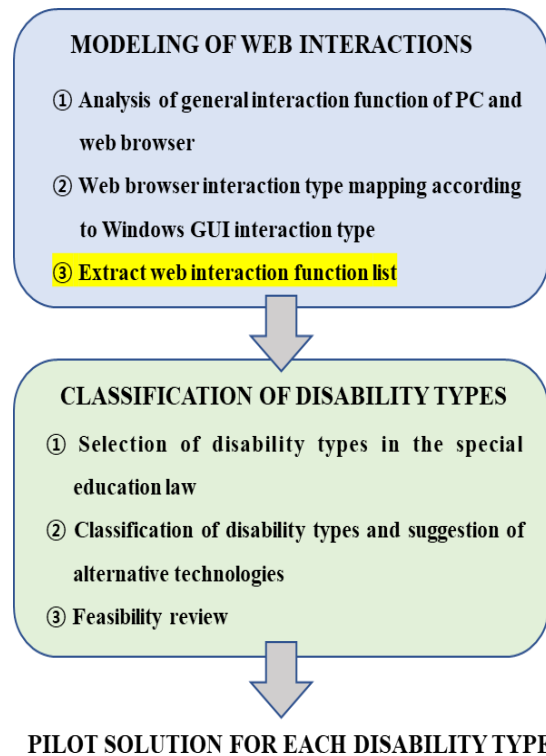
PILOT SOLUTION FOR EACH DISABILITY TYPE

Fig. 1. Overview of research flow.

To extract the web browser interaction function, we used the service task from the analysis of the digital information level of the disabled 0 in Korea. The surveyed service tasks were information and news search, e-mail communication, multimedia content service use, Social Networking Service, and cloud service. We investigated the interaction functions that occurred while using the Internet service and recorded the number of occurrences per interaction.

Based on the results obtained by matching the interaction types of the two environments, seven interaction types for accessing web content were discovered: the menu, move–grow, text, trace, object select, display control, and mixed interactors. The seven newly classified types of interactions excluded the angle and new pointer interactors from the Windows GUI. The angle interactor is an interaction type that performs angle calculation, whereas the new pointer interactor creates graphic objects such as rectangles. However, the analyzed web browser interaction functions cannot work similar to these two interactions types. The object selects, display control, and mixed interactors, which

250

were newly observed in the web browser environment, were extended.

As the final step in the modeling of interactions, we extracted the interaction types and functions that are frequently used in the newly classified web interactions. The frequency of use is the number of times the web browser service is recorded in the previous section. Only the interaction functions with a score of at least three points from a total score of five points were extracted. The interaction types and functions of the interaction modeling are presented in Table 1.

Table 1. Interaction Types and Functions of Interaction Modeling.

Type of Interactors	Detailed Interaction Function
Menu-Interactor	Web page menu execution
	Link execution
Move-Grow-Interactor	Browser menu execution
	Scroll
Text-Interactor	Short text input
	Edit text
Trace-Interactor	Move pointer
Display-Control-Interactor	Zoom in and out

Table 1 details the interactions for controlling the GUI result by moving the pointer and Pointer executing (clicking). This means that the movement and execution of pointers are the most important considerations for manipulating the GUI. Therefore, pointer movement and execution were determined as the reference points when designing the interface for each disability type. Finally, we analyzed the problems in performing web interactions according to the categorized disability types and suggested some solutions. The validity was confirmed by interviews with experts and user groups 0.

III. CLASSIFICATION OF DISABILITY TYPES

This section classifies the disability types based on the smart-device usage environment and derives the characteristic technology for each type to compensate for the limitations of web interaction control. Traditionally, disabilities are classified based on physical aspects; however, this study reclassifies the disability types based on Korea's Disability Welfare Act and device accessibility requirements. Device accessibility requirements were applied to the smart-device usage environment to perform the necessary web interactions derived in Section 2.

3.1. Classification of Disability Types and Targets

The classification targets were derived based on two criteria. First, the disability types specified in the Disability Welfare Act of Korea, which have no restrictions regarding the operation of the device, were excluded. Consequently, visual impairment and upper-limb disorders of retardation and brain lesion disorder were derived.

Second, functions that require the execution of essential web interactions (pointer movement and execution) are classified into visual perception and hand operation ability, as shown in Table 2, and the disability types selected in the previous stage were classified according to the degree of visual perception and hand operation ability.

Table 2. Types of Disability and its Range.

Criteria	Level	Range
Visual Perception (Display)	Blind	Do not see
	Low vision	Inconvenient low vision and presbyopia
	Normal vision	Non-disability
Hand Manipulation Ability (Control)	Hand unavailable	No operation
	Some hand available	Uncomfortable or some reaction
	Hand available	Non-disability

3.2. Analysis of Interaction Function and Suggestion of Alternatives by Disability Type

Based on the device accessibility criteria, assistive devices, and natural user interface (NUI) technology, we investigated whether essential web interactions can be performed according to the reclassified disability types. The device accessibility-criteria analysis included checking whether the software screen can be recognized, and the viewer menu operated. The auxiliary device and NUI technology were analyzed based on the possibility of auxiliary device operation and voice and gaze interface use. Finally, based on the analysis results, solutions were presented for each disability type. Table 3 presents the results of analyzing the possibility of performing essential web interactions and the use of alternative technologies according to the disability type.

In addition, the speech-based characteristic technology was derived because speech can be used for all disability types. The results are as follows: some hand-available user groups and hand-enabled groups in the blind group provided simple auxiliary controls and voice input and output. The low vision group suggested that the contents can be identified by expansion and voice output technology. The group with upper-limb disability, whose hands cannot be used but who had

normal vision, suggested the eye-tracking and voice-command technology as an alternative to control smart devices without the limitation of hand usage.

Finally, the following can be described as the disability type grouping. Validity of the disability type grouping and the characteristic technology of each group were examined through interviews

with experts and real user groups. The results indicated that the user group classification based on device usage function was appropriate, as was the classification and technology of the derived user group.

Table 3. Results of Analyzing the Possibility of Performing Web Interactions and the Use of Alternative Technologies.

User	Display Recognition	Menu Operation	Auxiliary Equipment Operation	NUI		Solution
				Voice	Gaze	
Blind, Hand unavailable	X	X	X	O	X	Simple auxiliary controls and voice input and output
Blind, Some hand available	X	X	△	O	X	
Blind, Hand available	X	X	O	O	X	
Low vision, Hand unavailable	△	X	X	O	X	Magnify content and voice out
Low vision, Some hand available	△	△	△	O	X	
Low vision, Hand available	△	O	O	O	X	
Normal vision, Hand unavailable	O	X	X	O	O	Eye tracking and voice commands for hands-free control
Normal vision, Some hand available	O	△	△	O	O	
Normal vision, Hand available	Non	Non	Non	Non	Non	

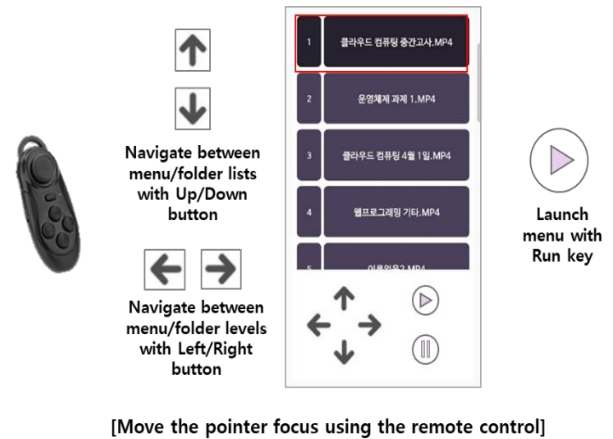# IV. DEVELOPMENT OF PILOT SOLUTION FOR EACH DISABILITY TYPE

4.1. Solution

Based on the results of the study presented in Sections 2 and 3, we developed a multimodal interface with solutions based on the types of disabilities.

Android-based mobile memo applications were developed to enable free voice memos and control for blind people [18]. The application instantly recognizes menus with voice output and applies Bluetooth remote control and voice functions to freely navigate through multilevel menus or folders.

We designed the menu control interface for multi-level menu navigation and execution by mapping it with the buttons on the remote control. The remote control is provided with up, down, left and right direction buttons and execution and pause button. The multi-level menu operation and execution control interaction was designed using the corresponding button. The up and down buttons used to navigate the menu list displayed on the same level. Left and right buttons move between upper and lower menus. Left button moves to upper menu, right button moves to lower menu. At this time, the right button moves to the sub-menu and does not execute the menu. Actual execution of the corresponding menu is performed by the execution button, and the pause button is a stop for executing the sub-menu during voice recording.

[Move the pointer focus using the remote control]

Fig. 2. Screenshot of remote control voice output application and method of control using Bluetooth remote controller.

For people with low vision, we developed a voice browser for mobile environments based on Android. A selective focusing technique that only selects the desired area within the web content and expands it in the selected order or outputs it by voice was applied.

Previously, if the content was enlarged by the radius of the moving area of the pointer, this study expanded the range of selection to access the content of the web document and select only the desired part by item or sentence unit. Here, the content is set to the content of the menu and body area of the web page that the user can browse. As content selection methods, individual element selection and range selection methods were presented. The individual element content selection is designed to individually select the content of the screen in units of sentences or paragraphs. The range selection was designed to be selected at once, even to an ancestor node containing a selection sentence through a double tap function. In addition, a yellow background color was designated for the

selected area so that it was intuitive to know whether it was selected.

We developed a PC-based eye-tracking and voice-command web browser called Eye-Voice for persons with upper-limb disability [19]. The pointer execution method of the existing gaze tracking technology causes a malfunction of pointer execution. These malfunction problems cause the problem of unintended execution of objects and difficulty of pointing due to the small size of execution objects. The eye-tracking technology was used to move the pointer, whereas the voice-command technology was used to perform pointer clicks. Subsequently, we developed a function that automatically expanded only objects that can be clicked in the path of the user's eyes.
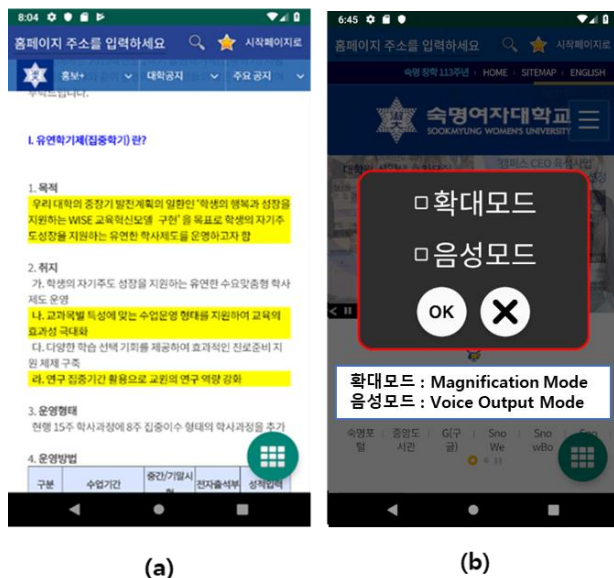

Fig. 3. Screenshot of selective focusing application (a) Select individual element contents by touching (b) Select screen Magnification mode or Voice output mode.
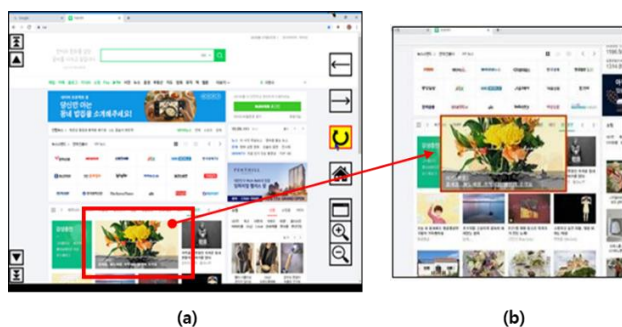

Fig. 4. Eye-Voice Browser (a) Original content size (b) Magnify content when on the path of eye movement.

4.2 Evaluation

The usability of the pilot solution for each disability type was verified. Tests were conducted to perform evaluations under the approval of the Institutional Review Board, which

has been established and operated to secure human bioethics and safety for human subject research conducted at Sookmyung Women's university in Korea, as prescribed by the Act on Bioethics and Safety. To ensure voluntary participations, subjects were recruited through a recruitment notice posted on the researchers' university bulletin board and on the website of a relevant research center.

There were a total of 10 participants in each experiment; they wore an eye patch for the blind test. In the experiment involving participants with low vision, those with myopia removed their glasses. In experiments involving participants with upper-limb disability, the participants were restricted from using their arms. The experimental results are presented subsequently.

Table 4. Results of the test for the blind.

Task (#)	Voice Note		Voice Memo	
	Duration (sec.)	Retries (count)	Duration (sec.)	Retries (count)
1	27.55	1.5	36.41	1.36
2	15.77	0.64	31.82	0.86
3	19.23	0.82	40.50	0.77

To verify the usability of the Bluetooth remote control voice output application for blind people, we conducted an experiment comparing the efficiency and accuracy of the menu with those of the "Voice Note" application. The result indicated that the Voice Note application required 20.85 s and 0.98 times, whereas Voice Memo (our application) required 36.24 s and 1.00 time. The number of retries between the two applications did not exhibit any significant difference in a T-test of t = 0.31. This means that Voice Memo will require a slightly longer time while the blind person is using the remote control, compared to a sighted person using the screen; however, the accuracies were the same. Table 4 is the average of the results measured by conducting the experiment twice for each task.

To verify the effectiveness of the selective focusing mobile voice web browser for the visually impaired, a comparison experiment with the "android talkback" service was conducted. In the experiment, the accuracy of element selection, shortening of reading time, and satisfaction were compared. The experimental results are shown in Table 5.

The results indicated that the talkback service required 1.02 times and 56.48 s on average, whereas the voice web browser required 0.29 times and 40.16 s. The results of the T-test confirmed a significant difference (p = error rate 0.042, execution time = 0.025 s). Compared to the talkback service, the voice browser reduced both the error rate and

execution time such that only the desired area can be read quickly and accurately. The satisfaction assessment confirmed the effectiveness of the selective focusing technique and reduced user fatigue.

Table 5. Results of the test for the low vision.

Task (#)	Complexity	Voice Browser		Talkback	
		Times (Sec.)	Retries (times)	Times (Sec.)	Retries (times)
1	Simple	16.4	0.20	29.15	2.2
	Medium	26.67	0.25	28.37	1.15
	Complex	27.35	0.82	37.42	1.7
2	Simple	34.68	0.2	39.4	0.1
	Medium	35.77	0.12	41.21	0.62
	Complex	37.37	0.43	48.33	0.62
3	Simple	47.7	0.05	65.77	0.63
	Medium	69.29	0.1	107.47	1.25
	Complex	66.25	0.5	111.2	0.92

Comparative experiments on the Eye-Voice web browser for people with upper-limb disability were conducted to verify the reduction in the malfunction of pointer execution with existing gaze interfaces (blinking, dwell time). Table 6 shows the results of experiments for each interface.

Table 6. Results of the test for people with upper-limb disability.

Task (#)	Dwell-Time Interface		Blink Interface		Eye-Voice Interface	
	Number of Attempts	Duration (sec)	Number of Attempts	Duration (sec)	Number of Attempts	Duration (sec)
1	1.35	68.90	1.38	57.99	1.23	50.19
2	1.73	83.15	2.01	109.80	1.70	82.85
3	2.50	162.05	2.57	147.25	2.24	117.90
Overall Average	1.86	104.70	1.99	105.01	1.72	83.65

The results indicated that the dwell-time interface required 1.86 times and 104.70 s; the blinking interface required 1.99 times and 105.01 s; the Eye-Voice interface required 1.73 times and 83.65 s. According to the T-test for the number of retries, no significant difference was observed between the Eye-Voice and the dwell-time interfaces; however, a significant difference was observed in the comparison with the blinking interface ($p = 0.08$; $p = 0.002$). In the T-test, the measured execution time was significantly different from that of the Eye-Voice at both interfaces ($p = 0.0006$; $p = 0.02$). This means that the Eye-

Voice reduced the malfunction rate of the pointer execution, thereby verifying its effectiveness.

V. CONCLUSION

In this study, problems of interaction control in the web environment were analyzed, and solutions for people with various disability types were suggested. Essential interactions in the web environment were derived, and the disability types were reclassified in terms of device accessibility. In addition, we analyzed the problems for each disability type when performing web interactions and suggested solutions. Furthermore, we developed a pilot multimodal interface to apply the solution.

We developed a remote-control operation voice interface for blind people and confirmed that no significant difference was observed from sighted people using the screen. By developing a voice output interface applying the selective focusing technique for low vision, we confirmed that the selective focusing function was fast and effective and reduced user fatigue. Finally, we developed a gaze-tracking and voice-command interface for GUI operations for people with upper-limb disability and confirmed that the malfunction rate of the pointer execution reduced in comparison with those of existing gaze-tracking systems.

This study confirmed that improved usability and accessibility enabled people with different disability types to control digital devices more easily. These findings offer independent learning for the disabled through digital devices and the web, thereby leading to improved economic stability for the disabled.

REFERENCES

[1] G. Jung, S. Ahn, and J. Lee, "Night school teachers' perceptions and current status about assistive technology application in night school for people with disabilities," *Journal of Special Education & Rehabilitation Science,* vol. 55, no. 3, pp. 281-304, Sept. 2016

[2] H. Park and H. J. Kim, "Disability Economic Activity Survey Report," Korea Employment Agency for Persons with Disabilities, Seoul, Dec. 2017

[3] K. S. Hong and H. K. Min, "A Study on the smart device of accessibility for persons with disabilities", *Journal of*

Rehabilitation Welfare Engineering & Assistive Technology, vol. 9, no. 1, pp. 23-28, Feb. 2015

[4] Y. K. Yoon and J. H. Wi, "A Study on the Evaluation and Analysis of Low Visual Satisfaction for Domestic Portal Mobile Web," *Journal of Cultural Product & Design*, vol. 54, no. 0, pp55-65, Sept. 2018

[5] VoiceNote, Dec. 2019; https://play.google.com/store/apps/details?id=xyz.may nine.voicenote

[6] AudioNote, March. 2020; https://play.google.com/store/apps/details?id=com.lum inant.audionote

[7] Voicemon, Jan. 2020; http://www.voicemon.com/

[8] J. W. Kim, Y. K and K. M. Kim, "Table Structure Recognition in Images for Newspaper Reader Application for the Blind," *Journal of Korea Multimedia Society*, vol. 19, no. 11, pp. 1837-1851, 2016.

[9] VoiceOver, "IOS Accessibility," Jan. 2020; https://www.apple.com/kr/accessibility/

[10] Talkback, "Android Accessibility," Jan. 2020;https://support.google.com/accessibility/android/ answer/6006564?hl=ko&ref_topic=6007234

[11] Alastair Chetcuti and Chris Porter, "Butterfleye: Supporting the Development of Accessible Web Applications for Users with Severe Motor-Impairment," in *Proceedings of the 30th International BCS Human Computer Interaction Conference*, Poole, pp. 1-3, July. 2016.

[12] Menges, Raphael, Chandan Kumar, Daniel J Müller and Korok Sengupta. "GazeTheWeb: A Gaze-Controlled Web Browser," in *Proceedings of the 14th Web for All Conference on The Future of Accessible Work*, Perth, Article No. 25, April. 2017.

[13] J.H. Park, "Multimodal interface to improve digital device accessibility for the people with disabilities in web environment," Ph.D. dissertation, Department of IT Engineering, Sookmyung Women's University, Seoul, Korea, 2020.

[14] B. A. Myers, "A taxonomy of window manager user interfaces," *IEEE Computer Graphics and Applications*. vol. 8, no. 5, pp. 65-84. Sept. 1988

[15] Brad A. Myers. "A new model for handling input," *ACM Transactions on Information Systems (TOIS),* vol. 8, no. 3, 289–320. July 1990

[16] Information Society Agency in Korea, "2018 Digital Information Gap Survey," Ministry of Science and ICT in Korea, Seoul, Feb. 2019

[17] J. H. Park, S. B. Lim, J. H. Yook, and J. W. Lee, "An analysis on the disability types and requirements for developing daisy reading assistive devices," *Journal of Special Education & Rehabilitation Science*. Vol. 56, no. 3, pp. 503-520, Sept. 2017

[18] S. B. Lim, M. Lee, E. Choi, J. H. Yook, J. H. Park and J. W. Lee, "Mobile Voice Note File Management Service For Improving Accessibility of the Blind," *Journal of Korea Multimedia Society*, vol. 22, no.11, pp.1215-1222, Nov. 2019

[19] J. H. Park, M. H. Park and S. B. Lim, "A Proposal of Eye-Voice Method based on the Comparative Analysis of Malfunctions on Pointer Click in Gaze Interface for the Upper Limb Disabled," *Journal of Korea Multimedia Society*, vol. 23, no.4, pp.566-573, Apr. 2020

Authors

Joo Hyun Park received her BS and MS degrees in the Department of Multimedia Science from Sookmyung Women's University, Korea, in 2010 and 2012, respectively. In 2020, She received a PhD degree in the Department of IT Engineering from Sookmyung Women's University, Korea. In March 2020, she joined in the Research Institute of ICT Convergence, Sookmyung Women's University and Research Center of Digital Equity 4 All, Korea where she is a senior researcher.
Her research interests include Accessibility for the disabled, UI/UX, Accessibility, Web/Mobile Multimedia and Multimodal Interface.

Jongwoo Lee is a Professor of IT Engineering at Sookmyung Women's University in Seoul, Korea. He received his BS, MS, and PhD degrees in Computer Engineering from Seoul National University in 1990, 1992, and 1996. From 1996 to 1999, he worked for Hyundai Electronics Industries, Co. He was an Assistant Professor of Division of Information and Telecommunication Engineering at Hallym University in Chooncheon, Korea from 1999 to 2002. He was a development director of Inixsoft from 2003 to 2004. He got a research scholar at Stony Brook University in 2008. He was a head of the intelligence office in Sookmyung Women's University from 2012 to 2013. He is a visiting professor at the City University of New York: John Jay College from 2015 to present. His research interests include mobile system software, storage systems, computational finance, cluster computing, parallel, Computing for the blind people and distributed operating systems, embedded system software.

Soon-Bum received his BS degree from Seoul National University, Korea, in 1982 and his MS and Ph.D degrees in computer science from KAIST(Korea Advanced Institute of Science and Technology), Korea, in 1983 and 1992, respectively. From 1989 to 1997, he was the engineering director for the font technology and printer division at Human Computer, Inc, and Trigem Computer, Inc. From 1997, he was an assistant professor in the Dept. of Computer Science at Konkuk University in Korea. From 2001, he is currently a Professor of IT Engineering at Sookmyung Women's University in Seoul, Korea. His main research interests are computer graphics, web and mobile multimedia contents, user interface, and electronic publishing such as font, eBook, and XML documents.