SnakeWindows: A New Way to Manage Windows on AR Glasses

Cen Yao
Lenovo
No. 6 Shangdi Xi Road,
Beijing, China
yaocen1@lenovo.com

Yucheng Jin
Lenovo
No. 6 Shangdi Xi Road,
Beijing, China
jinyc2@lenovo.com

Abstract
Unlike mobile devices and PCs, AR glasses offer an extendable interaction space where a large collection of content could be presented more efficiently. In this paper, we present *SnakeWindow*, a new way to organize information in AR environment by leveraging Windows - a frequently used content container. In particular, our work includes content navigation, windows management, and windows adjustment. We prototyped our design by taking the advantages of interaction space of AR glasses, our work offers an insight into efficient information organization in AR environment and enhances the windows management on AR glasses.

Author Keywords
Windows management, Augmented reality, AR glasses, user interface, interaction

ACM Classification Keywords
H.5.2 [User Interfaces]: Graphical user interfaces (GUI).

Introduction
Augmented reality (AR) technology has been becoming increasingly popular. Many companies have implemented AR technology on head-worn devices such as AR glasses. The AR glasses have an extendable interaction space and support multi-modal interactions. According to the input

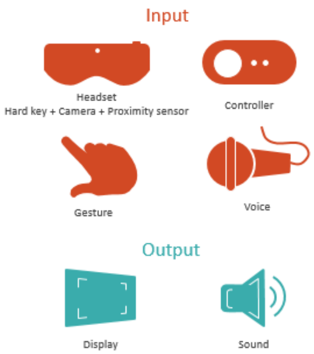

Figure 1: Common interaction modalities on AR glasses.

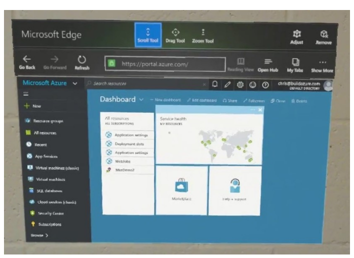

Figure 2: The windows designed in HoloLens.

and output tasks, a proper interaction method will be selected [1]. For AR glasses, the input methods mainly include head gaze, physical buttons on the glasses, mid-air gestures, and voice interaction. And the output should include a visual display, voice feedback, and haptic feedback (see Figure 1). Usually, head gaze and gesture are regarded as more stable interactions than voice interaction on AR glasses. Therefore, *SnakeWindows* mainly works with gaze and gesture controls.

Related work

Nowadays, to enhance the experience of using AR glasses, the AR researchers intend to extend the narrow field of view (FOV) of AR glasses and improve display effects for clear image [3]. Nevertheless, the user interfaces design for the AR glasses still follow the UI design guidelines for PC and mobile. Like windows in HoloLens, it still keeps the main layout used on PC system and put the content into a window with the back button and other adjustment buttons on the top. See Figure 2. Several AR researchers have widely discussed the fundamental differences between traditional GUIs and AR based user interface and the necessity of having specific design principles for AR based interface [4]. Window is the main container for organizing content, and overlapping/ dragging, resizable windows on desktop computers have become the standard for most users [6]. However, due to the limited display size of mobile devices, content navigation relies on the back and forward operations in a single window, which does not take full advantage of extendable AR design space. Based on the extensive investigation of the windows design on existing AR glasses, we found they are quite similar to the windows on traditional devices without considering the broader interaction space and user scenarios of glasses [2, 5]. E.g., adjusting the distance of the window and managing a large number of windows in 3D space are challenging on AR glasses.

Therefore, we intend to improve the window management on AR glasses from these aspects. First, we present a new way to navigate the user to different content by creating a new window for each content page in an application and wire them using dotted lines. Second, we provide visual guides and feedback to facilitate the windows adjustment in 3D space such as resizing, rotating, adjusting the distance. Third, we offer a shortcut key for clustering related windows in 3D space quickly.

The concept of window

In our work, window is defined as a container for a *single* content page in an application, which means every time when the user is navigated to another page a new window will be created. As a result, based on an activity flow of an application, several windows will be displayed in space; and the dotted lines indicate the parent-child relations between windows (See Figure 3).

We give an example of using *SnakeWindows* in an AR application. As shown in Figure 3, a user opens the remote assistance application. Initially, the home page window contains a contact list (see Figure 3a).Then the user selects an expert from the contact list, a new window showing a tutorial video (see Figure 3b) will appear at the center of FOV, and the home page moves to the left. If the user clicks the icon for showing more available expert, a new window will be created (see Figure 3c). Moreover, these created windows are wired by dotted lines which will be highlighted when the user drags the wired windows. When close a window, the windows belong to which should be closed as well. If an application contains multi-threading tasks, multiple sub-windows for the tasks can be displayed simultaneously. In fact, *SnakeWindows* has a **flat structure for managing content**, which allows the user to go through the content more efficiently than using back and forward navigation.

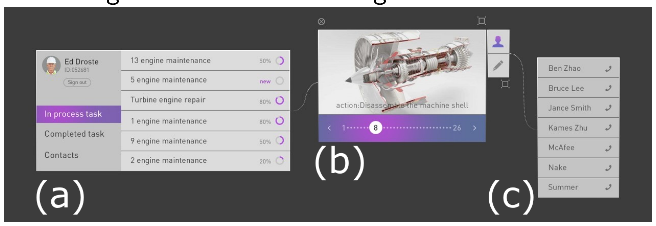

Figure 3: An example of SnakeWindows - remote assistance.

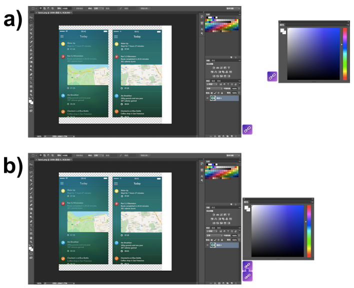

Figure 4: The "link-link" button to gather the related windows. a): two separated windows without pressing the "link-link" button. b): two connected windows after pressing the "link-link" button.

There are three advantages of *SnakeWindows*. Firstly, it makes full use of extendable space to show massive content in AR space. Secondly, multiple sub-windows allow different task flows to work at the same time, which facilitates the performance of multi-threading tasks. Third, by moving the head, the user can quickly go through the content in different windows instead of using back and forward to navigate step by step.

Window management

It is challenging to manage a large number of windows on AR glasses. We therefore provide a manager of application windows which allows the user to quickly gather all the windows of an application, and the windows of an application will be piled behind the application icons (see Figure 5).

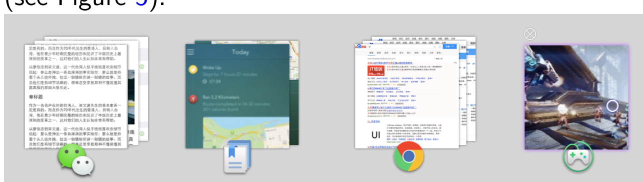

Figure 5: Manager of application windows.

If the users gaze focuses on a windows group, the windows will be spread for further operations (see Figure 6). Then, a window can be selected to open or close easily.

Figure 6: Multiple windows in an application

In addition, the windows of an application may be distributed randomly in space. Most of the time, they are not displayed in users view simultaneously due to the limited FOV. Thus, if the user needs to coordinate the multiple windows to perform a task, she needs to manually drag the windows from the different places. While we offer a link-link button beside each window. By clicking it, the related windows will be attached to the active window (see Figure 4).

In this way, searching for one specific window and coordinating multiple windows are more efficient.

Windows adjustment

Due to the poor human perception of size and distance in 3D space [7], adjusting windows on AR glasses is difficult. Therefore, by providing the visual guides, we intend to facilitate the adjustment of the window size, window distance, and its position.

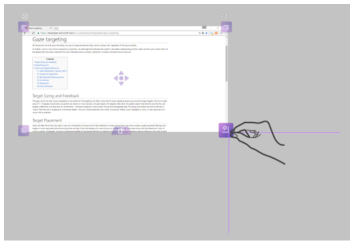

Figure 7: The controls for resizing the window.

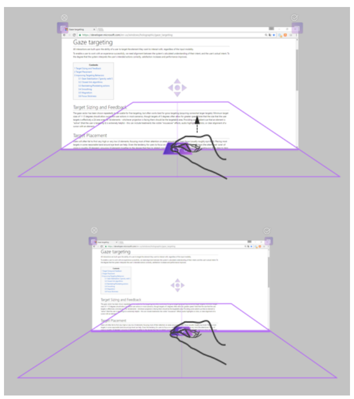

Figure 8: The controls for adjusting the distance of window along Z-axis.

There is an adjustment button on the toolbar of a window. After clicking this button, the editing mode will be activated. The system will reduce the brightness and highlight the adjustment controls. The buttons at four corners of the window are used to adjust the window size (see Figure 7). Besides, the button at the center can be dragged to change the windows position. The button at the bottom of the window is used to adjust the window distance (see Figure 8), which translates a non-intuitive operation along Z-axis to the vertical adjustment on a 2D plane. It is worth noting that the visual guides appear in the process of adjustment, which provides real-time feedback to the user.

Conclusion

In this paper, we present *SnakeWindows*, a new way to manage the windows on AR glasses. This new design is dedicated to an efficient way to navigate the content of an application. Moreover, the visual guides allow the user to adjust the windows easily and precisely. When the number of windows increases, the link-link button can be used to gather related windows quickly. More importantly, unlike using back and forward to navigate the content, our design employs a flat structure to organize information more efficiently on AR glasses. This kind of layout can make full use of the extendable AR space, and present the trend of border-less design and flat interface. Hopefully, our work provides a new insight and guidance to practitioners and designers for building applications on AR glasses. In the future, we will apply the *SnakeWindows* to a complete application and run a usability test.

References

[1] New interface for smart glasses goes live. https://www.otgplatforms.com/blog/2014/04/new-interface-for-smart-glasses-goes-live/.

[2] Profiles in inovation: Virtual & augmented reality understanding the race for the next computing platform. http://www.goldmansachs.com/our-thinking/pages/technology-driving-innovation-folder/virtual-and-augmented-reality/report.pdf.

[3] Billinghurst, M., Clark, A., Lee, G., et al. A survey of augmented reality. *Foundations and Trends® Human–Computer Interaction 8*, 2-3 (2015), 73–272.

[4] Dϋnser, A., Grasset, R., Seichter, H., and Billinghurst, M. Applying hci principles to ar systems design. *HIT Lab NZ, University of Canterbury, New Zealand* (2004).

[5] Poupyrev, I., Tan, D. S., Billinghurst, M., Kato, H., Regenbrecht, H., and Tetsutani, N. Developing a generic augmented-reality interface. *Computer 35*, 3 (2002), 44–50.

[6] Shneiderman, B. *Designing the user interface: strategies for effective human-computer interaction*. Pearson Education India, 2010.

[7] Wann, J. P., Rushton, S., and Mon-Williams, M. Natural problems for stereoscopic depth perception in virtual environments. *Vision research 35*, 19 (1995), 2731–2736.