VolGrab: Realizing 3D View Navigation by Aerial Hand Gestures

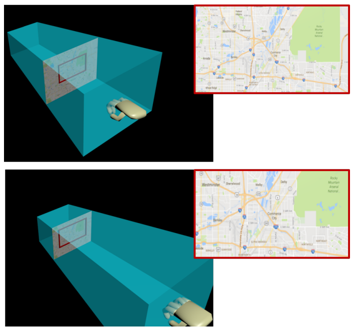


Figure 1: Operation of VolGrab

Shun Sekiguchi
Saitama University
255 Shimo-Okubo, Sakura-ku,
Saitama City, 338-8570, Japan
sekiguchi@is.ics.saitama-u.ac.jp

Takashi Komuro
Saitama University
255 Shimo-Okubo, Sakura-ku,
Saitama City, 338-8570, Japan
komuro@mail.saitama-u.ac.jp

CHI'17 Extended Abstracts, May 06-11, 2017, Denver, CO, USA

ACM 978-1-4503-4656-6/17/05.
http://dx.doi.org/10.1145/3027063.3053108

Abstract
In this paper, we propose a user interface that combines the existing touch interface with 3D view navigation by aerial hand gestures and that realizes simultaneous zooming and scrolling. While the touch interface enables only planar operation, aerial hand gestures enable 3D operation with an increased number of axes, and enables smooth navigation that integrates zoom and scroll operations by assigning the direction that is normal to the screen to zooming and the direction that is parallel to the screen to scrolling. We implemented a map application based on the proposed interface to confirm the effectiveness of the proposed interface.

Author Keywords
Zoom interface; gesture input; map navigation

ACM Classification Keywords
H.5.2 [User Interfaces]: Input devices and strategies

Introduction
Recently, touch screens are widely used as input interfaces for computers such as PCs and smartphones. Since the amount of contents that can be displayed on a screen at a time is limited due to the screen size, there are demands for changing the size of displayed letters or images on the

screen depending on the situation. View navigation such as zooming and scrolling is used for such a purpose. While some view navigation methods on a touch screen have been proposed such as specifying a zoom area by touching a touch screen with two hands [4] and performing zoom operation by rubbing a touch screen [9], the method of scrolling by dragging and swiping on a touch screen and zooming by pinching like in Google Maps has become common in these days. On the other hand, there have been research of view navigation methods without using a touch screen such as those using a mouse [2, 3, 13] and aerial gestures [1, 14]. There has also been research of changing the contents to be displayed during navigation to make the operation easy such as changing the zoom ratio depending on the scroll speed [6]. As the research that extends the existing touch interface, the system that enables multi-layer operation by setting an IR sensor on a touch screen [12] and the system that obtains the posture of a touching finger using three laser planes at different heights [11] have been developed.

User interfaces that combine the touch interface with aerial gestures have also been proposed. As an extension of existing view navigation methods, there has been the methods that assigns aerial finger gestures to zooming and scrolling together with the existing touch interface [5, 7, 10] and the method that combines operation on a touch screen with operation by aerial finger movement obtained by using motion capturing [8]. However, there is a problem that it is difficult for a user to perform zoom and scroll operations simultaneously since most of the methods assign each of zoom and scroll operations to an independent gesture command.

Therefore, in this paper, we propose a user interface that combines the existing the touch interface with 3D view navigation operation by aerial hand gestures. Since zooming and scrolling are performed frequently in view navigation, smooth navigation is realized by enabling users to perform these operations simultaneously. Since the proposed interface includes all functions of the existing touch interface, and in addition to that, the interface gives the function of 3D view navigation as an extension to the touch interface, it is expected that the proposed interface allows users smoother operation than the existing touch interface.

VolGrab

VolGrab(Volume Grabbing Interface) is a user interface that enables 3D view navigation by aerial hand gestures. While the touch interface enables only planar operation, aerial hand gestures enable 3D operation with an increased number of axes, and enables smooth navigation that integrates zoom and scroll operations by assigning the direction that is normal to the screen to zooming and the direction that is parallel to the screen to scrolling. In VolGrab, the view navigation function is enabled only when the user's hand is in a grabbing posture. This enables the user to perform consecutive zoom and scroll operations by repeating grabbing and releasing, which is a similar way to the ones in dragging a mouse and swiping a touch screen. Additionally, since navigation is disabled when the hand is not in a grabbing posture, the function of view navigation by aerial hand gestures does not interfere with the existing touch interface. Therefore, to the user can perform view navigation by aerial gestures and then perform selection operation by touching the screen with keeping the view.

3D view navigation by aerial hand gestures

VolGrab enables view navigation by grabbing and moving information volume in the shape of a pyramidal frustum as shown in Figure 2. This information volume is a virtual and invisible object that consists of a pile of images with dif-

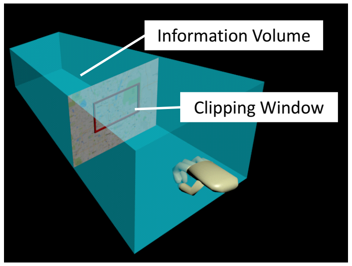


Figure 2: Concept of VolGrab

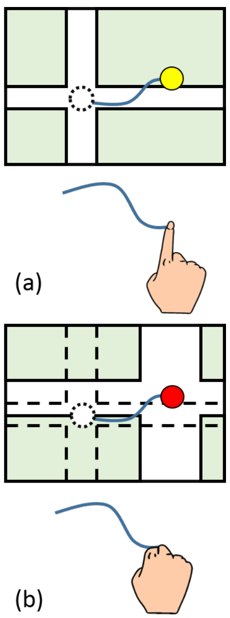


Figure 3: 3D navigation method (a)Non-grabbing posture, (b)Grabbing posture

ferent scales. The information volume is translated by the amount of movement of the hand in a grabbing posture. The part cropped by a red frame called a clipping window in the information volume is displayed on the screen. By making the information volume that consists of a pile of images resized in varying scales, the movement of the information volume in the direction that is normal to the clipping window is assigned to zooming and the movement in the direction that is parallel to the clipping window is assigned to scrolling. This enables integrated zoom and scroll operations by moving the information volume in a 3D space.

Figure 1 and Figure 3 show how a user performs 3D view navigation using aerial hand gestures. Figure 1 shows the user grabbing and moving the information volume to perform zoom and scroll operations. Figure 3 shows the user performing view navigation using aerial hand gestures. The circle on the screen is a pointer which is drawn at the position where the position of the hand is orthogonally projected on the screen. In Figure 3 (a), where the hand is in the non-grabbing posture, only the pointer is moved, whereas in Figure 3 (b), where the hand is in the grabbing posture, both the pointer and the view is moved.

When the shape of the information volume is pyramidal as in Figure 2, the change of zoom ratio is proportional to the amount of orthogonal movement. On the other hand, exponentiall zooming is also possible by making the information volume that consists of resized images in exponentially varying scales. Furthermore, when the information volume consists of images that have not only different scales but also different levels of detail, it is possible to change displayed contents according to the scale in the applications such as a map viewer.

Usage scenarios

There are two patterns of VolGrab's usage scenarios: the case when the target of view navigation is the entire screen, and the case when the target is a part of the screen. A typical usage in the former case is map viewing. Map viewer applications such as Google Maps are often used to search a particular place and confirm the way to the place. Figure 4 shows the scenario that the user wants to find the way from the zoo to the airport. To find the way from the zoo to the airport, the user needs to zoom in the map around the zoo (Figure 4(a)) and around the airport (Figure 4(c)), and to zoom out to see the overview (Figure 4(b)). In such a case, zoom and scroll operations have to be performed frequently. Using the proposed interface, the user can perform zoom and scroll operations by one motion, which reduces the number of required steps from those using the existing mouse or touch interface.

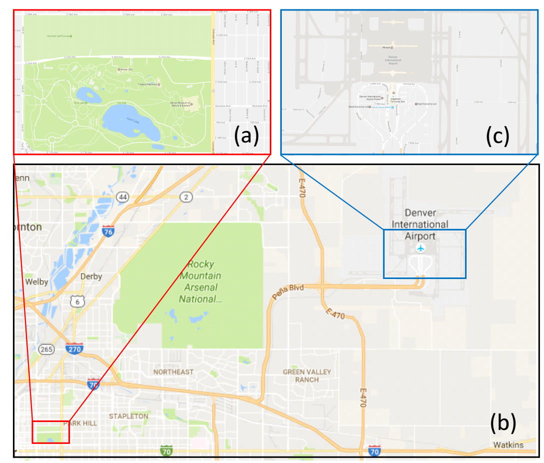


Figure 4: Usage scenario on a map application: (a) Zoo, (b) Overview, (c) Airport

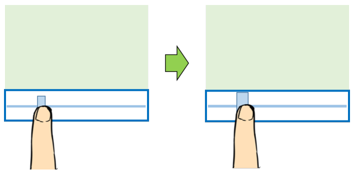


Figure 5: Zooming a slider

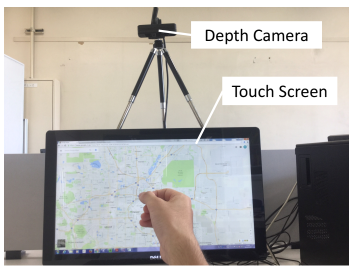


Figure 6: System configuration

On the other hand, a typical usage in the case when the operation target is a part of the screen is operation of UI components such as buttons and scrollbars. Small UI components such as buttons in a toolbar often cause unintended operation such as erroneous selection. Such errors can be reduced if the UI components are enlarged temporarily. For instance, the width of the UI components such as a scrollbar or a slider can be changed by zooming in/out the bar or slider as shown in Figure 5, which makes it easier to finely adjust the position.

Implementation

We implemented the VolGrab interface on a prototype system. In the following subsections, we describe the system configuration, the method for detecting grabbing hand postures and an example of use.

System configuration

The configuration of the prototype system is shown in Figure 6. In this prototype, we assume the UI for desktop PCs since desktop PCs that allow touch operation have increased recently. A 21.5 inch full HD multi-touch monitor is used for obtaining touch operation. A depth camera DS325 (Softkinetic) that is fixed above the screen using a tripod is used for obtaining aerial hand gestures. The size of depth images captured by the depth camera is 320×240 pixels, the range of obtainable depths is 0.15−1.0m, and the frame rate of capturing depth images is 60 fps. Using depth images from the depth camera, it becomes easier to detect the position and posture of the hand than using color images In order to detect the grabbing hand posture, a hand region is detected from a captured depth image, and whether the hand is in the grabbing posture or not is determined from the shape of the hand. In addition, the hand movement relative to the screen is obtained by using the positions of the detected hand and the screen, and the amount of zooming and scrolling is calculated according to the movement.

Figure 7: Grabbing posture detection:
(a) Non grabbing posture, (b) Grabbing posture

Detection of grabbing hand postures from depth images

In this implementation, we define the state in which the finger is stretching as the non-grabbing posture, and not stretching as the grabbing posture. Figure 7 shows an example of grabbing posture detection. The hand region is scanned from the screen side to the user's side and the region from where the fingertip is detected to where the finger width becomes larger than a threshold, which is the maximum of the assumed finger widths, is defined as the finger region. The extracted finger regions are shown in red in Figure 7. The length of the finger region is defined as the finger length. If the obtained finger length exceeds a certain threshold, the state is regarded to be the non-grabbing posture, and otherwise the grabbing posture.

Example of use

We created an application that allows view navigation of Google maps by aerial hand gestures that works in a web browser. In this application, the information volume that varies the zoom ratio exponentially is used. Figure 8 shows the user actually performing view navigation in the case of going from the zoo to the airport. First, the detail view around the zoo (Figure 8(a)) is displayed, then the overview that covers both the zoo and the airport (Figure 8(b)) is displayed by zooming out, and finally the detail view of the airport (Figure 8(c)) is displayed.

Conclusion

In this paper, we proposed a user interface that combines the existing touch interface with 3D view navigation by aerial hand gestures and that realizes simultaneous zooming and scrolling. We implemented a map application based on the proposed interface to confirm the effectiveness of the proposed interface. Future work includes implementation of zooming and scrolling of UI components and the usability testing with participants.

References

[1] Amartya Banerjee, Jesse Burstyn, Audrey Girouard, and Roel Vertegaal. 2011. WaveForm: remote video blending for VJs using in-air multitouch gestures. In *CHI'11 Extended Abstracts on Human Factors in Computing Systems*. 1807–1812.

[2] Benjamin B Bederson. 2000. Fisheye menus. In *Proceedings of the 13th annual ACM symposium on User interface software and technology*. 217–225.

[3] Benjamin B Bederson and James D Hollan. 1994. Pad++: a zooming graphical interface for exploring alternate interface physics. In *Proceedings of the 7th annual ACM symposium on User interface software and technology*. 17–26.

[4] Hrvoje Benko, Andrew D Wilson, and Patrick Baudisch. 2006. Precise selection techniques for multi-touch screens. In *Proceedings of the SIGCHI conference on Human Factors in computing systems*. 1263–1272.

[5] Xiang Anthony Chen, Julia Schwarz, Chris Harrison, Jennifer Mankoff, and Scott E Hudson. 2014. Air+ touch: interweaving touch & in-air gestures. In *Proceedings of the 27th annual ACM symposium on User interface software and technology*. 519–525.

[6] Takeo Igarashi and Ken Hinckley. 2000. Speed-dependent automatic zooming for browsing large documents. In *Proceedings of the 13th annual ACM symposium on User interface software and technology*. 139–148.

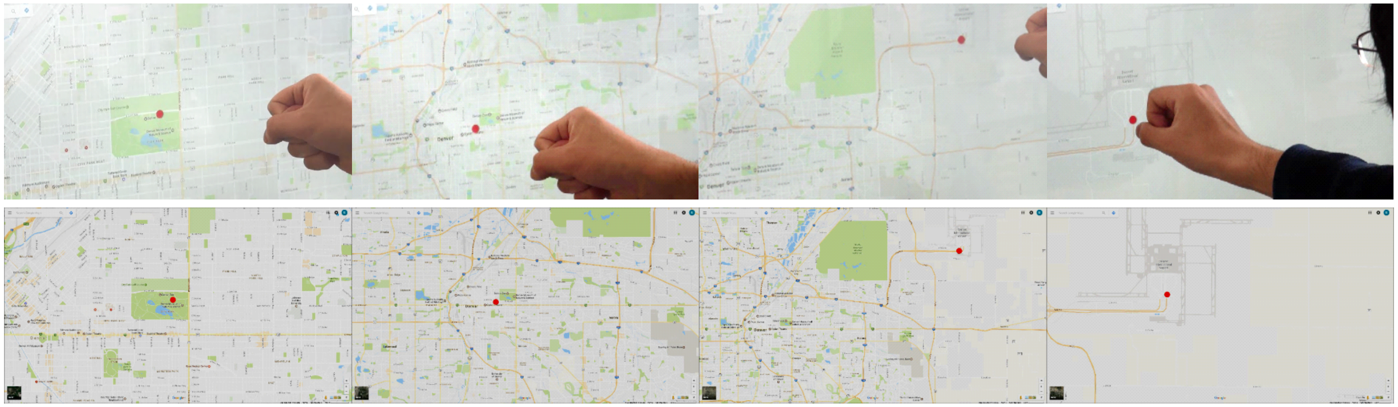
[7] Sven Kratz and Michael Rohs. 2009. HoverFlow: expanding the design space of around-device interaction. In *Proceedings of the 11th International Conference on Human-Computer Interaction with Mobile Devices and Services*. 4.

[8] Nicolai Marquardt, Ricardo Jota, Saul Greenberg, and Joaquim A Jorge. 2011. The continuous interaction space: interaction techniques unifying touch and gesture on and above a digital surface. In *Human-Computer Interaction–INTERACT 2011*. Springer, 461–476.

[9] Alex Olwal, Steven Feiner, and Susanna Heyman. 2008. Rubbing and tapping for precise and rapid selection on touch-screen displays. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 295–304.

[10] Jie Song, Gábor Sörös, Fabrizio Pece, Sean Ryan Fanello, Shahram Izadi, Cem Keskin, and Otmar Hilliges. 2014. In-air gestures around unmodified mobile devices. In *Proceedings of the 27th annual ACM symposium on User interface software and technology*. 319–329.

[11] Yoshiki Takeoka, Takashi Miyaki, and Jun Rekimoto. 2010. Z-touch: an infrastructure for 3d gesture interaction in the proximity of tabletop surfaces. In *ACM International Conference on Interactive Tabletops and Surfaces*. 91–94.



(a) (b) (c) (d)

Figure 8: Operation example of map application:
(Top row) Operation, (Bottom row) View on a screen

[12] Yujin Tsukada and Takeshi Hoshino. 2002. Layered touch panel: the input device with two touch panel layers. In *CHI'02 Extended Abstracts on Human Factors in Computing Systems*. 584–585.

[13] Fangzhou Wang, Yang Li, Daisuke Sakamoto, and Takeo Igarashi. 2014. Hierarchical route maps for efficient navigation. In *Proceedings of the 19th international conference on Intelligent User Interfaces*. 169–178.

[14] Xing-Dong Yang, Tovi Grossman, Pourang Irani, and George Fitzmaurice. 2011. TouchCuts and TouchZoom: enhanced target selection for touch displays using finger proximity sensing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 2585–2594.