Points of Interest Density Based Zooming Interface for Map Exploration on Smart Glass

Doyeon Kim[1], Daeil Seo[1(✉)], Byounghyun Yoo[1,2], and Heedong Ko[1,2]

[1] Center for Imaging Media Research, Korea Institute of Science and Technology, Seoul, Republic of Korea
ehdus0219@imrc.kist.re.kr, {xdesktop,ko}@kist.re.kr, yoo@byoo.net

[2] Department of HCI and Robotics, University of Science and Technology, Daejeon, Republic of Korea

Abstract. Smart glass devices have become popular in our daily lives, and many people have applied its advantages to their application fields. In particular, smart glass has the advantage of receiving desired information by maintaining their context of the real world without dispersing their attention. Smart glass is also widely used to obtain information about points of interest with map interfaces. However, smart glass has an insufficient user interface with which to interact with its map applications via the smart glass. This problem restricts the map's ability to provide specific functions and makes users feel constrained compared to hand-held devices. In this paper, we focus on a map exploration method for smart glass to solve the limitations of map interaction via the smart glass. We propose an interaction design with zooming interface based on region's point of interest density. This approach is intended for a region search without any other interaction tools. Users search and browse their regions of interest with the smart glass by using the proposed method.

Keywords: Point of interest · Wearable user interface · Wearable user interaction · Smart glass · Region-based navigation

1 Introduction

The advent of mobile computing technology has popularized the use of smart glass devices in our daily lives. Wearable applications for smart glass devices allow people to obtain information through optical see-through display in front of their eyes. This characteristic helps people obtain information by maintaining their context of the real world. Thus, many wearable device applications have been released and have received attention in recent years.

One of the most common application areas for smart glass is map services; people use the map services to browse their surroundings and obtain geometric information including points of interest (POIs). For example, Google Maps service [1] helps people find geometric information such as location of POIs, route information, and detailed information about POIs. These map services provide a keyword search with a speech

S. Yamamoto (Ed.): HIMI 2017, Part I, LNCS 10273, pp. 208–216, 2017.
DOI: 10.1007/978-3-319-58521-5_16

recognition technique to obtain the desired information about POIs by speaking the name of POIs into the microphone integrated in the smart glass device. However, it is difficult to obtain information about regions of interest (ROIs), because current map service applications for smart glass do not provide zooming and panning capabilities. Because of this limitation, users cannot identify their interesting information in ROIs. For example, a user visits New York City for the first time and searches the central park with a keyword search. The map service changes the center of the map to the central park to show a search result. However, the user is not able to browse the surroundings around the central park, because the map service does not provide panning and zooming user interfaces for nearby locations.

In this paper, we propose a POI density based zooming interface for map exploration on smart glass that enables users to zoom and pan their map to examine a desired region. The proposed method is designed with both a keyword search and region-based search. We have implemented the keyword search with speech recognition technique to identify keywords and move the map to the appropriate place. In addition, we provide region search with a POI density based zooming algorithm according to the number of POIs in a region.

This section has introduced and provided the motivation for the work; the rest of this paper is structured as follows. The state of the art and the problem statement are given in Sect. 2. We explain our proposed approach in Sect. 3 and describe architecture and prototype implementation in Sect. 4. We then conclude the paper and consider future works in Sect. 5.

2 Related Work

2.1 User Interface on Smart Glass

User interface designs for smart glass devices have been researched for a long time, ever since Sutherland [2] proposed the first head-mounted display (HMD). In recent years, many smart glass devices have been released in the commercial market, such as Google Glass [3] and Vuzix M100 [4]. These devices have unfamiliar user interfaces [5], and thus researchers have studied how to improve user experiences of smart glass devices [6].

Kerr et al. [7] introduced a hand gesture interface with a colored marker glove to interact with users and their systems. Similarly, Caggianese et al. [8] proposed a hand gestural interface with a depth camera to detect and track a user's hands in the physical space. The study also adopted a Windows, Icons, Menus, and Pointers (WIMP)-style user interface for their proposed system. However, interfaces with hand tracking require additional devices to detect a user's hand. In addition, users with the interfaces have to remember gestures and pay attention to their interactions. Baldauf et al. [9] proposed an vision based eye gaze tracking method that uses computer vision algorithm and mobile sensors to guide geo-referenced digital information. The eye gaze interface has an advantage that easily obtains geo-referenced information around users, but it is difficult to explore a remote site.

In another work, Mayer et al. [10] proposed an approach for controlling smart things recognized by the camera in smart glass devices, controlled via a smart watch, and described with a smartphone. The method provides easy interaction method by applying third party remote controllers as interaction tools. However, the method requires additional devices and only uses smart glass device to identify physical objects.

Previous studies applied third party devices and complicated interfaces to interact with smart glass devices. In addition, the works are not appropriate for exploring a map. To solve the existing limitations, we use the user interface of smart glass devices without any additional devices, and design user interface for map exploration.

2.2 Map Interaction Methods

In traditional handheld and desktop environments, interaction devices such as a two-dimensional multi-touch interface, mouse, keyboard, or trackball are used to communicate with applications.

In handheld devices, map services use a two-dimensional multi-touch interface to zoom or pan a map, and provide a virtual keyboard that enables to search interesting places and areas (e.g., Google Maps [11] and Foursquare [12]). As a representative example, the Google Maps service for handheld device uses a multi-touch interface to zoom a map. A user pinches with his/her thumb and another finger or taps with two fingers to zoom out the map. To zoom in the map, the user spreads two fingers or double taps on the touch pad of the device. The service also provides a keyword search using a keyboard interface.

Users use a mouse wheel and a keyboard as interaction tools to interact with map services in desktop environments. The Google Maps for desktop employs a wheel mouse and a keyboard. The user scrolls the mouse wheel for zooming, or presses the plus or minus keys on the keyboard to zoom in or out. The WIMP interface supports a keyword search and a region search. The WIMP interface enables map services to provide better ease of use for their interactions.

In contrast with handheld and desktop devices, numerous smart glass devices only provide a touch interface, and lack of interactions becomes a limitation. For example, commercial smart glass devices such as Google Glass and Vuzix M100 do not support mouse, keyboard, or multi-touch interfaces. Map services for smart glass have gradually increased to enable smart glass users to search for places, and the map service providers have studied interaction methods to control the map. Map2Glass [13] provides map services to explore interesting places, and the service receives a keyword with keyboard on the smartphone. However, users do not maintain their real-world context because they have to pay attention to the smartphone whenever they search for places. Altwaijry et al. [14] suggested an approach that recognizes landmarks with a computer vision algorithm and provides a guide to the recognized landmarks using the smart glass device. The approach delivers guide information when a user looks at the landmark. However, it is difficult to obtain information at distance, because the approach only works for landmarks that are visible from the user's perspective. Google provides the Google Maps service for Google Glass [1], which searches for places with voice command input. Google Glass users search their desired place by speaking a keyword. However, it is not possible to zoom regions. We split the current region to sub-regions for zooming in an attempt to solve this problem and the smart glass user chooses a ROI by selecting sub-regions.

3 Methodology

The proposed approach in this paper is defined in two ways: The one is a region search with a POI density based zooming interface that is used to divide a given region into sub-regions according to their number of POI groups. The other one is a keyword search using a voice user interface (VUI). In this section, we introduce our proposed map exploration method and interaction design to enable users to explore a map using smart glass. We apply a POI density based region division method for zooming the map and speech recognition technique for panning the map.

3.1 Map Exploration with POI Density

We design our approach to reduce user interactions and maximize the smart glass usability. Because the resolution of the smart glass display is too small to give much information, we cluster POIs into distance based groups to give abstracted information. Figure 1 depicts the comparison between POIs with and without clustering. POIs as shown in Fig. 1(a) occlude the map and other POIs, because there are too many POIs such as historical sites, transportations, and accommodations in the given region boundary. Consequently, a user is unable to find the desired place quickly, thus confusing the user. Therefore, we cluster POIs into groups to clearly see map and each group. The POI clustering method provides an abstraction layer of POI information. Figure 1(b) shows our POI clustering result, which is clustered by distance among POIs. Each POI group has a different color and number; the color represents the density degree of a POI group and the number in the circle icon is the number of POIs in the POI group. For example, the red circle icon shown in the middle of Fig. 1(b) represents the range from 100 to 1000 POIs, and the exact number is 293. The POI groups are recalculated whenever the map change zoom level or boundary.

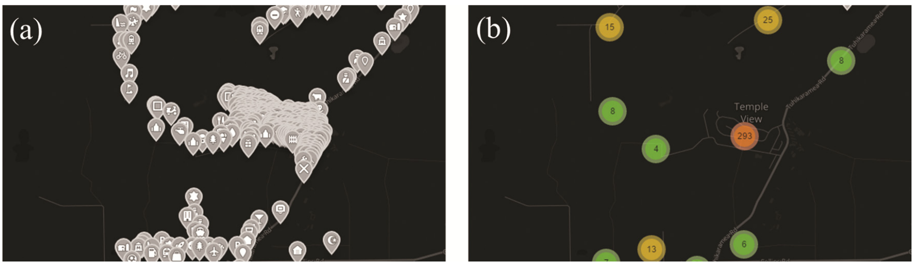

Fig. 1. Comparison between (a) without clustering and (b) with clustering

Figure 2(a) describes the concept of our approach for zooming with smart glass. The user explores a map by zooming with our proposed approach. In our approach, the current ROI is divided into four sub-regions according to the POI density. The regions are used for selecting the next interesting region, and the user selects a sub-region by moving ROI. The selected sub-region is expressed as the orange rectangular box, as shown in Fig. 2(a), and the box is used for zooming the map.

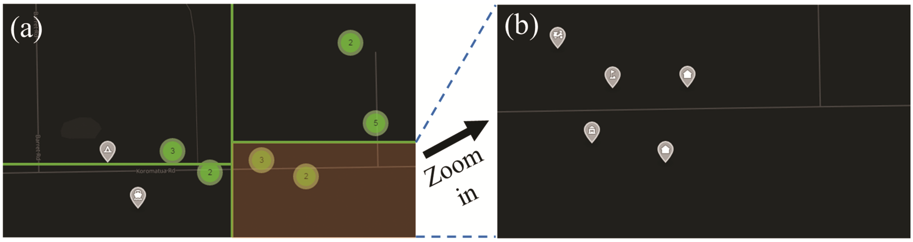

Fig. 2. POI density based zooming interface for smart glass device: (a) an interesting region at the bottom right is selected and (b) the region is zoomed in.

We apply a POI density based zoom interface that divides the region into sub-regions according to the number of POI groups. In other words, each sub-region has an equal number of POI groups. By this approach, a user can find the desired POI more quickly rather than sub-regions divided by the equal region size. For example, the user does not want to explore a region that has no POI such as the top left region as shown in Fig. 2(a). As an aside, POIs are sometimes crowded in a specific region and the user examines the region. The selected interesting region is zoomed in as shown in Fig. 2(b) and the region is divided again as shown in Fig. 2(a). The process is repeated until the user stops the zooming in action because the user finds their desired POI.

3.2 Interaction Design on Smart Glass

In this paper, we apply a VUI in the smart glass to provide a keyword search interface. The keyword search enables a user to find a named POI such as the Eiffel Tower, Disneyland, or Central Park. The advantage of VUI is natural and intuitive for smart glass applications [5]. The proposed system detects a user's speech with a voice recognition technique through the microphone in the smart glass and converts to text keywords using the speech-to-text (STT) technique. The text keywords are sent to the map service to pan the place of interesting on the map.

In addition, the system allows a user to zoom in and out on the map for a region search interface when the user examines an ROI or the area around interesting places. The region search helps the user obtain geometric information or POIs within the ROI. The system provides a touch interface in the smart glass to interact with the map for providing a region search. It minimizes the modification of traditional map interactions with the smart glass and provides a user-friendly interface for smart glass users. Table 1 is an example description of interaction mapping between touch events in the Google Glass and actions on the map with the proposed approach.

Table 1. Mapping of map interactions on the smart glass

Gesture	Action on map
Swipe left	To left region of interesting
Swipe right	To right region of interesting
Tab	Zoom into a selected area

Gesture	Action on map
Long press	Zoom into the center of the current region
Two tab	Zoom out the map

4 System Design

4.1 System Architecture

The proposed approach employs a touch interface and voice command in smart glass device. With the interfaces, the user searches desired for POIs and retrieves the related information about the POI and the region surrounding it. The retrieved POIs appear as several POI groups when there are too many POIs to display on a map.

The proposed system performs the method proposed in this paper in three-parts, as shown in Fig. 3: the smart glass, the POI server, and the map service provider. The map service provider supplies geometric information such as the shapes of buildings and roads. The POI server provides information about POIs in ROI. The smart glass device is connected to the map services provider and the POI server. When a user inputs commands, the smart glass perceives the user's input, and shows geometric information and POIs via the smart glass display.

The input handler receives the user inputs via the microphone and the touch pad of the smart glass. The STT engine converts the user's voice commands to text keywords

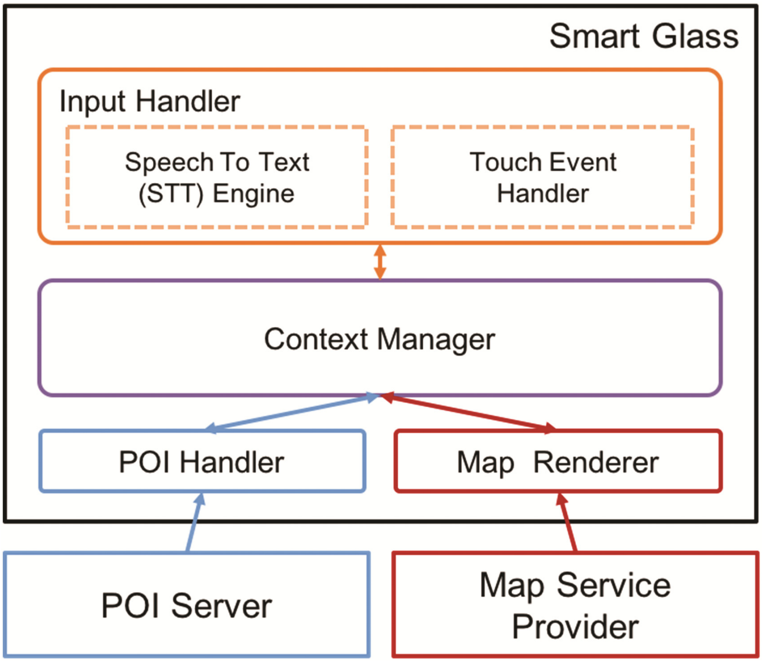

Fig. 3. System architecture

for the keyword search. The touch event handler perceives the user's touch actions for the touchpad on the smart glass. These inputs are sent to the context manager. The context manager manages the user's context. The context of the system is its current zoom level, the position of the ROIs, and the previously selected interesting region. The context is changed depending on the user's commands when user's input is detected. The context manager sends the context to the map handler and POI handler. The map renderer is connected to the map service provider to obtain map tiles, renders the map tiles on the map, and shows POIs in the ROI on the smart glass display. Whenever the map is panned or zoomed, the POI handler requests information of the POIs in the current ROI. Once the POI handler obtains POIs from the POI server, the POI server sends the result to the map renderer through the context manager and the map handler clusters obtained POIs by distance.

The POI server stores the digitalized POI contents, which are used to provide geo-referenced information about POIs to the user. The digitalized POI contents are in JavaScript Object Notation (JSON) format, and the POI schema is described in Table 2. The *ID* field identifies each POI, and it is composed of a unique string. The *name* field contains the representative name in the real world. The *type* field is the category of the POI such as museum, art gallery, or restaurant. The *type* field is used to display POIs on the map as different icons according to the characteristics of the POIs. The *location* field contains geo-location information, which is useful for locating POI icons on the exact location.

Table 2. POI object schema in the POI server

Field	Type	Example
ID	String	ebeff401-9c75-482e-99f2
Name	String	Museum of London
Type	String	Museum

Location	JSON	location: {
		type: "Point",
		coordinates: [
		-0.0967782,
		51.5176183
		]
		}

4.2 Prototype Implementation

The proposed system is developed with Mapbox [15] and Leaflet marker cluster plugin [16] on Google Glass device. Mapbox is a map service provider that provides map tiles and a map event handler based on Leaflet library [17]. Figure 4 shows the process of prototype implementation based on our proposed method. In this scenario, a user visits Seoul, Korea and tries to find the Gyeongbok-gung Palace via their smart glass device. The user speaks the name of the attraction into the microphone on the Google Glass as shown in Fig. 4(a), and the map pans to the place, and POI groups in Gyeongbok-gung Palace are shown as colored circle icons that represent POI groups, as shown in Fig. 4(b).

When a user examines a specific region, the user selects the desired region by moving the position of the orange-colored rectangular box, as displayed in Fig. 4(c) and (d) which shows the result of zooming interaction. After finishing sightseeing Gyeongbokgung Palace, the user visits other places around his/her current location. When the user explores the wider area, the user zooms out the map to the desired zoom level, and the map is zoomed out to the previous ROI, as shown in Fig. 4(e). The user explores ROIs via the smart glass based on their interest level by repeating the process.

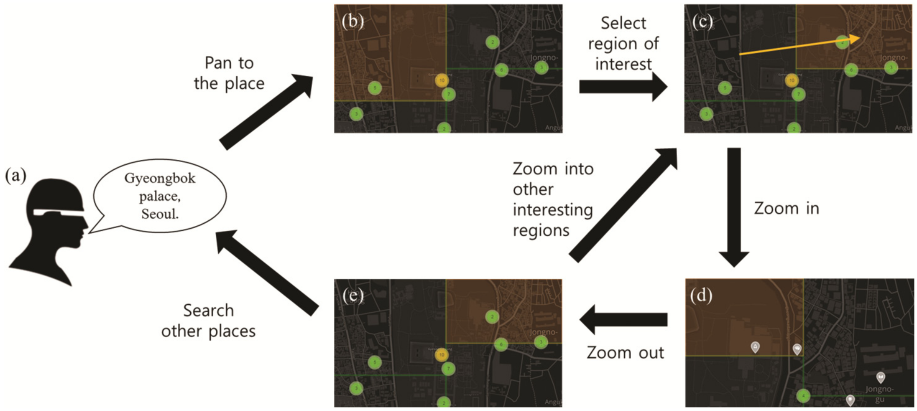

Fig. 4. Process overview of prototype implementation in our approach: (a) A user speaks the name of a POI. (b) The map pans to the location of the POI. (c) The user chooses a ROI. (d) The user zooms in the map to look closely at the interesting region, and (e) The user zooms out the map to browse the wider area.

5 Conclusion and Future Work

This paper presents the concept of our approach and prototype implementation to enhance the user experiences when the users explore a map via smart glass device. This approach use a region segment algorithm based on a POI density based zooming interface with the smart glass user interface to zoom in/out the map. We apply a speech recognition technique to interact with the map services to pan the map. The proposed approach allows users to search for POIs and navigate the map using the smart glass user interface without additional interaction devices or techniques. Therefore, we maintain the advantages and complement the disadvantages of smart glass. We expect that our proposed method will help people to feel at ease when obtaining geometric information using a smart glass device.

Regarding future work, we will conduct an evaluation to verify our approach's usefulness and effectiveness, which will help us improve its quality. Another future work is adding user-defined POIs to gather more POIs because it would be more useful to gather meaningful POIs for individual people. User-defined POIs also allow people to make private maps, and users can use private maps in their daily lives.

Acknowledgments. This work was supported in part by the National Research Council of Science & Technology (NST) grant by the Korea government (MSIP) (No. CMP-16-01-KIST) and the Korea Institute of Science and Technology (KIST) Institutional Program (Project No. 2E27190).

References

1. Google.: Google Glass Mirror API – Location (2013). https://developers.google.com/glass/develop/mirror/location. Accessed 18 Apr 2016
2. Sutherland, I.E.: A head-mounted three dimensional display. In: Proceedings of the Joint Computer Conference, San Francisco, California, pp. 757–764. ACM (1968)
3. Google.: Google Glass (2012). https://www.google.com/glass/start/. Accessed 17 July 2015
4. Vuzix.: Vuzix M100 (2013). http://www.vuzix.com/consumer/products_m100/. Accessed 17 July 2015
5. Malu, M., Findlater, L.: "OK Glass?" A preliminary exploration of Google glass for persons with upper body motor impairments. In: 16th International ACM SIGACCESS Conference on Computers and Accessibility, Rochester, New York, USA, pp. 267–268. ACM (2014)
6. Betancourt, A., Morerio, P., Regazzoni, C.S., Rauterberg, M.: The evolution of first person vision methods: a survey. IEEE Trans. Circ. Syst. Video Technol. **25**, 744–760 (2015)
7. Kerr, S.J., Rice, M.D., Teo, Y., Wan, M., Cheong, Y.L., Ng, J., Ng-Thamrin, L., Thura-Myo, T., Wren, D.: Wearable mobile augmented reality: evaluating outdoor user experience. In: International Conference on Virtual Reality Continuum and Its Applications in Industry, Hong Kong, China, pp. 209–216. ACM (2011)
8. Caggianese, G., Neroni, P., Gallo, L.: Natural interaction and wearable augmented reality for the enjoyment of the cultural heritage in outdoor conditions. In: De Paolis, L.T., Mongelli, A. (eds.) AVR 2014. LNCS, vol. 8853, pp. 267–282. Springer, Cham (2014). doi: 10.1007/978-3-319-13969-2_20
9. Baldauf, M., Fröhlich, P., Hutter, S.: KIBITZER: a wearable system for eye-gaze-based mobile urban exploration. In: ACM International Conference on Augmented Human, Matthias Baldauf, France, pp. 1–5. ACM (2010)
10. Mayer, S., Sörös, G.: User interface beaming – seamless interaction with smart things using personal wearable computers. In: 11th International Conference on Wearable and Implantable Body Sensor Networks Workshops, Zurich, pp. 46–49 (2014)
11. Google.: Google Maps (2005). https://maps.google.com/. Accessed 09 Feb 2017
12. NYC & SF.: Foursquare (2009). https://www.foursquare.com/. Accessed 16 July 2015
13. Zheng, L.: Map2Glass (2014). http://map2glass.com/. Accessed 31 Mar 2016
14. Altwaijry, H., Moghimi, M., Belongie, S.: Recognizing locations with Google Glass: a case study. In: IEEE Winter Conference on Applications of Computer Vision, pp. 167–174 (2014)
15. Mapbox.: Mapbox (2010). https://www.mapbox.com/. Accessed 09 Feb 2017
16. Agafonkin, V.: Marker Clustering plugin for Leaflet (2012). https://github.com/Leaflet/Leaflet.markercluster. Accessed 10 Feb 2017
17. Agafonkin, V.: Leaflet (2011). http://leafletjs.com/. Accessed 10 Feb 2017