Mobile Visualisation Techniques for Large Datasets

Motebang Lebusa
Department of Computer Science
Rhodes University
Grahamstown, South Africa
g14l4400@campus.ru.ac.za

Hannah Thinyane
Department of Computer Science
Rhodes University
Grahamstown, South Africa
H.Thinyane@ru.ac.za

Ingrid Siebörger
Department of Computer Science
Rhodes University
Grahamstown, South Africa
I.Sieborger@ru.ac.za

ABSTRACT

The mobile platform is faced with a challenge of limited computing resources. With the increasing popularity of mobile devices and data demands, it makes sense to provide mobile users with tools to absorb information on their devices. Visualisations can be quick and powerful aids for data analysis and information acquisition. This paper discusses the visualisation and interaction techniques that exist on the mobile platform. It further discusses the design and development of a visualisation application that uses categorical and geographic data in the mobile platform and concludes by giving a brief overview on further work that will arise from the visualisation.

Categories and Subject Descriptors

D.2.2 [**Design Tools and Techniques**]: User Interfaces; H.5.2 [**Information Interfaces and Presentation**]: Interaction styles

General Terms

Design

Keywords

Mobile visualisation, mobile interactions techniques, mobile devices

1. INTRODUCTION

As the mobile platform gains popularity, it makes sense to provide mobile users with as much information as possible wherever they are. The huge amount of data generated from various computing devices leaves users unable to digest as much as they would like. This effectively results in what is termed *information overload*. Visualisations can assist in relieving information overload. They have been used for a long time to enhance the amount of information and insight that can be gained from data [5]. This is unsurprising given the old adage "*a picture is worth a thousand words*".

The computing resources in mobile devices are limited [14]. As such, designing applications, including visualisations, for these devices poses a challenge to designers [14]. The visualisation application in this paper attempts to show appropriate visualisation techniques for large geographic and categorical datasets. The goals of this research project are threefold, to i) evaluate mobile visualisation and interactions techniques for large datasets, ii) develop visualisations based on the findings from related work and iii) evaluate the visualisations for effectiveness and usefulness amongst other factors. In this paper, we present

work done on the first two objectives of the study. A discussion on background information in mobile visualisation will follow in Section 2.

Data used for this visualisation is based on the sample data from the larger project called MobiSAM from which this research project derives. MobiSAM monitors service delivery in the Makana Municipality. It supports the community to participate in local government by reporting problems related to service delivery on its website[1]. The visualisation uses example data from MobiSAM. The data constitutes the responses to the polls taken on the website regarding various aspects of water service delivery in different suburbs around Makana Municipality. Polls feature options such as water quality, from which a poll calls for responses that describe water colour, taste, pressure or smell.

A lot of factors have to be taken into consideration in the design and development of the visualisation for this project. Unlike the desktop platform, the mobile platform faces other factors that are not common in the desktop platform. For example, the distinctive mobile nature of mobile devices dictates that an application should require less attention from a user as the attention is split among a number of activities at a time [4]. Factors that are involved in the design of mobile visualisations will further be discussed in the section on design, Section 3.

The implementation of the visualisation has been achieved through the use of various tools and resources. Since the data used is typically geographic, adding maps to the application was achieved through the use of Google Maps Android API v2. The API allows the addition of Google Maps to an application, together with other features that were found to be necessary from the design section of the system. Such features include the ability to draw the various boundaries of and colour the regions represented by the suburbs within the Makana Municipality. Section 4 will expand more on the implementation of the system.

The last section will summarize and will also include further work will that will arise from the visualisation.

2. RELATED WORK

Several studies have been conducted in the field of mobile visualisation [13], [11], [3] and [10]. The following sections discuss various aspects that provide background understanding prior to developing visualisations for the mobile platform.

---

[1] Available at: http://www.mobisam.net

2.1 Visualisations in the mobile platform

The mobile platform faces additional challenges in comparison to the desktop platform when it comes to developing visualisations. While the visualisation spectrum might have been fairly developed for the desktop platform, the primary challenge on the mobile platform lies in the limited resources such as small screen and processors within the devices [4]. Other challenges include mobility, which adds a lot of problems that need to be addressed in the design of visualisations such as short attention spans, multitasking and changing lighting conditions [4]. Despite these challenges, the rapid technological developments offer an opportunity to develop effective and interactive visualisations for the platform.

2.1.1 Hardware and software resources

The available resources on mobile devices have an impact on the visualisations that can be rendered by these devices. Below is a brief description of key resources to consider for visualisations:

1) The screen – this is key to a visualisation since it is the interface between the user's eyes and the visualisation. Touchscreens are a common type of screen in modern mobile devices.

2) Processor and RAM – performance of a visualisation depends on these two resources as they determine the speed and size of what can be cached and accessed as a user interacts with a visualisation.

3) Connectivity – different connectivity technologies are often available in one mobile device. These determine the speed data can be transferred across networks.

4) Input/output (IO) resources – interacting with mobile devices primarily depends on IO resources. The mobile platform has a variety of IO resources, including speakers, microphones, sensors, screens, etc. Some of these resources provide unique opportunities to the platform. For example, GPS sensors enable devices to interact with the geographical locations they are located in. Exploiting these resources can enhance interactivity in visualisations.

2.1.2 Design considerations

Previous studies proposed and developed some guidelines that could be used for designing effective visualisations. For example, Chittaro's [4] work discusses guidelines specific to visualisations, while Fling's [6] work gives general design guidelines to mobile design that could be adapted for visualisations.

Guidelines by Chittaro [4] are described below:

1) mapping – a visualisation design should allow for important aspects of the data being visualised to be obvious to the user's eyes.

2) selection – the small screen requires careful attention in choosing what to put on the visualisation layout, while ensuring that important aspects of the dataset are left out.

3) presentation – approaches such as Focus + Context (F+C) [2] and Overview + Detail (O+D) [4] have been developed to tackle presentation, but were not effective due to limited screen landscape [4] as they require multiple views from the small screens of mobile devices. Some approaches [3] were however effective, though clutter remained a problem as they introduced other features on the screen.

4) interactivity – an effective visualisation should allow a user some level of interactivity during the exploration of data being visualised. This actively engages a user during the visualisation process.

5) human factors – unlike the desktop platform, where users would typically be trained in using a system prior to deployment, mobile users are so widespread that the possibility of training users would not be feasible. As such, the target audience's computer and literary skills and cognitive capabilities should be established and understood ahead of time for a visualisation to be highly usable [5, 9].

6) evaluation – usability of a visualisation should be evaluated by users as they interact with the it [5].

Guidelines by Fling [6] include:

1) context – the environment in which a user is using a mobile device, together with the mental state of the user as they interact with the visualisation.

2) message – the visualisation needs to be informative to the user (related to the mapping point from guidelines by Chittaro [4] above).

3) look and feel – such aspects as colour, layout, typography and graphics are crucial in conveying meanings of different aspects of a visualisation.

2.2 Visualisation data types

Various studies have proposed visualisations based on different and sometimes similar aspects of the underlying data to be visualised [15], [4] and [12]. Shneiderman [12] proposes the following visualisations: i) 1D type that deals with only one variable; ii) 2D type that deals with two variables; iii) 3D type dealing with three variables; iv) N-D type that deals with n variables (where n > 3); v) temporal type dealing with data related to time; vi) tree type that shows hierarchical relationships with the datasets and vii) network type that shows links between groups of items in the dataset. Ware [15] proposes three data types, namely: i) categorical – which assigns labels to data in the datasets; ii) integer – which involves data that can be ranked in a sequence and iii) real-number type that involves intervals and ratios in the datasets.

2.3 Interaction techniques for mobile devices

To engage a user with a visualisation, some level of interactivity has to be offered. The hardware (as discussed in Section 2.1.1) available on the devices supports a variety of interactions. Examples of interactions include touch/tap, motion detection, speech recognition, location-awareness and orientation interactions.

2.4 User tasks in visualisation

In order for a visualisation to be effective, a user needs to be focused more on their intended goal rather than on the 'how' part of achieving the goal. To achieve this, Shneiderman [12] came up with the seven fundamental tasks to be provided to a user by a visualisation. These are:

i) overview – a user should have a way of interacting with or accessing the entire dataset

ii) zoom – a user should be able to closely visualise regions of interest

iii) filter – details that are not of immediate interest to the user should be easy to remove or hide from the visualisation

iv) details-on-demand – a user should be able to add details to the visualisation as they require

v) relate – relationships within the dataset should be explored and patterns be observable

vi) history – users should be able to undo tasks as they wish, and

vii) extract – parts of a visualisation data should be able to be extracted onto the user's device.

2.5 Summary

It is important to understand the various aspects that affect the visualisations on the mobile platform in order to design a suitable visualisation for the platform. The limited resources are a challenge with respect to the design and development processes. The wide interaction techniques with the platform can enhance the visualisation process if they are properly incorporated within the design of a visualisation.

3. DESIGN

As mentioned in Section 2.1.2, some visualisations (F+C, O+D [2] and Halo [3]) were either ineffective or introduced clutter on the screens. To avoid this, a number of considerations went into the design of the visualisation. The visualisation design is described in the following subsection.

3.1 System Design

From a high level perspective, the desired output should provide a user with at least three features. The most important feature for this project is the ability to provide the user with a map on which data overlays and other features would be superimposed. Displaying the boundaries of the individual suburbs is also important for the user to be able to interact with them individually. The visualisation would show an overview of the dataset by colouring each suburb with a colour that represents the specific category, and such colour will be one that represents a category with the highest number of responses from a particular suburb. To illustrate, suppose a suburb Vukani has a total responses classified according to Table 1 below. Out of all the responses, the highest number of responses from that suburb is in the category of cloudy water and so the suburb that represents Vukani will be coloured with a colour that represents cloudy water. The other aspect related to colour is the intensity of the colour. The design is such that intensity increases with the percentage it represents, such that the smaller percentages are less opaque than larger percentages.

Table 1. Responses for Vukani suburb

Category	Responses	% Responses
cloudy	4997	30,8
brownish	4213	26,0
Smells/tastes chlorine	3899	24,1
Smells/tastes offensive	3103	19,1

A complementary bar chart is placed at the top of the visualisation, always showing the overall results of the entire Makana Municipality to highlight the overall situation as one of

the user tasks by Shneiderman [12] described in Section 2.4. The chart displays the top two categories and sums the rest (if they exist) as 'Others' and displays their corresponding percentages, total responses and colours.

Interactivity has been shown to be one of the key success factors in designing visualisations [4]. To provide interactivity in the visualisation, the suburbs can be interrogated (through tapping them) to display their information both on the map and in textual format. Another interaction with the map is the zooming feature, it is ideal to let a user zoom in and out if they wish so that they can closely inspect their regions of interest.

Another feature is the incorporation of a time dimension within the dataset, which lets a user select a window of time from which to visualise the dataset, instead of the entire dataset. The time feature enables a user to view a suburb's (or all suburbs') details in a window of one up to ten days previously as the user slides the time slider.

The Balsamiq Mockups application was used for the design of the system. The application is a graphical user interface builder that allows designers to build mock-ups using widgets in a WYSIWYG editor. The design of a visualisation was developed using the Balsamiq Mockups as it provided the required widgets for the visualisation application. Figure 1[2] shows a prototype of the visualisation. The prototype was developed as it makes the design process faster. Designing using a mockup application is faster than coding. Since little time is spent on the design, it becomes easy to change the design based on user feedback.
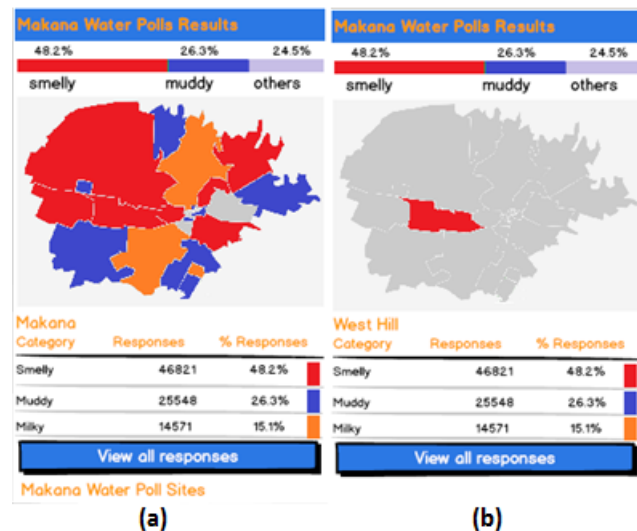

Figure 1 (a) Screen for displaying results for all suburbs b) screen displaying results for one suburb

4. IMPLEMENTATION

To realise the application design, some techniques and tools were used, guided by factors such as access to tools, ease of use, familiarity with the tools and general support from both the tool developers and the online community.

---

[2] The suburb layout used for the design does not match the Makana suburbs but was sufficient for the design. The actual suburb layout was used at the time of coding.

4.1 Development Tools

4.1.1 Eclipse ADT[3]

As Android was chosen as a target platform, the Eclipse Android Developer Tools (ADT) Bundle was chosen to assist with the development task. The first reason is that it is freely available. The other reason is that it includes an Android software development kit (SDK) integrated with the Eclipse integrated development environment (IDE) that provides tools for developers to develop, test and debug Android applications from the API libraries in the IDE [1]. All coding was done using the Eclipse ADT through its extensive suite of libraries.

4.1.2 Google Maps Android API v2[4]

As mentioned in Section 3, a map is the most important feature for this project because of the geographic data in the dataset. Google Maps (in comparison with competitors such as deCarta, Open Street Maps, Apple Maps and Bing Maps) was seen as the most ideal map to incorporate a map within the visualisation application due to its integration within the Android platform. Google Maps Android API v2 provides developers with easy integrations of Google Maps data to their apps. Google Maps Android API v2 provides designers with the capability to embed maps in their applications and add other features like markers, polygons, polylines, etc that are typically available on Google Maps through the API [7]. The API uses OpenGL ES version 2 for displaying the maps [7]. The API also handles automatic tasks like connecting to Google Maps server and responding to map events [7]. In order for any application to communicate with the API, a Google Maps Android API key is required and is freely available from Google APIs console. Google Play services SKD is also required to use the API. A detailed process of installing the API is available on the developer site[4].

4.1.3 Microsoft Azure[5]

To host the data used for the visualisation application, Microsoft Azure was used for its data management service that enables creation of databases and tables for storing data. A database was created and tables added for the project. The other important use of Azure is the mobile service it provides to developers of mobile applications. Mobile services provide a back-end for mobile applications for various purposes like push notifications, authentication of users, storing data on the cloud and adding custom business logic to the application [8]. Azure provides some level of security of the mobile service in order to protect the data from vulnerabilities that may not be foreseen during development. For example, permissions for the HTTP methods (e.g. POST and GET) used in a custom API can be set for various user types such as 'everyone', 'anyone with the application key', 'only authenticated user' and 'only administrators'. For this project, the level was set to anyone with the application key.

4.1.4 Other resources

The coordinates for the vertices of the suburb boundaries were downloaded from MobiSAM website. One important method that is not built-in to Google Maps Android API is determining if a clicked point is inside a polygon and was sourced from the developer site[6]. The method was adapted to match the Latitude/Longitude point format used by Google Maps Android API.

4.2 The Application

The final visualisation application includes: i) a bar chart that displays the overall results for the municipality; ii) an option to allow users to choose which service they wish to visualise (water, electricity, roads, refuse, etc); iii) a map that displays Grahamstown, together with the suburbs boundaries; iv) the suburbs coloured according to the category with the highest number of responses within a particular suburb; v) a time slider that allows a user to select a window of time from which to visualise the dataset and vi) a listview that displays information related to suburbs/categories. A screenshot for the visualisation is shown below in Figure 2 as an example. Figure 2(a) shows the overall view of the dataset, with each suburb coloured with the top category from that particular suburb. The details in the listview widget show total numbers for different categories together with their corresponding percentages. Figure 2(b) shows details of a chosen suburb for the past 9 days, obtained by moving the time slider to 9 days. Other factors such as security were not in the scope of this project though they were important to consider during development. For example, the only input from the user is a map click (achieved by tapping the screen) for selecting a suburb or a tab (for choosing to visualise data by suburbs or categories).
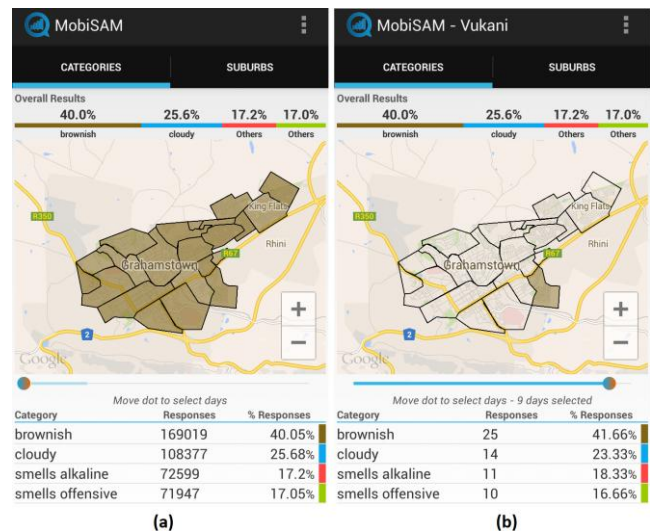

Figure 2 (a) Visualisation screen displaying results for all suburbs b) screen displaying results for one suburb

---

[3]   Available at http://developer.android.com/tools/sdk/eclipse-adt.html

[4]   Installation details available at: https://developers.google.com/maps/documentation/android/start#getting_the_google_maps_android_api_v2

[5]   Documentation available at: http://azure.microsoft.com/en-us/documentation/articles/mobile-services-android-get-started/

[6]   Available at http://www.ecse.rpi.edu/~wrf/Research/Short_Notes/pnpoly.html

5. CONCLUSION AND FUTURE WORK

Interaction techniques largely depend on the type of target device for the visualisation. That said, the developed visualization application indicates that visualisations for large datasets can, indeed, be created for mobile devices at least on the Android platform. It has also showed that geographic and categorical datasets can be used for visualisations techniques. Making use of APIs not only saved development time but improved performance as well, as APIs are optimised for performance and therefore take cognizance of the limited resources of mobile devices.

The effectiveness and usefulness of the visualisation is yet to be evaluated through user study as future work. It will be explored how users may make their own determination on what to visualise. Other data types, for instance text-based data, could also be interesting to use in developing visualisations and on other platforms such as iOS and Windows.

6. REFERENCES

[1]   ANDROID DEVELOPERS. ADT Plugin, 2014. Available from http://developer.android.com/tools/sdk/eclipse-adt.html. Accessed on 11 Sep 2014.

[2]   BAUDISCH, P., GOOD, N., AND STEWART, P. Focus plus context screens: combining display technology with visualization techniques. In *Proceedings of the 15th annual ACM symposium on User interface software and technology* (2001), ACM, pp. 31–40.

[3]   BAUDISCH, P., AND ROSENHOLTZ, R. Halo: a technique for visualizing off-screen objects. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (2003), ACM, pp. 481–488.

[4]   CHITTARO, L. Visualizing information on mobile devices. *Computer 39*, 3 (2006), 40–45.

[5]   DIX, A., FINLEY, J., ABOWD, G., AND BEALE, R. *Human-Computer Interaction*. Pearson Prentice Hall, 2004.

[6]   FLING, B. *Mobile design and development: Practical concepts and techniques for creating mobile sites and Web apps*. O'Reilly, 2009.

[7]   GOOGLE DEVELOPERS. Getting Started, 2014. Available from https://developers.google.com/maps/documentation/android/start. Accessed on 11 Sep 2014.

[8]   MIROSOFT AZURE. Get started with Mobile Services, 2014. Available from https://azure.microsoft.com/en-us/documentation/articles/mobile-services-android-get-started/. Accessed on 11 Sep 2014.

[9]   PAELKE, V., REIMANN, C., AND ROSENBACH, W. A visualization design repository for mobile devices. In *Proceedings of the 2nd International Conference on Computer Graphics, Virtual Reality, Visualisation and Interaction in Africa* (2003), ACM, pp. 57–62.

[10]   PARTRIDGE, K., CHATTERJEE, S., SAZAWAL, V., BORRIELLO, G., AND WANT, R. TiltType: accelerometer-supported text entry for very small devices. In *Proceedings of the 15th annual ACM symposium on User interface software and technology* (2002), ACM, pp. 201–204.

[11]   ROBBINS, D. C., CUTRELL, E., SARIN, R., AND HORVITZ, E. ZoneZoom: Map navigation for smartphones with recursive view segmentation. In *Proceedings of the Working Conference on Advanced Visual Interfaces* (2004), ACM, pp. 231–234.

[12]   SHNEIDERMAN, B. The eyes have it: A task by data type taxonomy for information visualizations. In *Proceedings of 1996 IEEE Symposium on Visual Languages* (1996), IEEE, pp. 336–343.

[13]   SÖRÖS, G., SEICHTER, H., RAUTEK, P., AND GRÖLLER, E. Augmented visualization with natural feature tracking. In *Proceedings of the 10th International Conference on Mobile and Ubiquitous Multimedia* (2011), MUM '11, ACM, pp. 4–12.

[14]   VAN TONDER, B., AND WESSON, J. Using adaptive interfaces to improve mobile map-based visualisation. In *Proceedings of the 2008 annual research conference of the South African Institute of Computer Scientists and Information Technologists on IT research in developing countries: riding the wave of technology* (2008), ACM, pp. 257–266.

[15]   WARE, C. *Information visualization: Perception for design*. Morgan Kaufmann, 2013.