

# Enhancing Traffic Visualizations for Mobile Devices (Mingle)

Ken Knudsen  
Computer Science Department  
University of Maryland, College Park  
ken@cs.umd.edu

## ABSTRACT

Current media for disseminating traffic conditions to motorists are often infrequent and lack sufficient detail. Mingle, a mobile application for visualizing historic traffic information in conjunction with current conditions, was developed to alert users to problematic areas and allow them to make informed decisions with respect to altering their route. Mingle layers select traffic datasets on top of a map API and offers controls for filtering data and adjusting layer rendering properties. Preliminary work is presented, as an evaluation has not yet been performed.

## 1. INTRODUCTION

Monitoring traffic has become a daily routine for motorists. Many rely on radio reports of the latest accidents and traffic conditions, as well as advice on alternate routes throughout the day. These reports are often infrequent and lack sufficient detail. Increasing availability of mobile GPS devices, such as cellular phones and automotive navigation systems, offers convenient access to information on local surroundings.

This paper introduces Mingle, a mobile application framework for visualizing multiple data sources having geospatial attributes on a map interface. This framework encodes information from each data source into a distinct layer and then lays these layers over the map for easy comparison. Additional rendering controls give users the power to adjust the transparency and display properties for each layer.

To illustrate its capabilities, Mingle compares three different traffic datasets, historic traffic accidents, speed data and live incident data to alert users toward problematic areas. With this knowledge, users can make an informed decision with respect to altering their route.

## 2. RELATED WORK

Mobile devices impose constraints on applications due to hardware limitations. Applications must accommodate small screen size, low resolution, limited memory and computing power [10]. Furthermore, mobile applications must be simple to navigate and intuitive to users. Chittaro [6] defines six issues to consider when developing for mobile devices: *mapping*, *selection*, *presentation*, *interactivity*, *human factors* and *evaluation*. The

rest of this section describes techniques for handling some of these issues.

## 2.1 Navigation and Selection

### 2.1.1 Touch-screen Interaction

Most mobile devices interface either through the keypad or the screen. Touch-screen displays are gaining popularity in the cellular phone market; however, most phones still interface through the keypad. Touch-screens handle navigation and selection similar to the mouse on personal computers. Touching an object selects it, dragging along the screen moves the object and tapping the screen clicks an object.

Touch-screen displays impose additional restrictions over cursor based displays. Users can interact with touch screens using a stylus, their finger or their thumb. Parhi et al conducted a study on one-handed thumb interaction for mobile touch-screen displays to determine the optimal icon size required for efficient selection [7]. The study found that icons needed to be larger for touch-screens than cursor displays.

Additional techniques are required to handle data selection when the data is clustered or occluded. Occlusion is a major issue facing densely populated displays where underlying information is easily hidden or indistinguishable from its surroundings. "Escape" introduces gestures to distinguish between close points by making gesture on the screen in one of four directions, corresponding to a data point [12]. Each object has a hot spot region and a corresponding gesture; when the region is triggered, it listens for the direction and then selects the desired point. Karlson et al propose ThumbSpace, a technique for selecting out of reach objects by defining a subspace region easily accessible to the thumb and scaling the actions accordingly [5].

### 2.1.2 Keypad Interaction

ZoneZoom and Jump and Refine are two techniques proposed for handling navigation and selection using the keypad [9][4]. Briefly, ZoneZoom divides the display into a 3x3 grid corresponding to the keypad layout and zooms into the section respective to the key selected. Jump and Refine moves the cursor at two different rates, allowing for course movements to the general vicinity and refining the cursor position at a finer rate over the target. Mingle does not handle either of these techniques at this time.

## 2.2 Visualizing Data on Mobile Devices

Visualizations projected on small, low resolution displays must conform to the restricted display space and provide effective color contrast that enable users to quickly see the information they are interested in. Small screen displays clutter quickly with relatively modest datasets, thus visualization approaches utilizing coordinated displays such as Overview with Details on Demand are a burden to use on small devices and should only be provided when there is enough space and sufficient benefit to the user [3]. Mingle displays all of the information in a single panel and opens menus over this panel to provide additional content or menu options.

## 2.3 Map APIs and Traffic Visualizations

Current web-based map APIs, such as Google, MSN and Yahoo, layer content from multiple data providers on to a single map panel [13][14][15]. These services provide strict visualizations and controls, restricting users to simple, predefined tasks, such as displaying current traffic conditions or incidents as icons, or overlying directions between two locations. The APIs use strict color schemes that are intuitive to the majority of users; however, users are not given the option to “emphasize” any particular layer through changes in icon size, color intensity or transparency. For instance, all three APIs show traffic speeds using colored “link” along major roadways. Links are preferred by users over point locations, such as icons or pushpins [1]. They offer better understanding as to the location and areas affected. Each API displays these links proportional to the map zoom where the lines become thicker as the zoom is increased. Users have no control over the size and intensity of the overlay, the translucent links on MSN and Yahoo blend into the background map.

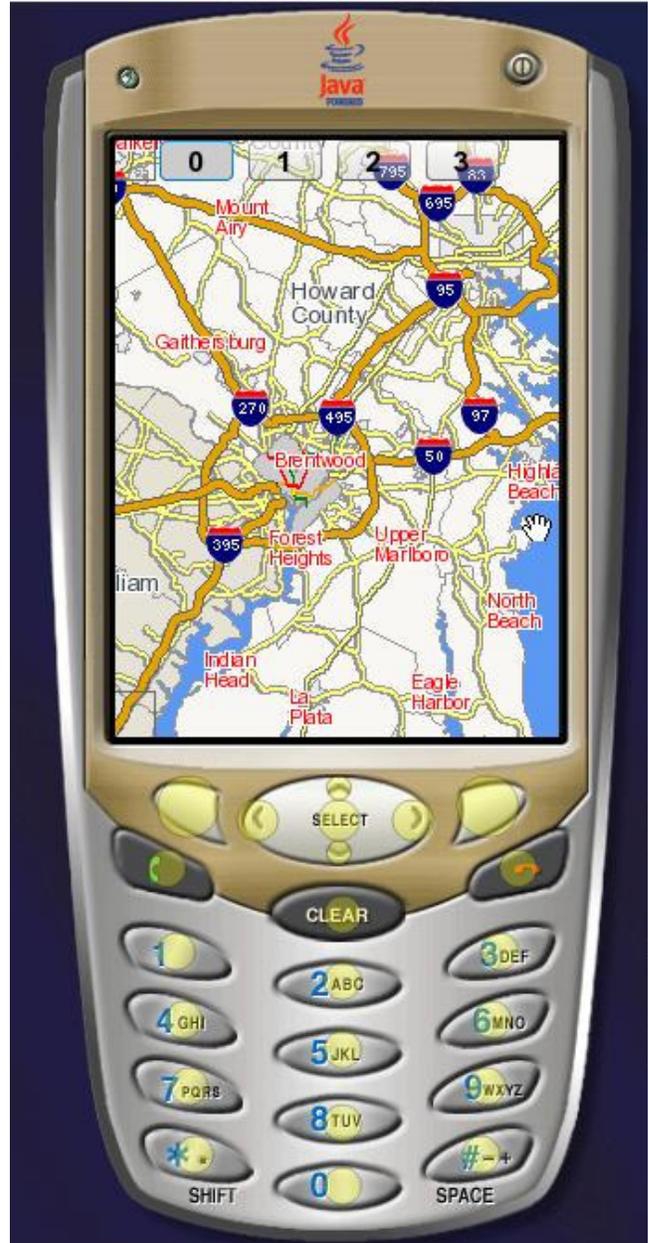
Traffic “Heat Maps” aggregate clusters of data points and projects them as a temperature map (color gradient) representing the intensity or number of points at each location. Mingle uses the algorithm for heat maps from Fervor to display incident data [11]. This aggregation technique suites mobile devices well, since it reduces the memory and processing load when rendering occluded points.

## 3. DATA

Three data sources are used for the examples in this paper, “accidents”, “incidents” and “speed”. Incident and speed represent current conditions and compared with the history of accident data.

Traffic accident and speed data were provide by the Center for Advanced Transportation Technology Laboratory for the Maryland highway system. The accident data contains information on the time and location for each incident along with the type of incident, severity and road conditions as reported from local transportation authorities for the time period October 2003 through December 2005. The traffic data contains speed and flow recorded from sensors along the roadways between January 1, 2006 and March 31, 2007.

Live incidents are queried from Yahoo through RSS. Yahoo includes a description of the incident in addition to the time and location. Incidents include work zones, road closures, accidents and disabled vehicles that have been reported.



**Figure 1. Simulated mobile interface for Mingle. Layers are activated through the buttons on the top of the display screen. There are two buttons on the left corners of the screen to (1) toggle the filter panel (top) and to (2) toggle the rendering options (bottom).**

## 4. USER INTERFACE

Mingle is displayed as a single screen with a map API, figure 1. Traffic information is added to the map as overlays; each overlay is a layer of encoded data points for a single data source, i.e. icons at incident locations. Buttons along the top of the map activate the various overlays. The active overlay, referred to as mode, sets the rendering controls, filter options and keypad mappings. Additionally, two buttons along the left corners of the screen toggle the filtering (top left) and rendering control panels (bottom left).

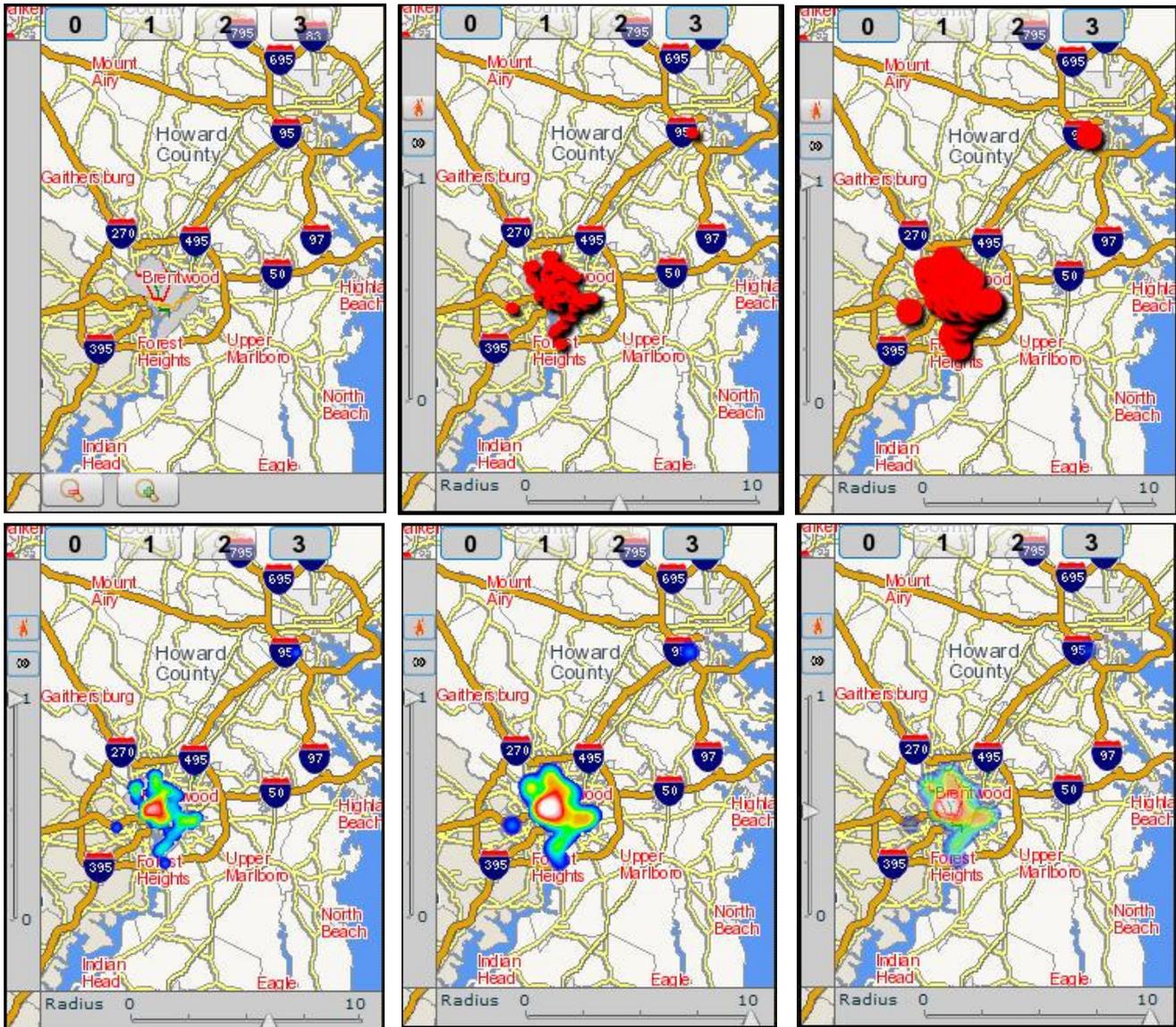


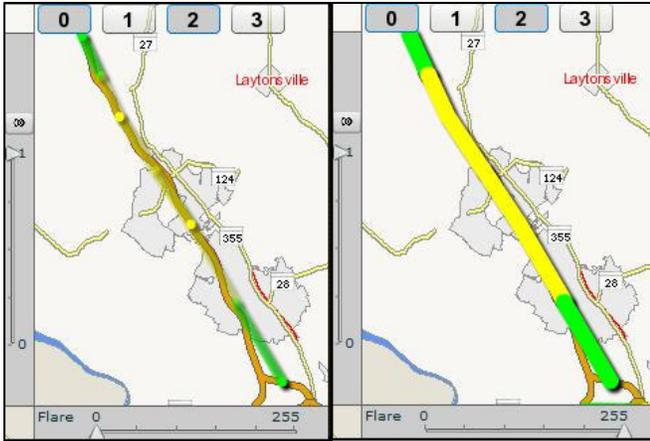
Figure 2. Mingle rendering controls. Map mode (top left) shows zooming controls for the map, while Incident mode (top center) displays common layer rendering options in the left panel and specific icon size control in the bottom panel. The Incident layer visibility is toggled through the eye icon in the left panel (second button down). The displayed incidents are rendered as icons which are adjustable through the radius slider in the bottom panel (top center and top left). Selecting the Heat Map rendering option in the left panel (top button) renders the incidents as a heat map (bottom row). The sizing control now changes the sensitivity of the heat map (bottom left and center). Finally, the transparency control blends the heat map in with the map background, revealing the occluded routes (bottom right).

## 4.1 Navigation

Users navigate Mingle through simulated touch-screen actions or limited keypad events to control map panning and zooming. The touch options are similar to mouse actions for web-based APIs where dragging pans and tapping zooms. The directional keys pan while the 1 and 3 keys zoom the map out and in respectively. Additional navigation controls activate depending on the active mode. Map mode (0) allows panning and zooming of the map using the keypad, figure 2; whereas, Incident mode (3) alters the keypad controls for exploring incidents.

## 4.2 Render Panels

The render panels provide quick visual controls specific to each data source. The left panel generally contains display options that are common throughout all layers, such as rendering the data as a heat map or icons, showing and hiding layers, and adjusting the transparency the transparency. The bottom panel contains controls relevant to specific layers, for instance, the map layer provides zoom controls, while the incident layer provides a control to adjust the icon size. These controls provide users a means to emphasize certain sources through intense, opaque objects, while blending other layers into the background through lower transparency.



**Figure 3. Flare icons for traffic speeds. The sensors are opaque circle icons with a transparent gradient decreasing as the distance from the sensor increases. The flare intensity is adjustable from showing the sensor location (left) to a solid opaque link (right).**

The "Traffic" overlay displays speed information for sensor values encoded as colored links along the roads. Mingle displays these links as "flares", a variation of the link method described by Plaisant. Flares function similarly to links as they are encoded by color and span the length of roads; however, flares are opaque at sensor locations and decrease in transparency as the distance from the sensor increases. The Traffic layer provides transparency control over the intensity of the flare, allowing users to vary the intensity of the flare from solid links through just the sensor locations suitable to users' preferences, figure 3.

### 4.3 Overlays

Overlays are data layers encoded into icons displayed on the map API, adding an additional dimension of the data such as speed or incidents to other sources. Layers are fixed according to the data

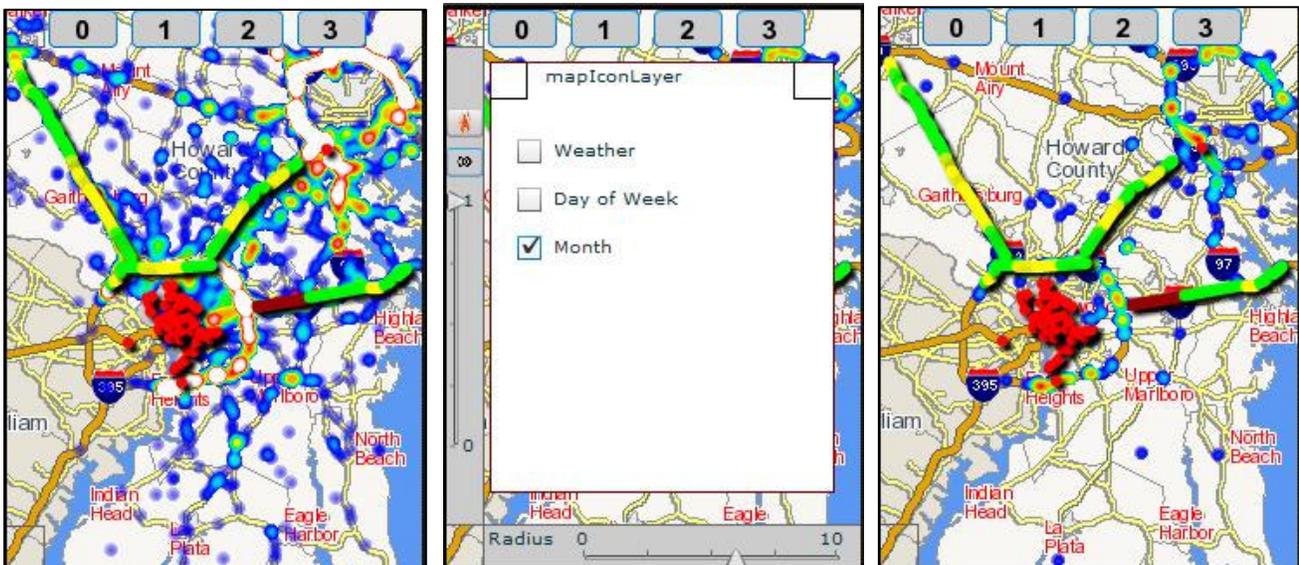
source order providing visual consistency. Layers are activated through the buttons along the top of the screen. The ascending button order displayed in the figures represents the order of the data sources. The first button (0) is the map API and does not have options for hiding the map or changing its transparency since it is the base layer. The other buttons correspond to the data sources used, (1) Accident data, (2) Speed data and (3) Incident data.

Each layer can be added individually as seen in figures 2 and 3 until all sources are visible. Figure 4 demonstrates the full layering for the sources listed above though filtering. The Accident data are rendered as heat maps, while the Incident data as red icons and the speed as green, red and yellow links (lines parallel to the roadways).

### 4.4 Filter Panel

The incident and speed data sets have multiple attributes: date/time, numeric and categorical. Since Mingle runs on mobile devices, users are presented with simplified aggregation options, such as month, day of the month, day of the week, time of day, and morning or afternoon. Initially, all incidents are displayed on the map without any filters (preferences and prior searches fall under future improvements). Selecting the filter panel by touching the top left of the screen opens the filter panel over the map, figure 4.

The filter panel provides aggregate query selections for the active layer, limiting the number of options displayed to the panel size. Changing modes while the panel is open changes the filter panel to the active layers' options. The active layer can also be changed by selecting the forward or backward buttons on the top right and left of the panel respectively. For instance, activating the Accident layer displays controls for filtering and aggregating on date/time attributes: a time period slider (not shown) and sets of grouping options using checkboxes.



**Figure 4. Filtering in Mingle. Start with three layers of data where the heat map displays accident data, links speed data and red icons recent incidents (left). The filter panel shows options for filtering accidents by weather condition or time period (center). Filtering on the current month May reduces the data to accidents which occur in May of any year (right).**

## 5. CONCLUSION

Mingle provides a simple interface for visualizing geographical data through a layered approach. Users explore data through layering and filtering, and adjust rendering properties to emphasize interesting features, figure 5. The accident data is blended into the map to show the roadways, the traffic links are flared slightly, showing traffic conditions while allowing the accident data to radiate through, and finally the incident icons are decreased in size since they cluster in a general location so that they do not distract from the other two layers. Through this example, motorists should be able to make an informed decision based on the current traffic conditions and the expected probability an incident may occur.

The framework provides allows users to visualize and explore multiple sources of geographic information through a simple interface.

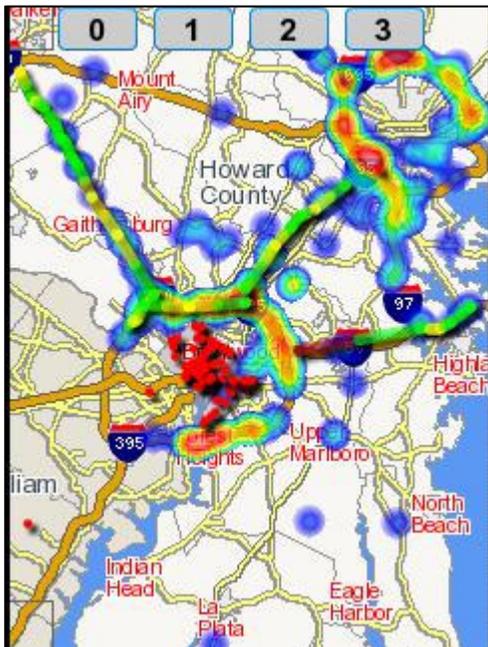
## 6. PROPOSED FEATURES

The following features were proposed for Mingle but were not finished for the initial release.

Currently Mingle returns to the initial view when started. Users should be able to save the overlay states and filters so that they can quickly recall the displays they use the most bookmarks.

With the increasing availability of traffic information sources, users should have the ability to specify data sources to retrieve and examine.

Additionally, users should be able to submit reports on suspected hot spots and monitor certain areas for activity and then filter the



**Figure 5. Final Mingle image a user may reach. The accident heat map is blended with the background to show roads, traffic speed icons are slightly flared to show the traffic conditions without occluding the accident data much, and the incident icons are small since they are clustered and should not distract from the rest of the data.**

data accordingly.

Only basic keypad functionality made it into Mingle, requiring more work to finish connecting all of the button actions. These features with additional support for ZoneZoom and Jump and Refine techniques would enhance the interface for users without touch-screens.

## 7. ACKNOWLEDGMENTS

I wish to thank Jong Chang-Han, Michael Lam, Michael Lieberman and Aaron Shulman for their thoughtful insights and to Darya Filippova, Joonghoon Lee, Andrea Olea, Michael VanDaniker and Krist Wongsuphasawat for their support and contributions developing Mingle.

## 8. REFERENCES

- [1] Bhamidipati, P., Plaisant, C., Tarnoff, P.. "Vehicle Speed Information Displays for Public Websites: A Survey of User Preferences". *Proceeding of Conference on Intelligent Transportation Systems*. 2001.
- [2] Bridle, R. and McCreath, E. 2006. "Inducing shortcuts on a mobile phone interface". *Proceedings of the 11th international Conference on intelligent User interfaces* (Sydney, Australia, January 29 - February 01, 2006). IUI '06. ACM, New York, NY, 327-329. DOI=<http://doi.acm.org/10.1145/1111449.1111526>
- [3] Büring, T., Gerken, J., and Reiterer, H. 2006. "Usability of overview-supported zooming on small screens with regard to individual differences in spatial ability". *Proceedings of the Working Conference on Advanced Visual interfaces* (Venezia, Italy, May 23 - 26, 2006). AVI '06. ACM, New York, NY, 233-240. DOI=<http://doi.acm.org/10.1145/1133265.1133310>
- [4] Hachet, M., Poudroux, J., Tyndiuk, F., and Guitton, P. 2007. Jump and refine for rapid pointing on mobile phones. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (San Jose, California, USA, April 28 - May 03, 2007). CHI '07. ACM, New York, NY, 167-170. DOI=<http://doi.acm.org/10.1145/1240624.1240651>
- [5] Karlson, A., & Bederson, B. B. 2007. ThumbSpace: Generalized One-Handed Input for Touchscreen-Based Mobile Devices. *Proceedings of INTERACT 2007*. [In press]
- [6] Chittaro, L., Visualizing information on mobile devices, *Computer*, vol.39, no.3, pp. 40-45, March 2006.
- [7] Parhi, P., Karlson, A. K., and Bederson, B. B. 2006. Target size study for one-handed thumb use on small touchscreen devices. In *Proceedings of the 8th Conference on Human-Computer interaction with Mobile Devices and Services* (Helsinki, Finland, September 12 - 15, 2006). MobileHCI '06, vol. 159. ACM, New York, NY, 203-210. DOI=<http://doi.acm.org/10.1145/1152215.1152260>
- [8] Plaisant, C. Survey of websites providing real time traffic information on maps. March 2000. <http://www.cs.umd.edu/hcil/highway/review/>, retrieved 04-30-2008.
- [9] Robbins, D. C., Cutrell, E., Sarin, R., and Horvitz, E. 2004. ZoneZoom: map navigation for smartphones with recursive

- view segmentation. In *Proceedings of the Working Conference on Advanced Visual interfaces* (Gallipoli, Italy, May 25 - 28, 2004). AVI '04. ACM, New York, NY, 231-234. DOI= <http://doi.acm.org/10.1145/989863.989901>
- [10] Seong, D. S. 2006. Usability guidelines for designing mobile learning portals. In *Proceedings of the 3rd international Conference on Mobile Technology, Applications & Systems* (Bangkok, Thailand, October 25 - 27, 2006). Mobility '06, vol. 270. ACM, New York, NY, 25. DOI= <http://doi.acm.org/10.1145/1292331.1292359>
- [11] Filippova, D., Lee, J., Olea, A., VanDaniker, M., and Wongsuphasawat, K. 2008. Exploring Clusters in Geo-Spatial Datasets. May 2008. [https://wiki.cs.umd.edu/cmssc734\\_08/images/4/46/HeatMaps.pdf](https://wiki.cs.umd.edu/cmssc734_08/images/4/46/HeatMaps.pdf), retrieved 05-8-2008.
- [12] Yatani, K., Partridge, K., Bern, M., and Newman, M. W. 2008. Escape: a target selection technique using visually-cued gestures. In *Proceeding of the Twenty-Sixth Annual SIGCHI Conference on Human Factors in Computing Systems* (Florence, Italy, April 05 - 10, 2008). CHI '08. ACM, New York, NY, 285-294. DOI= <http://doi.acm.org/10.1145/1357054.1357104>
- [13] Yahoo Maps. <http://maps.yahoo.com/>, retrieved 04-30-08.
- [14] Google Maps. <http://maps.google.com/>, retrieved 04-30-08.
- [15] MSN Map. <http://maps.live.com/>, retrieved 04-30-08