Touch & Interact: Applied to a Tourist Guide Prototype

Robert Hardy[1], Enrico Rukzio[1], Matthias Wagner[2], Massimo Paolucci[2]
[1] Lancaster University, UK, {hardyr, rukzio}@comp.lancs.ac.uk
[2] NTT DoCoMo Euro-Labs, {wagner, paolucci}@docomolab-euro.com

A video illustrating the interaction technique and the prototype can be found at
http://www.comp.lancs.ac.uk/~rukzio/video/multitouch.wmv

Abstract

An ongoing issue for mobile application developers is the limited output capabilities of mobile phones [1]. For this reason, current mobile phones may still fail to fully address the requirements of map, multimedia and information browsing applications. This paper presents Touch & Interact: an NFC (Near Field Communication) interaction technique that utilizes the capabilities of mobile phones and the screen size of public displays. Using the Touch & Interact interaction technique, an NFC phone can touch the display at any position in order to perform selections. During the interaction, both the phone display and public display share the display space. The shared display space is especially useful for separation of public and private information by presenting sensitive information on the phone display. In addition to an auxiliary display, the phone provides extra modalities (e.g. joystick and keypad), storage and additional feedback (audio and haptic). A tourist guide application was developed based on Touch & Interact in order to show the advantages of this interaction technique. Touch & Interact is influenced by traditional desktop interactions and interactions extending the traditional desktop environment such as pick-and-drop.

1. Technical Solution

The supporting hardware configuration (Figure 1.a) for the application uses:

1. A dynamic display consisting of a 10x10 mesh of NFC tags (Trikker BL38 from toptunniste.fi) covered by an A2 size paper sheet.
2. A video projector which projects onto the tags for dynamic feedback.
3. A Nokia 6131 NFC (Near Field Communication) phone. This is used to read NFC tags on the dynamic display.
4. A laptop (acting as a server) which receives tag events sent by the phone and is connected to the video projector.

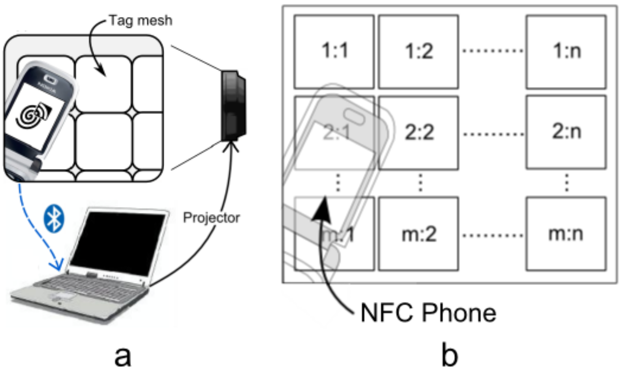

Figure 1: a - Emulation of a touchable dynamic display, b - NFC phone touches mesh of NFC tags.

Since the tags are covered by the paper sheet, a semi-transparent, virtual tag layer is projected onto the display to identify their locations. The 10x10 tag resolution is used to support a preliminary exploration of the potential of the concept and should not be considered the maximum resolution possible. Each tag in the mesh has its coordinates pre-stored (Figure 1.b), and when the tag is read by the phone, the coordinates indicate the location of the display currently touched. The tag size is 4x4cm (similar to the width of the phone used) and with spacing, the cumulative display area is 50x50cm. The tags store 512 bytes of data and support a read range of approximately 3cm.

The mobile phone client was implemented in Java ME (CLDC 1.1 / MIDP 2.0). The Contactless Communication API (JSR 257) was used for accessing the NFC tags, and the Java Bluetooth APIs (JSR 82) were used for the communication between the phone and the laptop. The server application (running on the laptop) was implemented in Java SE. The Bluecove API [2] was used for communication with the Nokia 6131 NFC.

In order to select a tag, the phone needed to the moved within read-range of the tag. Once in range, the phone reads the coordinates from the tag and sends a notification (using Bluetooth) to the server. This notification contains the tag coordinates along with identification of a possible phone key (pressed in conjunction with the tag read). When the server receives a notification from the phone, it fires an event which is handled by the appropriate listener depending on the server state. The server responds to the event by updating the necessary server state and showing the resulting feedback on the large display. The configuration supports the following generic interactions:

- **Hovering** – Using the hovering technique, a phone can be moved within read range of a tag and additional information is displayed on the phone screen.
- **Single Selection/De-selection** – When a tag is hovered, the user can press a specified key on the phone to select/de-select the tag.
- **Multi-selection/de-selection** – If the user holds the key, they are able to select/deselect multiple tags.
- **Polygon-select** – Polygon points can be plotted by holding a specified key and touching the appropriate tags. When the key is released the tags within the polygon area are selected.
- **Pick-and-drop** – Selected items are ‘picked up’ using the phone and can be dropped elsewhere on the screen.
- **Context menu** –The context menu is displayed on the phone. Using the phone’s directional keys, different options can be selected. Showing the menu on the phone avoids possible occlusion of the main screen options and interaction is very similar to menus on typical phones.
- **Remote Clear –** This interaction de-selects any currently selected tags remotely. Incorporating remote interactions into the prototype reduces arm fatigue which builds with prolonged use with pointing interactions.

2. Application Description

A tourist guide application (Figure 2) was implemented which presents users with a map of the local area with nearby markers (hotels, restaurants and events). The prototype would ideally be deployed in a tourist office, train station or town/city centre. The application offers a range of functionality to users and through phone interactions the user is able to manipulate markers and navigate the map. The application has two main modes: selection mode and view mode. Selection mode is used to select tags, find more information about tags and enter a context menu for a tag. View mode is used to pan/zoom the map.

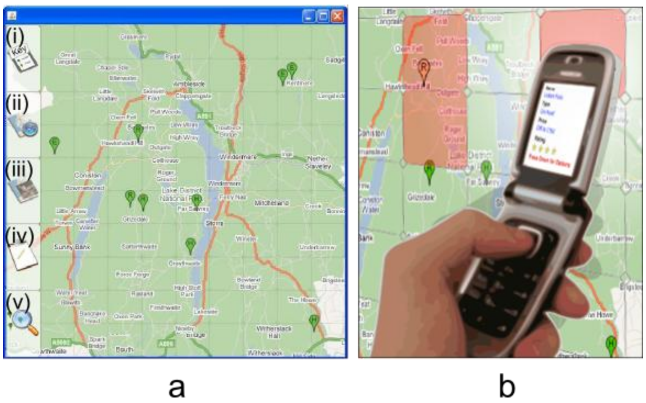

Figure 2: Tourist guide prototype, a - information showed on the public display, b - mobile phone selecting point of interest.

In selection mode, markers can be hovered and selected through a variety of methods. Using the hovering interaction, the phone display shows contextual information about a particular marker, for instance, price range, name and rating of a hotel. Whilst hovering, the user can press a phone key to enter a context menu which provides three options: showing more information about a specific marker, sending the marker VCard to the phone and finding the distance between two markers. Markers inside selected tag areas are also selected and the phone shows a list of currently selected markers on its display. From the user’s perspective, the phone appears to have picked up the marker which can be dropped into an itinerary using the itinerary menu option (Fig 2a.iv). The itinerary is a list of markers the user wishes to visit for the day. Once markers have been dropped into the itinerary, they can be viewed publicly or privately (using the phone display). When viewed publicly on the large display, markers can be removed and reordered with regards to visiting order.

The view mode allows the users to pan around the map to view markers outside the current purview. The user can also zoom into the map for viewing in higher detail. Both operations require tag selection on an area of the map combined with the corresponding key press. In view mode, the phone display shows a satellite view of the map. When zoomed in or panning, the satellite view shows the area the user is focused on using a rectangular graphic. The map can also be panned remotely via Bluetooth using the directional keys and the satellite view.

A side menu can be toggled on/off remotely using the phone and provides various options. The top option is a map key (Fig 2a.i) which can be hovered to indicate what each marker represents using the phone display. The next option (Fig 2a.ii) can be used to switch to view mode. Selecting the satellite overlay option (Fig 2a.iii) toggles the Google satellite imagery and a filter option (Fig 2a.v) allows markers to be filtered (e.g. show only restaurants).

3. Innovation of the Concept

There has been significant research in the area of NFC interaction with static displays such as smart posters and maps [3, 4]. These consist of static, paper displays which are augmented with RFID/NFC tags. However, our prototype extends work by Johannes Vetter [5] using a dynamic display. The advantage of using a dynamic display is the ability to show dynamic information which provides a broader scope for applications. Updated information does not have to result in re-manufacture of a static display but simple data manipulation. Variable data such as special offers, transport times, weather and sport scores can be supported. Interfaces can take multiple views which help to manage large volumes of information and can show different perspectives of the information based on the user access privileges or viewing preference.

The prototype inherits interactions from a traditional desktop environment such as widget hovering, left-click and right-click (context menu). Furthermore, phone storage capabilities are also used to create a new interaction called pick-and-drop which extends from the desktop interaction environment. In this interaction, markers are picked up onto the phone and dropped elsewhere on the display. This concept could be extended to support data to be moved between separate systems [6] or dropping contextual data (collected from the phone) to the system. The phone also has various input modalities allowing it to function as a 'smart stylus' whereby the phone can take the role of various tools with a single key press. Aside from phone features, the prototype is unique in exploring dynamic tag roles. The physical tags are represented by the server as Java tag objects. The tag objects were implemented so they could adopt multiple tag roles based on the state of the system. For example, tags in the application perform button, map area or list item roles. These roles can dynamically change throughout the interaction period.

4. Creativity of the Technical Implementation

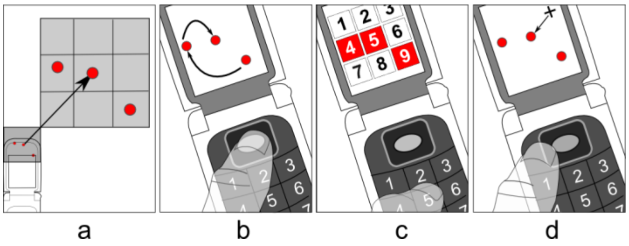

Figure 3: Solutions to overcome the problem of the limited input resolution and occlusion.

The prototype has currently a relatively low input resolution of 10x10 touchable areas. Although the NFC tag sizes can be much smaller [7], the size of the phone is likely to be an ongoing challenge for possible applications requiring high resolution input. For example, the phone used in the application had a width of approximately 5cm (slightly larger than size of the tags used). Developing a map application provided a perfect opportunity to test a number of approaches to address granularity issues as there may be multiple markers within a single tag. For instance, there could be three nearby events in a single tag area. The chosen method was to enlarge the tag into nine tags on the large display and markers can be selected using the normal approach in the enlarged area (Fig 3.a). There are a number of alternatives, for example, selecting a single marker using a phone key to iterate through markers (Fig 3.b). This process could be slow if the desired marker is the last iteration of many markers. The advantage of this approach is that it is negligible to input resolution. A further approach is to divide the tag into nine elements which map the phone’s keypad numbers (Fig 3.c). It would also be possible for the phone display to show the markers in the selected tag and the phone joystick could be used to control a curser on the phone display for selection of a marker (Fig 3.d).

Integrating a Google map into the application shows the adaptability of the Touch & Interact package whilst significantly furthering the functionality of the tourist guide application. To integrate the Google map, an open source project GMap-Viewer [8] was modified. The modifications customized the Google map and markers to suit the tourist application. Hotel, restaurant and event markers extended the abstract markers with additional information. The GMap-Viewer project was also modified in relation to user input as it was originally designed for use with a mouse. To adapt the package for use with the tag interactions, all the mouse event-functionality was removed and tag interactions were mapped to map controls. For applications in which the source could not be modified, the tag interactions could be mapped to mouse interactions using a Java Robot API [9]. The server display interface containing the map was implemented using the Piccolo library [10]. This is package has a higher level set of API’s than the Java 2D Graphics package making it easier to manage multilayered graphics. The ability to easily layer different parts of the display meant that they could be painted individually consequently saving.

An essential outcome from the project is to have developed foundations for future work. In order to support this development, the system must be extensible and reusable. By following the MVC (Model View Controller) pattern, there is a separation between the model and the display so that different displays can be developed to show different parts or representations of the model. *TagController* Java objects encapsulated the essential functionality for tags and are designed to be extended. The *TagController* is implemented around the concept of a JButton [10] in that they share similar properties – location, size, text, image and listener functionality. The *TagController* object also has grouping properties so tags can be grouped together to increase the target size of an object. These similarities with JButtons help developers to quickly become familiar with tag objects and thus easier to implement and extend them. Tag state is held in an array whereby each element in the array represents to current state of the tag for each role. This reduces any complexity involved in keeping a state history which the present state can return to. The reduced cross-coupling concerns and complexity makes it easier to extend and reuse the system. A tag event-model was implemented to fire and handle different types of tag events. The model is abstract, thus simple for developers to extend the system with new events or change the way events are handled. New event models for the map and button tag roles were easily extended for the tourist application. Also, as the overlays are self encapsulated as objects, adding additional information screens such as menu options and enlarged views can be done so with minimal change.

5. Business Potential

The last few years have shown a significant trend in the field of mobile services towards the common use of Near Field Communication to extend the reach of mobile phones into a ubiquitous service environment. This trend has been set by technologies for ubiquitous tagging, for instance based on NFC, as described above and deployed in our demonstration of Touch & Interact. Further near field solutions to support the linkage of digital information with virtually any object in the real world include RFID or 2D bar codes (QR). In Japan and Korea, this technology trend has already started to materialize in commercial services and applications. As an example, in Japan more than 20 million customers of NTT DoCoMo are today using phones from DoCoMo's Osaifu-Keitai line of mobile terminals that are already equipped with NFC chips to support services like Toruca (ubiquitous tagging) or Felica (mobile micro payment).

Most of the currently available commercial applications are based on simple interaction paradigms, which associate one single object with exactly one tag to provide one single link to exactly one service at a time (Touch-‘n-Go service interaction). As an extension to this rather limited setting, multi-tag interactions and applications, like Touch & Interact, are driven by the idea of augmenting objects through many tags that may even provide links to various different services. Given the current commercial uptake of basic Touch’-n-Go services, the concept of multi-tag interaction promises to hold great potential for new mobile services that provide further added value to the customer and be a driver for new markets.

Through projects like MULTITAG [14], demonstrators like Touch & Interact as well as the underlying research and development efforts, we hope to pave the way for future commercial cases of multi-tag interaction. Not only could the described prototype be directly deployed in commercial settings as a dynamic display with features superior to standard display solutions, it could also support the rapid prototyping of more static NFC applications, such as NFC-based posters. In the latter case, the proposed technology would have to be made available for developers as part of their tool support with easy ways to restrict functionality to what is commonly available on a static poster. As a result, development and time to market for a wide range of NFC solutions would be reduced.

6. Conclusion

Touch & Interact is a new interaction technique that can be used to overcome the limited screen size of a mobile device (a disadvantage when using map, browsing or multimedia applications) with the usage of a dynamic screen. Through the interplay of the two screens, mobile device and dynamic screen, richer interaction techniques such as pick-and-drop can be realized that offer an added value to the user. In addition, using mobile phones in interactions with displays increases interaction possibilities through phone input modalities, additional feedback, contextual potential and storage capabilities. The storage capabilities are an important feature allowing data to be taken away from the system (e.g. a list of directions and contact details) rather than the user remembering or writing down details. Regarding deployment as a public display, large size touch screens (approximately 42 inch) can expect to retail around £4000 [12]. Even using a mirror projector (e.g. NEC WT610 [13]) the hardware costs are less than half whilst also providing screen sizes of up to 100 inches. Moreover, the tags used are cheap and robust, offering advantages with regards to vandalism.

7. References

[1] Forman, G. H. and Zahorjan, J. The Challenges of Mobile Computing. IEEE Computer. 27 (4). 1994.

[2] Bluecove, http://code.google.com/p/bluecove/

[3] Gregor Broll, Sven Siorpaes, Enrico Rukzio, Massimo Paolucci , John Hamard, Matthias Wagner, Albrecht Schmidt. Supporting Mobile Service Usage through Physical Mobile Interaction. In Proceedings of the Fifth Annual IEEE International Conference on Pervasive Computing and Communications, March 19-23 2007, White Plains, NY, USA.

[4] Reilly, D., Rodgers, M., Argue, R., Nunes, M., Inkpen, K.: Marked-up maps: combining paper maps and electronic information resources. In: Personal Ubiquitous Comput., 10 (4), pages 215-226, ISSN 1617-4909, 2006.

[5] Johannes Vetter. Near Field Communication based Mobile Interaction Techniques. Project thesis. Media Informatics Group, University of Munich, 2006

[6] Jun Rekimoto. Pick-and-drop: a direct manipulation technique for multiple computer environments. UIST’97. Banff, Alberta, Canada, 1997.

[7] Hitachi Global News Release, http://www.hitachi.com/New/cnews/030902.html

[8] GMap-viewer project, http://code.google.com/p/gmap-viewer/

[9] Java Robot API, http://java.sun.com/j2se/1.4.2/docs/api/java/awt/Robot.html

[10] Piccolo Homepage, http://www.cs.umd.edu/hcil/jazz/

[11] Class JButton, http://java.sun.com/j2se/1.4.2/docs/api/javax/swing/JButton.htm

[12] ELO 4220L 42" Touch Monitor, http://www.primemark.com/epos/touch-screensl.php

[13] NEC WT610, http://www.nec-pj.com/products/wt/index610.html

[14] Multitag project, http://eis.comp.lancs.ac.uk/multitag/