



**HAL**  
open science

## **TangiMap - A Tangible Interface for Visualization of Large Documents on Handheld Computers**

Martin Hachet, Joachim Pouderoux, Pascal Guitton, Jean-Christophe Gonzato

► **To cite this version:**

Martin Hachet, Joachim Pouderoux, Pascal Guitton, Jean-Christophe Gonzato. TangiMap - A Tangible Interface for Visualization of Large Documents on Handheld Computers. Graphics Interface, 2005, Canada. pp.9 - 15. <hal-00308239>

**HAL Id: hal-00308239**

**<https://hal.science/hal-00308239v1>**

Submitted on 30 Jul 2008

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



HAL Authorization

# TangiMap – A Tangible Interface for Visualization of Large Documents on Handheld Computers

Martin Hachet

Joachim Pouderoux

Pascal Guitton

Jean-Christophe Gonzato

IPARLA Project (LaBRI - INRIA Futurs)  
University of Bordeaux - France

## Abstract

The applications for handheld computers have evolved from very simple schedulers or note editors to more complex applications where high-level interaction tasks are required. Despite this evolution, the input devices for interaction with handhelds are still limited to a few buttons and styluses associated with sensitive screens.

In this paper we focus on the visualization of large documents (e.g. maps) that cannot be displayed in their entirety on the small-size screens. We present a new task-adapted and device-adapted interface called TangiMap.

TangiMap is a three degrees of freedom camera-based interface where the user interacts by moving a tangible interface behind the handheld computer. TangiMap benefits from two-handed interaction providing a kinaesthetic feedback and a frame of reference.

We undertook an experiment to compare TangiMap with a classical stylus interface for a two-dimensional target searching task. The results showed that TangiMap was faster and that the user preferences were largely in its favor.

*Key words:* Handheld computers, interaction, tangible interface, large document visualization, evaluation.

## 1 Introduction

Handheld computers such as PDA and cell-phones are becoming more and more popular. They allow the users to benefit from computing capability for mobile interactive applications. The available applications on handheld computers have evolved from simple schedulers and note editors to more complex applications such as GPS based navigation assistants or 3D games. This evolution is strongly linked with the increasing computing capability and screen resolution of these devices. Although the current applications for handheld computers imply very different user tasks, the available input devices for interaction are still limited to a few buttons and direct pointers such as the styluses associated with sensitive screens.

The buttons are well suited for discrete inputs such as character typing. The efficiency of the stylus for pointing tasks makes it irreplaceable for WIMP interfaces. How-

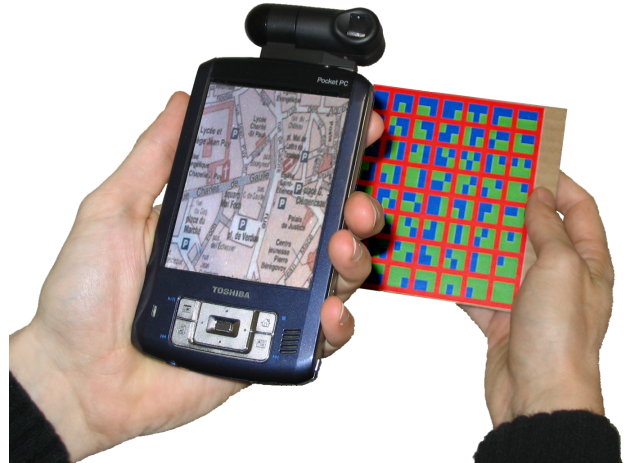


Figure 1: City map visualization with TangiMap.

ever, we must ask ourselves if other input devices should not be preferred for other tasks. Jacob and Sibert[13] argued that the structure of the input device should be adapted to the structure of the task. Therefore, we can presume that the stylus is not best adapted to the tasks implying more than 2 degrees of freedom (DOF). These tasks include pan-and-zoom and 3D interaction tasks.

Whereas others chose to design adapted user interfaces from the stylus inputs, we preferred to investigate the use of a novel input device for interaction with handhelds. The primary task that motivated our work was the interactive visualization of large documents like maps or pictures on the small screen size of the device. We developed a 3 DOF interface where the user directly holds the document through a tangible target. We called this interface TangiMap. The 3D position of the TangiMap target is tracked in real-time using the video stream of the embedded camera. Our interface and its software implementation has been designed to be fast and simple in order to maximize CPU capability for the application.

TangiMap is not limited to the visualization of large maps. In [10], we investigated the use of a tracked target for interaction with 3D environments. More precisely,

the 3 DOF of the interface were used to easily manipulate 3D objects and to navigate in 3D scenes. Many other applications could benefit from TangiMap, like information visualization applications or video games for example.

In this paper, we concentrate our study on the visualization of large documents. Maps are good examples of such data. After the presentation of the work that inspired our interface in section 2, we describe TangiMap in section 3 and some key applications in section 4. In section 5 we describe an experiment that evaluates the differences between our interface and a classical stylus interface. Finally we conclude and introduce our future work.

## 2 Previous work

TangiMap is linked to several works in different research areas. These works relate to the visualization of large information space from 2D input, handheld computer dedicated interfaces, Augmented Reality (AR) applications and 3D user interfaces.

### Large documents visualization from 2D input

The issue concerning the visualization of documents that are larger than the visualization area did not appear with the development of handheld devices. Consequently, many interfaces based on 2D input have been developed to help the users in their navigational tasks. Among these interfaces, Zooming User Interfaces (ZUI) allow the users to continuously access the data by zooming specific locations. The data is visualized through several levels of refinement. Pad [15], Pad++ [2] and Jazz [3] are examples. Another approach for accessing the different levels of refinement is an automatic zooming interface such as the one proposed by Igarashi [12].

### Handheld computer dedicated interfaces

In the specific context of handheld computers, dedicated interfaces have been proposed. They are generally based on the movements of the device itself. For example, the ScrollPad [6] is a mouse-mounted PDA allowing the visualization of large documents while scrolling on a table. The Peephole display [21] is a comparable interface where the user moves the PDA in space allowing users to build up spatial memory. Rekimoto [16] and Hinckley et al. [11] use tilt sensors to navigate maps or document lists. Inputs are provided by moving the device from its original orientation. These works are inspired from the pioneering investigations of Fitzmaurice [8, 9]. Another approach consists of using external input devices. For example, Silfverberg et al. [19] evaluated the benefit of using an isometric joystick for handheld information terminals.

### Augmented Reality applications

Handheld devices have been used for AR applications, too. The camera of the handheld is used to locate the device from specific markers fixed in the real environment. For example, Rekimoto and Nagao [17] tracked their NaviCam by means of color-code IDs. Wagner and Schmalstieg [20] use ARToolKit with a PDA camera for their AR applications. Currently, Rohs [18] uses visual codes for several interaction tasks with camera-equipped cell phones. Similar to the mobile AR applications, TangiMap is based on the analysis of the video stream of the embedded camera for interaction.

### 3D user interfaces

Finally, TangiMap has been inspired by previous research in the field of 3D user interfaces.

TangiMap benefits from two-handed interaction [4]. By moving one hand in relation to the other, the user benefits from a kinaesthetic feedback, i.e. the user knows where his dominant hand is according to his non dominant hand. The non dominant hand provides a reference frame [1].

TangiMap can be seen as a tangible interface too as the user physically holds the data. Tangible user interfaces provide a haptic feedback in addition to a visual feedback, which can increase interaction. Indeed, physically touching the data that is visualized provides an additional sensitive feedback and an increase in perception. An example of such a tangible interface is the Active-Cubes from Kitamura et al. [14].

## 3 TangiMap

### 3.1 Motivation

The two main motivations behind the development of TangiMap were:

- a task-adapted interface,
- a device-adapted interface.

### Task-adapted

We wanted to adapt the structure of the interface to the structure of the task. As our main task was the visualization of large two-dimensional documents such as maps, we developed an interface where 3 DOF can be managed at the same time. Indeed, navigating in large 2D documents requires at least 3 DOF. 2 DOF relate to pan movements while the third relates to zoom operation. These 3 DOF can be controlled by means of 2D input devices such as mice and styluses. In this case, specific user interfaces have to be chosen in order to map the 2 DOF of the device to the 3 DOF of the task. Among these user interfaces, there are the classical sliders, the ZUI and the

automatic zooming techniques described above. Intrinsically, these techniques lead to an indirect way of working, resulting in cognitive loads. To allow an easier and faster interaction with the data, we developed an interface that has as many DOF as the task requires.

### Device-adapted

We developed TangiMap according to the handheld's characteristics. To deal with the problem of the small visualization area, we developed an interface that does not occlude – even partially – the screen of the handheld computer. Indeed, the main drawback of the styluses which are the main input devices for interaction with PDA is that they occlude the small size of the screen, which conflicts with the visualization tasks. Occluding the visualization area may result in some problems for the user. Moreover, moving the handheld computer to interact as described previously leads to an attenuated perception of the displayed images, as the screen has to be permanently tilted.

Mobile setting is another characteristic of handheld computers that motivated the choices for the design of TangiMap. Indeed, we developed an interface that can be used anywhere, without the need of specific installation and without being dependent of a network.

Finally, the limited computing capability of handheld computers leads us to the development of a very efficient algorithm for our interface. This algorithm described in the following insures a real-time interaction.

### 3.2 Implementation

In this section, we sum-up the quite simple implementation of the TangiMap interface. For more details, the reader can refer to [10].

TangiMap consists of a tangible target that the user holds in one hand while holding the camera-equipped device in the other. The target, shown in Figure 2, is a  $12 \times 12$  cm wide square paperboard divided into an array of  $8 \times 8$  cells. These cells are color-codes composed of  $3 \times 2$  code-units. In order to minimize the dependence on the light conditions, the target is drawn with only 3 pure colors: red for the background, blue and green for the codes. Each cell codes its position in the target using a binary representation where blue is assigned to 0 and green to 1. The three top code-units of a cell relate to the column number while the three bottom ones relate to the row number.

In order to insure a real-time computation, the target tracking is done by analyzing only a few pixels in each frame of the video stream captured by the camera. To detect the current position of the target, we first consider the pixel at the center of the frame. From this starting pixel we look for the cell's border by looking at every pixel in the 4 cardinal directions until a red pixel is found

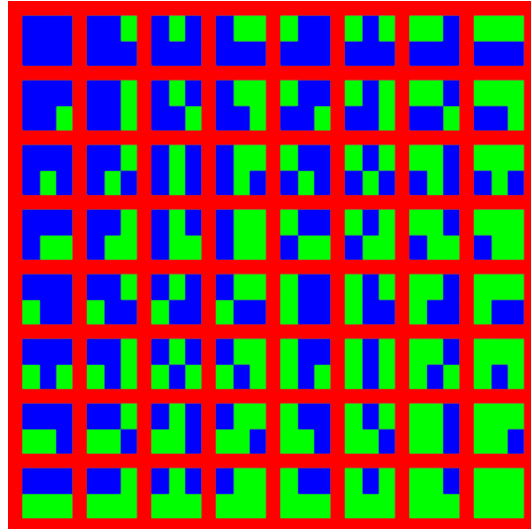


Figure 2: The target: an array of color codes drawn with 3 colors, Red (background), and Green and Blue (code).

(see Fig. 3). The row and column numbers are then directly inferred from the green or blue color of the 6 pixels relating to the code-units.

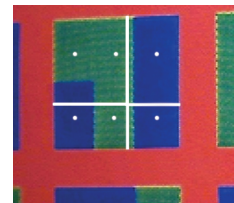


Figure 3: Color code analysis. The 6 code-units are directly accessed from the borders of the cell.

From the detected codes and the relative position of the starting pixel in the cell, the  $x$ - $y$  position of the target can be computed. The  $z$  coordinate is inferred from the cell width in pixel unit. Indeed, the farther the target from the camera, the smaller the distance  $z$ . A simple filter based on the previous records is used to avoid the jittering coming from the hand of the users.

We designed a very fast algorithm to maximize CPU capability to the end-user application. The target position can be estimated in very little time from the input images – less than one quarter of millisecond on our 400MHz PDA. Consequently, our approach allows a real-time interaction and can be used without penalizing the application.

## 4 Navigation in 2D data

The design of TangiMap has been initiated from the need to visualize large city maps on PDA. Let's imagine you have planned a trip to a city you don't know to take part in a conference. From the conference web site, you download a map onto your handheld computer, which indicates the location of your hotel and the location of the conference 2 miles away. In the street, planning your route with your PDA, you need to visualize the starting and ending point in a global view. At this level, the problem is that you cannot see the name of the streets. Consequently, you have to zoom-in to the starting point to focus on a reduced area. Then, you want to move the map to the end point, but you don't know exactly where you have to go as you do not see the end point anymore. Therefore, you have to zoom-out to have a global view, and zoom-in, and so on. With a stylus, zoom-in and zoom-out are generally performed by way of widget buttons, resulting in many coming and going operations. The same problem is also encountered with the majority of the online map browsers.

With TangiMap, the route planning is made more naturally because you virtually hold the whole map in your hand. The zoom operations are continuously performed by bringing your arm closer or farther from your device.

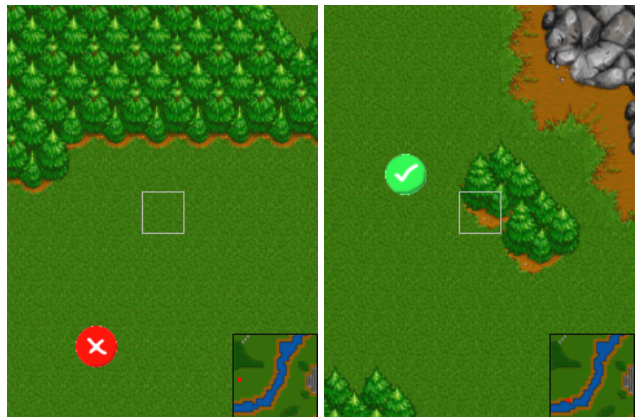
Using TangiMap for the visualization of large text documents such as PDF files or web pages as done in [12] is not appropriate. Indeed, reading long text on a small screen is mainly a linear task requiring only 1 DOF. Therefore, 1 DOF interfaces extended with a zooming control are much more adapted than TangiMap. TangiMap is well suited for the visualization of 2 dimensional documents such as maps or photos.

TangiMap is also particularly well suited to multiscale navigation as done in [2]. The navigation in the different order of magnitude is directly controlled by the 3 DOF of TangiMap. Similarly, applications using Scalable Vector Graphics (SVG [7]) can benefit from TangiMap, especially when adaptive level-of-detail are used [5]. For example, an architect who wants to check technical drawings (e.g. electrical circuits) on site with his PDA will be able to easily zoom-in to focus on details and zoom-out to have a more global view.

## 5 Evaluation

We wanted to evaluate the benefit of our interface as opposed to a classical stylus approach for a 2D searching task. Thus, we have set up an experiment where the subjects had to find targets on a map. In the first step, we only wanted to compare the differences in performance when the 2 pan DOF was evolved. Indeed, as styluses have only 2 DOF, an arbitrary technique has to be chosen

to control the third zooming DOF. In order not to compare our interface to a particular zooming technique, we chose to perform an experiment where only 2 DOF have to be controlled by the user. The classical drag-and-drop technique was used to control pan with the stylus. Pan was directly controlled by the TangiMap movements. A button of the handheld was used to activate or release the tracking of the target. Our hypothesis was that even with only 2 DOF to control, the user performance with our interface would surpass the user performance with the stylus for a 2D searching task.



(a) The red target has not been caught yet, its position on the map is drawn in the bottom-right minimap.

(b) The green target has been touched and the position of the next target is drawn in the minimap.

Figure 4: Screenshots of the experiment.

### 5.1 Task and procedure

The task consists of searching for a small target appearing on a map. We choose a quite simple map with some general landmarks such as a river, rocks and forests. As illustrated in Figure 5, the visible part of the map is displayed on the full screen while a small representation of the whole map appears as a minimap on the bottom-right corner. No indication about the location of the viewpoint appears on the minimap. Consequently, the users have to look for the target on the main map according to its global location on the minimap.

In a previous experiment, the current viewport was to drawn on the minimap. We noticed that the subjects didn't look at the main map anymore but rather focused their attention on the minimap. Therefore, the task was reduced to move the viewpoint box to the highlighted target without looking at the main map. This task was closer to a 2D pointing task than a task where the users had to find targets on a map. That is why we chose not

to display any information about the current location in the minimap. Moreover, clicking on the minimap with the stylus to jump to specific locations was not allowed. Indeed teleportation on the map implies a very limited knowledge of the environment, which is not well suited to a searching task. For example, directly jumping to the final destination does not make sense in the routing application we have previously described.

For each trial, a red target appears randomly on the map, and the corresponding point is highlighted on the minimap. The subjects are asked to position as quickly as possible the target to the center of the screen, by moving the map. Once the target is reached, the current target becomes green and the experiment continues with a new target.

16 subjects performed the experiment, with both the interfaces. All of them (10 males and 6 females) were graduated students between 22 and 27, and none of them used a PDA more than 5 times. Half of them began with the stylus while the other half began with TangiMap.

For each interface, the subjects performed 5 training trials and 15 measured trials, resulting in 240 records per interface. We measured completion time. We didn't try to measure the accuracy of the interfaces (i.e. the error rate for precise positioning). Indeed, there is no doubt that a stylus interface is more accurate than TangiMap. In the experiment we have performed, we focused on the ability of the users to quickly find targets on maps that cannot be visualized in their whole.

After the experiment, the subjects were asked to complete a questionnaire.

The experiment has been performed using a PocketPC Toshiba e800 PDA (XScale ARM at 400MHz) equipped with a CompactFlash FlyCAM camera with a resolution of 160×120 pixels. Our application has been coded in C using the Microsoft GAPI (Game API) to access the video memory.

## 5.2 Results

We used the paired t-test for the statistical analysis of the obtained means. The results have shown that TangiMap was significantly faster than a stylus approach.

	Stylus	TangiMap
Completion time means (s)	109.81	85.93
Standard deviation	39.37	29.36
Significance	$t(15) = 3.89, p = 0.001$	

These results are illustrated in Figure 5.

## 5.3 Discussion

The results can be explained by the structure of our interface. Indeed, by holding the tangible map, the users benefit from the kinaesthetic feedback in addition to the visual feedback. For example when a target appears on the top-left corner, the subjects have only to directly move the TangiMap target to a corresponding location. Then, the target can be found quickly. When a new target appears, the subjects move with their dominant hand the TangiMap target to the new location according to their non-dominant hand.

On the one hand, stylus interaction appears less efficient as the users have to move the map without knowing exactly where to go. The only feedback they have is the one provided by the visual cues. Consequently, subjects have to pay attention to the landmarks, which induces cognitive overhead.

As an informal experiment, we tested with a few subjects the same experiment without any background map. The users were totally lost with the stylus because they didn't have any visual feedback. On the other hand, they managed to find the targets more easily with TangiMap.

A faster technique with the stylus would be to directly click the location to go on the minimap. This technique has not been chosen as our goal was to visualize and to understand large documents such as maps. Indeed, as we have previously explained, direct jumps to specific locations do not allow a global knowledge of the environment.

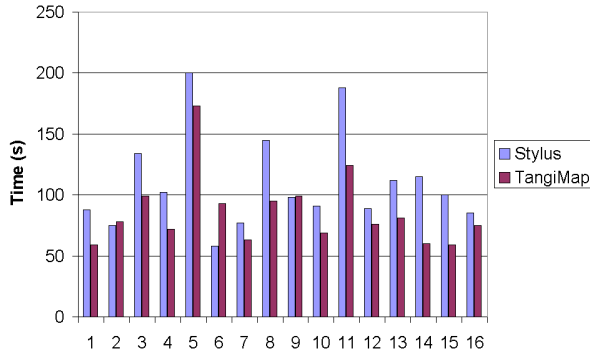
Finally, in this evaluation, the given map was not very large. A direct mapping between the map and the target of TangiMap was possible. For very large documents, such a direct mapping is not well suited anymore. Consequently, a first order transfer function has to be introduced to enable relative movements.

### Subjects comments

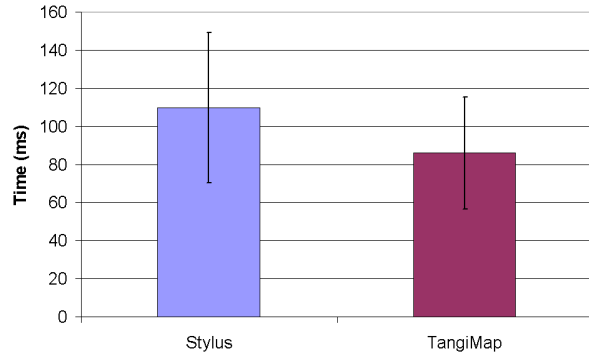
12 of the 16 subjects (75%) preferred using TangiMap rather than the stylus interface (Figure 6). They particularly appreciated the continuous and direct movements of TangiMap compared to the stylus. Indeed, for large displacements of the map, users had to apply several dragging movements with the stylus, which results in jerks. With TangiMap, users only had to make one main movement with their hand, resulting in a more direct interaction.

A technique consisting of moving the stylus from the center of the screen could have induced more continuous interaction with the stylus interface. However, the main drawback of such a technique is that it partially but permanently occludes the screen. Moreover, this technique requires a necessary learning period.

None of the subjects indicated that fatigue was a drawback of TangiMap in the free comment section. When



(a) Completion times of the subjects.



(b) Means of the completion times.

Figure 5: Experiment results.

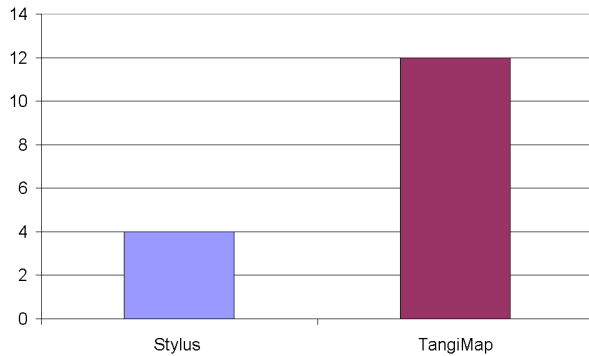


Figure 6: User preferences.

they were explicitly asked, 8 of the 16 subjects (50%) found that TangiMap induced more fatigue than the stylus interface. Half of the subjects found that applying several dragging movements was inducing more fatigue. During the experiment, we noticed that all the users were bringing the TangiMap target as close as possible to the camera. The closer the two hands, the more comfortable the position. Two issues limit the proximity between the TangiMap and the camera. First, at least one cell printed on the TangiMap target has to be visible in its whole for the tracking technique. The results of the experiment favor reducing the size of the cells on the TangiMap target. The second issue relates to the limitation of the low-quality camera. Indeed, the closer the TangiMap, the lower the quality of the image. With the new version of TangiMap, our tracking algorithm operates well starting from a distance of 5cm, which allows a comfortable use. The interface still operates well when the arm of the user

is tightened. Finally, we noticed that only one subject moved the handheld rather than moving the TangiMap target.

## 6 Conclusion and future work

Buttons and styluses are very well suited for many basic tasks on handheld computers. However, these input devices are maybe not the best solution for some higher-level tasks. In particular, we showed that even with adapted user interfaces, using a stylus to navigate in large documents suffered from several drawbacks.

We propose TangiMap, a new interface designed for interaction with mobile handheld devices. TangiMap is a 3 DOF interface benefiting from two-handed interaction, kinaesthetic feedback, and frame of reference. As the users hold the data, TangiMap can be seen as a tangible interface. We developed a very light algorithm enabling a real-time interaction using very few CPU ticks.

The experiment we performed showed that TangiMap was faster than a classical stylus approach and that the user preferences were widely in favor of our interface. It could be interesting now to compare TangiMap with a higher-level stylus user interface such as the speed-dependent automatic zooming interface proposed by Igarashi [12].

The success of TangiMap for navigation in large documents motivates us to extend the use of this new interface to the field of graph visualization. Indeed, directly holding the data thought TangiMap could favor the understanding of the abstract information. Moreover, specific information visualization techniques could benefit from the third degree of freedom. For example, a classical fisheye technique could be extended to a 3 DOF fisheye technique where the distance of TangiMap from the

camera would control the size of the fisheye.

## References

- [1] Ravin Balakrishnan and Ken Hinckley. The role of kinesthetic reference frames in two-handed input performance. In *Proceedings of the 12th annual ACM symposium on User interface software and technology*, pages 171–178. ACM Press, 1999.
- [2] B. B. Bederson and J. D. Hollan. Pad++: A zooming graphical interface for exploring alternate interface physics. In *Proceedings of the 7th Annual ACM Symposium on User Interface Software and Technology (UIST'94)*, pages 17–26, 1994.
- [3] Benjamin B. Bederson, Jon Meyer, and Lance Good. Jazz: an extensible zoomable user interface graphics toolkit in java. In *UIST '00: Proceedings of the 13th annual ACM symposium on User interface software and technology*, pages 171–180. ACM Press, 2000.
- [4] W. Buxton and B. Myers. A study in two-handed input. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 321–326. ACM Press, 1986.
- [5] Yi-Hong Chang, Tyng-Ruey Chuang, and Hao-Chuan Wang. Adaptive Level-of-detail in SVG. In *SVG Open 2004: 3rd Annual Conference on Scalable Vector Graphics*, 2004.
- [6] Daniel Fallman, Andreas Lund, and Mikael Wiberg. Scrollpad: Tangible scrolling with mobile devices. In *Proceedings of the 37th Annual Hawaii International Conference on System Sciences (HICSS'04) - Track 9 (To appear)*, 2004.
- [7] J. Ferraiolo, F. Jun, and D. Jackson. Scalable Vector Graphics (SVG) 1.1 Specification, 2003.
- [8] G. W. Fitzmaurice. Situated information spaces and spatially aware palmtop computers. *Communications of the ACM*, 1993.
- [9] George W. Fitzmaurice, Shumin Zhai, and Mark H. Chignell. Virtual reality for palmtop computers. *ACM Trans. Inf. Syst.*, 11(3):197–218, 1993.
- [10] Martin Hachet, Joachim Pouderoux, and Pascal Guitton. A camera-based interface for interaction with mobile handheld computers. In *I3D'05 - Symposium on Interactive 3D Graphics and Games*. ACM Press, 2005.
- [11] Ken Hinckley, Jeffrey S. Pierce, Mike Sinclair, and Eric Horvitz. Sensing techniques for mobile interaction. In *Proceedings of the 13th annual ACM symposium on User Interface Software and Technology (UIST 2000)*, pages 91–100, 2000.
- [12] Takeo Igarashi and Ken Hinckley. Speed-dependent automatic zooming for browsing large documents. In *Proceedings of the 13th annual ACM symposium on User Interface Software and Technology (UIST 2000)*, pages 139–148. ACM Press, 2000.
- [13] Robert J. K. Jacob and Linda E. Sibert. The perceptual structure of multidimensional input device selection. In *CHI '92: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 211–218. ACM Press, 1992.
- [14] Yoshifumi Kitamura, Yuichi Itoh, and Fumio Kishino. Real-time 3d interaction with activecube. In *CHI '01: Extended abstracts on Human factors in computing systems*, pages 355–356. ACM Press, 2001.
- [15] Ken Perlin and David Fox. Pad: an alternative approach to the computer interface. In *SIGGRAPH '93: Proceedings of the 20th annual conference on Computer graphics and interactive techniques*, pages 57–64. ACM Press, 1993.
- [16] Jun Rekimoto. Tilting operations for small screen interfaces. In *ACM Symposium on User Interface Software and Technology (UIST 96)*, pages 167–168, 1996.
- [17] Jun Rekimoto and Katashi Nagao. The world through the computer: Computer augmented interaction with real world environments. In *ACM Symposium on User Interface Software and Technology (UIST 1995)*, pages 29–36, 1995.
- [18] Michael Rohs. Real-world interaction with camera-phones. In *2nd International Symposium on Ubiquitous Computing Systems (UCS 2004)*, Tokyo, Japan, November 2004.
- [19] Miika Silfverberg, I. Scott MacKenzie, and Tatu Kauppinen. An isometric joystick as a pointing device for handheld information terminals. In *GRIN'01: No description on Graphics interface 2001*, pages 119–126. Canadian Information Processing Society, 2001.
- [20] Daniel Wagner and Dieter Schmalstieg. First steps towards handheld augmented reality. In *Proceedings of Seventh IEEE International Symposium on Wearable Computers*, 2003.
- [21] Ka-Ping Yee. Peephole displays: pen interaction on spatially aware handheld computers. In *Proceedings of the conference on Human factors in computing systems*, pages 1–8. ACM Press, 2003.