Spotlight Navigation: a pioneering user interface for mobile projection

Stefan Rapp
Conante
Fellenoord 130
5611ZB Eindhoven, The Netherlands
rapp@conante.com

alternative address
Dr. Stefan Rapp
Uhlandstr. 10
78559 Gosheim, Germany

ABSTRACT
Spotlight Navigation [4, 3] is - to the best of the author's current knowledge - historically the first general purpose user interface (UI) that has specifically been designed for mobile projection devices. From a user perspective, access to information through the projection device is performed by panning over a projected information space by moving the projection device as a whole, and by zooming into interesting regions using dedicated controls or specific gestures. The rendering component of Spotlight Navigation updates the projected image in real time according to the movements of the device, such that for users it appears as if they would interact on a huge virtual screen of which only the illuminated part within the projector's light cone becomes visible. In the paper we discuss fundamental aspects that differ from traditional desktop or hand held device interaction and describe interaction techniques that have been found suitable for mobile projection. All features and techniques presented here have been implemented on a working prototype that served as a basis for evaluation and elaboration of the techniques. We also discuss the setup and implementation architecture of our prototypes in more detail, and report on previously unpublished features of the prototype device, such as specific gestures, ripples and a text selection widget.

1. INTRODUCTION
Mobile projectors can be considered just a miniaturization of projectors for business presentation or home cinema use. They may also be seen as just another form of display technology for mobile devices, one that allows for larger screen space, despite a small form factor. Albeit all that is basically true, it is missing the essence: mobile projection devices are a new device class that deserve to have a specifically tailored user interface, to maximize its usefulness to the benefit of users. Spotlight Navigation has been developed along these lines, triggered by a simple question of how users may access and alter ultra high resolution information content when on the move. By using Spotlight Navigation as an interface, mobile projectors can be useful as an efficient, general purpose information processing device, and not just as a device to show movies or slide presentations, when plugged to a phone. So if users want to do more – how can they best operate a mobile projection device? Will they be using dedicated controls such as keys or a joystick on the device? Will they be using a pointing device, or interact directly on the projected image? In the author's view, none of these 'standard' ways of interacting with computers that are now common place for PC, tablet/PDA, (smart)phone, table-top surface or powerwall are best suited for mobile projection. As the usage context is so different from desktop or smart phone use, it is in order to rethink also some fundamental interaction principles.

2. USAGE CONTEXT
But let us start by looking at the usage context of a mobile projector. We postulate that users will want to use a projection device in ad-hoc situations, at various places and intermittent or occasional over a working day or during their free time, to quickly access information, or show information to others. As they will probably not carry a projection screen with them, they will want to project on any realistically suitable surface, such as a table, a wall, a sheet of paper, the floor, a book, or, if all else fails, a shirt, a curtain, a sleeve or the palm of their hand. If we look around, there are always suitable projection surfaces in convenient reach, and they tend to be surprisingly large, much larger than what current technology can handle with respect to brightness and contrast. The projection distance will vary, but it will be limited to comparably small distances due to the inferior light levels that are achievable with a battery-powered device of today's technology. We observe that there will not always be the possibility to lay down the device in an appropriate place, so in general, the projection will be from the device in the user's hand to a rather homogeneous, flat surface. In this situation, due to the naturally occurring small movements of the muscles, the so called tremor, projection is not perfectly stable. In consequence, if the physical resolution is relatively high, say XGA, it is advisable to prepare content as if for SVGA-resolution or lower, so that the information is still well readable also under tremor condition. The problem with tremor is not the translational variance but the rotational. If the projector is jittering left-right perfectly parallel (that is, without being rotated) by half a millimeter, the projected image is also jittering half a millimeter only. But if a projector of 10 cm length is only slightly rotated around its center, so that the front is shifted half a millimeter to the left, and on the back half a millimeter to the right, the projected image will shift already for 1 cm at one meter projection distance. It is also evident that any operation of keys or other controls that require force inevitably leads to movements of the projected image, as it can not be avoided to induce a tiny rotation on the device during operation. Rotations dominate translations. On the other hand, it is very easy to adapt the projector to a good position with respect to the projection surface, by intuitive and direct translation and rotation of the projector – everybody who ever operated a flashlight can do that, and it is much simpler than with stationary projectors. Also scanning over an extended surface or changing from surface to surface is instant and intuitive when the projector is in the user's hand.

3. THE FLASHLIGHT METAPHOR
People interacting with Spotlight Navigation appear as if they are searching for something with a flashlight. They shine a light here and there, stay on some spot for some moment and continue further. Alas, while acting in the real world, they are searching in the virtual world, and it is a portion of the virtual data space that they see in the projected light spot. While operating the device, its posture is continually measured and a corresponding image of the virtual data space is generated in real time in such a way, that the illusion is generated that the projected content appears stable and fixed to the surface, as if it would be a part of the real world. This linking of

location between the virtual and the real world is a major factor in the device's intuitivity. The projected light is unveiling a part of the virtual landscape, like making invisible ink visible with ultraviolet light. Searching for information gets as easy as using a flash light.

Pan and zoom user interface. Access to information within Spotlight Navigation is in general achieved by panning over a (theoretically) unlimited and resolution-less two dimensional information space. The panning is achieved directly by moving the device as a whole, to steer the light spot into the direction of interest. As previously explained, rotation dominates translation, so the typical style of panning the information space is to slightly rotate the device in the hand, while keeping its position. With effortless small movements, large distances on the projection surface can be quickly travelled. As the content stays at the same spot, it is easy to build a mental model of the information space, to quickly re-access previously seen items that are no longer visible with a flick of the hand. Once an interesting aspect in the data space is pointed at, users can zoom into the data space, either by a gesture or by simple controls on the device. Such a gesture could be approaching the projection surface with the whole device, or a gesture on a track pad or other touch sensitive surface on the device. In the Spotlight Navigation prototypes, the only way to zoom is via a scroll wheel on top of the device and under the user's thumb. Zooming in means that the area pointed at is enlarged and more details or additional information is becoming visible. While being independently developed, Spotlight Navigation resembles in many aspects Raskin's 'ZoomWorld' concept [5]. Whereas ZoomWorld is operating on a standard monitor and with standard input devices, it is the author's firm believe that through the direct pointing gestures of Spotlight Navigation, also to 'off-screen' locations, the panning and coordination with zooming is much more intuitive, and that this is a key enabler for making the pan and zoom interface actually work. **Clutching.** Panning and zooming is sufficient to travel across the unlimited data space even on a limited projection surface. If longer distances need to be travelled, zooming out, panning in the miniaturized view and zooming in again works. If needed, the sequence can be repeated after panning back. This operation is not very intuitive for beginners, so we introduced 'clutching', that can be used when through panning and zooming users have reached the end of a suitable projection area. During clutching (triggered by pressing the scroll wheel) the tricky processing of keeping things where they were is temporarily switched off, and the area currently shown is following with the device's movements over the projection surface. With it, the whole data space is translated to the new position: clutching is dragging the whole data space to a more convenient position, similar to moving a sheet of paper when sketching. Clutching becomes similarly intuitive and unconscious after a very short time. Panning, zooming and clutching together are a powerful way to explore and access an unlimited data space along visual cues.

4. IMPLICATIONS FOR INTERACTION

Having an unlimited, unbounded work area immediately makes several fundamental assumptions void, that we are so used to from our PC or mobile phone experience. Over the past decades, screen real estate has more or less stayed in the same order of magnitude, despite the advancements in computational power or memory. The predominant user interface paradigm can be seen as all about saving pixel space: windows are overlapping because there is not enough room for all; they are iconified to ease the management of screen space; there are menus that drop down or up so that they do not waste space while not in use; menus must be made hierarchic to save space; there are scroll bars as we do not have enough space for larger data sets or images etc. With an intuitive panning and zooming access to an arbitrarily large and resolution-less workspace, many if not all of these interface design choices should be revisited, to see if they are still reasonable. A reason to keep them might be that people are used to the concepts from the PC or phone, and that it may be easier to port existing applications or develop platform independent solutions. To fully leverage the potential of the interface, it is advisable to go for alternative approaches: Rather than scrolling with a scroll bar, in Spotlight Navigation, users can intuitively use panning and zooming directly. While scroll bars give only little help in where in the overall picture the current view port is, as the visual feedback is rather small and indirect, with Spotlight Navigation there is a direct and prominent feedback via the location of the projected light spot. Rather than overlapping windows, Spotlight can use tiling of windows so that nothing is obscured. Applications can run concurrently side-by-side, and switching between them needs nothing more than a flick of the hand, no need for task managers or a task bar. There is also no explicit form of iconification, of course items get smaller when zooming out and larger when zooming in. This may be viewed as some form of iconification, but it is gradual, in contrast to switching from one state to the other.

Focussed work style. In principle getting rid of the various forms of hiding information to save space should make it easier to navigate, as everything is readily visible. Of course, due to the zooming, not all details will be always visible, but it is easy to zoom into and check for the details, also see 'semantic zooming' below. Hiding also has the task of structuring (as in menus or folder hierarchies) as well as focusing on relevant information and filtering out the irrelevant for the task at hand (such as opening only a tiny portion of the documents available on storage). With Spotlight Navigation, as we have unlimited workspace, the decision on what is relevant and what is not is done by the user, selecting which portion of the data to view. In Spotlight Navigation, we call this focused work style. There are no categorical boundaries, everything is seamless: users may focus on a set of architectural plans, on the layout of the children's rooms in the first floor, or on the placement of a window or door. There are no abruptly changing views in spotlight navigation. Recent developments such as the UI for the iPhone take care that transitions in appearance are always animated, so that the users get a clue on what is happening, that they do not get too much confused by changing screen content. Spotlight Navigation preconceived this and implements it throughout: it is the users who select the good view and appropriate level of detail in their constant, daily routine operations. For example when working with the calendar, no explicit interactions are needed to switch from a week to a month view or change to the next week, everything is accomplished by the same panning and zooming used throughout the interface. This smoothness and being in control makes the behavior of artifacts in the virtual world more similar to what we are used to in the real world. Users can easily lay out or rearrange information to their needs, and can utilize their spatial memory to organize things. It is advantageous to also include visual cues, or landmarks, in the data space so that visual orientation is easier while navigating through it. The interface thereby takes care to not overload users with irrelevant information through semantic zooming – depending on the zoom level, the way that information is displayed can be different. For example the clock, when viewed from a distance, only shows the hour and minute hand, whereas if users zoom in, more and more information is shown, such as the second hand, a day-of-month display etc. Similarly, in the calendar, the individual scheduled appointments only get visible when users have zoomed in sufficiently that this makes sense. Otherwise, every day is only represented by its day-of-month number. In the Spotlight Navigation prototype, semantic zooming is often implemented by mipmapped textures, that have different information at the various mipmap levels. The rendering hardware then automatically selects the appropriate representation based on the currently visible size, and even implements a smooth blending over at no extra implementation cost.

Gestures. Apart from the deictic pointing gestures for selecting the view or point-and-click interaction, further gestures are used. First, the orientation of the working surface in relation to the projection device is defined by holding the device orthogonal to the projection surface during power on. The device will store this orientation until it is switched off. In practice, small misalignments are not problematic for interaction. Second, there is a specific home gesture. When users get lost in the data space by zooming in or out too far, quickly pointing to the ground and up again brings the data space scaled back full screen in the projection. Finally, there is a gesture for swapping walls that is especially useful for working in the corner of a room. As the device has no sense of where the corners of a room are, the trick is to build on the assumption that in most cases, adjacent walls are orthogonal to each other. In order to swap walls, users can simply point to the other wall, and as soon as they exceed somewhat more than 90 degrees, the virtual data space is swapped to the other wall by rotating it by 90 degrees. If walls are not perpendicular to each other, users have to switch off and on again to define the orientation of the new projection surface. Swapping walls also works with floor and ceiling. By turning the device up, as soon as the zenith is exceeded by some degrees, the data space is flipped on the ceiling, or similarly on the floor after pointing a bit behind the stand point. On floor and ceiling, it is also possible to rotate the data space by clutching and rotating the device in the appropriate direction. Here, clutching not only translates the data space but also rotates it, which is specifically useful, if users want to show information to others sitting on the opposite side of a table.

Pointer interaction. Spotlight Navigation can build on many successful interaction techniques developed in the last decades. For example, point-and-click or direct manipulation techniques, as well as drag-and-drop can directly by used. Rather than having a pointing device that can be moved within the current view to define the action point, Spotlight Navigation has a cross in the center of the view that defines it, so manipulating the virtual world is tightly coupled to the routine pan and zoom gestures. As an example, in our prototype's calendar application, scheduled appointments can easily be moved to some hours later within a day by pointing to it, pressing a button, dragging to the right time and releasing the button (just like in most PC applications). In contrast to other applications, through the panning and zooming, exactly the same method can be used to move it also to the next day, week, month or a completely new day all across the whole year. Similarly, other interactions known from traditional UIs can be directly utilized in Spotlight Navigation, like pressing a button, moving a lever, adding a check mark to an accomplished task and so on.

Scribbling. For text input, Spotlight utilizes a style similar to pen input, called scribbling. Similar to writing on a wall with a laser pointer, virtual ink is written in the projected area, following the hand's tiny rotational movements. It is comparably easy to write that way, more difficult than when using a pen directly, but seemingly simpler than writing with a mouse, as informal experimentation with a group of users has shown. This might be attributed to a tremor cancellation component that was developed that smoothes out the movements, so that the writing becomes less jittery and more readable and appealing. While this form of input is not suitable for continued text input, it is sufficient to take small notes, annotations or take down an appointment in the calendar, or write a small message. While in the Spotlight Navigation prototypes, just the strokes of the scribbled input are stored, it would also be possible to utilize a handwriting recognition component like with PDAs.

Text selection widget Another common user interface task is the selection among options, for example a file from a directory, or the selection of a country name from a drop-down-box in an address form. The latter is quite common, although list- or drop-down boxes are not the first choice when implementing a selection among so many entries such as all the countries of the world. A hierarchical grouping along continents may be preferable for instance. But the example illustrates, that developers tend to think along functional and not necessarily along usability lines. The goal for a text selection widget was therefore, to offer a single solution that scales well so that usability is included 'by design', to work reasonably well for a handful, several hundred or even thousands of entries, with the same fundamental mechanism. The solution is inspired by a text selection widget presented by Toshiyuki Masui et al. [2] but is adapted for usage with the Spotlight Navigation gestures. The solution takes advantage of the projected zooming interface in that it initially renders only a part of the entries, while hiding the majority in 'folds' that are also visualized graphically. Then, using zooming, more and more items can be made visible by the users, while they pan to the fold or folds of interest. Thus, instead of scrolling through an overly long list linearly, access to the wanted item is much faster: it is more like a logarithmic than a linear access (depending on the nature of the folds). For this to work efficiently, the way of 'folding' is important, and, that there is an ordering of the items such as the alphabetic order. Traditionally, in a long list, the first few entries are presented to a user, while all others are hidden, or, in the terminology used here, are in one large 'fold' at the end. Like in Masui's work, and in contrast to the common way, in Spotlight's text selection widget, we start with several folds, ideally of comparable size. So, if we are to show the list of countries, instead of showing the first 10 Afghanistan, Albania, ... down to Armenia, we show for instance Bolivia, Cyprus, Georgia, ... down to Zimbabwe, with stylized folds in between. If you want to look up China, say, pan to the fold between Bolivia and Cyprus and zoom in. After only a few zoom steps, China will become visible. Now, either click on it or pan over it and zoom out until there is only China left and thus selected. An obvious improvement is to not use blindly every nth entry, but items of high probability, if you have the additional knowledge. In our example, this may be population count (leading to Bangladesh, Brazil, China, India, ...), number of internet users in that country, or a distribution of existing customers etc. Also the component could be made adaptive in that it keeps a statistic on which items are selected by users, to make more frequent choices appear as early as possible, while maintaining also a quick access to infrequent cases (by balancing the size of the folds).

User notifications. Sun rays. Ripples. With the continued working in a partial view of the data space, there is sometimes the need to notify the users of events that are currently not in their view space, or locationless in nature. If Spotlight Navigation users send off messages by dropping a scribbled note over a business card, they are informed that the message has been sent by a temporary semitransparent overlay message in the screen center. In contrast to alert boxes, as they do not need a confirming button press and due to the semitransparency, it is possible to see through the message and continue working, but the message is rather obtrusive, which is OK for fundamental, infrequent process information or confirmation of an achieved user goal. It is equally suitable for really important events, such as energy running out in a few seconds. For a more unobtrusive notification, the information can be placed in the periphery (and in line with the trend in traditional desktop UIs). In Spotlight Navigation also two further ideas were developed, sun rays and water ripples, to notify the user of off-screen events (i.e. that also have an origin in the data space). An example is an alarm for an upcoming event in the user's calendar, the source of which can be located at the place of the respective calendar entry. Users can be notified in a scalable manner, from unobtrusive to highly visible, and also the location of the event is communicated. With sun rays, rays are cast from the origin that illuminate the current view in stripes (making the content under a stripe a little bit brighter). Through the size of the angle between stripes it can be directly inferred how far away the source of disturbance is and in which direction it lies. The notification can be scaled in intensity (how much the stripes are lightening up the content below, or by a 'glistening' pattern where neighboring stripes rapidly change in intensity). The idea that was finally implemented in the Spotlight Navigation prototypes is ripples. Here, the generating event literally makes waves that spread over the data space. Similar to Halos [1], from the diameter of the wave front users can judge the distance of the disturbance. Of course also the propagating direction gives a direct cue to the direction, and this is why ripples were preferred over sun rays. Technically the effect is implemented by rendering the current view on a polygon mesh that is distorted by some time dependent sinusoidal functions. Even though that water ripples on a wall do not have any real world correspondence, the visualization is sufficiently convincing, so that the metaphor works also on non-horizontal surfaces. Similarly to sun rays, the effect can be scaled from almost unobtrusive to a point where no reasonable working except tracing the origin of the disturbance is possible. This is done by simply adjusting the amplitude of the sinusoidal function and the decay / the number of wave fronts.

5. PROTOTYPES

There exist two hardware prototypes of Spotlight Navigation, manufactured in summer and autumn 2003 respectively. They utilize a monochrome XGA resolution projection engine developed by Sony's optical research group. It is using transmissive LCDs and a high power LED light source the light of which is collimated using a light pipe. The 2x2 LED array is passively cooled by means of a heat pipe. The achievable light output is comparable to current pico projector devices, about 10 Lumen, at about 5–7W electrical power. The prototypes are wired prototypes: the handheld projector is wired to a PC running the software. Also the signal conditioning for the LCD is done with an evaluation board outside the handheld case. Apart from the projection engine and some controls (the scroll wheel and two additional buttons taken from a mouse and also interfaced like that), the mobile case hosts an inertial navigation module (MT9 resp. MT9B from xsens). While the second prototype is designed for small appearance, the first prototype is component-based so that individual blocks for projection, orientation sensing, distance sensing, camera tracking, interaction controls can be snapped together in different configurations. The software is implemented in C++ using OpenGL for real time graphics generation. A standard (for 2003, now outdated) 3 GHz Pentium 4 hyper-threaded PC with Radeon 9800PRO graphics runs the software at full 60fps. Individual modules such as the orientation tracking, the tremor cancellation and the main visualization component communicate via a UDP socket. The software allows placements of images, has a calendar application with almost fully implemented functionality reading PDA calendar files directly, a clock and a VNC client through which web browsers or PC-desktops can be accessed. A rudimentary note pad and contact manager are also implemented to show drag and drop of scribbled messages. All other features mentioned above were also implemented, only the sun rays were left out in favour of the ripples, and not all possibilities in the text selection widget have been implemented, such as the adaptive behavior, or a sophisticated balancing of the folds. The software can also simultaneously show an overview of the whole data space and the currently chosen view, for instance for explanation or debugging purposes, or as a feedback to the user.

6. CONCLUSIONS

Spotlight Navigation is a fascinating and intuitive user interface that demonstrates that pico projectors can be used for more than just showing videos, pictures or slide presentations. With Spotlight Navigation it is efficient to access information from a vast information space quickly, and to manipulate data without the boundaries of a limited screen space. While legacy point-and-click applications using standard interface widgets can be integrated into Spotlight Navigation easily, it is beneficial to rethink applications ground-up from a user perspective, to use natural representations or metaphors, a layout taking advantage of panning and zooming, natural gesture and specifically designed widgets such as those described in the paper. This will in general lead to a simpler and cleaner yet more powerful design. A consequence of holding the device in the hand is, that there are only limited possibilities to enter textual information. While the device is probably unsuited to enter extended texts, the proposed scribbling is sufficient for small annotations or editing tasks as well as composing short messages or to use it for sketching. The strengths of the device are in exploring and manipulating vast data spaces such as access to structured information, maps and plans, or large media archives. Currently, the PC-based prototype is ported to a mobile platform [3].

7. CHRONOLOGY

Spotlight Navigation has been invented on 18th November 2002 by Dr. Stefan Rapp. It was presented to a group of co-workers, all user interface professionals at Sony research labs Stuttgart the next day. After the management has been convinced to support practical investigation further, a small project team of 4 researchers was put together, that started implementation work from April 2003. A first prototype was presented in September 2003 to managers from across the world inside Sony during a lab evaluation in Stuttgart, and in December 2003 to hundreds of visitors including the Sony top management at the Sony technology and exchange fair (STEF) in Tokyo. The first presentation to the general public was at Pervasive 2004 in Vienna.

8. ACKNOWLEDGEMENTS

The author wants to acknowledge the core team members Martin Barbisch, Ronan Bohan and Zica Valsan for contributing to the Spotlight Navigation prototype software, Martin Osen for conceptual work as well as the interaction and industrial design of the prototypes, Jason Williams for usability advice, Georg Michelitsch for conceptual work and discussions, Martin Emele for discussions and organization, Haruyoshi Suzuki for taking the funding decision, supporting us at Sony and enabling the spin-out, Manuel Teijido and Frederic Ludley for the smooth collaboration on the optical engine, and the people at Sony Alsace Factory for helping with producing the prototypes in time.

9. REFERENCES

[1] P. Baudisch and R. Rosenholtz. Halo: a technique for visualizing off-screen objects. In *CHI '03: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 481–488, New York, NY, USA, 2003. ACM.

[2] T. Masui, M. Minakuchi, G. R. Borden IV, and K. Kashiwagi. Multiple-view approach for smooth information retrieval. In *ACM Symposium on User Interface Software and Technology*, pages 199–206, 1995.

[3] S. Rapp. Spotlight Navigation and LumEnActive: Ubiquitous projection user interfaces. In *Pervasive 2010 Adjunct Proceedings (demo sessions)*, Helsinki, May 2010.

[4] S. Rapp, G. Michelitsch, M. Osen, J. Williams, M. Barbisch, R. Bohan, Z. Valsan, and M. Emele. Spotlight navigation: Interaction with a handheld projection device. In *Advances in Pervasive Computing: A collection of Contributions Presented at PERVASIVE 2004*, pages 397–400. Oesterreichische Computer Gesellschaft, April 2004.

[5] J. Raskin. *The Humane Interface: New Directions for Designing Interactive Systems*. Addison-Wesley Professional, 2000.