

DIZI: A Digital Ink Zooming Interface for Document Annotation

Maneesh Agrawala and Michael Shilman

Microsoft Research, One Microsoft Way,
Redmond, WA 98052
{maneesh, shilman}@microsoft.com
<http://research.microsoft.com>

Abstract. Pen computing devices provide a natural interface for annotating documents with freeform digital ink. However, digital ink annotations are usually larger and sloppier than real ink annotations on paper. We present DIZI, a focus+context interface that zooms up a region of the underlying document for inking. Users write in the zoomed region at a comfortable size for the device. When the zoom region is shrunk back to normal page size, the digital ink shrinks to an appropriate size for the underlying document. The zoom region covers only a small portion of the screen so that users can always see the overall context of the underlying document. We describe several techniques for fluidly moving the zoom region to navigate the document. We show that DIZI allows users to create digital ink annotations that more closely mimic the look of real ink annotations on physical paper.

1 Introduction

One of the goals of pen computing devices, such as Tablet PCs and PDAs, is to provide an inking surface that emulates physical paper. Ideally, using such devices should feel just like writing on paper and the resulting digital ink should look just like real ink. However, a variety of ergonomic factors make it difficult to fully achieve either of these goals. These factors include:

Slip. Digital screens are smoother and slipperier than paper.

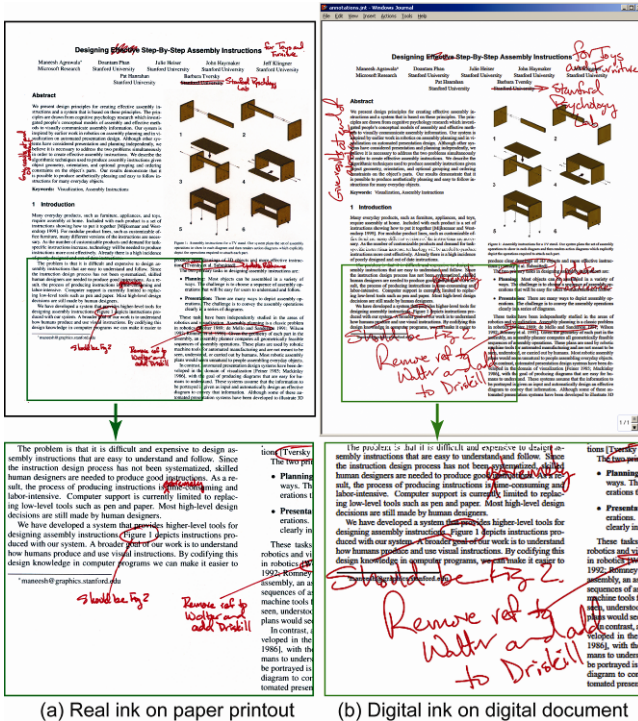
Resolution. Digital screens have less visual resolution than paper. In addition, digitizers usually track digital pens at lower resolution than the finest human hand movements.

Screen Size. Digital screens are often smaller than standard letter size paper.

Parallax. Digital screens often contain a protective layer of plastic that creates parallax between the tip of the digital pen and the screen.

Size and Weight. The physical size and weight of pen computing devices makes it hard to position and interact with them like paper.

While hardware designers continue to improve the feel of pen-based devices and make them more like paper, a substantial gap remains.



(a) Real ink on paper printout

(b) Digital ink on digital document

Fig. 1. Digital ink annotations are usually much larger and sloppier than real ink annotations on paper. As a result digital annotations may be difficult to read, especially if the reader did not originally write the annotation.

The ergonomic limitations of pen computing devices force users to change the way they write when using such devices. As shown in Fig. 1, digital ink is usually much larger and often sloppier than real ink annotations on paper. Creating legible digital ink is especially difficult when white space is limited, as it usually is when writing between lines, marking up individual characters or words, making comments in margins, or writing long comments. The resulting, larger-sized digital annotations often overlap the underlying text of the document and waste screen space because they are much less dense than real ink annotations.

To maintain context users typically need to see an entire page at a time. Yet, today’s pen computing devices have screens that are smaller than a standard sheet of paper, and so pages must be shrunk to fit on the digital screen. The shrinkage only exacerbates the legibility problems due to larger-sized digital ink.

We present DIZI, a focus + context interface that allows users to create digital ink annotations that mimic the look of real ink on paper. DIZI does this by magnifying a region of the underlying document for inking so that users can write at a size that is comfortable for the device. As we will show, we have experimentally found that a scaling factor of 2.0 for the zoomed region allows users to write comfortably and still produces legible natural looking ink when the region is shrunk back to the normal page

size. In addition, the zoom region is designed to cover only a small portion of the screen so that users can always see the overall context of the underlying document.

DIZI supports two primary forms of navigation, or moving the zoom region with respect to the underlying document; the user may move the zoom window explicitly to focus on a different part of the document, or the system may move the zoom window implicitly as the user is writing to ensure that the zoom region is always under the user's pen. We consider two strategies for explicit navigation, and we show that there is only one useful strategy for implicit navigation. We describe a simple but effective approach for combining implicit and explicit navigation to give the user full navigational control. Finally we show that the DIZI interface facilitates the creation of natural looking digital ink annotations.

2 Related Work

DIZI builds on two areas of previous work; zoomable user interfaces and free-form digital ink interfaces. We consider each of these in turn.

2.1 Zoomable User Interfaces

Zooming is a standard operation in many applications including document editing (Microsoft Word), photo manipulation (Adobe Photoshop) and vector-based design (Adobe Illustrator). However, these applications usually magnify the entire document and the surrounding context is not visible after the zoom. Thus, navigating in the zoomed view requires tediously scrolling through the document.

A better approach is to allow users to directly navigate a multi-scale focus + context view of the document, in which a magnified focus region show details while the surrounding context also remains visible. Researchers have developed a vast number of focus + context systems. for visualizing different kinds of documents including Fisheye Views[6], Perspective Wall[11] Document Lens[15], and Table Lens[14].

Researchers have also considered navigation in multi-scale space. Magic lenses[4] allow users to directly position a focus region (or lens) over the underlying document. Similarly, zoomable user interfaces such as Pad[12], Pad++[2] and Jazz[3], have applied panning and zooming navigation to a variety of tasks, ranging from viewing web pages, timelines, disk usage and maps, editing text and spreadsheets and presenting slideshows[8]. Users can pan and zoom to quickly reach any point in the multi-scale document space. Furnas and Bederson[7] use space-scale diagrams as a framework for describing this type of multi-scale navigation. Igarashi and Hinckley[9] also use space-scale diagrams to develop an automated zooming algorithm for browsing large documents. However, none of these systems consider the constraints on multi-scale navigation required to support digital inking.

2.2 Digital Ink Interfaces

One common use of freeform digital ink is to annotate digital documents. Schilit et al.[16] have developed xLibris, the first complete paper-like "active reading" system, in which ink annotations are used for collaboration and information retrieval. Follow-on systems include Adobe Acrobat Reader, Microsoft Windows Journal, Multivalent

Documents[13] and Callisto[1]. While some of these systems provide zoomable views and others support digital ink annotations, none of these systems provide both focus+context zooming and digital inking.

Another use of digital ink is note-taking. In work that is most closely related to ours, Davis et al.'s NotePals[5] addresses the problem of writing scale versus display scale. Users specify a focus cursor in the overview area at the top of the screen. As they write in a focus area at the bottom of the screen, the ink is shrunk to 40% of its original size and placed at the focus cursor. Users must explicitly create new writing space when they fill the focus area, which can break the flow of writing. Seni's Treadmill Ink [1] similarly uses a focus area for ink input. In this system the focus area is constantly scrolling from right to left thereby continuously clearing space for new ink. But, users must modify their writing style to accommodate the scroll. Both NotePals and Treadmill Ink are targeted at inking on small-screen devices like PDAs and cell phones. In other work, Lank[10] describes a fisheye lens-based inking system for Personal Digital Assistants. None of these systems is designed for annotating documents.

In this paper we apply the zoomable interface approach to annotate documents with free-form digital ink. We adopt the scale-space framework to explore several strategies for navigating the multi-scale space and we show when each of these strategies may be useful while annotating documents.

3 User Interface

As shown in Figure 2(a), DIZI allows users to interactively magnify a portion of the underlying document for inking. Annotating the document with DIZI in this manner involves a sequence of three interactions.

Initiating the Annotation. To indicate the zoom region users perform an initiation gesture. While holding the pen button users touch a point on the screen, which we call the zoom origin. DIZI generates a rectangular zoom window around this origin and the transition from the unzoomed state to the zoomed state is animated so that the zoom window appears to grow out of the zoom origin. As described in Figure 2(b,c) we have developed a flicking gesture as well as a rectangle gesture to allow the user to specify the orientation and size of the zoom window.

Writing the Annotation. Users then annotate the document by writing within the zoom window. We set a magnification scale factor of 2.0 within the zoom window. As we will describe in the section on implementation design details, we empirically chose this scale factor so that users can write at a size that is comfortable for the device and so that the digital ink appears at a natural size when the zoom window is shrunk back down into the underlying document.

Completing the Annotation. To finish the annotation, users tap the document while holding the pen button. An animated transition shrinks the zoom window along with the fresh ink back into the underlying document.

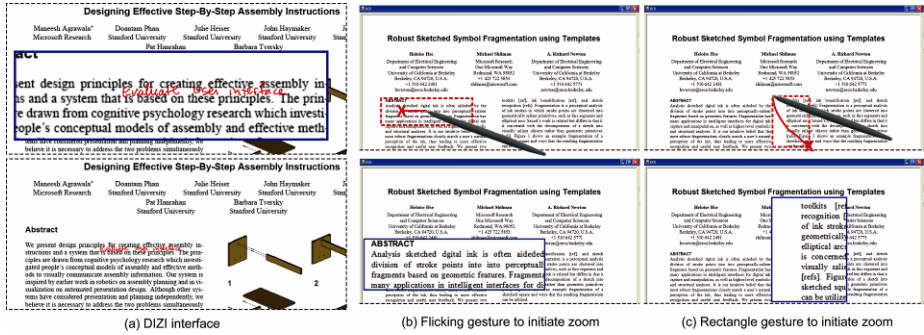


Fig. 2. DIZI interface (a) The user writes at a comfortable size within a magnified region of the document. When zoomed back down the writing appears at a natural size. Zoom initiation gestures (b,c) The user presses the pen button and places it at the zoom origin (red X). The path of the initiation gesture (solid red line from X to pen tip) denotes a region of the document to be magnified (dotted red box) (top). An animated transition shows the zoom window growing out of the zoom origin to produce the focus + context zoom window (bottom). Flicking gesture (b) The user quickly flicks the pen to create the zoom window. The direction of the flick sets the orientation (horizontal or vertical) of the zoom window. Because English is written left-to-right we offset the center of zoom windows so that 70% of the window is lying to the right of the zoom origin. Flicking upwards will generate a vertically oriented window which can be useful for writing in the side margins. Rectangle gesture (c) The user holds the pen button and directly specifies the rectangular region of the document to zoom. This approach allows the user to specify both the size and orientation of the zoom window.

3.1 Navigation

After users have initiated a zoom, they can choose from several strategies for navigating the document (i.e. changing the region that is zoomed). When the user actively chooses the navigation strategy we call this explicit navigation. Two such explicit strategies include:

Move Zoom Origin. One way to move the zoom region is to unzoom and zoom again at another location. However, this approach forces unnecessary switches between zoom mode and unzoom mode. Alternatively, we provide a control that allows users to remain in zoom mode and directly drag the zoom origin to a new location.

Move Document. Instead of moving the zoom origin users may want to leave the zoom window fixed, and simply change the portion of the document that is visible in the zoom window. The move document control allows users to drag the portion of the source document that is visible in the zoom window.

A key design goal for both of these explicit navigation strategies is that users should be able to reach any point in the underlying document. We will show how we achieve this goal in the implementation section of the paper.

In addition to these explicit navigation strategies there is also an implicit strategy that occurs without the user invoking a command:

Create Space. When the user’s writing approaches the edge of the zoom window, DIZI automatically repositions the window to create more writing space, as shown in

Figure 3. It is essential that this navigation strategy provide a smooth transition that doesn't disrupt the user's inking experience. As we will show in the implementation section, the only way to achieve this is to keep the zoom origin fixed as the zoom window is moved. Any other transformation will cause the surrounding ink and document to shift in the zoom window and break the user's writing flow.

While other navigation strategies are possible, we have found that these three suffice to accomplish common annotation tasks.

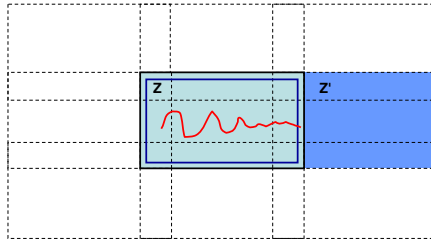


Fig. 3. If the user nears the edge of the zoom window while writing, DIZI automatically repositions the zoom window to create more writing space. We reposition the zoom window Z to one of the eight possible neighboring positions, based on the direction of the stroke. Here, as soon as the stroke exits the slightly smaller internal box (shown in dark blue) on the right side we reposition Z to Z' .

4 Implementation

Aside from inking, the primary form of interaction with DIZI is navigation, or moving the zoom region. In this section we first present a geometric framework for zooming interfaces. We then develop three navigation techniques based on this framework and show how two of them are combined in the default DIZI interface. Finally we describe several implementation design decisions we made in the process of taking DIZI from concept to usable prototype.

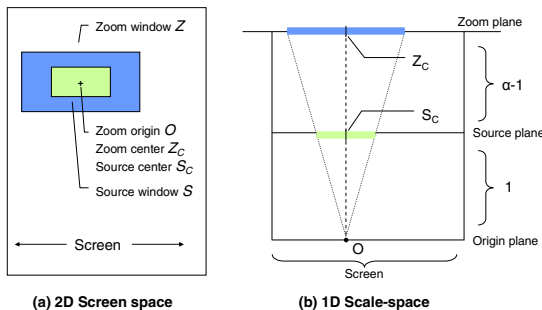


Fig. 4. A 2D screen-space diagram of our zooming interface (a), and a corresponding 1D space-scale diagram for the x-axis (b). The space-scale diagram shows that the mapping between the zoom origin O , the source window S , and the zoom window Z is a projection. The planes are spaced such that the projection scales any region on the source plane by scaling factor α .

4.1 Geometric Framework

Figure 4 shows a diagrammatic representation of our zooming interface. We adopt the space-scale diagrams of Furnas and Bederson[7] to depict the geometric relationship between the zoom region and the source document. In the space-scale diagrams zooming is treated as a projection from the zoom origin O , through the source window S , and into the zoom window Z . The planes are spaced such that the scaling factor α is given by:

$$\alpha = |Z| / |S| \quad (1)$$

where $|Z|$ and $|S|$ denote the width of the zoom window and source window respectively. By similar triangles, the center of the zoom window Z_C , the center of the source window S_C , and the zoom origin O are related by the following expression:

$$Z_C = O(1 - \alpha) + S_C \alpha \quad (2)$$

In fact this expression holds for any set of points Z_P and S_P lying on a projector line emanating from origin O . The space of geometric interactions with the system can be expressed in terms of these parameters. Assuming a given scale factor α , we can vary any two of the remaining parameters and the third is fully constrained.

4.2 Navigation

All three of DIZI's navigational strategies are achieved by setting the parameters of this geometric framework in different ways, as shown in Figure 5.

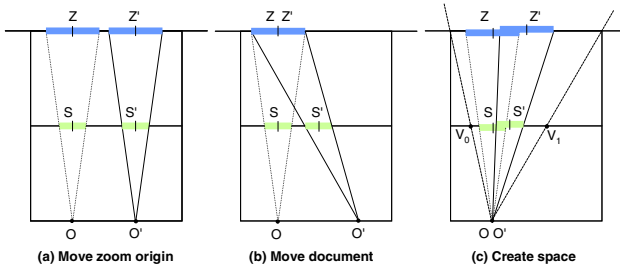


Fig. 5. Navigation strategies described using scale-space diagrams. (a) To move the zoom origin, the zoom window center O is locked to the zoom window center Z_C (b) To move the document, Z_C is fixed while S_C is dragged to S'_C . (c) To create space, the zoom center Z_C is moved to Z'_C while the zoom origin O remains fixed thereby preserving the mapping between the source and zoom planes. Note that since the zoom and source windows partially overlap we have slightly offset them vertically in the diagram. Both the move document and create space strategies shear the projection frustum to generate off-axis projections.

Move Zoom Origin. Our approach to moving the zoom origin is to lock the origin O to the zoom window center Z_C . As shown in Figure 5(a), when the user clicks at Z_C we set $O' = Z_C$. Using Equation 2 to solve for the new source center S'_C we see that $S'_C = O'$. As the user drags the zoom window, we translate all three parameters equally. This approach ensures that users can zoom up any point in the underlying document as they drag the zoom window.

Move Document. To move the document, the user drags SC to a new position SC' , as shown in Figure 5(b). In this case we keep the zoom window fixed by setting $ZC' = ZC$, and then solve for O' using Equation 2. This approach also ensures that users can zoom up any point in the underlying document but here they are dragging the source window instead of the zoom window.

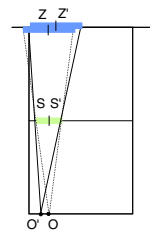
Create Space. When the user's writing approaches the edge of the zoom window we move ZC to ZC' such that the zoom window remains under the pen and provides more space to continue writing. In order for users to continue writing in a natural way, it is critical that the mapping between the source plane and zoom plane remains fixed. Otherwise the document would shift under the pen and disrupt the user. In the space-scale framework the only way to maintain this mapping is to keep the zoom origin O fixed. Therefore we set $O' = O$ and given the new ZC' we can compute SC using Equation 2. This approach allows the user to continuously draw strokes off the edge of the zoom window. However, because the mapping is fixed, in this case users can only reach a subset of the document as delimited by $V0$ and $V1$ in Figure 5(c). To reach other parts of the document users must unzoom and re-initiate the zoom in a new location.

Combining Navigation Strategies. To perform explicit navigation users must enter the appropriate navigation mode (either move zoom origin or move document) by tapping the corresponding button in a button bar. Initially this was the only way to perform explicit navigation. However, our early users pointed out that they sometimes wanted faster access to the move the zoom window mode in order to quickly magnify a new part of the underlying document. Based on this feedback we modified DIZI so that after the zoom window has been initiated, if the user places the pen outside the window, the system automatically goes into move zoom origin mode, and sets the zoom origin to the new location of the pen. The movement of the zoom window takes place immediately, without animation. We have found that this combination of navigation strategies in which we automatically create space when the user writes off the edge of the zoom window and we automatically move the zoom origin when the user places the pen outside the window, works extremely well in practice.

5 Design Details

In building a usable prototype of our zoomable annotation interface we made a number of low-level design decisions. We consider how these decisions affect the DIZI interface.

Initially Positioning Zoom Window. When the initial zoom origin is close to the edge of the document (as it typically is when making margin comments) the zoom window may extend off-screen. Our solution is to translate the zoom window and its center Z_C back on screen while leaving the source window and its center S_C fixed. Then we can again solve for O' using Equation 2. The inset figure shows how the projection geometry changes. If the zoom window is larger than the screen in either dimension, we translate the window so that



maximum portion of the zoom window is visible, under the constraint that zoom window center Z_C must always be in view.

Default Zoom Initiation Gesture. The default initiation gesture is flicking. In practice we have found that flicking is much faster and easier than the rectangle gesture. The rectangle gesture requires more effort and precision than the flicking gesture because the user has to actively think about the size of the window. With flicking the size of the zoom window is fixed and therefore the user only needs to consider where to place the zoom origin. However, in circumstances where it is important to control the size of the zoom window the rectangle gesture is more appropriate. We provide a checkbox to allow users to switch to rectangle gesture mode

Setting the Scaling Factor. To determine the appropriate scaling factor α , we conducted an informal study (3 participants) comparing the size of real ink annotations on paper to the same annotations written digitally on a TabletPC without the aid of our zooming interface. We scanned the paper documents and manually drew bounding boxes around each word of the both the real and digital annotations. We found that the digital annotations were between 1.4 and 2.7 times larger than the corresponding ink annotations. We then tried using several different scale factors within this range in DIZI and eventually settled on a scale factor of 2.0. We found that doubling the size of the document allows most users to write comfortably in the zoom window. Larger scale factors force users to deal with a lot more navigation to create long annotations.

The accompanying video shows how our implementation performs in practice. The video is in DivX 5.1.1 format and contains some visual artifacts due to compression.

6 Usage Experience and Discussion

We conducted informal trials in which 6 members of our lab first completed a short training task to learn the DIZI interface and then used DIZI to annotate a set of conference papers. The papers were edited a priori to include spelling and grammatical errors. Participants were instructed to both correct the errors and insert specific alternate sentences and comments.

Our goal was to determine the effect a zooming interface would have on the annotation process: how often users would use the zooming capability versus writing in the standard unzoomed, manner, where would zooming be most useful, how would the zooming interface affect the look of the digital annotations and would users link the interface. These trials confirmed some of the intuitions about the interface, but also included surprises.

While zooming is a primary feature of DIZI, the system also allows users to annotate documents without zooming. Figure 6 shows a typical page with zoomed and unzoomed ink rendered in red and green respectively. Users found zooming most useful when writing text or edit marks in areas of limited white space. Several users commented that their text annotations looked more like their written text on real paper. Users found it less useful to zoom when drawing long, figural annotations such as margin bars, underlines and circles. Such figural annotations usually require less precision than text annotations and can therefore be drawn without zooming.

We were surprised to learn that some users initially preferred the rectangle zoom initiation gesture to the flicking gesture. The rectangle initiation gesture provides full control over the area to be zoomed and is easy to learn because it is similar to selection gestures in many graphical interfaces. Talking with these users afterwards we found that their main complaint with the flicking gesture was that they found it difficult to guess the size of the zoom window. These users also found the gesture a little harder to learn, claiming that it was “unintuitive”.

Two of our users were given a longer training task that involved using both the flicking and rectangle gestures to initiate the zoom many times. These users preferred the flicking gesture to the rectangle gesture for its speed. While we believe flicking is a better choice as the default initiation gesture because it requires less effort and precision, these preliminary studies show that users must spend time learning how flicking works before they are comfortable with it. We plan to conduct more in-depth studies to evaluate the effectiveness of these initiation gestures.

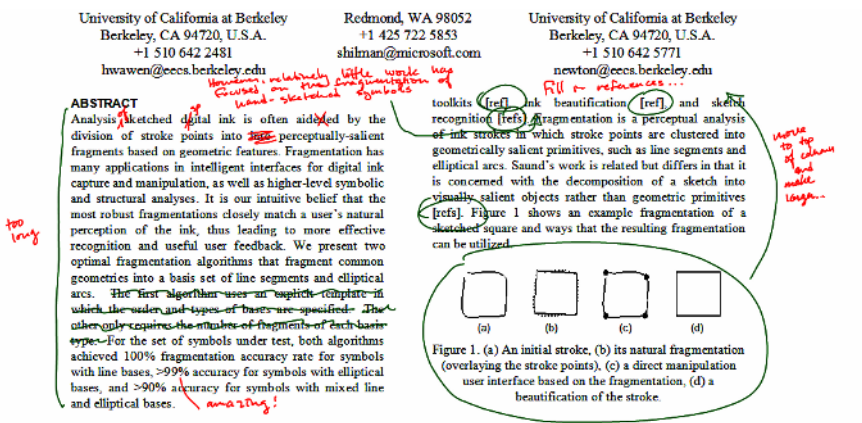


Fig. 6. Visualization of annotations created with DIZI. Text annotations (red) were created using the zooming interface. Figural circles, arrows, and strikethrough annotations (green) were created without zooming. In general we found that zooming was most commonly used in when writing text or edit marks in tightly constrained areas with limited white space. Most of the text in this example, including the long sentence starting, “However,...” would be impossible to fit in the limited white space without the zooming interface. These text annotations remain readable even after they are shrunk back down to page size. Long structural and figural annotations such as margin bars, underlines, arrows and circles are typically created without zooming because they require less precision than text annotations.

7 Conclusions

We have presented DIZI, a digital ink zooming interface that facilitates the creation of digital ink annotations with the same size and look as real ink annotations on paper. DIZI magnifies only a portion of the underlying document for inking so that user can write at a comfortable size of the device but still see the overall context of the page. We experimentally found that a scaling factor of 2.0 allows users to write comfortably and produces legible ink even when into the original document. We have explored

several different navigation strategies for moving the zoom region. Based on user feedback we combine an implicit strategy that automatically moves the zoom region so that it stays under the pen while the user is writing, with an explicit strategy that allows the user to directly move the zoom region to any part of the document. We believe that by allowing users to create digital ink that looks more like real ink our interface helps to bridge the gap between pen computing devices and physical paper.

Acknowledgments

We would like to thank Mary Czerwinski for her insightful comments and advice throughout the course of this work

References

1. Bargeron, D. and Moscovich, T. Reflowing digital ink annotations. *Proc. CHI 03*, (2003), 385-393.
2. Bederson B.B. and Hollan J. Pad++: A zooming graphical interface for exploring alternate interface physics. *Proc. of UIST 94*. (1994), 17-26.
3. Bederson, B.B., Meyer, J., and Good, L. Jazz: An extensible zoomable user interface graphics toolkit in Java. *Proc of UIST 00*. (2000), 171-180.
4. Bier, E.A., Stone, M.C., Pierce, K., Buxton, W. and DeRose, T.D. Toolglass and magic lenses: The see-through interface. *SIGGRAPH 93*, (1993), 73-80.
5. Davis, R.C., Landay, J.A., Chen, V., Huang, J., Lee, R.B., Li, F., Lin, J., Morrey, C.B., Schleimer, B., Price, M.N. and Schilit, B.N. NotePals: Lightweight note sharing by the group, for the group. *Proc. CHI 99*, (1999), 338-345.
6. Furnas, G. W. Generalized fisheye views. *Proc. CHI 86*, (1986), 16-23.
7. Furnas, G.W. and Bederson, B.B. Space-scale diagrams: Understanding multiscale interfaces. *Proc. CHI 95*, (1995), 234-241.
8. Good, L. and Bederson, B.B. Zoomable user interfaces as a medium for slide show presentations. *Proc of Information Visualization*. (2002), 35-49.
9. Igarashi, T. and Hinckley, K. Speed-dependent automatic zooming for browsing large documents. *Proc of UIST 00*, (2000), 139-148
10. Lank, E. and Phan, S., Focus+Context sketching on a pocket PC, Extended abstracts of *CHI 2004*, April 24-29, 2004, Vienna, Austria
11. Mackinlay, J.D., Robertson, G. and Card, S.K. The perspective wall: Detail and context smoothly integrated. *Proc. CHI 91*, (1991), 173-179.
12. Perlin, K. and Fox, D. Pad: An alternative approach to the computer interface. *SIGGRAPH 93*, (1993), 57-72.
13. Phelps, T.A., Wilensky, R. Multivalent documents. *Communications of the ACM*, 43, 6(2000), 82-90.
14. Rao, R. and Card, S.K. The table lens: Merging graphical and symbolic representations in an interactive focus + context visualization for tabular information. *Proc, CHI 94*, (1994), 318-322.
15. Robertson G. and Mackinlay, J.D. The document lens. *Proc. of UIST 93*, (1993), 101-108.
16. Schilit, B.N., Golovchinsky, G. and Price, M.N. Beyond paper: Supporting active reading with free form digital ink annotations. *Proc. CHI 98*, (1998), 249-256..
17. Seni, G. Treadmill ink - Enabling continuous pen input on small devices. *Proc. 8th International Workshop on Frontiers in Handwriting Recognition*, (2002), 215 -220.