

Virtual Notepad: Handwriting in Immersive VR

Ivan Poupyrev^{1,2}, Numada Tomokazu¹, Suzanne Weghorst²

¹ *IS Lab, Hiroshima University*
1-4-1 Kagamiyama
Higashi-Hiroshima, 739 Japan
{poup, tom}@isl.hiroshima-u.ac.jp

² *HIT Lab, University of Washington*
Box 352142
Seattle, WA, 98195 USA
{poup, weghorst}@hitl.washington.edu

Abstract

We present Virtual Notepad, a collection of interface tools that allows the user to take notes, annotate documents and input text using a pen, while still immersed in virtual environments (VEs). Using a spatially-tracked, pressure-sensitive graphics tablet, pen and handwriting recognition software, Virtual Notepad explores handwriting as a new modality for interaction in immersive VEs. This paper reports details of the Virtual Notepad interface and interaction techniques, discusses implementation and design issues, reports the results of initial evaluation and overviews possible applications of virtual handwriting.

1. Introduction

Writing is a ubiquitous everyday activity. We jot down ideas and memos, scribble comments in the margins of a book or an article, annotate blueprints and design plans. Using computers, we type documents, complete forms and enter database queries. However, writing, taking notes or entering text in immersive VEs is almost impossible. Cut off from conventional text input devices, such as keyboards, immersed participants are unable to effectively communicate textual information in VEs [1, 2]. Consider the following scenario: While walking through a virtual building and evaluating its design, the architect notices a flaw. Currently, this user is forced to exit the virtual environment in order to note observations or to access design plans. This is inconvenient, and important information and insights may be lost during the transition between the real and virtual workspace. Furthermore, on some occasions the architect may want to sketch design ideas or correct design plans while still experiencing the VE. However, the virtual workspace in which users are able to evaluate and experience phenomena of interest is currently separate from the workplace in which they perform their actual work.

We present Virtual Notepad, a collection of interface tools that allows the user to take notes, annotate documents and input text simply by writing with a pen, while still in an immersive VE. Using a spatially-tracked, pressure-sensitive graphics tablet, pen and handwriting recognition software, Virtual Notepad explores handwriting input as a modality for interaction in immersive VEs. This paper reports details of the Virtual Notepad interface and interaction techniques, discusses implementation and design issues, reports the results of initial evaluation and overviews possible applications of virtual handwriting.

2. Related work

The majority of current research on interaction techniques for VEs has focused on methods of direct manipulation, voice and gesture input as the prime ways of interacting in VEs. Although the importance of providing text input and note taking capabilities in virtual interfaces has been recognized [1, 3, 4], research investigating methods and tools for entering and manipulating textual data in immersive VEs remains sparse. Voice input with speech recognition has often been viewed as a possible method for the immersive input of text [1, 5]. However, text manipulation and editing is difficult to accomplish with voice alone, and current voice recognition technology suffers from poor performance in noisy environments and limited sensitivity to differences in the speaker’s voice and accent [5]. A virtual annotation system developed by Verlinden *et al.* [6] allows the user to add audio annotations to virtual objects. Annotations are represented as a small marker attached to the object, and can be selected later for playback. A similar system has also been developed by Harmon *et al.* [2]. The drawback of oral annotations is that although they are very easy to create, they are difficult to use. For example, editing and searching voice annotations can be a cumbersome task [2, 6]. Virtual keyboards have also been suggested as text input tools, but interaction with them has been reported as “clumsy”

and non-intuitive, and suffers from the lack of force feedback [1].

Although there have been no previous reported attempts to introduce handwriting into immersive VEs, there is a significant body of research investigating the application of handwriting recognition and pen input for conventional 2D interfaces (see, for example, [7, 8, 9]). There have also been a few attempts to apply pen input and 2D gestures to interaction with 3D environments. For example, the SKETCH system, developed at Brown University, allows the user to quickly sketch 3D scenes by drawing predefined 2D gestures using a mouse or pen [10]; and in systems by Billingham *et al.* [11] and Szalavári *et al.* [12] a pen and tablet are placed inside a virtual environment and are used as ubiquitous interface tools for creating and manipulating virtual objects.

3. The Virtual Notepad

To investigate the potential of virtual handwriting, we have developed Virtual Notepad, a collection of simple interface tools for note taking and text input in VEs. Virtual Notepad allows the immersed participant to write notes, annotate documents and images, and modify previously entered notes, as well as distribute them in VEs.

Virtual Notepad has been evaluated within a Virtual Reality Emergency Room (VRER) environment. A VRER has been developed at the Human Interface Technology Laboratory to quickly prototype and evaluate new medical interface metaphors and data representations [13, 14]. A task used to evaluate Virtual Notepad was to take notes and annotate medical images within the VRER.

3.1 The basics of the Virtual Notepad

Figure 1 presents the basic configuration used by Virtual Notepad. The participant is immersed using the head-mounted display and two Polhemus sensors, which track the position and orientation of the user's head and hand. The user is provided with a small pressure-sensitive tablet, which is tracked by a third sensor. Within the VE, we register the position of the virtual notepad with the position of the real tablet. Rotation and movement of the tablet in the physical world is translated into the corresponding movements of the virtual notepad in the immersive world (Figure 2).

Two major modes of interaction are supported. While the user is not touching the tablet with the pen, a virtual representation of the user's hand is provided so that the user can interact with objects in the VE as usual. When the participant touches the real tablet with the real pen, then Virtual Notepad is activated and the virtual hand is replaced with a virtual pen, which appears on the virtual notepad in the position corresponding to the position of



Figure 1. The physical setup of Virtual Notepad. The user is provided with a spatially-tracked tablet and pen for taking notes in the VRER.



Figure 2. The virtual notepad is spatially registered with the physical tablet. When the user draws on the physical tablet, the virtual pen “draws” on the virtual notepad.



Figure 3. Pen strokes can be removed by wiping an “eraser” over them. The red line indicates the path of the eraser on the notepad.

the real pen on the real tablet (Figure 2). This virtual pen provides the user with a constant visual reference for the location of the entry point. As the user draws on the physical tablet, the virtual pen “draws” on the virtual

notepad, creating the illusion of drawing directly on the virtual notepad.

3.2 Interacting with Virtual Notepad

The basic metaphor of Virtual Notepad is that of a hand held notebook. The user can write down notes on its pages, erase or modify them, flip pages to find a desired note, or tear pages out of the notebook. The pages of the notebook can be clean pages, pages with notes entered previously or any document or image which the user has added to the notebook. Below we discuss interaction techniques currently implemented in Virtual Notepad.

3.2.1 Taking notes in the virtual notepad. Like the Tivoli electronic whiteboard [8] and the Dynamite electronic notebook [7], Virtual Notepad uses digital ink without recognition approach. The user enters notes using freeform handwriting, which is captured as a collection of strokes—digital ink. This digital ink approach allows participants to take notes on any part of the virtual notepad, to write in any size and direction and to easily combine notes and drawings, an essential part of any note-taking process (Figure 5) [8]. One shortcoming of the pure digital ink approach is the lack of capability for text input, which is required to allow note indexing, searching, and input of commands or parameters (for example file names). To allow for input of text Virtual Notepad, therefore, also includes handwriting recognition capability, which is currently used only for command input. This functionality will be discussed further below.

3.2.2 Modifying notes in the virtual notepad. Mistakes by users while taking notes are inevitable, and editing and stroke manipulation gestures (meta-strokes) are therefore provided to allow for correction and modification of input [8]. In the current implementation of Virtual Notepad, we provide the user with a simple gesture for removing any individual stroke or set of strokes in the notepad. To remove unwanted strokes, the user simply flips the pen and erases them, by wiping the surface of the virtual notepad with the “eraser” located on the other end of the pen. As the eraser crosses a stroke, that stroke is removed (Figure 3). The red line indicating the path of the “eraser” on the notepad disappears when the user removes the pen from the tablet, cueing the end of the operation. Informal evaluation of Virtual Notepad has shown that this technique is intuitive and easy to use, since it capitalizes on our everyday experience with real pencils and paper. The ability to erase strokes seems sufficient for this application. If erasure is easy, it is much simpler to remove and rewrite faulty strokes than it would be to modify them.



Figure 4. The user can grab document from VE and “bind” it into the virtual notepad by intersecting virtual notepad with the document and pressing a button.



Figure 5. The user can annotate documents and images added to the virtual notebook. The digital ink approach allows for easy combination of notes and drawings.

3.2.3 Adding/removing pages in the virtual notepad.

The user can have as many pages in the virtual notepad as desired. These can be clean pages, pages with notes scribbled previously, or images and documents added to the notepad. Real-world documents can be added to the virtual notepad in either an off-line mode or while inside the VE. For example, if an architect wants to bring the plans of a building into the VE where this building is being evaluated, then she can scan the plans and instruct Virtual Notepad to add them when the application starts. If documents are already located in a VE, then the immersed user can simply grab them with the virtual hand and “bind” them into the virtual notepad. Binding can be performed by intersecting the virtual notepad with the desired document and pressing a button (Figure 4). Virtual Notepad

then adds the document into the notepad (Figure 5). The reverse operation can also be performed: the user can tear pages from the notepad and place them in the VE. This is accomplished by touching the virtual notepad with the virtual hand and pressing a button. The top page attaches to the virtual hand and can then be placed in any desired location in the VE (Figure 6).

3.2.4 Manipulating documents in the virtual notepad.

Virtual Notepad provides a set of commands that allow users to manipulate pages: flip through them, add blank pages or remove existing pages. We use the handwriting capabilities of Virtual Notepad, instead of using widgets, menus or other conventional means of command input in VEs. The user can simply write commands, anywhere on the virtual notepad, which are then recognized by the handwriting recognition engine and executed. For example, to access the next page of the notepad the user can write “N” or “n” (for “next”). To add a blank page, the user can write “A”, “a” or “add” and so on (Figure 7). One-letter commands were chosen to accelerate their input. To distinguish between the notes and commands we use “tense modes” [15] i.e., the user holds down the button on the pen while writing a command. A different ink color (blue) is used to visually distinguish written commands from regular notes. After execution, the command disappears.

3.3 Design and implementation

Virtual Notepad has been implemented using a custom VR software toolkit developed as an extension of the Sense8 World Toolkit version 6.0. An SGI Onyx RE2 graphics workstation, equipped with a Virtual Research VR4 head-mounted display and three Polhemus Fastrak 6DOF sensors is currently used. A Wacom Technology Corporation ArtPad II[®] 4” x 5” tablet is used as the handwriting input device. To avoid interference with the electromagnetic field of the tablet, the Polhemus sensor must be mounted about 5-8 cm outside the tablet (Figure 1). The CalliGrapher[™] handwriting recognition engine from ParaGraph International has been installed on a Windows NT workstation, and communicates with the SGI Onyx workstation using TCP/IP protocol. Below, we discuss some of the implementation issues of Virtual Notepad.

3.3.1 Implementation of the virtual notepad. Each page of the virtual notepad is implemented as a texture that is mapped on a single polygon and dynamically updated every cycle of the simulation loop with new handwriting input from the user. The highest possible resolution of the tablet (1000-2540 dpi, depending on the driver, accurate to 0.02”) is significantly higher than the current resolution of the texture (128 x 128 pixels), and the user’s input is

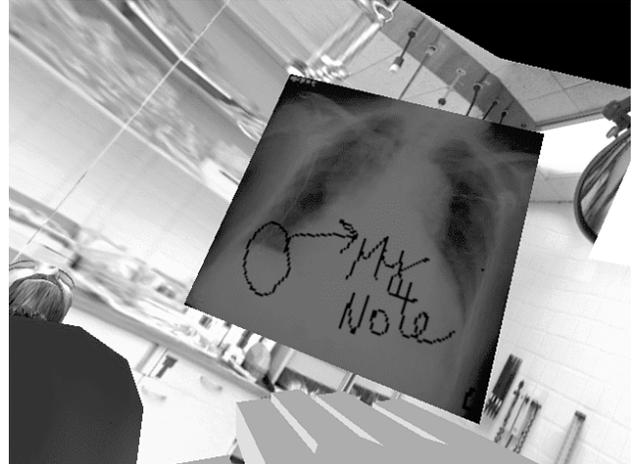


Figure 6. Pages can be “ripped” off the virtual notepad and placed anywhere in the VE.

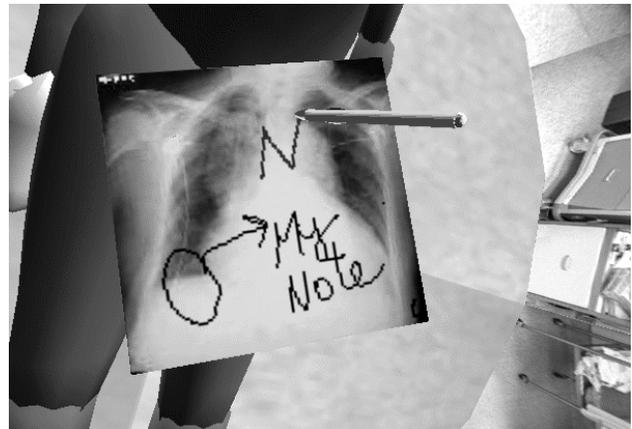


Figure 7. Handwritten commands are used to flip the virtual notepad’s pages. The user writes “N” to access the next page of the notepad.

therefore filtered to remove excess data. Increasing the resolution of the virtual notepad currently increases the lag between writing on the real tablet and the appearance of the writing on the virtual notepad. Initial evaluation shows that the current resolution is sufficient for simple note taking, since it allows for about 15-20 symbols per row of the notepad.

3.3.2 Modes of the virtual notepad. The user activates Virtual Notepad’s handwriting functionality by physically bringing the pen close to the tablet. Wacom tablets can accurately register the position of the pen within 4 - 5 mm from its surface. Therefore, Virtual Notepad is activated when the pen is close enough to be sensed by the tablet, rather than when the user actually touches the tablet with the pen. When the tablet senses the pen, the virtual hand becomes invisible and a virtual pen appears on the virtual notepad, allowing the user to write. Conversely, when the user removes the pen from the working range of the tablet, the virtual pen is hidden and the virtual hand reappears,

allowing object manipulation and other conventional VE interactions. Ability to sense the pen when it is not in physical contact with the tablet is crucial for seamless immersive handwriting. This is because when we write the pen is naturally lifted from the writing surface whenever we finish a stroke within a character, word, drawing, etc. If Virtual Notepad switched off each time the user lifted the real pen from the tablet, then the virtual pen would disappear after each stroke. Consequently, the user would not be able to accurately reposition the pen *between* strokes, because the visual reference for the entry point, the virtual pen, would not be available.

Virtual Notepad therefore supports two modes of interaction. When the physical pen is close enough to be sensed by the tablet, but is not touching it, then Virtual Notepad is in positioning mode, and the user can move the pen around without drawing on the virtual notepad. As soon as the pen contacts the tablet, Virtual Notepad switches to handwriting mode, allowing the user to write on it. Lifting the pen from the tablet (but staying within its working range) switches Virtual Notepad back to positioning mode, and so on.

3.3.3 Performance. The user has very fine control over the pen and a significant amount of handwritten data can be entered very quickly. To avoid losing data, we use a designated process that collects handwriting input, filters it to remove excessive data and then sends the resulting trajectory points to the VE rendering process. Unfortunately, there is noticeable latency between writing on the tablet and the stroke appearing on the virtual notepad. Evaluation of Virtual Notepad shows that, with practice, users can develop techniques to cope with this lag, but practical applications of immersive handwriting will require minimal latency. We are currently investigating various approaches to reduce it.

4. Evaluation and discussion

In order to identify the strengths and weaknesses of Virtual Notepad and to identify problems and possible improvements, an informal evaluation of the usability of the system was conducted. Immersive handwriting is just starting to be investigated, and informal observation of users was therefore considered an appropriate method for initial evaluation of Virtual Notepad.

This preliminary evaluation demonstrated that handwriting is an efficient, intuitive and enjoyable modality for interaction in VEs. As in many other interaction paradigms that employ real world metaphors, using handwriting allows us to take advantage of skills that users have developed in the real world and utilize them in the immersive environment. Virtual Notepad supports two-handed interaction with “tactile augmentation”, which results in

very intuitive interaction. The user can “feel” the virtual pen and notepad while writing, which enhances performance and the sense of presence within the VE [16]. Using two hands also provides physical support, as the user can rest his hand on the tablet while writing (Figure 1). Exact registration of the virtual notepad and tablet is not necessary, because the user can easily and unconsciously move his hands toward each other, and in relation to each other, in two-handed interaction [17]. The user can rely on kinesthetic cues to easily adjust for small discrepancies between the spatial positions of the tablet and the notebook.

The drawback of the real world metaphor is that while employing their real world experience for immersive interaction, users also base their expectations on their experience with a physical pen and paper. For example, most of the users expected an immediate response when writing on the tablet, which did not happen due to the latency of the system, and this often led to confusion. However, after practicing and receiving an explanation, users would learn to write more slowly and their performance would significantly improve. Generally, we found that visual realism was less important than realism of the pen’s responses and behavior.

Although exact registration of the virtual notepad relative to the tablet was not essential, its correct registration relative to the user’s body was important. When the user is holding the tablet in front of herself, she expects the virtual notepad to also be in front of her. Due to the kinesthetic cues mentioned above, the user never misses the virtual notepad, and can always write on it, but writing is uncomfortable and fatiguing when it is not properly registered with the body. The size of the virtual notepad is another important parameter. When the virtual notepad is too large, it obscures the VE and forces users to stretch their hands uncomfortably far forward, to allow themselves to see more of the notepad. Making the virtual notepad too small also makes writing difficult, because it requires very accurate pen movement and prompts users to keep the tablet as close to their face as possible to be able to see drawings in details. Further evaluation is important to develop guidelines for this size/distance trade-off for Virtual Notepad.

During the evaluation of the system we found that a simple set of handwritten commands that can be quickly jotted down and executed may be more expressive and easy to use for some tasks than virtual menus and widgets. In addition to eliminating the clutter of control objects and widgets, handwritten commands help to overcome some of the shortcomings of direct manipulation. For example, specifying operations on groups and subgroups of objects using direct manipulation methods can be a tedious task. An operation such as “remove all objects with names starting with ‘a’” would require manual selection of the objects that satisfy this condition [18]. On the other hand,

using a command language the user can easily specify the same operation as one command (for example “rm a*” for UNIX). Using Virtual Notepad and handwriting recognition, we can implement virtual command interfaces that are able to extend the traditional direct manipulation paradigm of VEs.

5. Conclusions and future work

In this paper, we present Virtual Notepad, a collection of interface tools which introduce and explore immersive handwriting interfaces. Using Virtual Notepad, the user can take notes while still *inside* a VE, experiencing and evaluating phenomena of interest. Furthermore, users can take documents from their physical workplace into their virtual workspace to be annotated. Thus, Virtual Notepad can potentially transform VEs from a place to gain experience into a place to do real work.

The potential applications of immersive handwriting go beyond note taking and include immersive collaboration, immersive modeling and content creation. Coupling handwriting input with recognition can provide natural and efficient virtual command language interfaces that enhance the traditional direct manipulation paradigm of VEs. Certainly, the tablet and handwriting do not replace direct manipulation, voice and gesture input, or other means of interacting in VEs. Instead, we see handwriting as an additional modality, which can be combined with existing VE interface paradigms to enhance their utility and result in effective and enjoyable interaction.

Our future research will both improve and add to the current functionality of Virtual Notepad. The problem of latency should be addressed by optimizing algorithms and the software architecture of the system. The current use of handwriting recognition is limited and more work should be done to investigate its potential. Our informal evaluation suggests that additional gestures for stroke manipulation might be useful, as well as tools that allow the user to magnify parts of images on the virtual notepad for detailed examination. The combination of handwriting interfaces with virtual widgets, direct manipulation techniques, voice and gesture input will also be investigated.

6. Acknowledgments

We thank ParaGraph International and the HIT Lab Virtual Worlds Consortium for their support of this project. We are also thankful to many individuals who helped us. Mark Billingham, Jennifer Feyma, Sisinio Baldis and Edward Miller contributed their time and effort at various stages of the project. Professors Ichikawa, Hirakawa and Yoshitaka provided important critical feedback, and Konrad Pollmann from Wacom Corporation explained the

technical subtleties of the Wacom tablet. We would also like to thank anonymous reviewers for their comments.

7. References

1. Angus, I., Sowizral, H., VRMosaic: web access from within a virtual environment. *IEEE Computer Graphics & Applications*, 1996. 16(3): pp. 6-10.
2. Harmon, R., Patterson, W., Ribarsky, W., Bolter, J., The virtual annotation system. *Proceedings of IEEE VRAIS'96*. 1996. IEEE CS. pp. 239-245.
3. Mine, M., Brooks, F., Sequin, C., Moving objects in space: exploiting proprioception in virtual-environment interaction. *Proceedings of SIGGRAPH'97*. 1997. ACM. pp. 19-26.
4. Bolter, J., Hodges, L., Meyer, T., Nichols, A., Integrating perceptual and symbolic information in VR. *IEEE Computer Graphics & Applications*, 1995. 15(4): pp. 8-11.
5. Rory, S., *The design of the virtual environments*. 1996: McGraw Hill. pp. 274.
6. Verlinden, J., Bolter, J., der-Mast van, C., Virtual annotation: verbal communication in virtual reality. *Proceedings of European Simulation Symposium*. 1993. SCS Ghent, Belgium. pp. 305-310.
7. Wilcox, L., Schilit, B., Sawhney, N., Dynamite: a dynamically organized ink and audio notebook. *Proceedings of CHI'97*. 1997. ACM. pp. 186-193.
8. Pedersen, E., McCall, K., Moran, T., Halasz, F., Tivoli: an electronic whiteboard for informal workgroup meetings. *Proceedings of INTERCHI'93*. 1993. ACM. pp. 509-516.
9. Wellner, P., Interaction with paper on the digital desk. *Communications of the ACM*, 1993. 36(7): pp. 87-96.
10. Zeleznik, R.C., Herndon, K.P., Hughes, J.F., SKETCH: an interface for sketching 3D scenes. *Proceedings of SIGGRAPH'96*. 1996. ACM. pp. 163-70.
11. Billingham, M., Baldis, S., Matheson, L., Phillips, M., 3D palette: a virtual reality content creation tool. *Proceedings of VRST'97*. 1997. ACM. pp. 155-156.
12. Szalavári, S., Gervautz, M., The personal interaction panel - a two-handed interface for augmented reality. *Proceedings of EUROGRAPHICS'97*. 1997. pp. 335-346.
13. Kaufman, N., Poupyrev, I., Miller, E., Billingham, M., Oppenheimer, P., Weghorst, S., New interface metaphors for complex information space visualization: an ECG monitor object prototype. In *Medicine Meets Virtual Reality: Global Healthcare Grid*, K. Morgan, et al., Editors. 1997, IOS Press. pp. 131-140.
14. Weghorst, S., Oppenheimer, P., Immersive prototyping of spatial medical information environments. Human Interface Technology Laboratory, University of Washington, Seattle, WA: *Technical Report R-96-1*. 1996.
15. Buxton, W., Chunking and phrasing and the design of human-computer dialogues. *Proceedings of IFIP World Computer Congress*. 1986. pp. 475-480.

16. Carlin, A., Hoffman, H., Weghorst, S., Virtual reality and tactile augmentation in the treatment of spider phobia: a case report. *Behavior Research and Therapy*, 1997. 35(2): pp. 153-159.
17. Hinckley, K., Pausch, R., Gobble, J., Kassell, N., Passive real-world interface props for neurosurgical visualization. *Proceedings of CHI'94*. 1994. ACM. pp. 452-458.
18. Gentner, D., Nielson, J., The anti-Mac interface. *Communications of the ACM*, 1996. 39(8): pp. 70-82.