

To be presented at 10th International Conference on Human-Computer Interaction (HCI2003), Crete, Greece, June, 2003.

Toward A Taxonomy of Interaction Design Techniques for Externalizing in Creative Work

Kumiyo Nakakoji

Yasuhiro Yamamoto

RCAST, University of Tokyo and PRESTO, JST

4-6-1 Komaba, Meguro, Tokyo, 153-8904

{kumiyo, yxy}@kid.rcast.u-tokyo.ac.jp

Abstract

Externalizing, the act of creating and modifying an external representation, plays a crucial role in creative design work. Whether a tool supports externalizing in a designer's desirable manner depends not only on what the tool does but also on its interaction design; how a user interacts with the tool through what representations. As an initial attempt to the development of taxonomy of interaction design techniques for tools that support externalizing, this paper examines seven tools for externalizing to illustrate why and how fine-grained interaction influences the process of externalizing.

1 Externalizing in Creative Work

The goal of our research has been to design and develop conceptual frameworks and computational tools that support creative design work (Nakakoji et al. 2000)(Yamamoto et al. 2000)(Fischer, Nakakoji 1994). This paper focuses on a designer's externalization processes, and uses a word *externalizing* to refer to the act of creating and modifying an external representation. Tools for externalizing provide a designer a space for the act of externalizing, supporting the designer's creative process and amplifying their creative ability.

There have been a number of studies that stress the importance of externalizations for human cognition. Scaife and Rogers (1996) use the notion of *external cognition* to emphasize the importance of graphical representations in helping human cognition. Arnheim (1969) states that visual perception is not a passive recording of stimulus material but an active concern of the mind and a viewer is engaged in problem solving while perceiving the visual stimuli. Zhang (1997) shows that different external representations help people differently in solving problems, and that choosing the right representation is critical to make problem-solving more efficient.

These studies, however, mainly focus on external representations as a given entity, and not on the externalizing process. A notable exception is Schoen (1983), who argues that an important aspect of the design activity is *reflection-in-action*. Differentiating *reflection-in-action* from *reflection-on-action*, he argues that for a designer, moving his/her hand while drawing sketches is crucial in the reflective thinking. In this sense, experiential cognition is not separated from reflective cognition (Norman 1993); they are intertwined. This suggests that activity and reflection should complement and support each other (Csikszentmihalyi 1990) at a very fine-grained level of interactions.

We use the term, externalize, to mean to cause state changes in a representation resulting in a different "what you see." Resultant externalizations are not merely expressions of mental imagery (Goldschmidt 1999), but rather, externalizing, the process itself, helps construct mental imagery, affecting what the person thinks, feels, knows and understands. Sketching is a typical example of

To be presented at 10th International Conference on Human-Computer Interaction (HCI2003), Crete, Greece, June, 2003.

the act of externalizing. Not only sketched objects, but also the sketching process itself helps a designer identify problems and formulate a solution space.

In order for a tool to support externalizing in a desirable manner, we must be concerned not only with what functionality the tool should provide, but primarily with how the designer interacts with the tool and through what representations. While sketching is a representation, holding a pencil and moving it while pushing its lead on a sheet of paper resulting in a black line is an interaction. The selection of sketching tools affects an architect's creative process (Lawson 1994). Seemingly subtle differences in a representation and interactions have a large impact on the effect of externalizing. The design of fine-grained representation and interaction, therefore, should be a central concern in the development of tools for externalizing.

Little is known, however, about what aspects of representations and interactions with tools are important in externalizing and how they are related to the externalizing process in promoting or disturbing a creative process. The rest of this paper examines seven systems as tools for externalizing, especially in early stages of design tasks. We discuss what aspects of the externalizing process each system does and does not support to address this issue.

2 Seven Tools for Externalizing: Illustrative Examples

Ideas. The Ideas system supports pen-based sketching and helps the designer in the manipulation of pages and resultant sketched objects (Hoeben 2001). The system mimics the conventional externalizing scheme of paper and pencil on a sketchbook, while extends it by providing a natural, smooth interaction through transparent operations on the representations.

Teddy. The Teddy system takes a user's 2D drawing action as an input and produces a corresponding 3D graphic object (Igarashi 1999). For instance, while a user draws a circle, a trajectory of a mouse appears forming a circle in a window (just like a conventional pen-based sketching interface), but when the user finishes drawing the circle (by creating a closed line), the circle is converted into a sphere in the same window. When the user adds another circle on top of the sphere, it is then converted into a sphere on top of the previous sphere. Repeating this process, the user can produce a hand-drawing 3D model consisting of objects, such as a Teddy bear. With Teddy, the externalizing process takes place at two levels. The first level is through a transparent operation. While drawing a circle, the mouse trajectory appears as a curved line and gives a feeling to the user that the user is directly drawing the curved line. The second level is at a larger granularity of interactivity (Svanaes 1999). Each time a user finishes a closed shape, Teddy converts it into a 3D object and displays it in the same window replacing the original shape. This externalizing process is not as closely integrated as the first level, but gives a user a feeling of externalizing.

VR Sketchpad. The VR Sketchpad system also takes a user's hand-drawing as an input, but produces a 3D architectural object in a VRML window, such as a wall, table, chair, or a TV set (Do 2001). The user can draw a floor plan on a sketchpad window, and when requested, the system first parses the sketchpad and identifies shapes (e.g., a circle), and produces architectural objects in corresponding locations (e.g. a table). VR Sketchpad supports externalizing process in two levels similar to Teddy. The first level is the same with Ideas and Teddy. The second level is even larger granularity of interactivity than Teddy. With Sketchpad, a user has to explicitly request for conversion to produce architectural objects from sketched objects. The converted 3D objects appear on a different VRML window and therefore do not replace original hand-drawings. The

To be presented at 10th International Conference on Human-Computer Interaction (HCI2003), Crete, Greece, June, 2003.

user may see the sketchpad as input and the VRML window as output. In this sense, the system might provide less feeling of immediateness in externalizing.

Silk. The Silk system takes hand-drawn configuration of user interface objects as an input, and produces functional, executable user interface objects as an output in a different window when requested by a user (Landay, Myers 2001). Similar to Teddy and VR Sketchpad, Silk supports externalizing process in two levels. The first level is the transparent operation as in the same as the above systems. In the second level, hand-drawings and the generated user interface components have a little more detached relationship than VR Sketchpad because the user has to specify constraints and behavior in order to produce functional interface components.

SketchAmplifier. The SketchAmplifier is a tool that amplifies the aspect of time spent in sketching. When a designer draws strokes in a canvas window (Figure 1-a), SketchAmplifier computes the time taken to draw the lines and generates a 3D representation by taking the time as the Z-axis (Figure 1-b). This representation is used to generate an animation (Figure 1-c) that illustrates how the lines have been drawn. With SketchAmplifier, the designer is not only able to immediately reflect on what has been drawn, but also on how it has been drawn in terms of the order and the speed of drawing, which is impossible with paper and pencil.

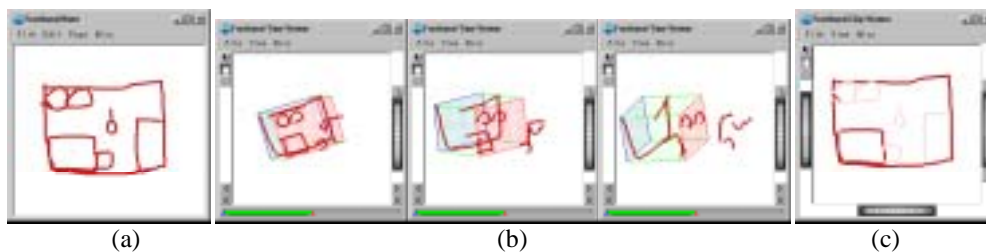


Figure 1: SketchAmplifier

ART#001. The ART#001 system aims at providing “sketches” for early stages of writing (Yamamoto, Nakakoji 2002). In addition to a regular way of externalizing words and sentences by typing a keyboard and applying copy-paste operations, the system allows a user to manipulate text as a chunk (element) with a 2D space view and a document view. Dragging a text element in the space view changing the relative position in the space will dynamically changes the order of the element in a document view. With ART#001, a user can get a feeling of directly manipulating text objects. Since the user is able to read elements in a document view in different orders by moving around the corresponding element in a space view without releasing the mouse (i.e., without making commitment), ART#001 allows the user to play what-if games by interacting with the representation.

SideView. The SideView tool allows a designer to compare results of applying different image manipulation filters in a GIMP-based image editor by providing previews (Terry 2002). Based on observations of graphic designers repeating a process of menu selection and undo to compare different effects of applying different image filters, the system is designed to extend each menu item with the previews of a result of applying the function to the current task. The focus of SideView is not explicitly supporting the user to externalize representations. However, how the user interacts with the system can be viewed as an externalizing process. A user takes an action (moving a mouse over a menu item), and the system shows a preview as a result of the user’s

To be presented at 10th International Conference on Human-Computer Interaction (HCI2003), Crete, Greece, June, 2003.

action. The user then is able to reflect on what would be the result of taking the action; just like drawing a rough line in a sketching pad and examines an outcome.

3 Toward Taxonomy

All of the above seven systems support externalizing in early stages of creative tasks by employing different types of interaction design techniques. They use different representations and interactions based on their purposes, tasks, goals, algorithmic constraints and the limitation of current computational capability.

As a first step toward the development of taxonomy of interaction design techniques for externalizing, we present a list of aspects that characterizes their approaches.

- *Representational Immediateness*: The first levels of externalizing discussed above (sketching in Ideas, VR Sketchpad, Teddy, Silk and typing in ART#001) provide low-level immediate representation for a user's action. In contrast, the second level of externalizing provide domain-rich, possibly more distant representational mapping of what a user acts to what the system shows as a result of externalizing.

- *Spatial Immediateness*: Where the user's action takes place and where the resulting externalization is displayed can be the same or different. Hand-drawing interface usually draws lines where the mouse cursor is. 3D objects generated in Teddy will appear in the same window, but representations generated in VR Sketchpad and Silk, and the animation of SketchAmplifier will appear in different windows. These decisions are primarily due to computational limitations (e.g., no algorithms can draw a 2D shape on a VRML window in an effective manner).

- *Temporal Immediateness*: What unit of interaction is interpreted as a meaningful unit by the system determines temporal immediateness. While simple mappings of actions (line-drawing and direct manipulation of objects) enable temporal immediateness, algorithmic constraints and computational power are two factors that affect the temporal immediateness. Algorithmic factors affect Teddy, for instance, which requires a user to finish drawing a stroke before the system can convert it into a 3D object. Available computational power affects SketchAmplifier, for instance, with which the speed of playing animation is limited.

- *Realism toward domain*: Both VR Sketchpad and Silk produce representations that are closer to the domain the user is engaged in by taking a user's hand-drawing. By focusing on a particular domain, such systems map a user's simple action to complex domain objects, allowing the user to have the situation talks back to them more grounded in the domain.

- *Realism toward verisimilitude*: Some tools for externalizing aim at providing representations that are more realistic. They include the real practice (paper and pencil in Ideas), physical objects (3D objects in Teddy), or temporal experience (sketch animation in SketchAmplifier).

- *Allowing what-if games*: Both ART#001 and SideViews turn "expensive" operations into affordable ones, allowing a user to play what-if games during the externalizing process.

4 Discussions

In order for tools to support an externalizing process, we have to consider: *what* a designer wants to externalize (representation), and *how* the designer wants to externalize it (interaction). Whether a representation is desirable depends on what interaction is possible with the representation, and what interaction is necessary depends on what representation is to interact with.

Existing HCI approaches, such as the notion of direct manipulation, virtual reality, interaction design, are necessary but not sufficient to help us address the concern. This paper gives our initial attempt to develop a conceptual framework and a language that can be used to describe what

To be presented at 10th International Conference on Human-Computer Interaction (HCI2003), Crete, Greece, June, 2003.

aspect of tools support *how* and *what* aspects of externalizing processes. We will continue this effort through the development of tools for externalizing in creative work and the examination of both success and failure cases of tools developed by others.

5 References

- Arnheim, R. (1969) Visual Thinking. University of California Press, Berkeley.
- Csikszentmihalyi, M. (1990) Flow: The Psychology of Optimal Experience. HarperCollins Publishers, New York.
- Do, E.Y-L. (2001) VR Sketchpad: Create Instant 3D Worlds by Sketching on a Transparent Window. Vries, B.d., Leeuwen, J.P.v. and Achten, H.H. eds. CAAD Futures 2001, Kluwer Academic Publishers, Eindhoven, the Netherlands, pp.161-172.
- Fischer, G., Nakakoji, K. (1994) Amplifying Designers' Creativity with Domain-Oriented Design Environments, Artificial Intelligence and Creativity, in Part V, T. Dartnall (Ed.), Kluwer Academic Publishers, The Netherlands, pp. 343-364.
- Goldschmidt, G. (1999) Design, in Encyclopedia of Creativity, Mark. A. Runco, Steven R. Pritzker (Eds.), Vol.1, Academic Press, San Diego, CA., pp.525-535.
- Hoeben, A. & Stappers, P.J. (2001) Ideas: Concepts for a Designers' Sketching-Tool, Extended Abstract of CHI 2001, Seattle, WA.
- Igarashi, T., Matsuoka, S. & Tanaka, H. (1999) Teddy: A Sketching Interface for 3D Freeform Design ACM SIGGRAPH'99, Los Angeles, pp.409-416.
- Landay, J.A. & Myers, B.A. (2001) Sketching Interfaces: Toward More Human Interface Design. IEEE Computer, 34 (3). pp.56-64.
- Lawson, B. (1994) Design in Mind, Architectural Press, MA.
- Nakakoji, K., Ohira, M., Yamamoto, Y. (2000) Computational Support for Collective Creativity, Knowledge-Based Systems Journal, Elsevier Science, Vol.13, No.7-8, pp.451-458, December.
- Norman, D.A. (1993) Things That Make Us Smart. Addison-Wesley Publishing Company, Reading, MA.
- Scaife, M. & Rogers, Y. (1996) External Cognition: How do Graphical Representations Work?, Int. J. Human-Computer Studies, no.45, pp.185-213, Academic Press.
- Schoen, D.A. (1983) The Reflective Practitioner: How Professionals Think in Action. Basic Books, New York.
- Svanaes, D. (1999) Understanding Interactivity: Steps to a Phenomenology of Human-Computer Interaction, Ph.D. Dissertation, Dept. of Computer and Information Science, Norwegian University of Science and Technology, Trondheim, Norway.
- Terry, M. & Mynatt, E. (2002) Side Views: Persistent, On-Demand Previews for Open-Ended Tasks UIST 2002, pp.71-80.
- Yamamoto, Y., Nakakoji, K., Takada, S. (2000) Hands-on Representations in a Two-Dimensional Space for Early Stages of Design, Knowledge-Based Systems Journal, Elsevier Science, Vol.13, No.6, pp.375-384, November.
- Yamamoto, Y., Nakakoji, K. & Aoki, A. (2002) Spatial Hypertext for Linear-Information Authoring: Interaction Design and System Development Based on the ART Design Principle Proceedings of Hypertext2002, ACM Press, pp.35-44.
- Zhang, J. (1997) The Nature of External Representations in Problem Solving. in Cognitive Science, pp.179-217.