DELIVERING INSTRUCTIONS FOR INHERENTLY-3D CONSTRUCTION TASKS: AN EVALUATION OF AUTHORING ENVIRONMENTS FOR MULTIMEDIA PRESENTATIONS

Guy Zimmerman, Julie Barnes, Laura Leventhal

Department of Computer Science
Bowling Green State University
Bowling Green, OH 43403 USA
Contact: gzimmer@cs.bgsu.edu

ABSTRACT

Today's WWW developers are faced with a myriad of format choices that range from text to still images to animations, movies and virtual reality (VR). In building presentations that incorporate one or more of these presentation formats, developers must not only choose the most effective multimedia combination of presentation styles, but also must select the tools that they will use to build their presentation. In this paper, we compare two authoring environments: Cosmo™Worlds and Flash. We created a standard presentation for the delivery of instructions for an inherently-3D construction task; we believe that a characteristic of this type of instructional presentation is that it will be enhanced by the presence of user-controlled 3D models. We compared both the efficacy of the two authoring environments and the effectiveness of the resulting presentations. We conclude that Flash was the more complete development environment for our presentation task. Both resulting presentations (VR and Flash) were equally effective.

1.INTRODUCTION

Recently, the number of presentation formats that are deliverable via the World Wide Web (WWW) have increased dramatically. Today's WWW developers are faced with a myriad of format choices that range from text to still images to animations, movies and virtual reality (VR). In building presentations that incorporate one or more of these presentation formats, developers must not only choose the most effective multimedia combination of presentation styles, but also must select the tools that they will use to build their presentation. In our recent work, we have found that multimedia presentations may vary greatly in their effectiveness for a given task. For example, we have found that for *delivering instructions for inherently-3D construction tasks*, presentations that include parallel textual and visual representations are comparatively more effective than those using text- or visual-only information. In particular we have found that when users are presented with and can control three-dimensional (3D) models and animations, their performance may be greatly enhanced. [2,7].

Once the developer chooses to build WWW multimedia presentations that include user-controlled 3D models and animations, how is (s)he to select the tool that will best suit her/his needs? The answer to the question is likely to be non-trivial. In choosing an authoring tool, the presentation developer will be making choices about both the development environment and the final product. In general, the designer will choose the tool that has the lowest cost in terms of resources and training. However, the choice of tool often determines the ultimate look and feel of the final presentation. From our prior work, if a developer wishes to include user-controlled 3D models and animations, their set of tool choices will be limited.

In this paper, we briefly overview the problem domain that is the focus of our work: delivering instructions for inherently-3D construction tasks. We argue that presentations that incorporate VR or 3D models are well suited to this problem domain. Next we describe two authoring tools, Cosmo™Worlds and Flash, that yield presentations of this type. We describe the two multimedia presentations that we built using these two tools. Finally we compare these tools along two dimensions: how well does the tool support presentation development and how effective were the presentations for the delivery of instructions in an inherently-3D task. We conclude that presentations built with these tools are very effective, but that there are differences between the tools themselves that should be considered when making a tool choice.

2.DELIVERING INSTRUCTIONS FOR INHERENTLY-3D CONSTRUCTION TASKS

There is a class of problems that we believe would benefit from the use of 3D representations. These problems involve the delivery of instructions to build a real-world object when the real-world object has some "inherently-3D" features. For example, assembling a model airplane is an inherently-3D construction task; delivering instructions for this task would fall under the class of problems that we are considering. Under our definition, inherently-3D construction tasks have the following characteristics: 1) The goal is to build a real 3D object. 2) Construction of the task requires a series of steps. 3) The object is asymmetric or changes symmetry during construction. In the case of a model airplane, there may be a passenger exit on only one side of the plane. See [5] for a more detailed description of inherently-3D construction tasks. Collectively the above characteristics suggest that instructions for an inherently-3D construction task would be enhanced by incorporating multiple views of the object at each step.

Instructions for inherently-3D construction tasks have traditionally been delivered in a paper form; these instructions typically use combinations of text and still pictures to present the instructions. When instructions for these types of tasks are ported to the WWW, they usually have retained their relatively static look and feel, incorporating text, still pictures and possibly a movie. However, in considering the inherently-3D properties of the real-world object in these types of construction tasks, we question whether these somewhat static presentations are ideal. Based on our prior results, we now believe that incorporating user-controlled 3D models, in the form of either: 1) VR and animations or 2) 2D models with multiple perspectives and animations would fit the problem space much better. [7]

In much of our work, we have focused on the construction of origami objects as our inherently-3D construction task. Origami folding satisfies our four criteria for an inherently-3D construction task. In addition, the complexity of folding origami objects ranges from the extremely simple to the highly complex, providing a myriad of target objects for instructional presentations. Instructions for folding origami objects are typically delivered in paper form, but a number of recent products have been marketed for computerized multimedia presentations [4,6].

3.AUTHORING TOOLS

The primary thrust of this paper is to investigate the efficacy of authoring tools to build multimedia presentations to deliver instructions for our target inherently-3D construction task domain. We required the tools to yield presentations with the following characteristics: 1) Supported the development and presentation of a data format for VR or 2D model with multiple perspectives. We considered this second case to be a sort of simulated 3D model. 2) Supported the development and presentation of user-controllable animations (play, stop, replay) 3) Provided a mechanism to integrate 3D models and animations with text and still images 4) Yielded a presentation that was web-deliverable and accessible via a web browser.

Two authoring tools, Cosmo™Worlds and Flash satisfied our requirements. Cosmo™Worlds provides a tool set from which the developer can build simple VR models, under the Virtual Reality Modeling Language (VRML) specification. VRML is a *de facto* Web standard for providing interactive 3D models. Flash permits the developer to build a Flash presentation; such a presentation may show a number of perspectives and thus yields a kind of 3D model. Both Cosmo™Worlds and Flash can be used to build user-controlled animations. The animations and VR or 2D models that emerge from these tools can be integrated with text and still pictures. Finally, the presentations that result are web-deliverable; both Flash and VRML have plug-ins for Web browsers.

3.1 Cosmo™Worlds: Authoring a presentation integrating text, still pictures and VRML models

To author our presentations that incorporated VRML models, we used the Silicon Graphics product, Cosmo™Worlds. Cosmo™Worlds provides both traditional 3D modeling tools and GUI tools to manipulate nearly every aspect of the VRML specification. In Cosmo™Worlds the author works in a development window. Within this window, the developer can manipulate the scene graph directly using an object browser facility or by direct manipulation (via mouse) of objects in the development window. VRML nodes can be grouped together into a Switch node. Cosmo™Worlds supports keyframe animation, which allows the developer to animate VRML models. Authors can easily set characteristic material properties, such as color, for the model. Authors can also make the VRML models more realistic by applying texture maps to a given surface.

One very useful component of Cosmo™Worlds is the PEP tool suite. This set of tools allows direct manipulation of individual Points, Edges and Polygons. For example, a selected polygon can be automatically split into two pieces. This was useful for our target task; with each new fold line one polygon becomes two new polygons. Finally, an optimization tool is available to reduce the number of polygons, file size and overall complexity.

We note that we did consider other formats and authoring tools for the development of integrated VR presentations. We specifically considered Java 3D and World Up. [10] We rejected Java 3D as being too low-level compared to VRML. World Up is a full-featured 3D system, but was inappropriate for several reasons. Unlike VRML, World Up is currently not a Web standard and the necessary plug-in is obscure. Finally, integrating WorldUp presentations with HTML was problematic. We did feel that the World Up toolkit may be a better overall development tool in the future for problems of this type, in no small part due to its integrated, object-oriented, scripting feature.

3.2 Flash: Authoring a presentation integrating text, still pictures and models from multiple viewpoints

Macromedia's Flash is a very popular authoring environment for creating interactive vector graphics and 2D animations for the Web. Flash provides an authoring environment for creating animations for Web pages. It also offers support for streaming audio and fast Web delivery. In addition, using Flash, it is possible to integrate text and/or still picture presentations. [9] Unlike Cosmo™Worlds and VRML, Flash is both a development tool and a format.

The Flash authoring environment provides the developer with a vector-graphics editor to create 2D graphical objects; these objects are combined into a Flash animation scene.

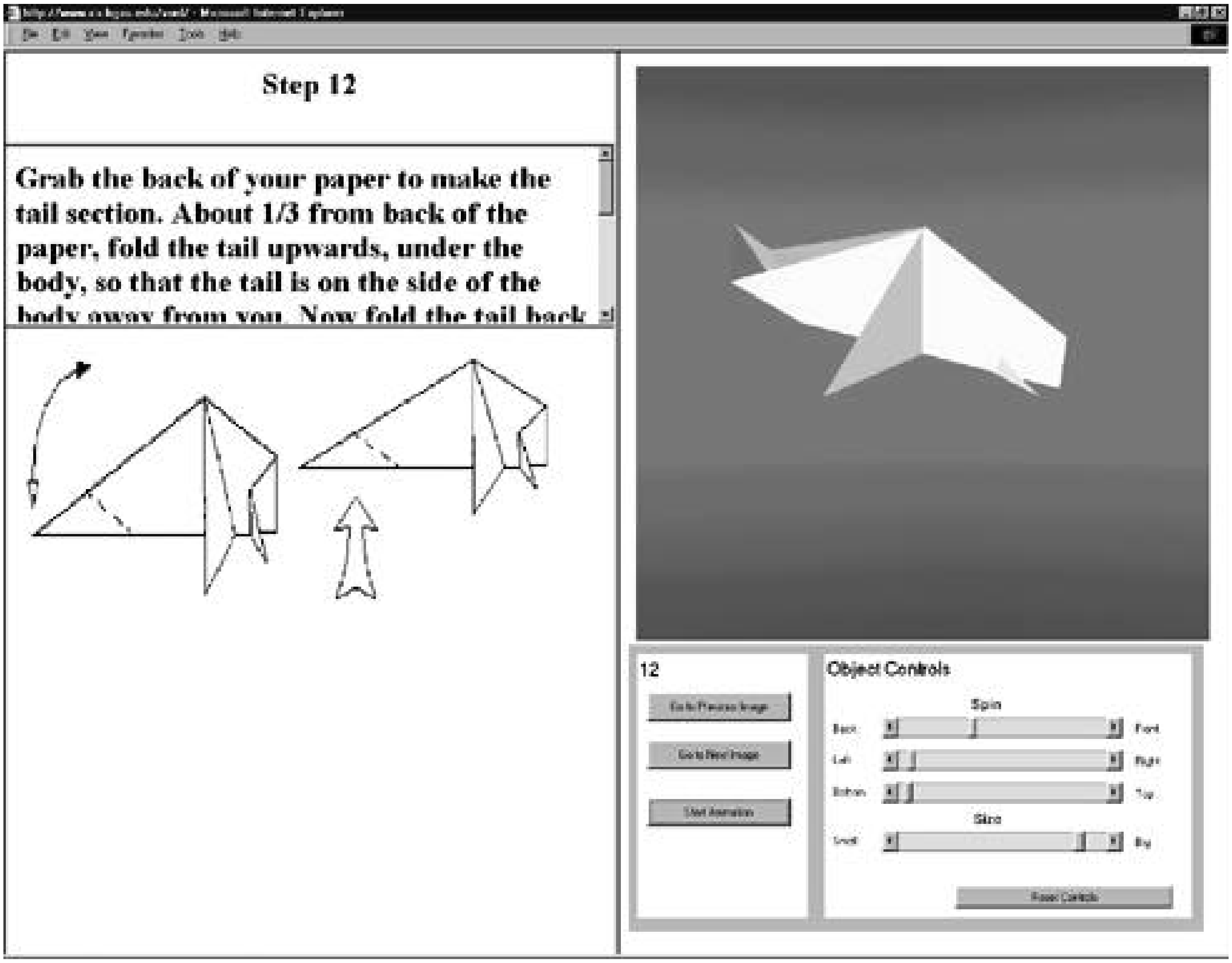

Figure 1.

The developer can create multiple layers for each scene, allowing the author to finely control the animation of different objects within a scene. Although a scene is composed of multiple layers and may suggest depth characteristics, its objects are still 2D. In order to show more than one perspective or orientation of an object, the developer must create several scenes. Flash models are not directly manipulable by the user in the sense that the user cannot choose an arbitrary perspective from which to view the object. An important feature of Flash is its ability to create interpolated or "tweened" animations. The developer may specify Shape Hints to guide the interpolation in shape tweening. The developer may specify a Path in a Layer Guide to assist the interpolation in a motion tween.

4. COMPARING AUTHORING TOOLS BY COMPARING DEVELOPER WORKLOADS: BUILDING A PRESENTATION IN COSMO™WORLDS VS. FLASH

In order to compare the two authoring environments, we defined a standard task and interface to construct within the environments. The benchmark task was to deliver the instructions required to fold an origami whale. The whale consisted of 12 distinct steps and 25 distinct folds. Steps 1 through 5 were folds or fold-unfold combinations on a flat piece of paper, and the remaining steps involved building the three-dimensional characteristics of the whale. Steps 1-3, 6 and 9 formed the body, Steps 4-6 and 10 formed the fins, Steps 7-8 and 11 formed the mouth, and Step 12 formed the tail.

We defined a standard interface for the overall presentation of our instructions. The left side of the presentation included (top to bottom) a label denoting the current step number, a scrollable text window, and a window containing a still picture. The right side of the presentation contained (top to bottom) a window for the 3D model (VRML) or 2D model with multiple perspectives (Flash) and controls for the user to manipulate the presentation. These controls included buttons to navigate through the different steps of the presentation and controls to manipulate the 3D model or Flash model. Both the VRML and Flash models could be animated.

The control interface for the VRML presentation, created in Cosmo™Code, contained three spin controls that permitted the user to rotate the model about the three standard axes. There was also a size control to scale the model. A start/stop button allowed the user to control the fold animation at each step. Additionally, users could return the 3D model to its original orientation and size for each step by using a reset button. Figure 1 shows the VRML presentation.

The Flash model included buttons so that the user could interact with the model or models. This interaction was limited to 1) starting and stopping the animation, 2) navigating between steps or scenes, and 3) changing viewpoints (selecting an alternate perspective of some scenes). The user could not directly manipulate the images of the paper model. The user could only replay the Flash animations that had been created. The individual images that we used in Flash were either imported GIF images or images that were created with the Flash drawing tool. For the inherently 3D steps of the construction, the Flash presentation included two different perspective views of the folds. Hence we were able to show an alternate perspectives of some of the steps. The user was still restricted to the viewpoints determined to be most meaningful by the author of the movie. Figure 2 shows the Flash presentation.

In terms of effort to author the presentations, Cosmo™Worlds has a complete set of features for generating objects, color and animations; as a tool for creating stand-alone VRML worlds it is excellent. In order to create some of the details of the VR model, such as foldlines, it was necessary to overlay texture maps onto the basic image. Overlaying the texture maps was possible within Cosmo™Worlds; however we were forced to create the texture images outside of Cosmo™Worlds and import and place them manually. This process was quite time-consuming.

Cosmo™Worlds was also limited in its ability to integrate the elements of the total presentation. We used a second tool, Cosmo™Code to develop Java code to manage a user interface and to allow the user to control aspects of the animation, because we required the functionality of a full-featured programming language for these cases. We were forced to use Java to create and to manage the separate elements of the user interface, such as the integrated text, picture and VRML components, because such a programming capability does not exist in Cosmo™Worlds. This deficiency in Cosmo™Worlds seems to be a limitation as an authoring tool. It is useful to note that other researchers have argued that combining VRML and Java can be effective in this way and thus would have also been limited by the use of Cosmo™Worlds. In [8], the authors discuss a CAD tool for the virtual assembly of furniture. Client/server applications using Java and VRML are discussed in [1].

By contrast, Flash was easy and fast to use to build the instruction set for the origami whale. The tweening feature reduced the number of images required to create animation. Color interpolation and shape tweening can give the effect of a paper fold. The step buttons permitted the user to step through our intermediate keyframes. As such, we were able to create our entire integrated presentation in Flash, rather than using a second tool, such as Java. The primary obstacle in building our Flash presentation was the 2D nature of the models. While individual models were themselves simple to build, building multiple orientations multiplied the workload. In fact, within our timeframe, we were only able to build multiple perspective presentations for 2 of the 12 whale folding steps. Finally we note that it took much less time to get up to speed using Flash than was the case with Cosmo™Worlds.

In summary, we conclude that Flash in its current form is the more complete authoring environment. Its primary limitation is simply that it is capable of building only 2D models. If 3D is required, the developer will be forced to build a number of 2D models in various orientations. By contrast, Cosmo™Worlds is an excellent tool for the development of stand-alone VRML. When the task is to generate an integrated VRML, Cosmo™Worlds is insufficient. The author will be forced to use other tools.

5. COMPARING AUTHORING TOOLS BY COMPARING THE EFFECTIVENESS OF THE PRESENTATIONS: COSMO™WORLDS VS. FLASH

In the previous section, we concluded that the Flash authoring tool is the more complete tool for developing our integrated presentations. It is likely to be more work to develop an integrated VRML presentation, but the Flash presentation will require a number of 2D presentations at each step to approach the 3D models that we were aiming for. In this section, we compare the effectiveness of the two presentations for building the origami whale. In particular, we are interested in determining if the more three-dimensional VRML models lead to better user performance and justify the additional work imposed in the use of the Cosmo™Worlds tool.

In our evaluation, users saw a presentation of instructions for folding the whale, in one of two treatments: (1) VR: containing text, still-images, and VRML and (2) FLASH: containing text, still-images, and Flash animations. There were 18 users for this study, with ten in the VR treatment and eight in the Flash treatment. Users were sophomores and juniors enrolled in computer science classes at Bowling Green State University. All were highly computer literate. The instructions were presented with a Silicon Graphics O2 computer with a

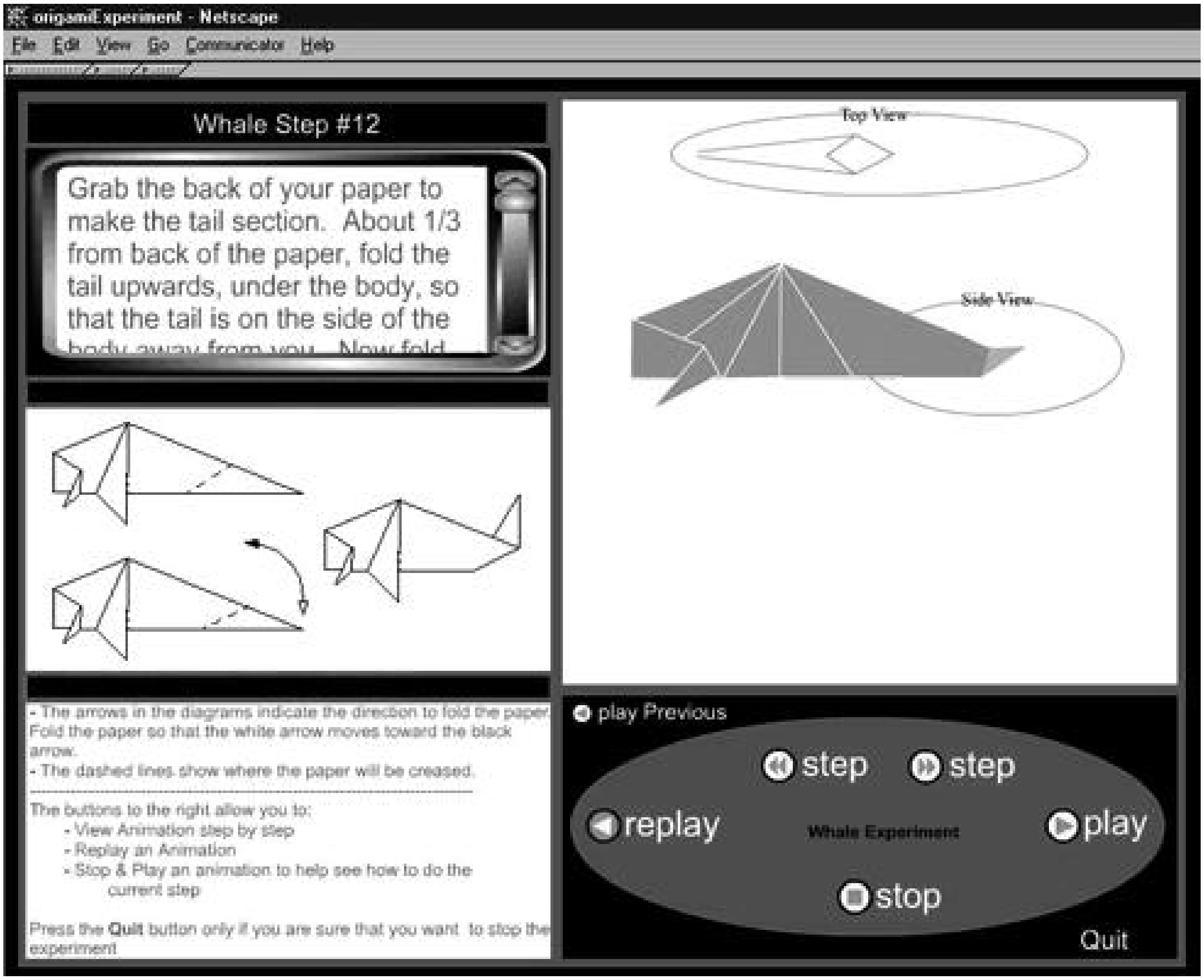

Figure 2.

17 inch monitor. Users viewed the presentations with *Netscape 4.0*.

All of the users received training to learn to use the tool set for their presentation. This training took about 8 minutes to complete, unless the users had questions. All users then received training in paper folding from computerized instructions. In this phase of the training, users folded a stylized paper airplane which had five steps and eight folds. The computerized instructions were presented in whichever of the treatments that the user was to receive. Users were told to fold the whale by following the presented instructions, using a special piece of origami paper. There were no time limits for how long users were given to fold the whale.

User performance was measured and assessed by the number of correct folds and the number of error folds in the whale. In order to assess the correctness of the whale folds, each existing fold was graded by three criteria: 1) Placement of the fold, 2) Direction of the fold, and 3) Size of the fold. In order to be a correct fold, the fold had to be correct on all three of these criteria. Recreased but correct folds were scored as correct folds, so it was possible to have more than one correct fold at a given point.

The users in both of the treatments did very well. The average percentage of correct folds was 82.4% of 25 possible folds. The average number of errors across both treatments was 5.9. There was no statistical difference by treatment for either of these dependent variables.

We conclude that users were successful in folding the whale, regardless of which presentation they saw. This result is consistent with our previous studies which indicate that presentations that include both visual and textual information are likely to be more useful than either visual or textual information alone. More importantly the results suggest that as the presentation with Flash was just as effective as the presentation with VRML, the additional developer overhead of using Cosmo™Worlds instead of Flash is simply not justified by any improvements in user performance.

6. SUMMARY AND CONCLUSIONS

In this paper, we raise the question of which authoring tool, Cosmo™Worlds or Flash, is most appropriate when developing presentations to deliver instructions for inherently-3D construction tasks. Delivery of this kind of instructions is a common problem and would lend itself to multimedia presentations on the World Wide Web. Our previous research has indicated that when delivering instructions for inherently-3D construction tasks, multiple media (visual and textual) are more effective than a single type of media. 3D models are one example of visual presentation that can be used effectively with text. Cosmo™Worlds supports the development of presentations with VRML models and Flash supports the development of presentations with multiple 2D models, which is a kind of simulated 3D model.

We find that as an authoring tool, Flash can be used to quickly develop a simulated 3D model, because it provides an integrated environment for developing for the Web. Also, Flash does not require the developer to create an actual 3D model to manipulate. By comparison, development of 3D models in VRML is much more difficult. While Cosmo™Worlds provides a VRML environment, it lacks features for integration as compared to Flash. In terms of the effectiveness of the presentation, we found that both presentations were equally effective and had a very high success rate for folding the origami whale.

Which tools should a developer select when delivering instructions for inherently-3D tasks? Based on our results we believe that for delivering instructions for simple construction tasks, Flash is the clear choice. Our results simply did not justify the extra work required by Cosmo™World as compared to Flash. Is this always the case for delivery of instructions for inherently-3D construction tasks? This question will be the subject of our future work.

7. ACKNOWLEDGEMENTS

We thank Keith Ebare, Lisa Weihl, Tom Stoltz and Brian Ameling for their assistance on this project

8. REFERENCES

[1] B. Jung and J. Milde, "An Open Virtual Environment for Autonomous Agents Using VRML and Java", The Fourth International Conference on the Virtual Reality Modeling, Language and Web 3D Technologies, February 1999, http://www.clab.de/vrml99/home.html.

[2] B. Teasley, K. Instone, L. Leventhal and E. Brown. Effective illustrations in interactive media: What works? Human-Computer Interaction: INTERACT '97, S. Howard, J. Hammond & G Lindgaard (Eds.)

[3] NewYork: Chapman and Hall, July, 1997, 197-204.

[4] C. Marrin and J. Couch, Specification of the VRML/EAI Interface. The web 3D Consortium, Online resource, January 1999, http://www.web3d.org/WorkingGroups/vrml-eai/.

[5] Cassady & Greene, Origami: The Secret Life of Paper, 1996, ISBN 1-56482-104-8.

[6] G. Zimmerman, J. Barnes, L. Leventhal, Building User-Controlled 3D Models and Animations for Inherently-3D Construction Tasks: Which Tool, Which Representation? Computer Science Technical Report, Bowling Green State University, CS01-Sept00, September 2000

[7] I. Honda, The World of Origami, Japan Publications Trading Co., Tokyo, 1965.

[8] L. Leventhal, J. Barnes, G. Zimmerman, and L.Weihl, Multiple Web Representations of Instructions on a Procedural Task: An Empirical Study, Computer Science Technical Report, Bowling Green State University, CS01-Sept99, September 1999.

[9] M. Nousch and B. Jung, CAD on the World Wide Web: Virtual Assembly of Furniture with BEAVER. The Fourth International Conference on the Virtual Reality Modeling, Language and Web 3D Technologies, February 1999, http://www.clab.de/vrml99/home.html.

[10] Macromedia Flash 4: Using Flash, Macromedia, Inc., San Francisco, 1999.

[11] World Up User's Guide, Sense8 Corporation, Mill Valley, Ca. 1997.