The Visual Program Project

Paul Juell
Department of Computer Science
North Dakota State University
juell@plains.nodak.edu

ABSTRACT

The Visual Program project is building visualizations to help students understand AI programs and techniques. These are Web based, providing easy access. The images are 3D interactive images. They are delivered in VRML with supporting Java programs. The images are course content visualizations that allow the student to explore information. The core task of these visualizations is to present a description of process. Perhaps the hardest part of explaining a process is to show changes over time. We are using three main techniques to show the effects of time. These are: movies, false coloring and representing time as a spatial dimension. One example of each of these are: movies of searches, false coloring time in a GA and representing time pictorially in a state space search. A basic search tree is used in the movies. The search point is depicted as a sphere moving from node to node of the search tree. A Genetic Algorithm (GA) search has a random property throughout its process. To help understand a GA, we use false coloring to show when during the search a point was inspected. The point searched is shown in 3 space by a small sphere. The color of the spheres ranges from red for early searches to blue for late searches. This image can help gaining an intuitive understanding of a complex search process. The image gives a perspective on the nature of the search and how the computation time is spent in the search, both of which can be missed in lecture presentations. This is presented as a navigable 3D world. This allows the student to work through the data in his or her own way. Time can also be shown as a spatial dimension. For state space searches, the states are pictorially displayed. Subsequent steps in time are shown in the same image further back into the image. This allows the student to see the developing pattern of the search by seeing all the state transitions in one image.

INTRODUCTION

The Visual Program project is building visualizations to help students understand AI programs and techniques. These are Web based, providing easy access and interactive 3D images. The images are in VRML with supporting Java programs. The images are course content visualizations that allow the student to explore information. The core task of these visualizations is to present a description of process. Perhaps the hardest part of process to show is time. We are using three main techniques to show the passage of time. These are: movies, false coloring and representing time as a spatial dimension.

The Visual Program project is part of NDSU's WWWIC (World Wide Web Instructional Committee) effort. WWWIC is funded primarily by NSF, but it also has received local and other funds. WWWIC [4] [2] is sponsoring various projects to develop technology based educational material for several academic disciplines. These include: The Virtual Cell for biology, The Geology Explorer [5] [6] [8] for geo-sciences and BlackWood for sociology. There are several

common themes for these projects, the primary one to to find ways to involve the student in the education process. All of the projects are delivering material by Web based interfaces. This material includes new modes of iteration with the course material and new modes of presenting the information.

THE PROBLEM DOMAIN - UNDERSTANDING AI PROCESSES

An AI course is a good problem domain for building educational visualizations. An AI course is a very rich domain in which numerous algorithms, techniques, representations and processes to solve problems are presented. Many of these items are complex and require an effort to explain. Even if any one of these items can be simply explained, the process of taking a student through a number of these can overrun the student's ability to cope with diverse information. Much of the material is abstract and needs clear visualization to aid the student in understanding the concepts.

This need for the visualization can be seen in the way many of the techniques are explained. For a number of the items, such as blocks world, semantic networks, neural networks the concept is tightly coupled to a particular image. Although, sematic networks can be explained as sets of FOPC rules, the normal presentation is a diagram. The various parts of the discipline bring up sets of of images. As powerful as these images are, they do not normally address the changes over time, or the bases of process. One way a book or instructor will try to explain process is to present a sequence of images, to show the steps. If a single image is used for the process, the sequence of steps are typically overlayed onto the single image. For example the nodes searched in a tree can be numbered to specify at which time each node is inspected. While these notations may be clear, there is value in being able to show the sequence of operations over time. The instructor can use this as way to bring home the points and the process. More important, the student can then see the correct sequence when working on his or her own. It can be hard to correctly record the sequence of a chauk talk.

Technology can help present images to the student that have some of the properties of the classroom chawk talk lecture. We have chosen to use VRML was a way to show the AI techniques and in particularly the changes to over time.

The Web, VRML and Java

Some of the techniques that Multimedia provides tools and techniques to present motion, sound, 3D images and interactive environments. These all can be used to provide additions to the static images found on the page of book. Our current preferred mode of delivery of information is the Web. Using the Web permits easy use of the material in instrumented classrooms and by the student out of class. We chose to use VRML because it provided Web access to all the above modes of information. VRML is a well supported standard [1]. In VRML you describe a what is called a world. These worlds are three dimensional and can be displayed within a Web browser using a specialized plug-in. For MS Windows, there is a plug-in called Cosmo. The VRML plug-ins allow navigating the 3D world using the mouse. The VRML world can have motions programmed into the objects. Sounds can be attached to the objects as can detailed images. One other important property is the ability to connect Java code to the objects in the world. The VRML world is described by a straight forward ASCII notation. This collection of features allows easily building complex worlds with rich behaviors. These worlds and the supporting programs can then be accessed by common Web browsers. Basically, VRML can be used to

build almost anything from simple images to immersive interactive virtual realities.

VISUALIZATIONS PROCESS 3D and TIME

A significant amount of work has been done in the area of visualization. However, many of the visualization tools are a designed for continuous processes and continuously varying variables. Two examples are Vis5D, developed to display weather data and VTK [7], a sophisticated tool kit with tools to trace flows in fluids and surface contours. Decrete values and decrete processes can be displayed, but this is not the primary aim of these tools. When you look at visualization for programs, you find tools that are designed to display decrete variables and processes [10] [3]. However, much of this work is designed to help debug a program and its logic. This is useful, however, the teaching tools we want are to address the logical and conceptual level to aid in understanding the material. This has led us to develop our own visualization. The visualization techniques we use are standard techniques, such as animation and false coloring. However, the emphases is on the what we consider the important AI concepts in teaching the material.

Using Movies

A simple way to show operations of a process over time is to present a movie. We did this for two of our visualizations. For one we used movement capabilities directly supported by VRML and for the other we built a Java program modeling the problem. A natural match with movies is the process of explaining searching techniques. At some point the instructor will simply walk through the steps and show the nodes inspected. We wanted to produce a image that had these properties and that could be used for lecture or independently by the student. For the straight forward tree walks, (depth first and breadth first) we developed a movie showing the path of inspected nodes. These were implemented in VRML. Figure 1 shows the depth first search example. There is a clickable node which starts the movie. The movie shows a red ball tracing the specified search path. This was implemented in VRML, not because we wanted the 3D support, but because VRML had support to describe movements of objects.
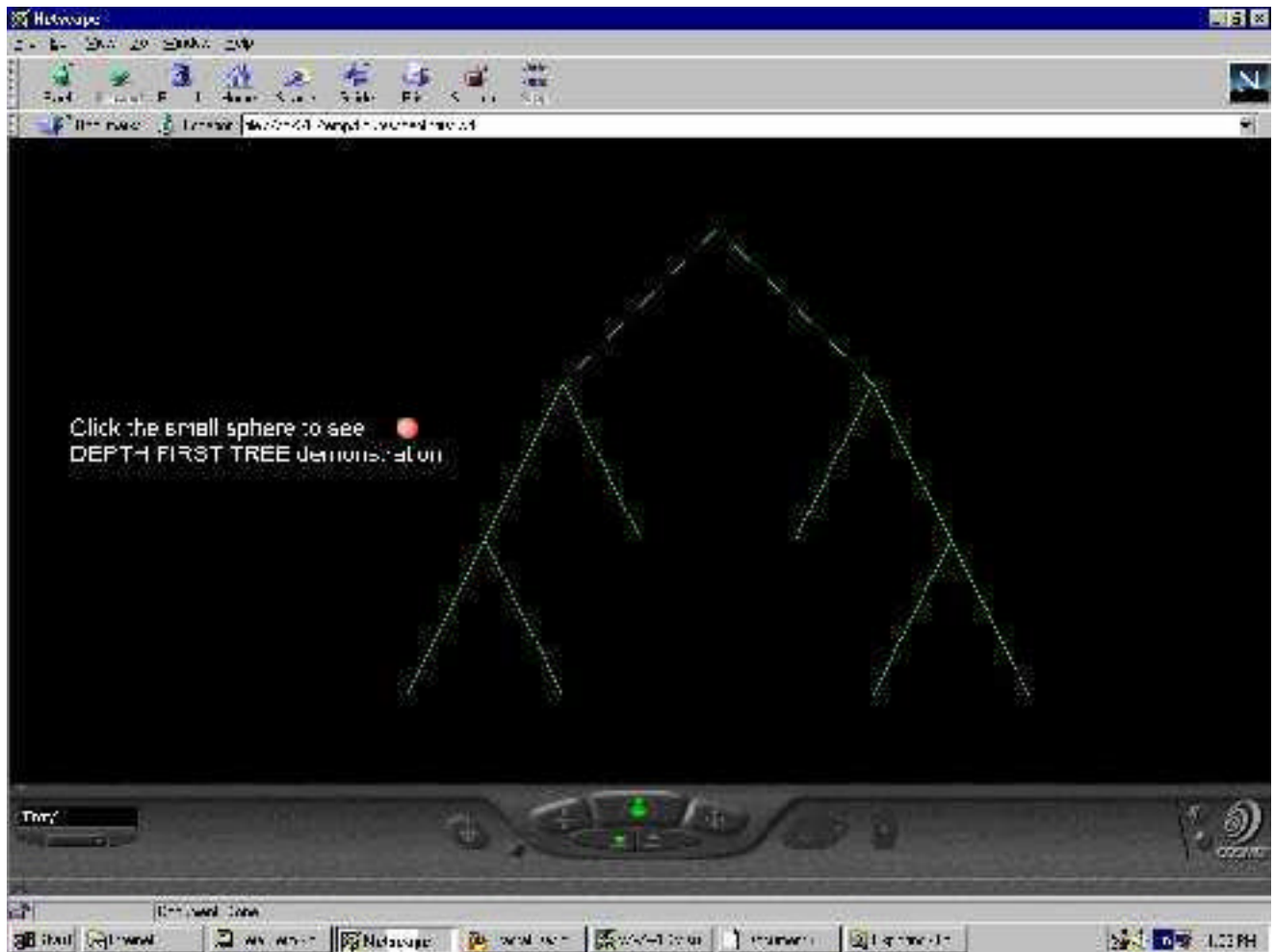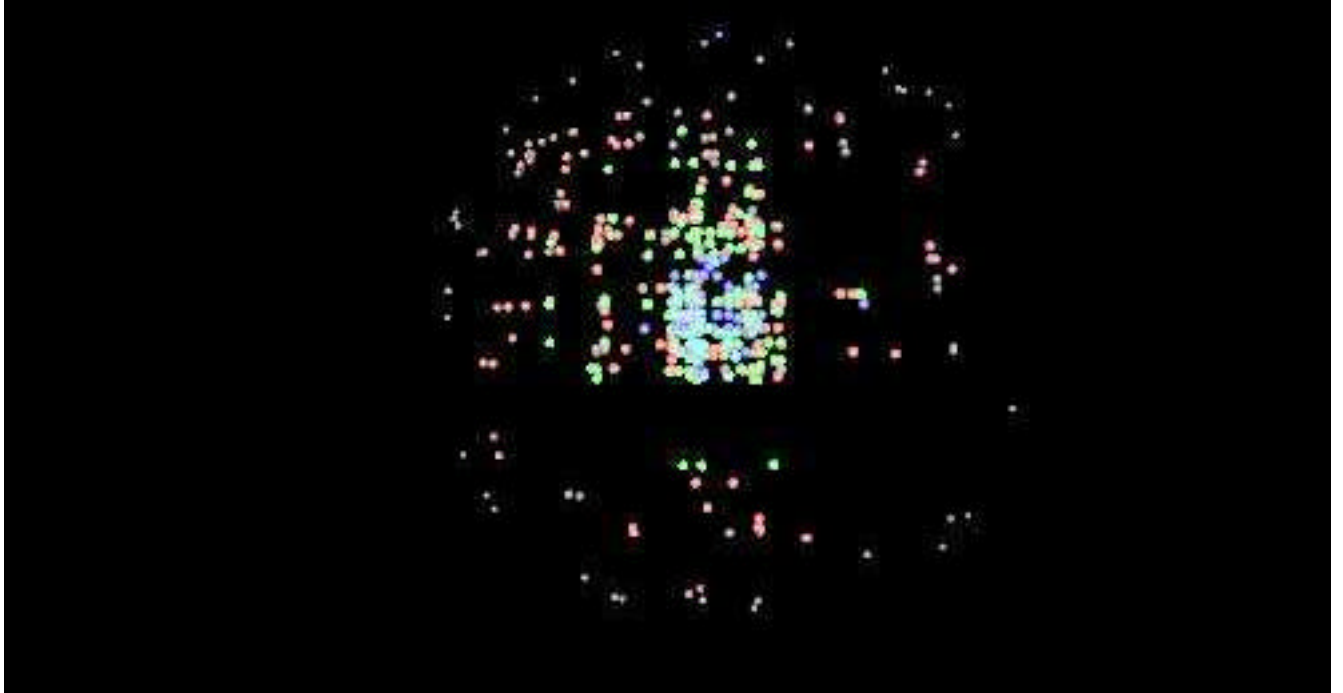
Figure 1: depth first search movie example

A more sophisticated program was developed to demonstrate the hill climbing search. Hill climbing is search that checks the local neighborhood to find the largest slope and then steps that direction. The process is then repeated at that new point. We developed a Java program and an image to present the search. The search path is displayed over the image. See Figure 2. The image is a version of the Grand Canyon. This is hand converted into a 12 by 12 grid of elevation values. When a person selects the search the search starts and displays the points searched and the path taken. As the search proceeds, a colored line shows when the space is sampled. This in shown stepping around the current point. Once the sampling has been preformed, the path is extended in the best direction and the process continues. The starting button specifies a starting point and a real hill climbing routine in the Java applet drives the display. In order to get the desired behavior, the elevation grid values and the starting points are carefully set to get the desired type of path. This then needs to look reasonable as an overly to the image.

FIgure 2: hill climbing movie example

OVERLAYING INFORMATION

A common visualization technique is false coloring. This is a way to add addition dimensions of information to an image. The additional data is indicated by the color used to display the regions or objects in the image. This could be used to show temperature on a map by showing cold areas as blue and hot areas as red. We chose false coloring as one of the ways to show the operation of a Genetic Algorithm (GA). The GA uses populations in the search and is driven by a directed random search. We felt we could help the students understand the process better if we could show the full search process in a picture. Both the number of points inspected and the order of the operations are important and different from "traditional" search techniques. We wanted to make these concepts clear in the visualization. Both of these characterization can be lost in looking at tables of numbers from the routines. One of the standard test and demonstration programs for GA is a 3 dimensional parabola, with the GA trying to find the lowest point. See figure 3. We choice to display all of the points searched, each as a sphere. Since we also wanted to see the order of search, the spheres were false colored with the time of the search. The early searches were colored red, the next searches were colored green and late searches were colored blue. Despite extensive work with GA's this author found the volume and patterns of search surprising and gave incites into the technique.

STATE SPACES AND TIME

An important AI concept is the state space search. The states are describable configurations and the search goes though these states trying to find an answer. The concept is simple, however, the actual behavior of the search can be lost in the volume of states searched. We wanted to let the student see the full process of these searches. We developed two versions to display these search, one for the blocks world and one for crossing type problems. The blocks world is a common AI example. There is a stack of blocks that are re-arranged by a robot arm. One problem is that it that the student often can not see am image of the state, but just the an ASCII description. We provided tools to convert a student's version of the block's world into an image. One measure of success of these tools, was that students discovered bugs in the starting program that other students had not previously discovered. When a block was placed on the table its properties were not correctly set. In effect, the blocks was not connected to the table or the rest of the objects. The blocks world has a natural translation into an image. That is simply convert the blocks in the program into VRML blocks in 3 space and display them. This was done by instrumenting the student's program and getting the program's description of the model world. This was then converted into VRML and displayed.

Another example of a state space search is the set of crossing problems. One of these is: there is a father, two sons and a boat. The boat can take 200 pounds. The father weights 200 pounds. Each of the sons weigh 100 pounds. How can all three get across the river?

This problem is interesting because it take several steps and some of the work has to be undone to get a solution. To allow the students to see the full process we provided a visualization with all of the states of the search in one image. Basically we converted time into the z dimension and the

state changes were then linearly presented. Each of the individuals in a state was presented as a tall box and a low box was used for the boat. As new states are presented, they are put further back in the image. Then the sequence of operations can be seen from the start (at the front) to the end (the furthest back in the image). See Figure 4. In addition, the starting side states were colored reb, the crossing group blue and the final side states green. This presents a colorful and dense image. The student can explore the image using the plug-in and can following the wrong steps taken and the development of the solution.

The student had to develop a larger scale crossing problem, the missionaries and cannibals problem. They instruments their program and produced the full search visualizations.
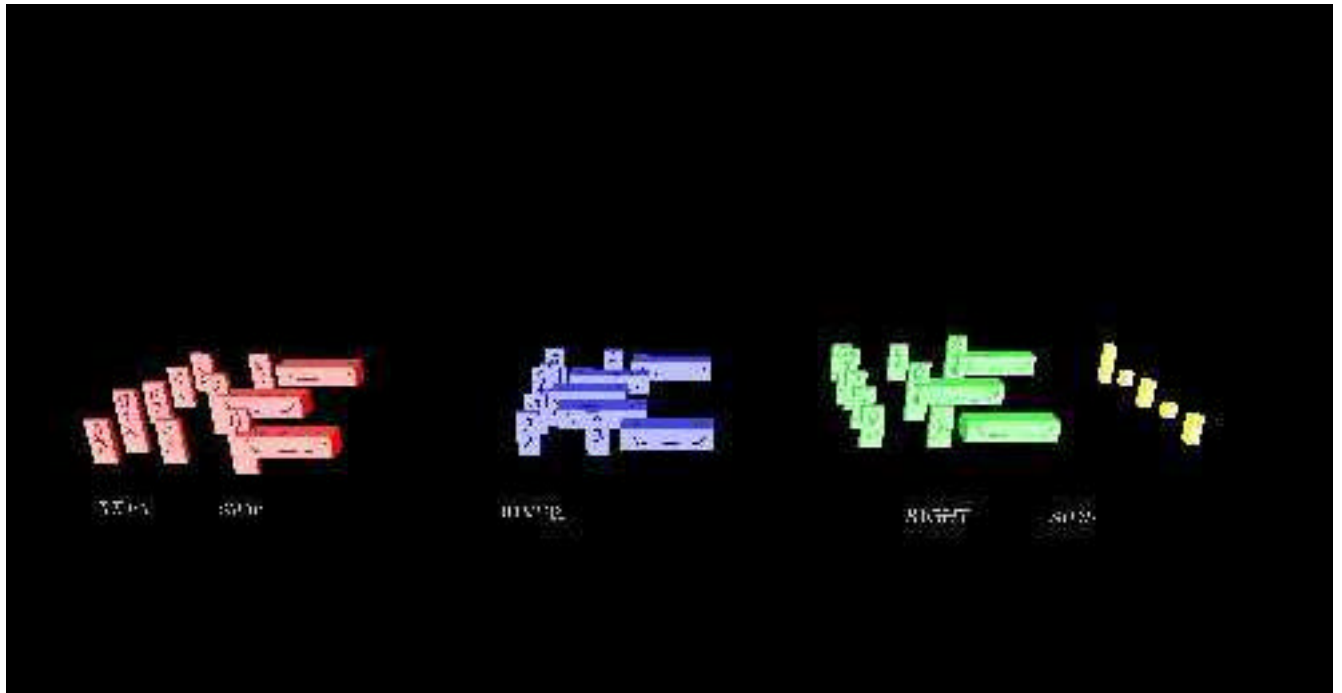
Figure 4: river crossing, time as dimension example

The Current State of the VP Project

The visualizations and visualization tools presented are at a beta level. They have been developed, and seem to be of value. They have been used in one class. A pretest was administered at the start of the class. Post tests on concepts and responses to the visualization are being administered some time after their use. We have yet to evaluate these results. We hope to then improve the tools and the visualizations.

CONCLUSION

We have presented some of the results of an ongoing project to develop visualizations for an AI class. We have been very happy with the images and think they are contributing to the student's learning and overall understanding of the concepts presented.

Literature References

1. Hartman, Jed and Josie Wernecke (1996). The VRML 2.0 Handbook: Building Moving Worlds on the Web. Reading, MA: Addison-Wesley Publishing Co. Silicon Graphics, Inc.

2. Hill, Curt and Brian M. Slator (1998) Virtual Lecture, Virtual Laboratory, or Virtual Lesson. In the Proceedings of the Small College Computing Symposium (SCCS98). Fargo-Moorhead, April. pp. 159-173.

3. Hoebel, Louis, William Lorensen and Ken Martin, Integrating Graphics and Abstract Data to Visualize Temporal Constraints, SIGART Bulletin, Winter 1998, Vol 9, No. 3&4, pp18-23.

4. McClean, Phillip, Donald P. Schwert, Paul Juell, Bernie Saini-Eidukat, Brian M. Slator, and Alan R. White (1999). Cooperative Development of Visually-Oriented, Problem-Solving Science Courseware. Proceedings of the International Conference on Mathematics/Science Education and Technology (M/SET-99). Association for the Advancement of Computing in Education, San Antonio, March 1-4.

5. Saini-Eidukat, Bernhardt, Don Schwert and Brian M. Slator (1998). Text-Based Implementation of the Geology Explorer, a Multi-User Role-Playing Virtual World to Enhance Learning of Geological Problem-Solving. GSA Abstracts with Programs, Vol. 30, No. 7, October, 29. Toronto.

6. Slator, Brian M., D. Schwert, B. Saini-Eidukat, P. McClean, J. Abel, J. Bauer, B. Gietzen, N. Green, T. Kavli, L. Koehntop, B. Marthi, V. Nagareddy, A. Olson, Y. Jia, K. Peravali, D. Turany, B. Vender, J. Walsh (1998). Planet Oit: a Virtual Environment and Educational Role-playing Game to Teach the Geosciences. In the Proceedings of the Small College Computing Symposium (SCCS98). Fargo-Moorhead, April. pp. 378-392.

7. Schroeder, Will, Ken Martin and Bill Lorensen, The Visualization Toolkit: An Object-oriented Approach to 3D Graphics, Prentice Hall PTR, Upper Saddle River, New Jersey 07458, 1996.

8. Schwert, Donald P., Brian M. Slator, and Bernhardt Saini-Eidukat (1999). A Virtual World for Earth Science Education in Secondary and Post-secondary Environments: the Geology Explorer.Proceedings of the International Conference on Mathematics/Science Education and Technology (M/SET-99). Association for the Advancement of Computing in Education, San Antonio, March 1-4.

9. Smith, Sean M. and Glen N. Williams, A Visualization of Music, Proceeding, Visualization'97, October19-24, 1997, Phoenix, Arizona, IEEE Computer Society Technical Committee on Computer Graphics, in cooperation with ACM SIGGRAOH, pp499-503.

10. Stasko, John, John Domingue, Marc H. Brown and Blaine A. Price, Software Visualization: Programming as a Multimedia Experience, The MIT Press, Cambridge, Mass, 1998.

Acknowledgments