An Intelligent Tutoring System for Remote Sensing and Image Interpretation

Aaron M. Bell
Stottler Henke Associates, Inc.
San Mateo, California
bell@stottlerhenke.com

Sowmya Ramachandran
Stottler Henke Associates, Inc.
San Mateo, California
sowmya@stottlerhenke.com

ABSTRACT

Remote sensing technology is playing an increasingly central role in military operations. The interpretation of satellite and aircraft photography has great utility in allowing forces to assess a situation from a secure, stealthy distance. However, performing effective interpretations of remote sensing photography is a complex science. Knowledge of a complex, highly-visual domain, such as that of remote sensing, cannot be learned from manuals. For an analyst to become proficient requires a large investment of training time from an expert.

Our paper describes an Intelligent Tutoring System (ITS) being built for NASA. The goal of this software is to provide the benefits of a one-on-one remote sensing instructor to teach Earth Science principles. The ITS integrates directly with the NASA Image2000 image processing software, the actual end environment in which researchers perform real-life tasks. Together, NASA Image2000 and the ITS provide students with a dynamic, hands-on educational experience that is cost-effective. It monitors and assesses the student's actions, provides immediate feedback as well as contextual guidance and challenges. A separate authoring tool allows new tutorials for the ITS to be developed by non-programmers. Thus, instructors can readily produce interactive, adaptive tutorials for an area of remote sensing interest. With regard to military application, tutorials can be constructed to teach the wide array of interpretation tasks: image enhancement, object classification, target discrimination, damage assessment, weather prediction, etc. The tutorials can focus on a specific geographical location.

This paper outlines our research of a remote sensing ITS system, its general framework, the artificial intelligence technologies it employs to model the knowledge of the student and the expert, and its application to the Earth Science domain.

ABOUT THE AUTHORS

Aaron M. Bell has a strong background in software development and has a devoted interest to applying artificial intelligence technologies to Earth Science problems. At Stottler Henke, he is currently project manager for the EarthTutor project, an Intelligent Tutoring System to teach the interpretation of satellite imagery. EarthTutor is being developed under the SBIR program for NASA Ames Space Center. Aaron is also an A.I. Software Engineer for the Aurora project, the scheduling engine for the space shuttle and International Space Station ground processing facilities at NASA's Kennedy Space Center. Before working at Stottler Henke, Aaron was a software development engineer at Microsoft Corp. for six years. Aaron holds a Bachelor's Degree in Computer Science from Stanford University in 2000.

Sowmya Ramachandran has a strong background in a wide variety of artificial intelligence techniques, including Intelligent Tutoring Systems and Machine Learning. She has extensive experience in developing educational software. Dr. Ramachandran has been involved in developing several Intelligent Tutoring Systems, including one for adult literacy enhancement and one to teach Algebra to at-risk high-school students. She is also currently developing a general-purpose authoring framework for rapid development of ITSes. Dr. Ramachandran earned a Ph.D. from the University of Texas at Austin in 1998.

An Intelligent Tutoring System for Remote Sensing and Image Interpretation

Aaron M. Bell
Stottler Henke Associates, Inc.
San Mateo, California
bell@stottlerhenke.com

Sowmya Ramachandran
Stottler Henke Associates, Inc.
San Mateo, California
sowmya@stottlerhenke.com

INTRODUCTION

As satellite and aerial photography has become a widely available resource, scientists and planners have been quick to recognize the value of the technology and involve it in their research. Researchers benefit from the powerful software tools used to acquire, process and manipulate images; however, to use these tools successfully, one must make meaningful interpretations of the image data. Unfortunately, remote sensing imagery is notoriously challenging to work with; it often contains misleading cues, erroneous data, occluded features, and false positives among other challenges. Effective interpretation requires a considerable amount of training. First, the student must become adept with a processing tool; second, the student must develop the understanding and insight to properly understand nuanced imagery.

The goal of this paper is to describe applied artificial intelligence (AI) methodologies as potential remote sensing learning aids. Currently, we are developing an Intelligent Tutoring System (ITS) called EarthTutor to teach Earth Science principles using satellite imagery for NASA Ames Research Center. Ultimately, we hope the system will demonstrate that an ITS can be useful in addressing the challenges presented by remote sensing education.

REMOTE SENSING TRAINING CHALLENGES

As the image processing software available to researchers becomes more advanced, the related educational innovation has lagged behind. Current training materials built for image processing software do not take advantage of the full potential of the computer context to train, immerse, and adapt to different students. Paper and web-based materials are limited by their medium of delivery. These approaches are simply not an effective teaching device in the highly-visual domain of remote sensing. On the typed page, image processing instructions, let alone typical computer operation instructions, are confusing and tedious to follow. Printed materials can't display live overlays and demonstrative animations essential to the domain. Moreover, printed materials lack interactivity. They are incapable of providing feedback when a student does something wrong or providing encouragement when a student does something right. Likewise, there is no way to adapt to the student's needs, provide extra guidance when a student is struggling, or pinpoint a student's trouble areas.

Typically, it is considered the teacher's role to provide the guidance the printed material cannot. An experienced teacher is on hand to lecture, answer questions, and provide assistance for all students. The teacher must share all his/her time among all students. However, in a large class, it is impossible to present a course that is customized to individual differences, with the result that some students understand and master skills quickly, while others don't acquire enough proficiency in the given amount of time. Depending on their experience and their learning speed, some students may need fewer hours of study than others. Moreover, in a domain with so many concepts new to a student (e.g., in remote sensing: spectral bands, histograms, lookup tables, etc.), different students need additional training in different topics. This setting is inefficient with respect to the student's time, because it cannot accommodate such differences. The only effective way to achieve such customized instruction is through one-on-one instruction, considered the gold standard in education. However, because of financial and human resource limitations, one-on-one instruction is typically impossible.

APPLICATION OF ITS TO REMOTE SENSING

Emerging artificial intelligence technology allows for novel software-based tutorials that accommodate highly-visual domains, like that of remote sensing, and offer enhanced multimedia, interactivity and adaptive capabilities. An Intelligent Tutoring System (ITS) is a software program that aims to offer the benefits of one-on-one instruction automatically and cost effectively. Like training simulations, ITS enables students to practice their skills by performing tasks within highly-interactive environments. However, ITS goes beyond training simulations by tailoring instruction and providing individualized guidance. Unlike other computer-based training technologies, ITS systems closely monitor students as they work in interactive environments and develop a model, called the student model, of each student's knowledge, skills, and expertise. Based on this model, ITSes determine instructional strategies, in terms of both the content and style, and provide explanations, hints, examples, demonstrations, and practice scenarios as needed.

EARTHTUTOR

Under an Advanced Educational Processes grant, NASA has made possible the research of an ITS system, EarthTutor, to train college students on remote sensing, image processing and Earth Science principles. This system is currently under development. The host for the learning experience is the NASA Image2000 (NI2k) application, a freely-available, extensible image processing software program developed for classroom computer labs. EarthTutor runs within and shares the same parent window as NI2k such that the student is immersed within same the program with which he/she will eventually be doing independent scientific study. From the student's viewpoint, EarthTutor is seamlessly integrated. However, the core of the ITS is generalized such that it can be plugged into the other image processing software and applied to other domains.

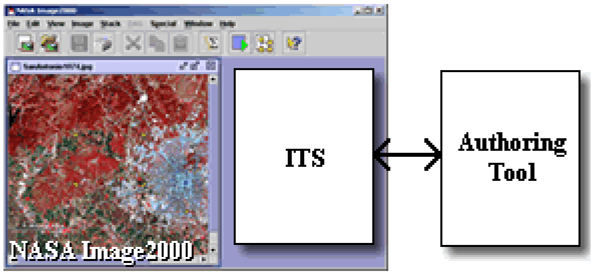


Figure 1. Host Integration

In addition to the ITS component of EarthTutor, a separate Authoring Tool allows an educator to construct new tutorials and scenarios that can be imported into the ITS system. The ITS and Authoring Tool communicate through sets of XML files.

How It Works

From within NI2k, the student elects to begin a tutorial which matches his/her desired area of study. The final version of EarthTutor will be packaged with a core set of Earth Science tutorials (e.g., geology, vegetation, urbanization, and oceans). Each tutorial is divided into chapters, which in turn are divided into interactive pages which guide the user through program tasks. Some chapters may be designated as prerequisites to other chapters. Initial lessons provide the student with introductory instruction. The lessons use artificial intelligence technologies based primarily on state machines (described later in detail) to monitor and make sense of the student's actions as he/she use the program. Initially, the tutor is highly visible, giving step-by-step guidance, prompting frequently with questions, providing hints, and correcting the student if he/she makes mistakes. At different stages, EarthTutor employs scenario-based reasoning (SBR) to provide the student with pre-constructed scenarios that match the student's learning deficiencies. In later lessons, EarthTutor becomes less visible and provides open challenges without detailed procedural instruction, which allows the student to work with more independence. However, the tutor continues mapping the student's actions to state machines so the ITS can provide context-based guidance if the student requests assistance.

SYSTEM ARCHITECTURE

Figure 2 below shows the high-level architecture of the proposed system. The components above the bold, dashed line form the ITS, which is responsible for providing instruction, while the components below the line constitute the content and objects related to the authoring environment.

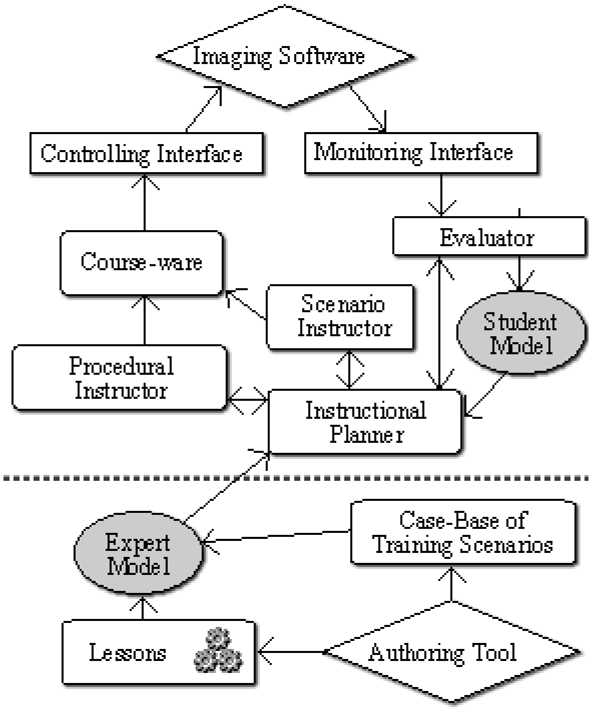
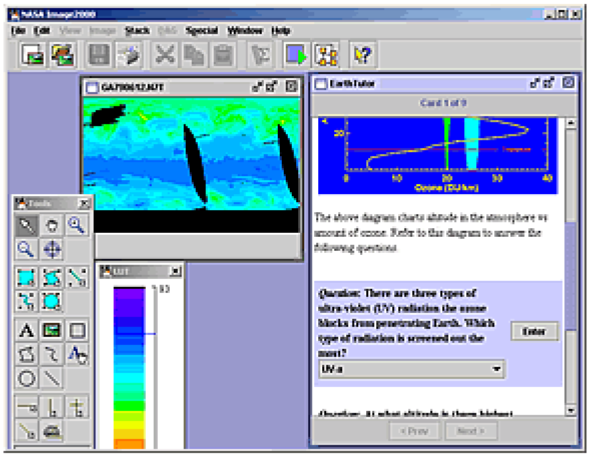


Figure 2. ITS Architecture

Controlling Interface

The Controlling Interface (CI) represents a set of classes that integrate directly into NASA Image2000's user interface. In figure 2, we denote the CI's set of instructional capabilities as "course-ware." The primary course-ware component is a panel that opens within NI2k (typically docked along the right-side of the main window) to display instructions, provide audio-visual guidance and present challenges to the student. A screenshot of the panel inside NI2k is shown in Figure 3 below.



The above diagram charts altitude in the atmosphere vs amount of ozone. Refer to this diagram to answer the following questions.

Question: There are three types of ultra-violet (UV) radiation the ozone blocks from penetrating Earth. Which type of radiation is screened out the most?

Question: At what altitude is there highest

Figure 3. Instructional Panel inside NI2k.

The panel always displays the current interactive page within the current lesson. This page contains instructional material and course-ware interspersed with numbered subtasks directed to the student to achieve a specific goal. An inexperienced student will be shown highly-detailed subtasks, while a more experienced student will be presented with more general material. The current task is highlighted. Through this panel, EarthTutor may also ask the student to answer fill-in-the-blank questions, multiple choice questions, or answer visually-oriented questions by selecting or clicking a "correct" region of the image being analyzed. The format for the panel's content is standard HTML. Thus, an author can provide links that open a web browser to a location at a public image archive or provide background information.

The CI is also responsible for demonstrating complex tasks. Much of EarthTutor's presentation and interaction occurs outside the panel described above, in the main imaging windows and toolbar palettes of NI2k. The ITS can move, click, drag the mouse and interact with windows and menus. The CI can also overlay animated graphics on the image being edited and display descriptive pop-up hint bubbles next to fields, buttons and image locations.

Monitoring Interface

The Monitoring Interface (MI) represents a set of classes that hook into the Image2000 core to make EarthTutor aware of actions the student performs. For example, when a student changes the selected tool, the system fires an event packaged with the relevant information (e.g., "user selected TextAnnotation tool") which is passed by the MI to all interested EarthTutor components, indicating that a new action has taken place. This information is then used to track the state of the program in the Evaluator, provide context to the Instructional Planner and update the Student Model.

The Evaluator

The Evaluator determines the progress of a student through a tutorial based on state machines, an AI method of representing behavior. Conceptually speaking, a state machine can thought of as a flow chart. More formally, it is a network of states connected by transitions; a user can follow a sequence from state A to state Z by satisfying all the conditions in between. EarthTutor's state machines are created using the Authoring Tool (which is described in detail below.) The monitor events described above trigger movement through the state machines. For example, assume the student is being asked to find the maximum Dobson Units (the most basic measure used in Ozone research) value for twelve images representing twelve months of Ozone readings. Within NASA Image2000, there are two ways the student could achieve this task:

> (a) the student could open up the twelve images as a stack, measure the maximum value for each image individually, then determine the largest value in his/her head, and record it; or
>
> (B) the student could perform the max stack operation on all twelve images, then perform a max on the resulting merged image.

An author would represent these different routes to achieve the task using states diagramed in Figure 4. As the student performs the different tasks, the Evaluator follows him/her through the states. As each condition is satisfied (e.g. "did the student add open the images as a stack?"), the student arrives at the goal. In this case, no matter if the left-side path or right-side path was followed, the student will ultimately be prompted to enter the final max value by entering it into a textbox or being asked to highlight (using the pinpoint tool) the pixel on the image where that max value occurs. These tasks would also be represented using a state machine.

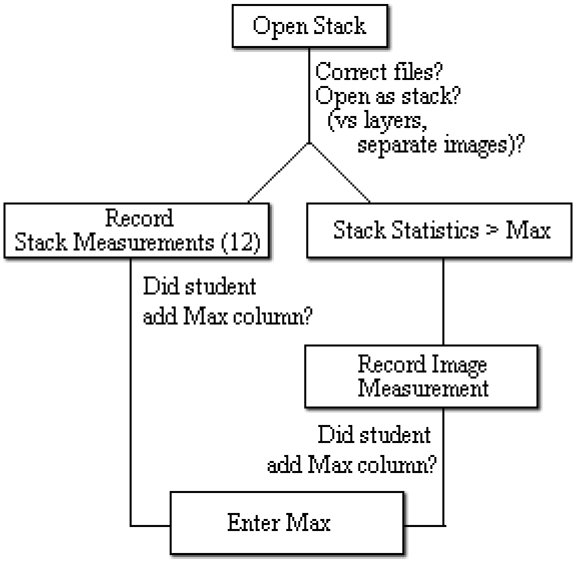


Figure 4. Entering max DU value for 12 mo. Ozone

The state machines and other lesson plan information are stored as knowledge in the Expert Model. This model is represented on disk as a set of XML files.

In addition to tracking the progress of a student through a tutorial, the Evaluator is also responsible for tracking the state of the images being edited. If a student crops or zooms in on a picture, the Evaluator must perform the equivalent translation and scaling operations respectively on its internal representation of the image so that it remains synchronized with the monitored student. This representation is a simple data structure that maintains basic properties (such as origin, scale, brightness, contrast, colors, etc.) for an image.

Student Model

Once the student completes a task, his/her new ability is updated in the Student Model. The Student Model represents knowledge using a hierarchical structure of student skills. At the most granular level, the model records the student's proficiency in demonstrative subtasks (e.g. can the student add measurement columns?). The proficiency in the subtasks propagates upwards to a calculated proficiency in a parent task (e.g. can the student record the max value for a stack?) The propagation routine follows the behavior of a standard Bayesian network. In other words, an author can specify the parameter probabilities that imply the level of understanding for a parent task based on the level of understanding for a subtask.

The Student Model hierarchy is built using the Authoring Tool described later. Skills that make up the Student Model are either global or tutorial-specific. Some skills are universal to almost all image processing activities (such as a working with the lookup table), and are part of the core skills defined and available in EarthTutor. Other specific skills (such as tropical vegetation dynamics) can be added as needed by an author. We are currently investigating the possibility of developing a centralized web database to manage and standardize Student Model components, so that they can be shared between tutorials.

In addition to the student's current abilities, the Student Model keeps a history of the student's past abilities and answers to questions. At the end of a lesson, the Student Model performs a final assessment of the student. The history as well as assessment are stored to disk and can be requested by an instructor.

Instructional Planner

The Instructional Planner reads lesson data from the Expert Model and incorporates this with student data from the Student Model to determine what content should be presented in the next instructional episode. For example, if the next lesson requires the student to apply a map projection to an image, the Planner will check the student's proficiency level at this task. If the student does not meet a threshold of understanding for the concept of map projections, the Planner will provide detailed instruction to the student outlining the task. Otherwise, it will provide limited instructional information and extra guidance only if the student struggles.

Additionally, in the course of a lesson, a well-constructed tutorial will consistently ask the student questions to ensure he/she fully comprehends the material before continuing. If a student has trouble answering correctly, the Planner will deliver more instruction to the student with regard to the trouble area. Overall, the page-to-page flow for the Earth Science tutorials is rather formally defined by the educator; however, the tutor is constantly monitoring the Student Model to see if additional instruction should to be inserted.

A teacher can set the default values for the initial Student Model for his/her class. For example, if the electromagnetic spectrum has been taught in lecture, the teacher can configure EarthTutor so that it does not need to be presented again. However, the Student Model is only a hypothesis of what the student truly knows. Thus, if a student begins showing difficulty in an area that EarthTutor infers to be related to the spectrum, the tutor will adjust the model accordingly, and deliver spectrum (or prerequisite) instruction once more. We are investigating another approach to initializing the Student Model by prompting the student upfront with assessment questions and setting the defaults based on the student's answers. Unfortunately, this would require the tutor to begin by barraging with a quiz on unfamiliar concepts which would likely have the undesirable effect of discomforting the student.

Because the student model data is persistent, a student can approach the packaged learning modules in any order. For example, if a student elects his/her first experience to be with the polar sea ice module, he/she will be presented with introductory instruction on the electromagnetic spectrum. The student's progress with regard to this introductory material will be stored. Later, when he/she begins with an urban growth module, EarthTutor will not squander the student's time by presenting redundant spectrum information again. Thus, the program allows the student to begin training on any subject and can provide prerequisite information as needed. This process requires a larger upfront investment time in the production of training materials and a standardization of common learning concepts, but allows greater flexibility and lesser time investment in actual training.

In our knowledge engineering for this project, we learned that it is not uncommon for a student new to remote sensing to badly botch an image. Often times the steps leading up to a catastrophic state are irreversible. In cases like these, EarthTutor has to be flexible and allow the student to rewind back to the instructional state in which the student first began with an image. Thus, EarthTutor has "go back" functionality which allows the student to start over (i.e., setting back the Evaluator) and updating the Student Model accordingly (marking a deficiency and being sure not to prompt again with the same questions.)

Procedural vs. Student-Directed Instruction

The Instructional Planner classifies instruction into two general methods: procedural and student-directed. Procedural methods teach using a step-by-step, linear approach while student-directed methods are more variable, allowing the student to explore with the tutor on hand to provide challenges and general guidance. Our research led us to the methodology that, for remote sensing instruction, it is best to first present introductory information with lots of hand-holding, being sure the student has a good handle for the materials, before proceeding into areas of study where you give the student more independence, allowing him/her to lead the program interaction. For example, a lesson on measuring distance and area would first present instructions to accomplish the task one step at a time. Steps more difficult to visualize, such as selecting a subregion, are graphically demonstrated with "watch me" mouse movements, text in bubbles, etc. These instructive steps are always followed by interactive open exercises. In the measuring distance and area example, the student could be challenged to measure the distance between a capital city and a hidden water source (that is revealed only by altering the satellite image). EarthTutor monitors the student's actions as he/she attempts to solve the challenge. The ITS can provide context-based guidance if it senses the student is having trouble or if the student asks for assistance. At exercise milestones, EarthTutor will ask the student to answer questions to assess the student's progress and provide immediate feedback to the student.

Scenario Based Reasoning (SBR)

The Instructional Planner continually examines a student's misconceptions and knowledge deficiencies and, between pages, uses similarity-based metrics to select scenarios that best fill the student's gaps. If a student at particular stage should not have a problem with concept x (e.g., identifying an alfalfa crop), but keeps struggling with it, then a deficiency is marked. EarthTutor will then begin tailoring instruction; in other words, it picks out pre-constructed instructions to reinforce learning that can satisfy the deficiency. For instance, there may be a pre-constructed scenario (content, tasks or questions) that provides a parallel examples that teaches x (e.g., identifying a potato nugget crop.) EarthTutor will present this scenario; however, if the student continues to struggle, EarthTutor will have to begin making new inferences. For instance, maybe the student is having trouble with x, because he/she doesn't truly understand related concept y (e.g., the basics of reflectance.) EarthTutor will then begin instructing on y.

The tutorial author will provide these scenarios ahead of time and can manually specify similarities. The ITS will also infer similarities by looking up what knowledge is gained by completing a particular scenario.

Authoring Tool

The Authoring Tool is a separate application that allows an educator to develop new tutorials. The tool is basic enough to allow teachers to construct tutorials to fit one's curriculum (e.g., localize images to his/her classroom's geographical region), while also powerful enough to allow advanced authors to create new interactive tutorials. This Authoring Tool is built in Java to allow operating system portability.

The Authoring Tool has four main editors as seen in Figure 5 below. First, the HTML editor allows an educator to edit the presentational content. An educator uses HTML to layout the content of the lessons. He/she uses common HTML tags to stylize content and make references to websites, images, and animations. EarthTutor uses a special set of tags to represent dynamic content. For example, an educator can use the tag `<openFile filename=”LATLONG.TIF” transparency=”true”>` to specify that this tag should be dynamically replaced with content that will guide the user through the process of opening the file “LATLONG.TIF” with a transparency flag set. This content would be linked to the state machines constructed in the Authoring Tool's second editor, the machine editor.

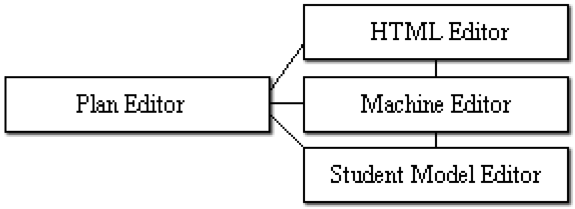


Figure 5. Authoring Tool Components

The machine editor (shown in Figure 6) allows educators to create new state machines that define the flow of the program tasks and user conditions involved in performing an image processing activity. A user has a canvas in which to layout and interconnect the constituent tasks and conditions. Available program tasks include presenting textual instruction, adding student knowledge to the student model, asking the user a question, informing the user of a mistake, providing the user with a hint (in a bubble), moving the mouse to a GUI element and displaying a layover on an image among others. Available user conditions include “has the user provided an answer for question x?”, “has the user opened file y?”, “does the student model contain knowledge z?”, and many other possibilities.

The machine editor comes equipped with numerous state machine templates that monitor common image processing activities. For example, the prototype has a template machine for recording measurements over a stack of images. These state machines can be nested within larger state machines constructed by an author. The goal of this editor is to model programmatic behavior without requiring the user to understand a programming language. Hence, the machine editor consists of a GUI, drag-and-drop interface which provides a visual interface on top of EarthTutor's processing logic.

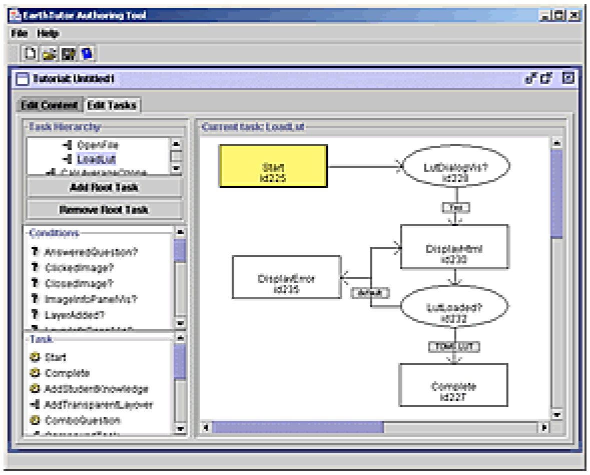


Figure 6. Machine Editor (Loading a LUT)

The third editor, student model editor, consists of a hierarchical tree of knowledge, as described in the student model section earlier. The branches and leaves of the tree map to image processing/Earth Science skills. The machines make reference to these skills as conditions. The information a machine displays as it runs depends on what student knowledge is fulfilled in the student model.

The fourth editor, the plan editor, ties the three previous editors together. Here the educator can blueprint the flow of presentational material. He/she specifies the order in which to present content, and which skills, as defined in the student model, must be attained at each stage. These skills get linked to scenarios (designed as HTML and machines in the first two editors) that can be selected dynamically to ensure a student reaches proficiency in a skill before continuing.

The audience of the Authoring Tool is Earth Science material providers. The tool does not require one to know a programming language, but it does require a significant investment of time to develop an effective tutorial. Developing an ITS tutorial is different than constructing a curriculum for a textbook; one has to take into account the many deficiency's a student may have, as well as the many routes a student may partake upon while using the program. We are currently the primary users of the tool as we will be developing all initial software content for EarthTutor in collaboration with Earth Science educators, who are delivering to us textual content, data, and knowledge of their experiences with prior students.

LESSONS LEARNED

We do not yet have statistically significant findings, but we have performed some very limited evaluations of the EarthTutor prototype with Earth Systems students at Stanford University. In a sample trial, we provided one student with an EarthTutor-based tutorial on global ozone readings. We provided another student with an equivalent paper-based tutorial. Both tutorials consisted of instructional material and a quiz on the characteristics of TOMS Ozone data, applying lookup tables, missing data, overlays, map projections, and calculating global and local measurements. The aim was to provide a clearer picture of the pros and cons of each learning device on similar material. We note that the student using the EarthTutor-based tutorial was able to complete an arguably more efficient course of study:

1. When the EarthTutor-based student got lost (e.g., opened an incorrect file), EarthTutor was quick to alert the student to this error, and the student was able to immediately remediate the situation. In contrast, the paper-based student at one point neglected a printed instruction that told to set the transparency index on an overlay to 255; therefore, the overlay clobbered the underlying picture. It took the student some time (after progressing a couple steps) to ascertain that her image did not look right and then figure out how to remove and reapply the overlay. EarthTutor would have prevented this confusion, but then again, going through this process can also be argued as beneficial to problem solving skills.

2. EarthTutor was able to filter out redundant instruction. When using NI2k, a student may have an overwhelming number of windows, panels and toolbars open at once. The student often closes windows they are not using. When presenting instruction, EarthTutor is aware of which windows are open. For example, if the Lookup Table window is already open, it won't instruct the student to needlessly select "Lookup Table…" from the View menu to make it visible. The print material does not have this luxury; it always instructs the user to open the window regardless of whether it is currently visible. Hence, the print-based student would choose the menu option, see nothing change on the screen, and then choose the menu option again assuming she did something wrong.

As our prototype advances, we will do more evaluations that ultimately attempt to quantify the efficacy of a learning system like EarthTutor in comparison to traditional methods. In the above trial, the EarthTutor-based student mentioned in the debriefing that he appreciated the fact that when he answered a question (e.g. "What is the Ozone value at 37ºN, 122ºW") the tutorial provided immediate gratification and (if necessary) help. In contrast, for the print-based student, we graded her answers at the end of the trial and handed them back. If she reviewed them at all, she no longer had the benefit of NASA Image2000 at her disposal to verify her work. However, she did add in her debriefing, that a great advantage of printed material is being able to refer to it later, and apply it to other work. Hence, we are investigating ways in which to translate EarthTutor's dynamic tutorials into a printed static form so the student can easily refer to them later.

CONCLUSIONS

Intelligent Tutoring Systems show promise in engaging remote sensing students in unprecedented ways. Preliminary evaluations indicate that EarthTutor has advantages over traditional web-based and print-based remote sensing materials. Further prototype development and large-scale evaluations will have to be conducted to demonstrate effectiveness.

ITS technology is innovative in the area of remote sensing in that it instructs within the software framework with which the student will eventually do independent research. The visual instructional environment is much richer, and thus much more effective, than traditional means of satellite imagery education. Artificial intelligence methods also provide for an adaptive experience. The ITS targets a student's problem areas, and can provide scenarios that address his/her most serious deficiencies.

FUTURE WORK

There are a number of AI-related areas in which we hope to further our research. In current form, EarthTutor can dynamically select the best pre-constructed scenarios to present to the student based on the student's deficiency. Our future work will include investigations of ways in which a tutor could use image processing classification techniques to generate new scenarios from scratch using stock imagery. This will be quite feasible for some tasks (e.g. asking the student to measure distances between randomly-selected, known features in an image), but much more challenging for others (e.g. determining features.)

We are also initiating research into ways EarthTutor technology can provide intelligence to the user outside the realm of formalized tutorials. To illustrate, a user might be doing personal research, and after performing a number of operations on image, find that the image appears not as expected. The user could then ask EarthTutor for assistance. If while monitoring the student, EarthTutor learned that they were using with Nimbus-7 Ozone data, but neglected to select a TOMS lookup table, it could suggest that select that table.

Another area of exploration is enhancing the dialogue between the student and the tutor. Currently EarthTutor is equipped only to ask multiple choice and short phrase or numerical fill-in the-blank questions. Recent years have seen great advancement in the research of natural language processing. We would like to incorporate this technology to enable EarthTutor to ask the student questions to be answered in sentences, which would then be interpreted semantically by the tutor. Similarly, it would be of great value if the student could ask EarthTutor a question (such as, “What does this dark area represent?”) and the tutor could respond with explanations and justifications in natural language.

OTHER APPLICATIONS

In the area of Earth Science, remote sensing ITS education looks to offer advantages over traditional educational methods. We are interested in extending our research to other image processing domains to see if it will be similarly successful. Potential areas in which an ITS may improve remote sensing education include urban planning, utility planning, geological exploration, resource exploration, map revising, archaeological digging, and weather forecasting, among others. Non-satellite potential areas include biomedical imaging, chemical imaging and security screening.

A remote sensing ITS also has potential military application. In the Gulf War of 1991, less than 25% of the bombs used were guided to targets (Friscolanti, 2002). In the Iraq War of 2003, nearly all missiles into Iraq were guided by satellite or lasers to encompass strikes of unprecedented, surgical precision. Due to skillful analysts and the high-resolution photography of satellites such as IKONOS and QuickBird, the U.S. military was able to prevent large numbers of civilian casualties. Today’s military relies heavily on effective satellite and aerial interpretation for reconnaissance, target discrimination, weather forecasting and countless other purposes. The training expenses for this pervasive technology are exorbitant and can be reduced by the use of Intelligent Tutoring Systems.

ACKNOWLEDGEMENTS

This research is supported in part by NASA Ames Research Center grant NAS2-03131.

REFERENCES

Friscolanti, Michael (2003, March 22). Undreamed-of precision: Satellites, lasers at the controls of Iraq bombardment. *The National Post*.

Guralnik, D. (1996) An Authoring Tool for Procedural-Task Training”. PhD Dissertation - Technical Report #71. *The Institute for the Learning Sciences, Northwestern University.*

Lillesand, T. and Kiefer, R. (1999). *Remote Sensing and Image Interpretation.* Toronto: John Wiley and Sons.

NASA. (2000) *Studying Earth’s Atmosphere from Space.* Retrieved June 20, 2003 from http://see.gsfc.nasa.gov/edu/ SEES/.

Ong, J and S. Ramachandran. (Feb 2000). Intelligent Tutoring Systems: The What and the How. *Learning Circuits on-line magazine.* Retrieved from http://www.learningcircuits.org/feb2000/ong.html.

Sabins, Floyd F. (1994). *Remote Sensing: Principles and Interpretation***.** Toronto: John Wiley and Sons.