Visual Learning for Science and Engineering

Judith R. Brown, University of Iowa
Steve Cunningham, Oregon State University
Werner Hansmann, Universität Hamburg
Michael B. McGrath, Colorado School of Mines

***Visual Learning** is the use of graphics, images and animations to enable and enhance learning.*

Abstract
The goal of this study is to make faculty in science, computer science, and engineering disciplines aware of the potential for learning through visual means and to encourage them to use visual methods for teaching. Exploiting the visual senses of students can engage their interests and enhance learning. These active visual methods have the potential to increase the number of students, especially women and minorities.

ACM SIGGRAPH and Eurographics have sponsored two events on visual learning in science and engineering, a Campfire in 2002 in the US, and a track at the Computer Graphics Education Workshop in China in 2004. Building on the results of these two workshops and other discussions, we hope to have fruitful discussions at EG05 and to encourage the development of visual learning in these fields. The outcome of these discussions will be a white paper for NSF that will include conclusions and recommendations for future work. This white paper will be in the ACM Digital Library and will also be published by Eurographics. This paper describes a need for meaningful assessment of visual learning methodology. It stresses the importance of drawing and sketching, gives some examples of current courses that use visual approaches, and ends with some conclusions and recommendations for future work.

1. Assessment of visual learning

While scientists and science instructors generally believe that visualization (visual presentations and experiments) can improve learning, including the learning of concepts, there is a lack of empirical evidence that visualization really does help and some evidence that it may not. Hegarty [Heg99], reports on an experiment showing that both a printed presentation and a hypermedia presentation of a mechanical system gave the same learning results. Kehoe [Keh01], however, describes a study that suggests that visualizations may be more valuable when they are used in homework or open-length study settings. Narayanan et al. [Nar04] report on a series of experimental results for both mechanical and computational systems, comparing multimedia presentations with printed presentations. These experiments are used to create a design process that includes a number of cognitive factors which are shown to lead to multimedia visualizations that give genuinely improved learning.

Visual learning needs a study to assess its value, and assessment of learning is always difficult. Assessment of visual learning is even more difficult because our usual means of assessment are verbal and written, which do not necessarily reflect visual learning. How do you test visual learning in science and engineering through the usual written or multiple-choice exams? How do we expect students to be able to express verbally something they have understood through visual means?

2. Importance of drawing

One topic that has surfaced throughout the disciplines is the importance of having students draw to help them think about and understand scientific and engineering concepts. Because many educators feel that drawing is essential, it is being required in many kinds of courses.

2.1. Drawing and sketching for engineering education

Drawing and sketching are important components of visual thinking and learning and have a major role to play in engineering education, as they can help develop the creative side of the student's mind. However, drawing and sketching have been neglected in the present engineering curriculum in favor of an emphasis on analytical thinking.

The goal of engineering education for many years has been to give students a foundation in good design and problem solving. While this is important, industry is also looking for students who can, in the popular jargon, "think outside of the box," or be creative. But how do you teach students to be creative? A proven effective way to develop creativity is to exercise what is commonly referred to as the "right" or visual and spatial side of the brain.

Robert H. McKim (2) developed a popular and successful course at Stanford in the 60's that incorporated this approach in support of design. His book, entitled "Experiences in Visual thinking," and the success of this course, provide convincing evidence of the value of visual thinking. This course was followed by the current course ME313 — Ambidextrous Thinking. The course description describes this course as a "focus on right mode or visual, spatial, kinesthetic and intuitive skills, that will foster a balanced or whole-person approach to problem solving."

An important method to develop creativity is to teach sketching and drawing, especially in support of design projects. Sketching and drawing has been shown to cause a shift to a different mode of thinking that seems to stimulate the mind to creativity. Kathryn Moore [Mor00] suggests that we use drawing imaginatively, to use it as Leonardo da Vinci proposes "as a way of enhancing and arousing the mind to various inventions". Art educators use this approach as a fundamental component in the artist curriculum. An interesting recent study [Hal03] using a Thinking Styles Assessment instrument on engineering students showed that these students and the engineering faculty had a strong preference for thinking in a visual manner. Such research in visual thinking and its potential makes a case for fundamentally rethinking and revising our educational system by including visual literacy to balance our overdependence on analytical approaches.

2.2 Picturing to learn

Some science and engineering faculty believe that learning happens when a person draws or creates an image. Felice Frankel, Massachusetts Institute of Technology, has discussed many aspects of imagery for learning science [Fra05]. She describes how drawings and other means of visual communication help students understand science.The student's drawing lets the teacher see if the student's thinking is correct or not. At the same time, the process of creating the drawing, including any mistakes made, clarifies the idea for the student who is making the drawing. Figure 1 shows frames from a student storyboard for a science study.

3. Examples of Courses Using Visual Approaches

There are different ways to use visuals in teaching science, computer science, and engineering. Here, we highlight a few examples.

3.1 Visual Approaches to Programming

There are some interesting recent developments in work with strong visual approaches to programming courses in computer science. Several computer science faculty, including Steve Cooper, St. Joseph's College, and Wanda Dann, Ithaca College, are using the Alice programming environment in the beginning programming course. In this environment, students use standard programming constructs to control the behaviors of virtual humans in a 3D space. This puts the student in a familiar human-scale place and makes programming a human, not an abstract, process.

In another approach, Mark Guzdial, Georgia Tech, has developed a programming course for non-computer science students entitled "Media Computation" in which students use standard programming techniques in the Python language to manipulate image, video, and sound data. The Python language is relatively easy to use, and the media data makes it exciting for the students to write programs that achieve familiar media goals.

Each kind of course finds its visual content critical to keeping students excited about their work and to improving student success. Each course also has a goal of getting students interested enough in computing that they will

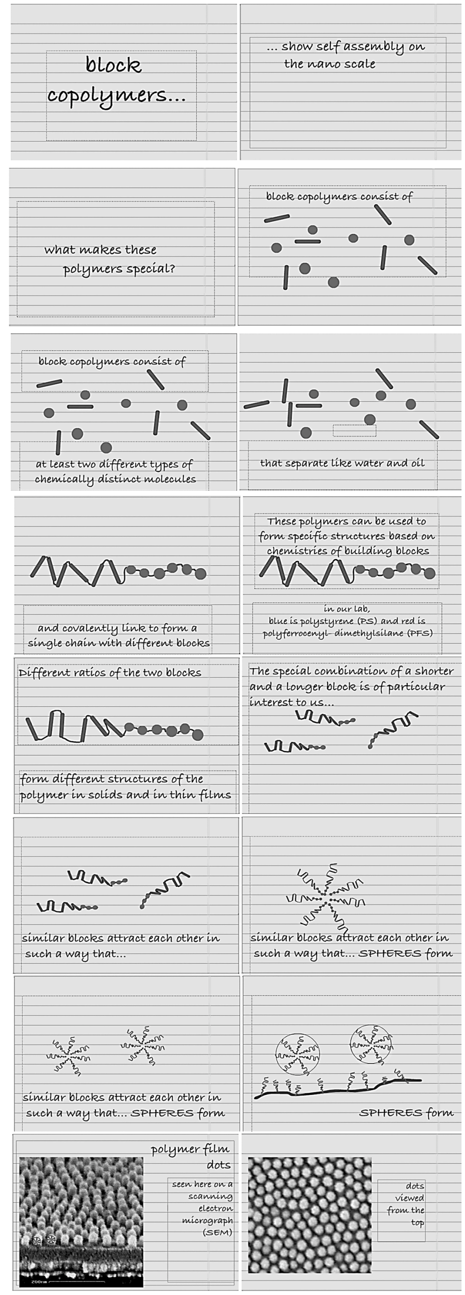

Figure 1. Frames from an animation illustrating block copolymers by Marianna Shnayderman, an undergraduate student.

choose to make computing a career goal. These classes show large increases in the number of women students who take and complete the beginning computer science course.

The real question for this kind of course is whether students whose introduction to programming comes through media computation, the Alice system, or another approach can make the transition into mainstream computer science work. This is making us look at the real learning that is needed in beginning programming and shows promise, though it is far from settled.

3.2 Visuals as a check on other processes

Visuals can inform students as to whether or not they have done a problem correctly. Clemson University has a unique approach to teaching computer science concepts through the use of a problem-based, computer graphics approach. *Τεχνι* (pronounced "techni") is the Greek word for art, and it shares its root with the Greek word for technology. Clemson established a Master of Fine Arts in Digital Production Arts (DPA) program in 1999 that bridges arts and sciences. The goal of *Τεχνι* is a redesign of the B.A. program in computer science that incorporates DPA content and computer graphics research results into all required computing courses. Figure 2 illustrates student work from this course. More information on these courses can be found at http://www.clemson.edu/~tadavis/

A trial course in the new program, Tools and Techniques for Software Development, has been completed. The intent of this course is instruction in programming methodology, and it is taught via the large-scale problem of constructing a ray-tracing system to render synthetic images. Images become the main tool for debugging the logic of a program. Because students can tell immediately if the image is accurate, and because the image points back to the underlying problem, they can easily correct their mistakes. Both correct images and mistakes enable learning. According to Tim Davis, Assistant Professor of Computer Science at Clemson University, the results of this course have been:

- Motivation has increased. Students worked far beyond the course requirements.
- The extent of the extra work performed was very surprising.
- Projects are more effectively engaging the students. This conclusion is based on the student evaluations, as well as the work performed.
- Students get very excited about this class. The course organizers are hoping this type of class will draw more women and minorities, as has been the case in the DPA program, where they get about 30% women and 15% African American students.

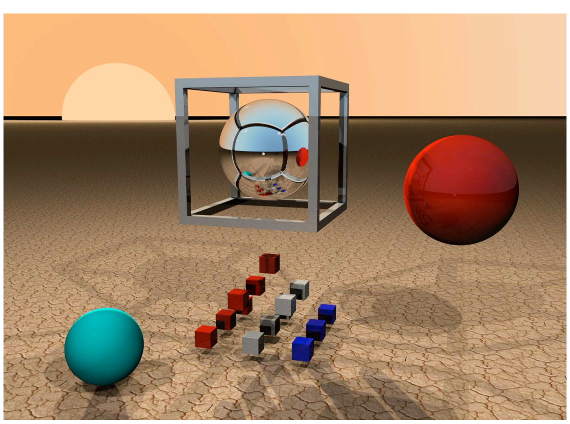

Figure 2.Student work from the Tools and Techniques for Software Development, courtesy of S. Duckworth.

3.3 Studio Courses

Studio courses are courses that focus on student activities, rather than lectures or traditional laboratories, and are examples of active learning in the sciences. In studio courses, the instructor may give a short demonstration or discussion to set up the day. Then the students work individually or in small groups, with the instructor answering questions or giving advice as needed by the students. This emphasis on student activities puts students more in control of their learning. The course activities may also include critiques of student work from the instructor or other students.

Studio classes can be organized in several ways. When this model is used in the sciences, the course usually focuses on students doing active exploratory work. An excellent model is "Studio Physics" where students use computation and visualization tools in their explorations. One exemplary course is Electricity and Magnetism MIT. This course uses Technology Enabled Active Learning (TEAL) (cf.
[http://evangelion.mit.edu/3DStudioPhysics/index.html]
and
[http://web.mit.edu/user/j/b/jbelcher/www/Belcher_physicsannual_fall_01.pdf]).

The course environment is a room containing tables set up for three groups of three students, with each group sharing a computer. Many of the explorations for the course use visualizations that are quite compelling [Bel01]. Figure 3 illustrates student work from this course.

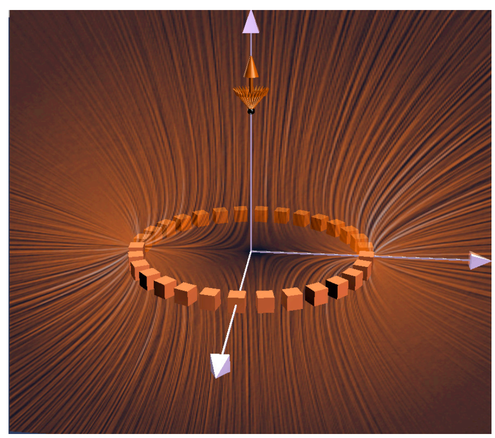

Figure 3: An electrostatic field with a ring of charges, with forces on a moving particle above the ring

3.4 Visuals in Computer Graphics and Geometric Modeling

Co-ordinate transformation in 3D space is one of the main tasks in both Computer Graphics and Geometric Modeling. While understanding the rather simple formal background of transformations is generally not a problem for advanced students of Computing Science who obtained a thorough mathematical foundation during their first undergraduate semesters, it is quite often rather complicated for them to imagine or anticipate the result of a series of transformations. This is particularly true when rotations in combination with translations, or rotations about arbitrary axes are involved. Good imaginative abilities are also required when it comes to mastering illumination and colors, the effects of reflection and refraction, and the application of textures. Again, the basic concepts of simple shading algorithms or even ray tracing and radiosity can usually be well understood, while skills beyond pure mathematics are needed for the proper use of these techniques.

It goes without saying that – as is certainly true for most Computer Graphics curricula – graphical illustrations and animations form an integral part of the presentations in the introductory Computer Graphics course within the Computing Science education at the University of Hamburg. In order to complement the lectures, and allow the students to practise and enhance their imaginative skills, an optional lab assignment is offered.

In teams of three the students have to create animated sequences using POV-Ray™ for photorealistic rendering of the individual frames, and the CSG modeler contained within POV-Ray™ to model the objects of the scenes. They may freely choose the topic of their movie. At least one of the objects should feature moving extremities; digital pictures are to be used as textures. As a scene description language POV-Ray™ may be a rather clumsy programming language; but students of Computing Science can be expected to handle it well. And the advantage lies in the fact that they are obliged to formulate every CSG modeling step and every detail in their scene descriptions explicitly; thus proving that they understood the underlying concepts and mathematics.

At the end of the semester they submit a CD containing the completed movie, the POV-Ray™ script, and a written documentation of their work. The movies are presented to the entire group of students who attended the course; together they act as jury which assesses the submissions. Figure 4 is an example of this work.

The creativity involved with this task and the immediate visual feedback results in a very high motivation, even enthusiasm. It also attracts students to this course who take Computing Science as a minor field of study, although their mathematical background is not always as firm as that of the Computing Science students. Students are generally very excited about this course; even some of those students who are not required to carry out the assigned task do it nevertheless, and with very good result. With additional soundtracks etc. most of the submitted movies exceed the requirements.

4. Conclusions

From our workshops and discussions with science and engineering faculty and students, we have found some common views on visual learning, as well as areas where more work is needed. We draw some conclusions from our discussions and classes.

Figure 4. „The AntHropoid," frame from a movie of an ant playing basketball, by Peer Stelldinger, graduate student

Conclusion 1. Students need to create their own visuals for understanding and learning. All science and engineering students should have some exposure to creating visualizations, and a science program should include a firm grounding in visual theory and practice.

Conclusion 2. Technology advances and cost decreases have opened up new possibilities for visual learning. Technology now allows us to create and demonstrate images and animations easily. Visualization equipment (both hardware and software) costs have dropped to where they are affordable for classroom use. There is no longer a major impediment to using visual techniques for teaching.

Conclusion 3. Engineering education must be fundamentally revised to include visual thinking throughout the curriculum. Drawing and sketching are important techniques to accomplish this and should be included in the students' experience.

Conclusion 4. We can blend old and new ways of knowing to build new "languages," or symbol systems, of visual communication. A visual language can communicate most effectively in some situations, just as verbal, mathematical, or musical "languages" communicate effectively in other situations.

Conclusion 5. Developing visual thinking is important in order to create the collaborative learning methodologies and enhance distance learning environments that are crucial to the future of learning.

5. Future Directions

Our experiences have illuminated additional work that needs to be done to accomplish our goals. We encourage NSF and other funding agencies to support this continuing work towards better understanding and facilitation of effective learning through the use of images and animations.

1. We need to collaborate across disciplines and cultures to articulate this visual methodology.

2. More studies that assess the value of visual learning need to be done. How do we assess visual learning? Can we assess visual learning through traditional oral and written exams, or do we need to develop visual testing methods?

6. References

[Bel01] J. W. Belcher and R. M. Bessette, "Using 3D animation in teaching introductory electromagnetism," *Computer Graphics* 35, 18-21 (2001)

[Fra05] F. Frankel, "Translating Science into Pictures: A Powerful Learning Tool," in *Invention and Impact: Building Excellence in Undergraduate Science, Technology, Engineering, and Mathematics (STEM) Education*, pp. 155-158, AAAS. Washington DC, 2005.

[Hal03] A. Halstead, *Thinking Styles Assessment for Enhanced Progression: Progress Report*, University of Hull, 2003

[Heg99] M. Hegarty, et al., "Multimedia Instruction: Lessons from Evaluation of a Theory-Based Design," *Journal of Educational Multimedia and Hypermedia*, v8, 1999, 119-150

[Keh01] C. Kehoe et al., "Rethinking the Evaluation of Algorithm Animations as Learning Aids: an Observational Study," *International Journal of Human-Computer Studies*, 54, 2001, 265-284

[McK72] R. H. McKim, *Experiences in Visual Thinking*, Brooks-Cole, 1972.

[Mor00] K. Moore, "BETWEEN THE LINES: the role of drawing in design," in *Environments by Design*, volume 4, 2000. See also http://www.lboro.ac.uk/departments/ac/tracey/comm/moore.html

[Nar04] M. Narayanan, et al., "Multimedia Design for Communication of Dynamic Information," *International Journal of Human-Computer Studies*, 57, 2004

This work was supported by the U.S. National Science Foundation under the grant DUE-0407830. All opinions, findings, conclusions and recommendations in any material resulting from this work are those of the principal investigators, and do not necessarily reflect the views of the National Natural Science Foundation.