Just Truss Me: User Interface

Alexis Chuck, Patrick Robinson, Chris Aikens, Martin Field, Travis Kosarek,
Drew Logsdon, Laura Murphy, Alyssa Nabors, Paul Taele, Stephanie Valentine
Erin McTigue, Julie Linsey, Tracy Hammond
Sketch Recognition Lab
Computer Science Department
Texas A&M University
911 Richardson
College Station, TX 77843-3112

ABSTRACT

The mechanics of trusses is an essential concept taught in introductory civil and mechanical engineering courses. Professors want to have students learn through sketching truss diagrams. However, such introductory courses are often very large, and sketches take substantial time to grade. As a result, it is extremely difficult for professors and teaching assistants to provide students with constructive, timely feedback on assignments involving sketched diagrams. We developed an educational program, Mekanix, to address the dilemma presented by time constraints. Mekanix provides immediate, incremental feedback on sketches of truss diagrams. We developed a highly intuitive user interface that avoids mode switching and simulates pen and paper as closely as possible. This allows its users to focus on learning the engineering concepts rather than software-specific tools. Through its immediate feedback and ease of use, Mekanix will benefit student and instructor alike.

INTRODUCTION

In their first semester, mechanical and civil engineering students learn the fundamental concepts of engineering. A large portion of the time spent in these introductory classes is devoted to solving statics problems. Statics problems usually require the student to draw free body diagrams and planar truss diagrams.

Trusses are used as supports in many structures such as bridges, houses, and other buildings. An excellent foundation of how to construct a truss is critical for a student's success as an engineer in the future.

In current practice, the most effective method for learning how to construct a truss is to draw the truss along with the forces acting upon it on pen and paper. This method works best when an active learning approach is taken, that is, a learner should be engaged and cognitively active while learning. Timely feedback should be given to the learner when a mistake is made to prevent the learner from adding false information into their knowledge framework.

While this method seems ideal, the large size of introductory engineering courses prevents hand-drawn solutions from being used because of time constraints on providing feedback to the students. To combat these time constraints, multiple-choice questions are the primary source of testing. In these courses, students are likely to receive only one or two hand-drawn assignments a semester.

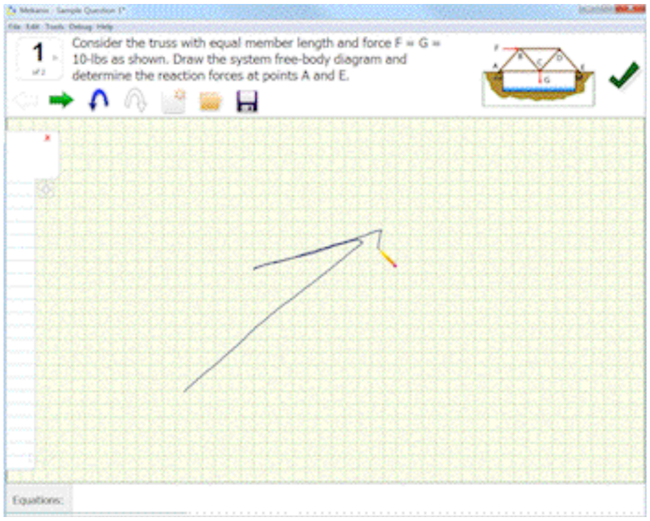
To stimulate the educational value of these courses, the need for a better method of grading hand-drawn truss diagrams is necessary. We created a program that uses sketch recognition to provide immediate, formative feedback on diagrams.

BACKGROUND

Educational Benefits of Sketching

Many educational theories indicate the physical act of sketching leads students to a better understanding of course material. Sketching is a kind of forced active processing that requires students to construct their knowledge by using what they already know in conjunction with the tools at hand – whether those tools are pen and paper or pen and tablet PC. In contrast to formats in which students have less control over their answer, as in the case of multiple-choice questions, assignments featuring sketches result in students becoming more engaged with the problem at hand [Kozma 1994]. Such engagement ensures a higher level of understanding, according to the Select--Organize-Integrate model of education [Mayer 1996]. While visual and auditory learning are already provided by texts and class lectures, sketching, as method of kinesthetic learning, introduces an additional channel of cognition. Using multiple cognitive channels in instruction allows for more learning [Sweller 1994]. The inclusion of sketching will greatly aid those students who are primarily kinesthetic learners.

Other characteristics besides its tactile nature make sketching an appealing method of instruction, especially in regard to truss diagrams. In contrast to a simple yes or no answer, a sketch reveals the many intermediate steps a student took to reach his or her final answer. This allows for corresponding intermediate feedback and partial credit.



Consider the truss with equal member length and force F = G = 10-lbs as shown. Draw the system free-body diagram and determine the reaction forces at points A and E.

Figure 1. Screen shot of Mekanix. Here a user is in the process of drawing an arrow.

Finally, sketching is a widely possessed skill. Professional engineers and architects prefer using pen and paper sketching to computer software [Kivett 1998; Carrato 2004; Rose 2005]. Having students sketch in the classroom not only reduces the time required to learn how to use the program, but will also better prepare them for their careers after graduation.

The Importance of Feedback

In large classes students often have to wait weeks, or even months, for an assignment to be handed back. If the professor or teaching assistant must grade hundreds of assignments, he or she likely will not have the time to provide individualized feedback. In this scenario, a student may have had a faulty understanding of a key concept for quite some time, which will have impaired his or her learning of new material. In addition, the absence of specific feedback will make it hard for the student to perceive his or her mistake. Finally, if assignments are not returned in timely manner, students cannot study them in preparation for exams. The advantage of receiving feedback before a course has moved on to different content has been confirmed by studies. Additionally, formative feedback -- given as students learn, guiding them to understanding -- is more effective than summative feedback, which only tests what has been learned [Black 1998].

METHOD

Simulating Pen and Paper

Truss diagrams include trusses, which are combinations of closed shapes such as triangles and squares, as well as axes and forces, which are represented with arrows. Since using pen and paper is a widely held skill, we reduce the learning curve of our program by creating an interface that simulates the experience of drawing a truss diagram by hand.

We avoid the traditional toolbar-based system; there is no line tool or triangle tool. Instead, students are free to draw truss diagrams in an unconstrained manner. There are no restrictions on the size or order of the strokes they use. We use free-sketch techniques of recognition, which focus more on what a shape looks like than how it is drawn to accommodate a wide range of sketching styles. As long as the final sketch reasonably resembles an expected element of a truss diagram (a line, polygon, arrow, etc.), Mekanix is able to recognize it.

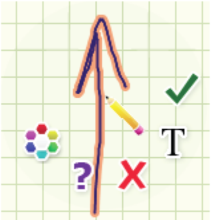
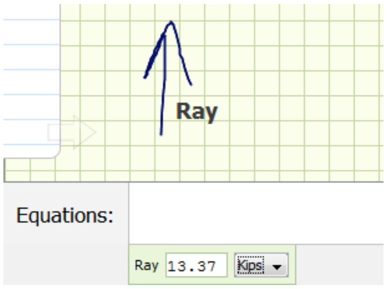


Figure 2. Highlighted arrow and the shape popup menu. The cursor is represented with a pencil graphic.



Ray 13.37 Kips

Figure 3. Reaction force and the corresponding panel.

Mekanix provides a tabbed note panel on the left-hand side of the application screen for scratch work. The panel can be dragged in and out as needed. Though its contents are not recognized, they can be exported to the main sketch area where recognition does take place.

Adding Information to a Sketch

As the user mouses over their sketch, existing shapes will highlight. Highlighting serves as the most basic form of feedback, providing visual confirmation of strokes being grouped into more complicated shapes.

If a user pauses on a highlighted shape, a pop-up menu opens offering a limited number of options. These include (from left to right in Figure 2): changing the color of the highlighted shape; changing how the shape is recognized (for example, if Mekanix has mistakenly identified a lumpy triangle as a circle); deleting the shape; labeling the shape; and, if the user is within the instructor mode (see below), adding specific information that must be provided about the shape, such as a maximum or minimum load. This menu disappears once the user has chosen an action, or if they move the cursor beyond the menu.

Mekanix is attentive to how users label their shapes. For example, if an arrow's name begins with a capital letter 'R' or 'F,' the arrow is recognized as a reaction force as this is the standard convention for labeling such forces. When a reaction force is first labeled, a new panel appears at the bottom of the window. This panel provides an area for users to enter a value to be associated with the force. The panels update if the shape's label is modified, and removed entirely if the user re-labels the shape with an empty string.

This method of adding panels only as needed, and without any additional action on the user's part, keeps the interface free of potentially distracting or confusing unneeded elements. It offers a way for the user to create an entry field for necessary information without doing anything more than label a shape.

Instructor Mode

The interface that instructors use to input questions and solutions is virtually identical to the one students interact with. Instructors draw the correct diagram, enter the correct equations, and assign the correct labels and values to appropriate forces. The result is saved as a correct solution. Since some questions leave room for multiple correct answers, instructors may add multiple solutions to a single question.

Checking Answers

A student can request feedback at any time by clicking a green check mark in the top right-hand corner of the application window. When he or she does so, Mekanix creates a model of the sketched truss and compares it with models of that question's solutions. Other elements of the diagram, such as axes and forces, and any equations or values are also compared.

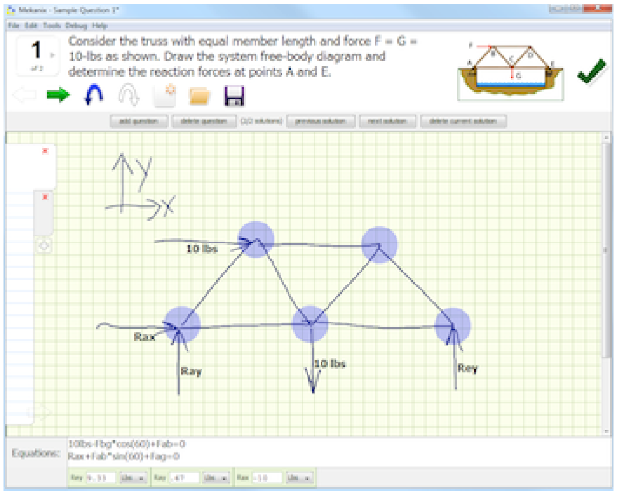
Differences between the student's sketch and the answer are stored in a priority queue according to the grievousness of the error. Only the highest priority difference is then reported to the user. A feedback panel slides down at the top of the sketch, displaying a message describing the problem for a few seconds before sliding back up. A component of the sketch, such as a node or force, or the equation text box, or one of the various reaction force panels is individually highlighted at this time.

If the instructor has provided multiple correct solutions, but the student's answer does not match any of them, Mekanix attempts to determine which correct solution is closest to the student's. It does so by checking the student's answer against all solutions and determining which pairing results in the smallest list of differences. The feedback students receive is thus specific and constructive. It is formative in nature since students can immediately attempt to correct their answer and re-check their answer. Mekanix avoids providing too much information at once, guiding students towards understanding by clearly defining the next step they need to take to arrive at the correct answer.

RESULTS

It will take at least a semester of classroom use to formulate quantifiable data on the educational benefit of Mekanix, so currently, we can make few meaningful conclusions. We have, however, performed several user studies with civil and mechanical engineering graduate students and professors to ensure that the software we send into the classroom is robust and intuitive. Each of the five studies helped us to iteratively assess the accuracy of our recognizers and the usability and flow of our interface.

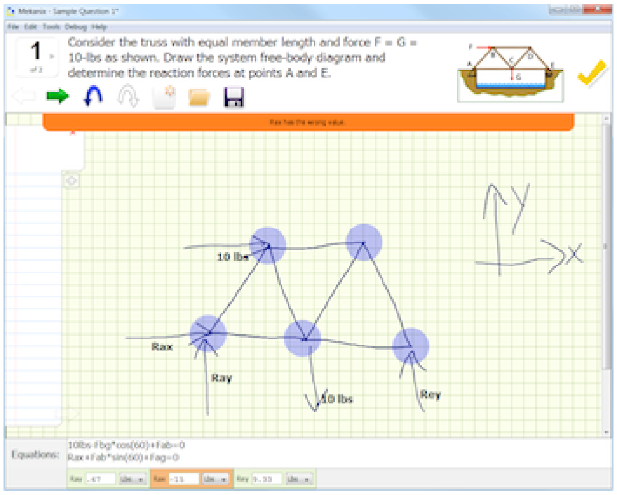
We have a formal classroom study planned to take place during the fall 2010 semester. In our study, we will gather about 75 participants from a single undergraduate engineering course usually taken by freshmen. Each student will receive extra credit for his/her participation in the study. We will split participants into four groups. Three of



Consider the truss with equal member length and force F = G = 10-lbs as shown. Draw the system free-body diagram and determine the reaction forces at points A and E.

10lbs-Fbg*cos(60)+Fab=0
Rax+Fab*sin(60)+Fag=0

Figure 5. Screen shot of Mekanix in instructor mode.



Consider the truss with equal member length and force F = G = 10-lbs as shown. Draw the system free-body diagram and determine the reaction forces at points A and E.

Rax has the wrong value.

10lbs-Fbg*cos(60)+Fab=0
Rax+Fab*sin(60)+Fag=0

Figure 4. Feedback for an incorrect answer. In this case, a reaction force has been given the wrong value.

the groups will attend five two-hour sessions (ten hours total) during which they will receive several problems to solve. One group will solve the problems using our Mekanix software, one will use WinTruss (a competitor software), and one will use simple pencil and paper. The fourth group will attend no extra tutoring sessions.

We will track the test scores of the participants before, during, and after the tutoring sessions to determine the impact of Mekanix's instant feedback on overall learning. We will also invite a sampling of the participants in the first three groups to a focus group discussion, where we will ask for feedback on the effectiveness, intuitiveness, and helpfulness of each method (Mekanix, WinTruss, and pen/paper).

RELATED AND PRIOR WORK

LADDER [Hammond 2005] is a sketch recognition language used for the recognition of shapes. It uses geometry-based recognition to define how a shape is formed. Recognizers can be defined by first drawing a shape. From the sketched shape, LADDER recognizes primitive component shapes such as lines, arcs, circles, etc. LADDER then automatically creates a recognizer for that shape based on constraints like “below”, “near”, or “coincident.” Using these constraints, LADDER can define or describe the higher level shapes.

PaleoSketch [Paulson 2008] is a sketch recognition library used to recognize hand-drawn primitives like lines, ellipses, arcs, curves, etc. To do this, PaleoSketch creates confidence values on what shape a stroke could potentially be, then chooses the shape with highest confidence value as the recognized shape.

WinTruss [Sutton 2000] is an application to design and solve truss diagrams. Before the user can begin drawing trusses, the application’s environment must be set up with specific information about units, grid spacing, and the materials being used to build the structure. After this is done, the system allows the user to use tools such as the “beam tool” to draw a beam on the screen, define the actual length of the beam, and label it as needed. After the external forces have been applied to the truss diagram, WinTruss can solve the member force values of the constructed truss diagram. The system is designed to allow the user to draw and simulate the forces acting on the truss. However, it does not provide instruction or feedback on how trusses should be formed.

Newton’s Pen [Lee 2007] is a “pentop computer” application, meaning that it runs on a processor inside the pen itself. The application uses vision-based sketch recognition to accept or reject very simple free body diagrams. To recognize shapes, the pen digitizes the ink that it inscribes on paper and compares the digitized strokes to a bitmap of the “perfect” configuration for that shape. The program runs as a finite state machine, so each piece of the diagram must be drawn in a specific order and configuration. The application gives basic feedback, but only to inform the user of the number of forces left to be drawn.

Hashigo [Taele 2009] is a system for assisting a user in learning the written technique of Japanese kanji. It provides feedback and critiques of the user’s sketch of the character that he or she wishes to learn how to write.

iCanDraw? [Dixon 2010] mixes facial and sketch recognition to assist the amateur artist in successfully rendering an accurate drawing of a human face. It compares the user’s sketch to an image of a face and provides feedback to help the user improve his or her drawing.

CONCLUSIONS AND FUTURE WORK

Mekanix recognizes and provides feedback on a student’s hand-drawn truss diagram in real-time. By using a highly intuitive interface, we hope to guide students to the understanding of concepts, not train them in the workings of a specific software.

The immediate feedback Mekanix provides allows instructors to regularly assign sketches as homework and test questions as it eliminates the burdensome time commitment presented by grading pen and paper sketches.

In the future, we hope to implement a system of logging the number of times a student checks their answer, and the differences present each time, before they arrive at the correct one. This record would then be reported to instructors who could define their own system of awarding points and grades.

We also hope to create a web system to process the correctness of sketches, thus removing the “key” sketches from the student's hands. The “keys” currently take little investigation to discover if a student is knowledgeable about the system.

Also, we would like to expand our system to allow the full spectrum of free-body diagrams, not just trusses.

ACKNOWLEDGMENTS

We thank the CRA-W/CDC Distributed Research Experience for supporting this opportunity.

This work funded in part by NSF IIS grants: NSF 0935219: Civil Engineering Sketch Workbook, NSF 0942400: Sketched-Truss Recognition Tutoring System, and NSF 0943999: REU Supplement for 0757557.

REFERENCES

1. [Dixon 2010] Dixon, D., Prasad, M., and Hammond, T., 2010, iCanDraw? – Using Sketch Recognition and Corrective Feedback to Assist a User in Drawing Human Faces, ACM Conference on Human Factors in Computing Systems.
2. [Hammond 2005] Hammond, T., and Davis, R., 2005,

LADDER: A Sketching Language for User Interface Developers, Computer and Graphics, Elsevier, pp. 518-532.

3. [Kozma 1994] Kozma, R., 1994, Will media influence learning: Reframing the debate, Educational Technology Research and Development, pp. 7-19.

4. [Paulson 2008] Paulson, B., and Hammond, T., 2008, PaleoSketch: Accurate Primitive Sketch Recognition and Beautification, 13th International Conference on Intelligent User Interfaces, pp. 1-10.

5. [Sutton 2000] Sutton, M., and Jong, I., 2000, A Truss Analyzer for Enriching the Learning Experience of Students. In *2000 ASEE Annual Conference Proceedings*.

6. [Taele 2009] Taele, P., and Hammond, T., 2009, Hashigo: A Next-Generation Sketch Interactive System for Japanese Kanji, 21st Innovative Applications Artificial Intelligence Conference.

7. [Mayer 1996] Mayer, R. E., 1996, Learning strategies for making sense out of expository text: The SOI model for guiding three cognitive processes in knowledge construction, Educational Psychology Review, pp. 357-371.

8. [Sweller 1994] Sweller, J., 1994, Cognitive Load Theory, learning difficulty, and instructional design, Learning and Instruction, pp. 295-312.

9. [Black 1998] Black, P., and William, D., 1998, Assessment and Classroom Learning, Assessment in Education: Principles, Policy & Practice, pp. 7-74.