SkCHI: Designing Sketch Recognition Interfaces

Tracy Hammond
Sketch Recognition Lab
Computer Science
Texas A&M University
3112 TAMU
911-912 Richardson
College Station, TX 77843 USA
hammond@cse.tamu.edu

Edward Lank
David R. Cheriton School of Computer Science
University of Waterloo
200 University Ave. West
Waterloo, ON N2L 3G1 Canada
lank@uwaterloo.ca

Aaron Adler
Raytheon BBN Technologies
10 Moulton Street
Cambridge, MA 02138 USA
aadler@bbn.com

Sketch Recognition Lab @ TAMU
http://srl.cse.tamu.edu

Acknowledgements:
The authors would like to thank Paul Corey, Brian Eoff, Josh Johnston, Joshua Peschel, and Paul Taele for their contributions towards the completion of this document.

Copyright is held by the author/owner(s).

CHI 2010, April 10–15, 2010, Atlanta, Georgia, USA
ACM 978-1-60558-930-5/10/04.

Keywords
Sketch recognition, pen-based computing, interfaces

ACM Classification Keywords
H.1.2 [Models and Principles]: User/Machine Systems--human factors; H.5.2 [Information Interfaces and Presentation]: User Interfaces

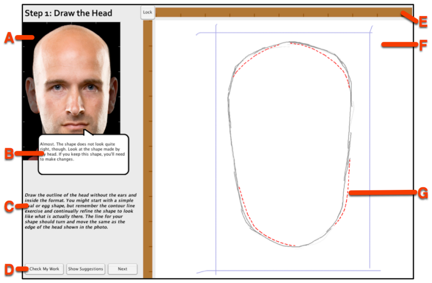
General Terms: Human Factors

Abstract

Sketch recognition user interfaces currently treat the pen in the same manner as a mouse and keyboard. The aim of this workshop is to promote thought and discussion about how to move beyond this to create natural and intuitive pen-based interfaces. To this end, the workshop will include panel discussions, group discussions, and even an instructional session on drawing sketches.

Motivation

Sketch recognition interfaces exist at the crossroads of artificial intelligence and human-centered systems; however, traditionally sketch recognition has focused on 1) the AI aspects of the field and 2) incorporating recognition into existing interface paradigms, sometimes at the cost of the user experience. Users of sketch recognition systems are frequently required to use a seemingly familiar tool within the restrictions of

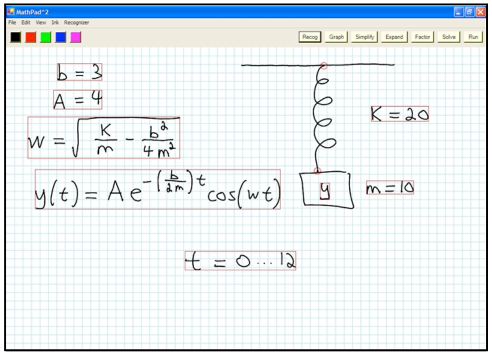


Step 1: Draw the Head

Almost. The shape does not look quite right, though. Look at the shape made by my head. If you keep this shape, you'll need to make changes.

Draw the outline of the head without the ears and inside the format. You might start with a simple oval or egg shape, but remember the contour line exercise and continually refine the shape to look like what is actually there. The line for your shape should turn and move the same as the edge of the head shown in the photo.

Figure 1: ICanDraw [6]



$b = 3$

$A = 4$

$w = \sqrt{\frac{K}{m} - \frac{b^2}{4m^2}}$

$y(t) = Ae^{-\left(\frac{b}{2m}\right)t}\cos(wt)$

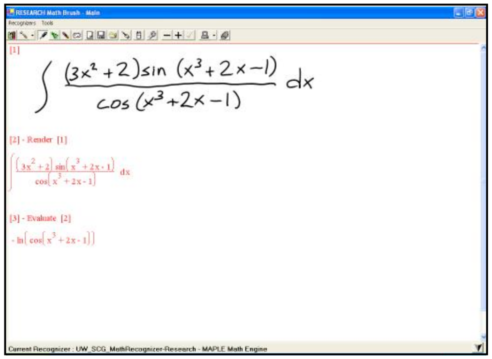
$K = 20$

$m = 10$

$\boxed{y}$

$t = 0 \dots 12$

Figure 2: MathPad2 [11]



[1]

$$\int \frac{(3x^2+2)\sin(x^3+2x-1)}{\cos(x^3+2x-1)}\,dx$$

[2] - Render [1]

$$\int \frac{(3x^2+2)\sin(x^3+2x-1)}{\cos(x^3+2x-1)}\,dx$$

[3] - Evaluate [2]

$$-\ln(\cos(x^3+2x-1))$$

Current Recognizer : UW_SCG_MathRecognizer-Research - MAPLE Math Engine

Figure 3: MathBrush [10]

the traditional WIMP interface paradigm whose input has been dominated and skewed by the proliferation of the keyboard and mouse. Commonly, the pen acts as a mere replacement device for a mouse, and sketch recognition researchers seem somewhat bound to this approach. While some work has been done seeking to ameliorate the oddity caused by the use of a pen in a potentially unnatural and unintuitive environment designed around a mouse and keyboard, a gulf still exists between the sketch recognition system and the user. To the user, a new mode of interaction is occurring, pen input; however, this conceptual model is inaccurate as the computer still interprets the pen under the mouse/keyboard archetype. No longer can the pen merely stand in for the mouse; rather, a new paradigm of human-computer interaction must be designed around the pen and recognition of the pen input. Pen-based interfaces should provide interpretation and feedback in a natural and intuitive manner, rather than locking the user into mouse-like interactions.

We have created a workshop called “SkCHI: Designing Sketch Recognition Interfaces”, which focuses on the computer human interface of sketch recognition (SkCHI is pronounced ‘Skye’, although some may think the name ‘Sketchy’). This workshop seeks to classify and redress the gap between user understanding and interaction. Current interaction techniques and paradigms will be examined for weakness and strengths, and be adapted or incorporated in the design of a new interaction model tailored for sketch recognition systems. We will investigate how current interaction methods have succeed and how they have failed, and what new methods are needed to create a natural and intuitive sketch recognition system. In short, we seek to envision a powerful new interaction paradigm in which the pen is the primary interaction tool.

Previous Research

Several researchers within the sketch recognition community have conducted preliminary research related to our goal of a unified sketch recognition interface framework; however, the primary focus of most research groups has been on recognition algorithms and accuracy. These initial steps provide a strong base from which the new sketch interface paradigm can be built. For example, Alvarado provides insight into the requirements of a sketch recognition interface within keyboard/mouse world in [2]; Hammond and Davis seek to formalize the underlying language of sketch interfaces in [8], Stahovich et al. employ sketch recognition for mechanical design [14]; Forbus et al study spatial reasoning through user recognized sketches [7]; Davis summarizes the power of sketch recognition interfaces in [5].

Sketch recognition interfaces draw heavily from research search done in pen-based computing. Bi et al. have investigated deeply how the pen can be used as an interaction tool [4]; Apitz et al. have proposed new methods for pen-based selection and other commands [3]; Saund and Lank investigate modality in pen-based computing [13]; Landay and Myers have developed sketch-based interface design tools [9]; Kurtenbach and Buxton have pioneered novel interfaces using pen input [12].

Workshop Topics

Users were asked, not to present papers on their research, but rather potent and reactive position papers

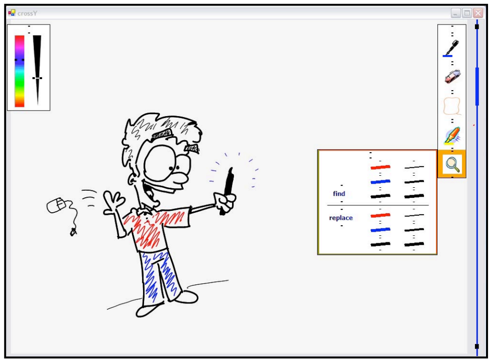


Figure 4: CrossY [3]

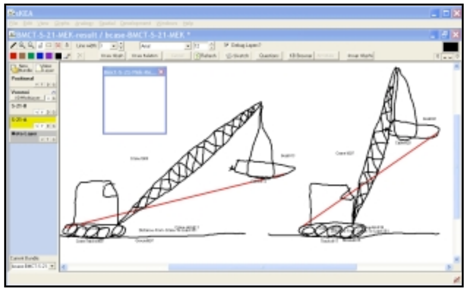


Figure 5: CogSketch [7]

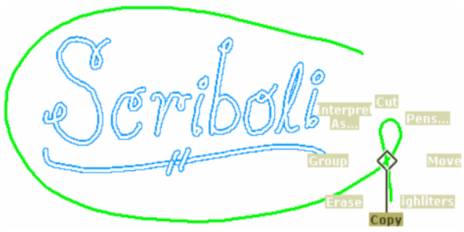


Figure 6: Scriboli [4]

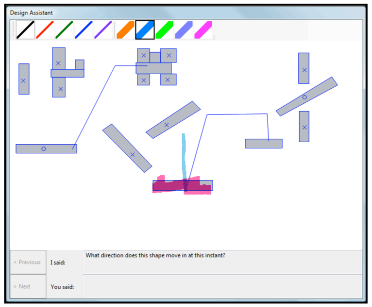
on the open research problems in the SkCHI community. Discussion topics in the design of this new interaction paradigm include, but are not limited to:

- What is an acceptable amount of user instruction and/or user training examples such that the system is still usable? How does the domain factor into the tradeoff between a ready-to-use system versus one requiring learning or training by a user?
- Feedback is invariably an important feature in sketch recognition systems. What are various possible methods of feedback? Which types of feedback are appropriate in which situations?
- Erasing is challenging to implement, especially when considering aspects such as stroke-level versus bitmap-level erasing, and the effects on re-recognition. What is the best way to implement erasing in order to provide a truly usable drawing experience?
- The images displayed on the screen are of primary importance to the user. Displaying a user's original strokes has significant benefits, as does displaying the system's interpretation. How can sketch recognition systems harness the benefits of both methods of display?
- Errors, caused by a recognizer or by a human, will occur. What is the best way to deal with error detection and recovery?
- Sketches of a single design contain a combination of drawing, editing, and attention strokes. An area of continuous debate has been mode versus modeless systems. What is the best way to distinguish between these stroke types from a user interface perspective?
- What editing capabilities are necessary to make a truly usable experience?
- Sketch recognition systems have been mostly limited to a specific type of domains. What other domains could benefit from diagram understanding? How could sketch recognition be applied to these domains? What are the difficulties (and possible solutions) for applying sketch recognition such a domain?
- Sketching is not just about drawing. When people sketch, they talk, they interact, they touch, they gesture. How can the benefits of multimodal interfaces be incorporated to create a usable application? What are some of the pitfalls of multimodal interaction? How can the pitfalls be overcome?
- Repeatedly, the question "What is the killer application?" is posed to the sketch recognition community. What factors of design and usability are necessary before any such killer application can exist?
- What else is holding back sketch recognition systems from being truly positive computer-human interaction experiences?

Workshop Goals

The goal of our workshop is to obtain potential solutions to user interface difficulties that have eluded sketch recognition system creators. But more importantly we want to provoke new avenues of thought for sketch recognition research. As such, the workshop will consist of research overviews and introductions, panels and debates, and even drawing sessions.

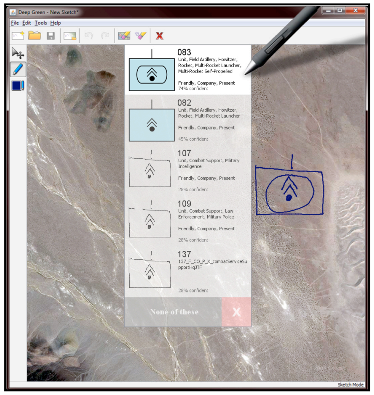
This workshop will dedicate one day to the examination of these open research problems. The day will start



I said: What direction does this shape move in at this instant?

You said:

Figure 7: MIDOS [1]



083
Unit, Field Artillery, Howitzer, Rocket, Multi-Rocket Launcher, Multi-Rocket Self-Propelled
Friendly, Company, Present
74% confident

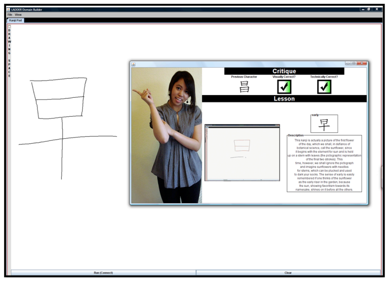
082
Unit, Field Artillery, Howitzer, Rocket, Multi-Rocket Launcher
Friendly, Company, Present
45% confident

107
Unit, Combat Support, Military Intelligence
Friendly, Company, Present
28% confident

109
Unit, Combat Support, Law Enforcement, Military Police
Friendly, Company, Present
28% confident

137
137_F_CO_P_X_combatServiceSupportHqJTF
28% confident

Figure 8: Sketch Interface for military course-of-action (COA) symbols.



Description

This kanji is actually a picture of the first flower of the day, which we shall, in defiance of botanical science, call the sunflower, since it begins with the element for sun and is held up on a stem with leaves (the pictographic representation of the final two strokes). This time, however, we shall ignore the pictograph and imagine sunflowers with needles for stems, which can be plucked and used to dark your socks. The sense of early is easily remembered if one thinks of the sunflower as the early riser in the garden, because the sun, showing favoritism towards its namesake, shines on it before all the others.

Figure 9: Hashigo [15]

with a 2-minute madness where each contributor presents her past experience and/or research contributions in the field of designing sketch recognition interfaces. Then, interesting and diverse opinions on the above topics will be chosen to briefly present their ideas and then lead a panel on the topic. Many of these sessions will involve people breaking up into groups of a manageable size to provoke innovative discussions. During the day we plan to divide the group into students and non-student to discuss the future of the field. We expect that the students of today will be the leaders of the field tomorrow, and as such will provide an interesting perspective on the future of the field.

Citations

[1] Adler, A. MIDOS: Multimodal Interactive DialOgue System. *Ph.D. Thesis for MIT.* 2009

[2] Alvarado, C. Sketch Recognition User Interfaces: Guidelines for Design and Development. In *AAAI 2004 Symposium on Making Pen-Based Interaction Intelligent and Natural*, (2004), 8-14.

[3] Apitz, G. & Guimbretière, F. CrossY: a Crossing-Based Drawing Application. In *Proc. of UIST-04,* ACM Press (2004), 3-12.

[4] Bi, X., Moscovich, T., Ramos, G., Balakrishnan, R., & Hinckley, K. An Exploration of Pen Rolling for Pen-based Interaction. In *Proc. of UIST-08,* ACM Press (2008), 191-200.

[5] Davis, R. Magic Paper: Sketch-Understanding Research. *Computer* 40(9), (2007), 34-41.

[6] Dixon, D., Prasad, M., & Hammond, T. iCanDraw? Using Sketch Recognition and Corrective Feedback to Assist a User in Drawing Human Faces. In *Proc of CHI-10,* ACM Press, (2010).

[7] Forbus, K., Usher, J., Lovett, A., Lockwood, K., & Wetzel, J. CogSketch: Open-domain Sketch Understanding for Cognitive Science Research and for Education. In *Proc. of Eurographics Workshop on SBIM*.(2008).

[8] Hammond, T., & Davis, R. LADDER: A Sketching Language for User Interface Developers. *Computer and Graphics*, Elsevier, (2005), 518-532.

[9] Landay, J.A. & Myers, B.A. Sketching Interfaces: Toward More Human Interface Design. *IEEE Computer*, 34(3) (2001), 56-64.

[10] Labahn, G., Lank, E., MacLean, S., Marzouk, M., & Tausky, D. MathBrush: A System for Doing Math on Pen-Based Devices. *The 8th IAPR International Workshop on Document Analysis Systems* (2008), 599-606.

[11] LaViola, J. & Zeleznik, R. MathPad2: A System for the Creation and Exploration of Mathematical Sketches, *Proc. of SIGGRAPH 2004*, ACM Press 23(3), (2004), 432-440.

[12] Kurtenbach, G. & Buxton, W. 1994. User Learning and Performance with Marking Menus. In *Proc. of CHI-94*, ACM Press, (1994) 258-264.

[13] Saund, E. & Lank, E. 2003. Stylus Input and Editing without Prior Selection of Mode. In *Proc. of UIST-03*, ACM Press (2003), 213-216.

[14] Stahovich, T.F., Davis, R., & Shrobe, J., Generating Multiple New Designs from a Sketch. *Artificial Intelligence*, 104 (1998), 211-264.

[15] Taele, P. & Hammond, T. Hashingo: A Next-Generation Sketch Interactive System for Japanese Kanji, In *Proc of IAAI* (2009).