conduit!

Volume 10, Number 1 | Department of Computer Science | Spring, 2001

Brown University

STATISTICAL PARSING FOR SPEECH RECOGNITION, or HOW I CAME BACK FROM A WORKSHOP WITH SOMETHING NEW TO WORK ON

Eugene Charniak

It is a common saying in academic computer science (and I suspect in most research areas) that you go to a conference not to hear the talks but to talk with other researchers in your area. We say this even though a lot of what we talk about is the usual academic gossip—who got a job where, how the industry is doing, etc. In fact, very important scientific information is being passed around, but this would not be obvious to a listener; I think even the scientific participants might not immediately recognize exactly what it is that they are getting out of these conversations, aside from the dirt about so-and-so whose spouse left him/her for...oh, never mind!

I have come to realize that what is being passed around is not so much specific information as world views. What are the problems that people see as important? How important is one line of research compared to another? Why does a particular researcher spend time working on X when to me, at least, it does not seem to be going anywhere?

All of this became clearer to me after a recent workshop at the Institute for Mathematics and Its Applications at the University of Minnesota. The conference was intended to bring together three groups of people who have overlapping interests but seldom attend the same standard academic conferences. I am a member of the "computational linguistics" clan. As a group we are interested in applying computers to problems in language. Most of us would see as our ultimate goal getting a computer to understand language. It is also relevant that I am a "statistical language processing" person. That is, in my research I use computers to gather statistics from large bodies of text ("corpora") and then write programs that use those statistics to process new text. Of particular importance here is the work I have been doing on statistical parsing, of which more later.

"I am a member of the 'computational linguistics' clan. Most of us would see as our ultimate goal getting a computer to understand language"

The second group of researchers were statisticians. Researchers in statistics have, among other things, developed a large body of tools for learning things accurately from repeated "trials", as they might put it. From their perspective, a "trial" might be a sentence and thus their techniques might let someone like me improve how I gather the statistical information my programs

Brown University, Box 1910, Providence, RI 02912, USA

need. As I left for the workshop, I was hoping to pick up some tips I could apply to my research.

The third group was the speech-recognition community. Formally, I suppose, one might put them in the computational linguistics community, since they too are interested in getting computers to understand language, in their case spoken language. Practically, however, only a small fraction of the communities overlap. There are two reasons for this. First, starting about twenty years ago, the speech-recognition community adopted statistical methods for their research, while the influence of these techniques within the rest of the computational linguistics community has been more recent. I started using statistics about nine years ago, and I was a comparatively early adopter. The second reason is that many of the concerns of the speech-recognition community, particularly the intricacies of how the human voice makes words, are unique to them.

There is, however, one particular point of potential overlap. All modern-day speech-recognition systems have a "language model"—a program that assigns to any string of words the probability of that string in the language. Thus "I went to the store" would have a relatively high probability, while that of the string "store the to went I" would be very much lower. Such language models are needed because often it is very difficult to decide what was said without knowing the context. Or, to put it another way, if you splice out a single word from a tape and ask someone to tell you what the word is, people cannot do very well, whereas with the complete sentence there would be no difficulty at all. The individual words help support one another because we implicitly know which combinations are reasonable and which are not. For example, in normal speech, when people are not paying particular attention to enunciation, the differences between "a" and "the" can be negligible. However, in the context "They appealed to a/the Supreme Court" we would have no trouble. The idea then is that one can combine the language-model probability, which tells us how well the words fit together, with the acoustic model, which tells us how well the acoustic signal supports the hypothesized words, in order to get what one hopes is the correct overall interpretation.

For many years now the speech-recognition community has been using the so-called "trigram" language model. In this model one estimates the probability of a word coming next in the string based on the two previous words. So in the above example we would ask what the probability is of seeing "a" (or "the") after "appealed to" and what the probability of "Supreme" after either "to a" or "to the". Presumably the probability of seeing "Supreme" after "the" would be higher and the correct decision would be made. Creating such a language model is tricky, but the basic idea is that one collects a large amount of text (or even better, a large corpus of transcribed speech) and gathers the statistics.

The trigram model works remarkably well. Indeed, there are constant rueful comments from the computational linguistics community about such a stupid technique working so well, while so far "smarter" techniques have pretty much failed to improve upon it. However, at this workshop the speech-recognition people were showing the limitations they run up against with the trigram model. Some of the examples they put on the board were:

`Too many American citizens fail to vote.` (vs.)

`To many American citizens voting is a chore.`

`Put the paper in/and the file.`

The trigram model works remarkably well. Indeed, there are constant rueful comments from the computational linguistics community about such a stupid technique working so well"

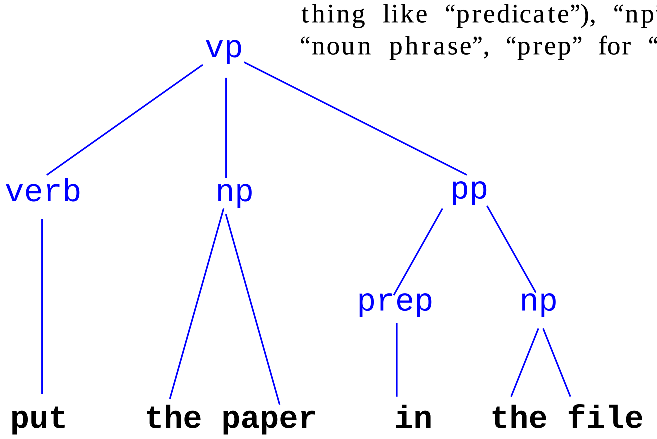
In the first pair of sentences, the first word that can help decide between “too” and “to” occurs four words later, and the differences in the acoustic signal between “too” and “to” are, to put it mildly, subtle. (The words “too” and “to”, of course, are pronounced exactly the same. However, if you try saying the sentences out loud you will find that the rhythm you use will be different—it is the difference between “TOO many American citizens ...” and “to MANY American citizens ...”. But current speech systems are nowhere near sophisticated enough to notice this.) In the third sentence all the three word combinations (which is all the trigram model ever looks at) are perfectly fine, i.e., “the paper in/and”, “paper in/and the” “in/and the file”.

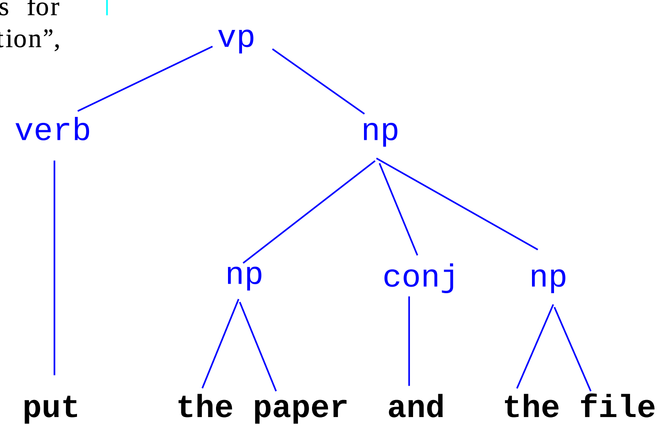
In these cases, however, we would normally think of the wrong word as making the sentence ungrammatical, and thus a program that looked at sentences in terms of grammar might have a good chance of picking the correct word. More formally, most statistical parsers work by assigning a probability to each sentence/parse combination. By a “parse” I mean some structure that elucidates the syntactic structure of sentences. It is like what I did in grammar school when I was told to draw a line between the “subject” and the “predicate” of a sentence (e.g., between “The dog” and “ate the steak”) and then to draw a half-line between the verb and the direct object (“ate” and “the steak”). A parser does the same thing but in more detail, and the results are expressed in a tree structure. So, for example, the parses for the two sentences “Put the paper in/and the file” are shown below.

Here “vp” stands for “verb phrase” (something like “predicate”), “np” stands for “noun phrase”, “prep” for “preposition”, and so on. The parser (in principle) finds all possible parses for the sentence and adds up their probabilities, and the result is the probability for the sentence. In practice these parsers find only a subset of the parses, and instead rely on techniques that find high-probability parses early on, so that they can ignore the very low-probability interpretations for a sentence. Because natural languages are so flexible, there are zillions of parses for moderate-length (20-word) sentences, but almost all of them make no sense and thus are given very low probabilities, so their contribution to the sum is negligible.

I am not the first person to think of using a statistical parser in this fashion. There are two very good pieces of previous work in this area, one by colleagues at Johns Hopkins (where I spent my sabbatical three years ago), and more recently by Brian Roark, a graduate student here at Brown in Cognitive and Linguistic Sciences. In both cases they found that their language model based upon a statistical parser performed slightly better than the trigram model.

Naturally I had known about this work prior to this workshop, and I also suspected that, since my parser was more accurate at parsing than either of the other two, there was a reasonable chance that it would perform better as a language model as well. Yet somehow, without the immediate contact with people to whom this was a pressing issue, it did not seem an important thing for me to work on, particularly since, for reasons I do not want to get into here, my parser would require significant revisions to make it applicable to this problem.





Let me now explain how statistical parsers work so you have some idea of how they might assign reasonable probabilities in the above cases. All statistical models of complicated real-world phenomena must make what we call "independence assumptions." So the trigram model assumes that the probability of a word depends only on the last two words and is independent of all of the remaining previous words. In current parsing models we first need to assume that the probability of a sentence is independent of the sentences that came before. We also make more complicated assumptions about which words can be ignored in computing various probabilities. Naturally, for both the trigram model and our parser, the assumptions are wrong. However, we have to make them because otherwise there would be too many different situations for us to gather reasonable statistics on. The goal in statistical language work is to make the required independence assumptions as natural as possible.

Statistical parsers assign probabilities to particular parses by multiplying together individual probabilities for everything that goes into the parse. For example, consider the noun phrase "the paper" in our sample sentence. The parser would ask, what is the probability that a noun phrase appearing as the direct object of the verb "put" would have as its main noun "paper"? What is the probability that, if the noun phrase has the main noun "paper", it will have the form "determiner noun"? How likely is "the" to be a determiner modifying "paper"? and so on. Of course, none of these probabilities are going to differ between the two parses, since the noun phrase "the paper" appears in both of them.

If you think about the two sentences you will soon realize that the intuitive reason that "in the file" sounds better than "and the file" is that the verb "put" almost always requires a prepositional phrase after the noun phrase to indicate the place where the thing should be put. The first of the two parses has this prepositional phrase because in this sentence the hypothesized word is "in", which is a preposition, whereas the second parse does not because the alternative is "and", which is a conjunction. Thus in the second sentence the parser hypothesizes that there is a conjoined noun phrase "the paper and the file". If we put this in

Max Salvas installs a cooling fan in one of the PCs he's assembling from commodity components. Says Director of Computer Facilities Jeff Coady, "Max is amazing at building or fixing systems. He is fearless when it comes to replacing boards and he hasn't fried one yet! There isn't much electronic or mechanical that he can't decipher, take apart, repair and put back together—from a toaster-oven to a mainframe computer." This project is a cost-effective way to supply the department with PCs that can be upgraded quickly and thereby more closely track enhancements in component technology. To identify Max's completed machines, a special 'Max Built' logo (above) has been designed.

Max Built logo by RI cartoonist Rick Billings terms of the probabilities that my parser considers, it comes down to this: what is the probability that a verb-phrase whose main verb is "put" will end with a noun-phrase, or instead will have a prepositional phrase following the noun phrase? My parser gathers statistics like that from a large "training corpus" of about a million words of human-parsed newspaper text (from the *Wall Street Journal*). According to its statistics, the probability of ending with a noun phrase is 0.029, whereas the probability of a following prepositional phrase is 0.8, almost thirty times larger. (If you are wondering how it is that even 3% of the verb phrases do not have a prepositional phrase, it is because another use of "put" does not use the pp: "No matter how you put it, it sounds fishy.") Thus, as you might expect, the sentence with "in" is given a much higher probability than the sentence with "and". Naturally, the parser never saw this particular sentence in its training examples, or even a sentence where the direct object of "put" was "the paper". However, when building the parser I assumed that the probability of seeing a prepositional phrase in a verb phrase, while dependent on the main verb of the verb phrase, was independent of all of the other words.

"The trigram perplexity for both my trigram model and those in the earlier papers...was about 167. The grammar perplexities for the two previous grammar models were 158... and 152... . My statistical parser had a perplexity of 131, which suggests that it should make a very good language model"

So, you ask, how well does my parser do? The short answer is very well, thank you. To get more precise I need to tell you something about how things are measured in this field.

Intuitively, we want a language model to assign high probability to sentences that are likely to occur and very low probability to sentences that are bad for some reason and thus unlikely to occur. One way to do this is to take a corpus of English that you assume is reasonably representative and see what probability the model assigns to the corpus. That is, take each of the sentence probabilities, multiply them together; the higher the result is, the better your language model. (Notice that the we do not need a corpus of "bad" sentences to check that they get low probability. If all the good ones get high probability, then the bad ones must be low, since the total over all sentences must sum to one.)

This is, in essence, exactly what we do, except that because the resulting numbers come out so small we modify the procedure first to take the *n*th root of the overall probability, where *n* is the number of words in the corpus. The idea is that since probabilities are typically combined by multiplying them together, the *n*th root gives us the average probability for each word. Finally, we take one divided by this probability. So if the average probability for a word is, say, .005, then we report the number 200. One way to think of this number is that, on average, guessing the next word is as difficult as guessing which of 200 doors has the prize behind it. The resulting number is called the "perplexity" of the corpus according to your language model. The smaller the perplexity, the better the language model, since it is making plausible words more probable, thus is assigning higher probability to good sentences, and thus will give better guidance to the speech-recognition system when it tries to decide between words that the acoustic system finds confusing.

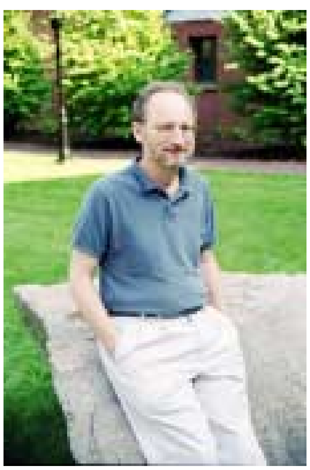
In the experiments I took the training corpus that had been used by prior researchers and trained (collected statistics for) both a trigram model and my parser. I then ran my parser over the same test corpus that had been used in earlier research and computed the perplexity. The trigram perplexity for both my trigram model and those in the earlier papers I mentioned above was about 167. The grammar perplexities for the two previous grammar models were 158 (for the Johns Hopkins researchers) and 152 (for the paper by Brian Roark here at

Brown). My statistical parser had a perplexity of 131, which suggests that it should make a very good language model indeed.

However, do not expect to see my parser incorporated into your favorite speech-recognition system soon. Several problems deserve mention. The first is that parsing is quite time-/space-consuming, so machines will have to speed up a good bit before parsers are practical for everyday speech recognition. Second, remember that current statistical parses need a training corpus from which to gather their required statistics. If the way you use words is quite different from the training corpus, the language model might not work very well for you. Or then again, maybe this will not be a problem. In fact, we simply don't know yet.

Finally, there is one problem that is unique to my parser and is tied up with why, I believe, it was able to outperform previous attempts. You may remember that in the "Put the paper in/and the file" example I noted that verb phrases headed by the verb "put" almost always require a prepositional phrase. More generally, my parser (and all the best statistical parsers) works on the principle that it should always condition probabilities on the head word of the phrase that the program is working on. This creates problems for speech recognition, however, in that speech systems like to work from left to right, following the time sequence of the words coming in. Unfortunately, sometimes the head word of a phrase comes in the middle or end of the phrase—noun phrases are the most common example, where usually the head noun is the rightmost (e.g., "key" in "the small room key"). Thus the earlier language parsing models did not use this principle, and I believe suffered because of it. There are several ways this slight incompatibility with the dictates of speech recognition can be overcome, and I am hoping that my parser's good performance will encourage research in this area.

THE 26th IPP SYMPOSIUM: 'WEB TECHNOLOGIES'


Host Eli Upfal

On November 2, 2000, we held our 26th Industrial Partners Program Symposium on "Web Technologies," featuring speakers from six leading companies in the Web industry. The talks covered a variety of Web-related technology issues ranging from search engines and user interfaces to security and content delivery.

The first speaker was **Prabhakar Raghavan, Chief Scientist and VP for Emerging Technologies at Verity, Inc**. He is better known to many of our students as the coauthor of a theory textbook on randomized algorithms used in CS155. Prabhakar was a researcher and manager in IBM's Research Division and only recently moved to an executive job at Verity, a small company specializing in software for intranet applications (intranets are Internet networks connecting users and servers inside a company or an organization). His talk on "Networks and Sub-Networks in the World Wide Web" discussed several structural properties of graphs arising on the Web, including the graph of hyperlinks and the graph induced by connections among distributed search servants. He presented a number of algorithmic investigations of the structure of these networks, and concluded by proposing stochastic models for the evolution of these networks.

The second speaker was **Tom Leighton, Chief Scientist, Akamai Technologies**. Tom was one of the two founders of Akamai, one of the most successful Web-related startups in recent years. His background is also in theory: when not involved in Web business, Tom is a professor in MIT's applied mathematics department and the head of the algorithms group at MIT's Laboratory for Computer Science. Akamai's business is to provide fast access to the content of its customers' Web sites. Although users may not notice this, when accessing popular sites such as Yahoo! or CNN.com the content is actually delivered to the user from Akamai's servers, not from the original sites. Tom gave an overview of the

Symposium speakers from l to r: Ted Tracy, Latitude Communications; Tom Leighton, Akamai Technologies, Inc.; Andrei Broder, Alta Vista Company; Stephen Uhler, Sun Microsystems; Eli Upfal, Brown; Prabhakar Raghavan, Verity, Inc.; Olin Sibert, InterTrust Technologies

complex technology involved in fast content delivery and discussed some interesting research questions related to further improvement of the process. It was also fascinating to hear a first-hand account of the two years that moved an abstract idea in theory to a successful multimillion-dollar company.

The last morning speaker was **Ted Tracy, Vice President for Product Development at Latitude Communications**, who talked about "Web Collaboration" and gave an overview of Latitude's main business, "Web-conferencing" technologies, i.e., tools and applications for group collaboration across Web networks. Web conferencing allows multiple remote participants to exchange voice, data, and video information across IP networks for general-purpose team collaboration as well as specific vertical applications. As the Internet continues to improve in terms of both total bandwidth and "quality of service" mechanisms,Web conferencing will become even more pervasive. Ted described Latitude's flagship MeetingPlace, which provides a powerful solution to the need for improved professional worker productivity and associated collaboration.

Following the usual excellent lunch and fancy cakes, the first afternoon speaker was **Stephen Uhler, Researcher, Web Applications Technologies, Sun Microsystems**. Steve described the "Brazil Project" on tools to integrate applications and data across multiple platforms on the Internet. Brazil provides components for generating content, management of requests, content transformation with tools such as TCL, Python and a scripting language, handling special-purpose devices and services, such as SmartCards and ActiveX/Com connectivity, and utilities, such as regular expression processors and proxy services. Brazil's components could be used to build applications involving the collection, filtering, and processing of data from multiple sources in potentially different formats and the reformulation and display of the data on Web pages.

The next speaker was **Olin Sibert, VP Strategic Technologies, InterTrust Technologies**, who talked about "Digital Rights Management in the Era of Napster." The modern digital world, in which computation and communication are (almost) free and (almost) unlimited, poses critical new challenges for management of rights and information. For example, there is much hullabaloo today about "the end of intellectual property" (postulating a world where information is completely uncontrolled) and "the end of fair use" (the opposite world, where information is tightly locked up). These and similar apocalyptic visions are inspired by an absolutist interpretation of various technologies; in reality, the picture is by no means this simple and clear-cut. The talk discussed the concepts and mechanisms of Digital Rights Management (DRM), InterTrust's solution to securing content and transactions, and how it can act as a moderating factor in such visions.

The last talk of the day was by **Andrei Broder, VP Research and Chief Scientist, Alta Vista Company**, who talked about "Trends in Search Technology." Andrei was a researcher at Digital's (later Compaq's) SRC Lab in Palo Alto, working on theory of algorithms and in particular randomized algorithms and analysis. SRC was also where the search engine Alta Vista originated. Andrei was fascinated by the complex algorithmic questions related to this project and when Alta Vista spun off from Compaq, Andrei joined as the leader of research and development. The challenge of the Web search industry, as described by Andrei, is to "meet the users' needs, given their poorly made queries and the heterogeneity of Web pages." Search technology is ubiquitous on the Web: from the major search engines that index hundreds of millions of pages to the tiniest e-commerce site, there is a search box on every site. These boxes are powered by a vast array of methods of varying sophistication, combining classic information-retrieval and linguistics techniques with Web-specific data and algorithms. On the other hand, users increasingly expect and actually receive a substantially uniform interaction style—basically unstructured, full-text search—no matter what search box they are using. The talk explored some of the technology trends and business developments that make this search paradigm so prevalent and powerful.

Peter Wegner, Professor Emeritus, received the ACM's Distinguished Service Award at a ceremony in San Jose on March 11.

"For many years of generous service to ACM and the computing community, including outstanding and inspiring leadership in publications and in charting research directions for computer science."

During his distinguished research career Peter Wegner has written or edited over a dozen books in the areas of programming languages and software engineering. For decades, he has been an initiator in ACM's educational and publication efforts, performing invaluable service to innumerable readers, researchers, practitioners and students. As editor-in-chief of ACM Press Books (1987-1992) and the *ACM Computing Surveys* (1995-1999), Dr. Wegner demonstrated innovative leadership and the ability to inspire and motivate others. In both of these editorial positions he innovated and improved the publications substantially, reaching out with originality, energy, and good taste.

Dr. Wegner has an exemplary history of service to the computing community and his efforts have inspired several generations of computer scientists. He was a leader in charting research directions for CS who continually found new ways of focusing the field's intellectual energy, helping to shape research agendas and funding programs. His commitment to CS research and publishing was matched by his commitment to education, and his focus on the state and the direction of education provided an immense contribution. At a time of splintering and specialization in CS, Peter provided an integrative force to the computing community.

LOOSE, ARTISTIC 'TEXTURES' FOR VISUALIZATION

Reproduced from IEEE Computer Graphics and Applications, *March/April 2001.*

In the November/December 2000 issue of *IEEE Computer Graphics and Applications,*[1] Vicki Interrante posed a visualization problem that she and I have been interested in for several years. The problem is that of visually representing a 2D field of data that has multiple data values at each point. For example, 2D fluid flow has a vector value at each location and derived values are often available at each location. Interrante suggests using natural textures to attack this problem, because the textures can potentially encode lots of information. She provides some intriguing examples and proposes a psychology-based approach for developing an understanding of how we perceive natural textures, like those that Brodatz[2] photographed. Understanding this can help us build better visualizations.

David Laidlaw

Based on Interrante's suggestions, I would like to posit and explore what is, perhaps, a less well-defined approach. Through evolution, the human visual system has developed the ability to process natural textures. However, in addition to natural textures, humans also visually process man-made textures—some of the richest and most compelling of which are in works of art. Art goes beyond what perceptual psychologists understand about visual perception and there remain fundamental lessons that we can learn from art and art history and apply to our visualization problems.

The rest of this article describes and illustrates some of the visualization lessons we have learned studying art. I believe these examples also illustrate some of the potential benefits of further study. While this approach is more open-ended than a perceptual psychology approach, both approaches are worthy of pursuit, and the potential benefits of using the less structured approach outweigh any risk of failure.

How Humans See and Understand

Scientific visualization, a term coined only a little over ten years ago, is the process of using the human visual system to increase our understanding of phenomena studied in various scientific disciplines. While the term is young, the process (modulo the computer) has been used since the beginning of science. Many scientists have created drawings or built 3D models to understand and communicate their science. The history of science and art can provide us with lessons we can apply to using computers effectively. Over time, artists have developed techniques to create visual representations for particular communication goals. Art history provides a language for understanding and communicating that knowledge.

Historically, the two disciplines approach the human visual system from different perspectives. Art history provides a phenomenological view of art—painting X evokes response Y. Art history, however, doesn't deconstruct the perceptual and cognitive processes underlying responses. Perceptual psychology, on the other hand, strives to explain how humans understand those visual representations. There's a gap between art and perceptual psychology—we don't know how humans combine visual inputs to arrive at the responses art evokes. Shape, shading, edges, color, texture, motion, and interaction are all components of an interactive visualization. But how do these components interact and how can they most effectively be deployed for a particular scientific task? Answers to these questions are likely to fill some of the gap between art and perceptual psychology.

1. V. Interrante, "Harnessing Natural Textures for Multivariate Visualization," *IEEE Computer Graphics and Applications,* vol. 20, no. 6, Nov./Dec. 2000, pp. 6-10.
2. P. Brodatz, *Textures*: *A Photographic Album for Artists and Designers*, Dover, New York, 1966.

As an example, the human-computer interaction (HCI) community is using and extending knowledge about perception to test and develop better user interfaces. We can find analogous inspiration for improved methods for scientific visualization in the gap between art and perceptual psychology. Many of these lessons will impact the visual representation of multivalued data.

Looking Up from our Monitors

A number of times over the last few years I've shepherded my students to art museums for guided tours by my artist collaborator davidkremers, the Caltech Distinguished Conceptual Artist in Biology. After initially searching for scientific visualization inspiration in art, these visits let us formulate a plan for finding and applying the concepts. Our initial focus was on oil painting, particularly from the Impressionist period, because these paintings are so visually rich. The multiple layers of brush strokes in the paintings provide a natural metaphor for constructing visualizations from layers of synthetic "brush strokes." Some of my colleagues look at me askance when I describe our research field trips, as if to say, "This is research?" But stepping out of the lab helps students build a new picture of what they can accomplish when they come back to the computer. It trains their eyes and minds to see differently.

During these field trips, we have studied, in particular, the works of three painters.

• Van Gogh, whose large, expressive, discrete strokes carry meaning both individually and collectively.

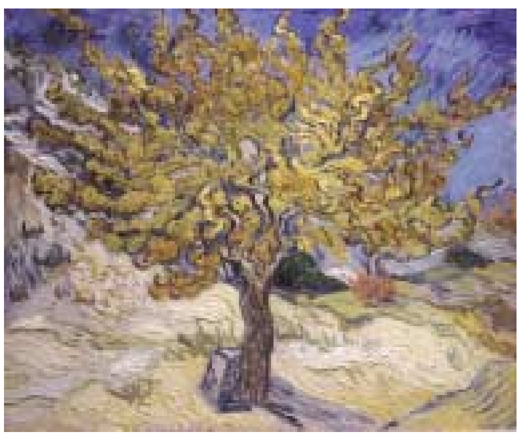
• Monet, whose smaller strokes are often meaningless in isolation—the relationships among the strokes give them meaning, far more than for van Gogh.

• Cezanne, who combined strokes into cubist facets, playing with 3D perspective and time within his paintings more than either van Gogh or Monet. His layering also incorporates more atmospheric effects. In a sense, his work shifts from surface rendering toward volume rendering.

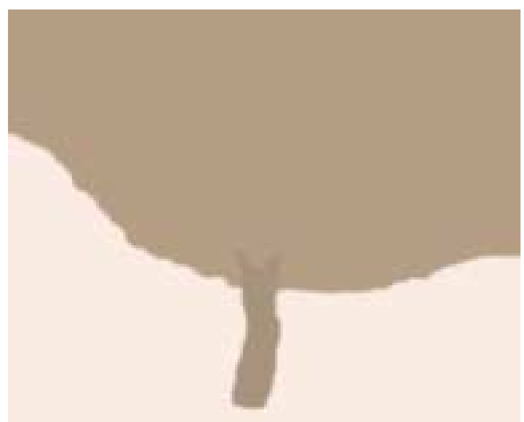
The three artists' work in this sequence builds in complexity and subtlety. In our field trips we studied all three, but most of our experiments thus far are limited to ideas we learned from van Gogh's work.

Van Gogh introduced us to the concept of underpainting, or laying down a rough value sketch of the entire painting. The underpainting shows through the overlying detailed brush strokes to define the anatomy of the painting. Fig. 1b shows underpainting for Fig. 1a. It divides the canvas into two parts—a primed lower region of hillside, rocks, and ground cover, and a darker upper region of tree, sky, and distant hills. Underpainting helped us present some overall parts of our data. We found that an analogous underlying

(1a)



(1b)



1(a) *Van Gogh's* The Mulberry Tree *(1889, oil on canvas) illustrates the visual shorthand that van Gogh used with his expressive strokes. Multiple layers of strokes combine to define regions of different ground cover, aspects of the hillside, and features of the tree.* ***(b)*** *An underpainting shows the "anatomy" or composition of the scene in broad strokes. (Image of* The Mulberry Tree *granted by the Norton Simon Foundation, Pasadena, California. Gift of Norton Simon, 1976.)*

form in our visualizations anchors and literally gives shape to disparate data components. Outlines around regions provide separation and emphasis and lend definition to our sea of data.

In van Gogh's *The Mulberry Tree* (1889, oil on canvas), brush strokes represent the solid trunk of the tree, bending branches, leaves blowing in the wind, and tufts of grass. We learned that there are many shorthand ways of depicting complexity using icons, geometric shapes, or textures that evoke a characteristic of the subject or the data—and with that comes the responsibility of choosing brush strokes that don't create opposing or unwanted secondary impressions. Beyond this direct representation, they also invite the viewer to experience the scene, not just view it passively. Similarly, brush-stroke size and proximity depict density, weight, and velocity. In our visualizations, we wanted to capture this marriage between direct representation of independent data and overall intuitive feeling of the data as a whole.

Back in the Lab

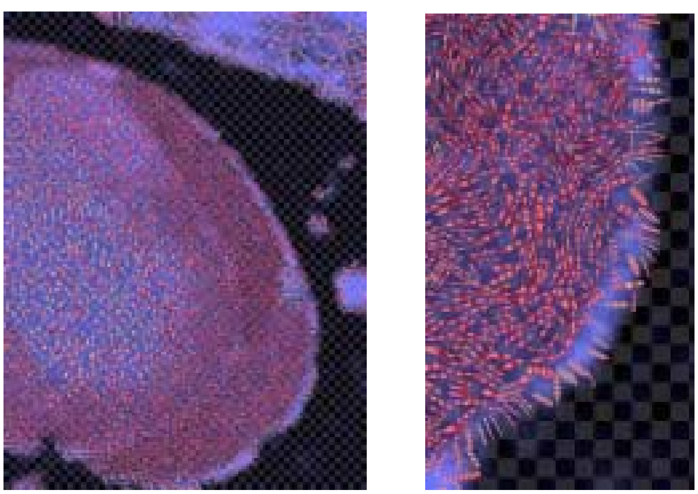
Returning to our computer lab, we tried to use some of the ideas we had gleaned, once again drawing mostly from van Gogh's work. We experimented with brush stroke shapes and ways of layering them. Our initial attempts were free-form and produced interesting results. Our next attempts were more directly applied to scientific problems. We show two of the images we generated in Figs. 2 and 3. The problem-directed approach led us to iconic-looking strokes.

In Fig. 2, we show one 2D slice of a 3D second-order tensor field, which has about six different data values at each point in the image. The image shows the right half of a section through a mouse spinal cord. To create the visualization. we used a layer resembling an underpainting with ellipse-shaped strokes on top of it. On each of the strokes, a texture represents more of the data. For more details on the scientific interpretation and the visualization, see Laidlaw et al.[3]

In Fig. 3, we show 2D fluid velocity together with a number of derived quantities. About nine values are represented at each spatial location in this visualization. We again used a layer resembling an underpainting with layers of ellipse, wedge, and box strokes on top. The ellipse strokes have a subtle texture superimposed. More details on the visualization appear in Kirby et al.[4]

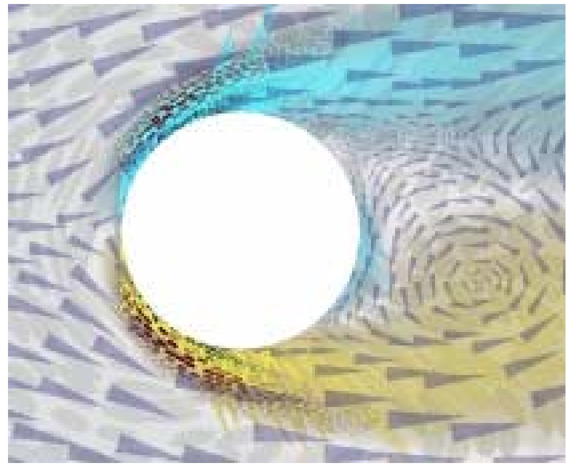
Space

We learned that paintings (and, in some cases, visualizations) are multiscale. They can be viewed from different distances and seen and understood differently. This raises interesting issues about the definition of texture. Let's consider van Gogh's *Mulberry Tree* (Fig. 1a). From a few inches away (look closely at Fig. 4b) you can see shapes from the bristles of the brush as well as colors mixed within a stroke. At this distance, the shape and color features might be considered texture, but they could also be interpreted individually. At a distance of 18 inches (Fig. 4b at normal reading distance of 18



2 *Visualization of half a section through a mouse spinal cord. The data is a symmetric 3D second-order tensor field, with the equivalent of six independent scalar values at each point. The detail on the right shows the lower right part of the section*

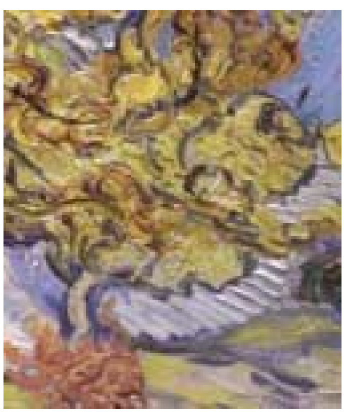
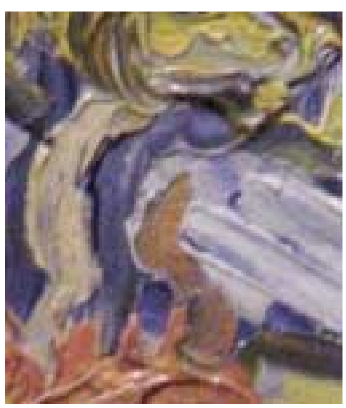
3. D.H. Laidlaw et al., "Visualizing Diffusion Tensor Images of the Mouse Spinal Cord," *Proc. Visualization 98*, IEEE Computer Soc. Press, Los Alamitos, Calif., 1998, pp. 127-134.
4. R.M. Kirby, H. Marmanis, and D.H. Laidlaw, "Visualizing Multivalued Data from 2D Incompressible Flows Using Concepts from Painting," *Proc. Visualization 99*, IEEE Computer Soc. Press, Los Alamitos, Calif., 1999, pp. 333-340.



3 *Visualization of 2D fluid velocity together with several derived data values. Approximately nine values are represented visually at each point in the image*

inches), these features appear smaller and resemble a texture on each stroke. The strokes themselves are still individual. At a distance of five feet (Fig. 4a), the strokes merge together to appear more like a texture. Finally, at 15 feet (Fig. 1a), the strokes blend together and become almost invisible individually.

We can use this lesson by encoding different information at different scales. Iconic information at one scale can turn into texture information at another scale. With care, we can design features at different scales into the same images. In the scientific visualizations of Figs. 2 and 3 we designed visual features at different scales. The texture on strokes is at a much finer

(4a)  **(4b)** 

4 *Variances in viewing van Gogh's* Mulberry Tree*. Viewed in this article from about 18 inches, Fig. 1a shows what you would see 15 feet from the painting. Comparatively, Fig. 4 shows the following: (a) a detail of what you would see 5 feet from the painting, and (b) a detail at actual size (what you would see from 18 inches). Look at (b) more closely for viewing distances less than 18 inches*

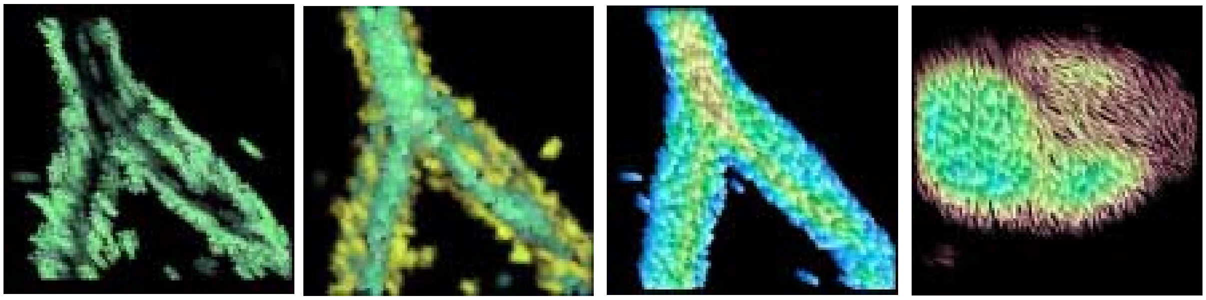
scale than the strokes themselves, and the dark box strokes of Fig. 3 are at a different scale than the other strokes.

To take full advantage of the multiscale nature of paintings and visualizations we have to have ways of interacting with them—that is, ways of changing our viewing distance. We rarely change the distance from which we view our monitors—only a bit more frequently do we do so with paper publications. That's why the same image is shown at different scales in Fig. 4 and in some of the other detailed figures. However, we do view images hung on a wall from different distances. And some images on paper—often artistic—inspire that sort of study. Projection systems like PowerWalls and CAVEs may be good options for encouraging this sort of exploration, as may other hangable large-format output media.

Time

We also learned that paintings (and, in some cases visualizations) have a temporal component. For instance, we see different aspects of an image at different viewing times. Some parts stand out quickly, like the overall composition or palette of a painting, and some take more time to become apparent, like the texture or shape of individual strokes. The scale and speed of recognition correlate, as do contrast and speed of recognition—but these are not the only factors.

To use this lesson, we can design our visualizations so that important data features are mapped to quickly seen visual features. For example, features that we want



5 *Loose texture examples*

to measure directly from an image are present for detailed study but don't intrude on the visualization's initial impression. The multiscale examples from the "Space" section illustrate this temporal impact. Fig. 3 gives another example: we can read the wedges more quickly than the ellipses because of a difference in contrast.

Studies of preattentive vision and knowledge about low-level vision are useful for designing the quickly seen visualization. It's more difficult to test the more slowly seen parts, which makes it more difficult to design them. Task-oriented experimental tests seem logical, but the tasks are often so complex that the performance variance is relatively high, making methods difficult to compare.

Our Initial Experiments

Our initial experiments were much looser than the examples shown in Figs. 2 and 3. Some examples in Fig. 5 show 2D or 3D fluid flow. Since I want to emphasize the overall texture and visual qualities, I won't go into detail about the mappings for each. To many, the images are visually compelling, yet it has been difficult to extract concrete visualization lessons from them beyond those I previously described. What people see in these images includes not only the mappings that were used for the data value, but also other visual characteristics. Despite being 2D, some images give an overall sense of depth. Some of the strokes appear to layer, like feathers or scales. One of our challenges with these looser images is in understanding what works, what doesn't, and (we hope) why.

Closing Thoughts

I've tried to illustrate some examples of looking toward art for inspiration in creating visualizations. Here we feature van Gogh and mention Monet and Cezanne for context. In your artistic searches, choose the artists in whom you have a passionate interest. I believe that any artist has lessons to offer to visualization.

Working on scientific visualization problems, we already interact with scientists and adopt their problems. As toolsmiths, we do better computer science through addressing scientists' problems on scientists' terms.[5] Similarly, we benefit from critical feedback from artists, despite the difficulty of creating and maintaining these relationships. I try to look at and understand art—early and often—and emulate it in scientific visualizations and get critical feedback from artists. I explain what I'm trying to do visually and have artists critique it. Then I iterate, iterate, and iterate.

> *"we benefit from critical feedback from artists, despite the difficulty of creating and maintaining these relationships. I try to look at and understand art and emulate it..."*

Of course, the scientists must be involved in this iterative process. Artists can help with inspiration and feedback on the visual and communicative aspects of visualization, but scientists define the tasks performed and therefore must ultimately

5. F.P. Brooks, "The Computer Scientist as Toolsmith II," *Comm. ACM*, vol. 39, no. 3, Mar. 1996, pp. 61-68.

evaluate the success of the methods. For instance, the fluid flow example in Fig. 3 may be aesthetically pleasing, but without explanation—perhaps via a legend or key—it's not scientifically useful.

Fig. 3 displays as many as nine values at each point of the image. With some research indicating that texture has roughly three independent dimensions, the ability to represent nine values is somewhat surprising—perhaps it's due to combining color with texture or layering textures at different scales.

Texture is hard to define. Understanding black and white natural textures like the photographs in Brodatz is a good start, but we also need to look broadly. Task-oriented user testing may help, and perhaps we can use the critiques that are part of artists' training. This might combine perceptual psychology and art to fill in the gap in our understanding of how humans see. By having artists cognitively analyze what is shown by more complex textures, we might come to a consensus on what works, what doesn't work, and why it does or doesn't work in the context of art and art history.

Acknowledgments

I'd like to thank my collaborators—in particular, artist davidkremers—as well as Clifford Elgin, Mike Kirby, and Matthew Avalos. I also want to thank Victoria Interrante and Barbara Meier for their encouragement and ideas. The NSF (CCR-9619649, CCR-0086065) and the Human Brain Project (NIMH and NIDA) supported some of the work described.

CS HOSTS INAUGURAL NEPLS SYMPOSIUM

Numerous New England academic and industrial groups have interests in programming languages and systems. This means many talks of broad interest take place at diverse locations and times, with no single venue where researchers in the area can gather to hear about recent results and work in progress.

The New England Programming Languages and Systems (NEPLS) symposium series was founded to create a periodic regional venue. It draws direct inspiration from a similar series held in the greater New Jersey area, and from other such events. Its goal is to provide a somewhat informal venue for talks and demos, where emerging work can receive feedback and younger researchers can hone their skills.

The first NEPLS event was held at Brown University on December 7, 2000 and was organized and hosted by Shriram Krishnamurthi. The 60-odd attendees represented universities, research labs and industry. Participants came from as far afield as SUNY Stony Brook, IBM's T.J. Watson Lab, Sun Labs, Williams, RPI and Cornell. The talks ranged broadly, covering topics from call-graph construction through mobile code to program transformation and novel semantics touching on various programming paradigms along the way. The Lubrano conference room and the atrium buzzed with the discussions of researchers and students. By all accounts, it was a successful inauguration.

Numerous institutions have since offered to host meetings. For more information on NEPLS, see *http://www.nepls.org/* (which is hosted at Brown).

Department of Computer Science Industrial Partners

SPOTLIGHT ON INTERNSHIPS WITH CS INDUSTRIAL PARTNERS

To highlight Partners in our Industrial Partners Program and to encourage students to do internships with them, we're initiating a regular column in which we describe students' work experiences with Partners. The WiCS group (Women in Computer Science) has compiled its own database of WiCS members' internships, available at *http//www.cs.brown.edu/orgs/wics/itern.html.*

LATITUDE COMMUNICATIONS

Albert Huang '03

"Last summer, I was a software engineering intern at Latitude Communications along with Brown students Melissa Cheng and my roommate Luke Ma. We were there for about 13 weeks. Since we were the first official engineering interns, we were treated more or less like normal employees. Work was a lot of fun, mirroring the media-described laid-back Silicon Valley atmosphere. Luke and I worked together for the most part on Web application development (SQL, VBasic [ASP], C++,) while Melissa worked on some Mac stuff. We didn't have prior knowledge of most of what we worked on, so we often ended up spending a day or two learning the material before starting."

Luke Ma '03

"As Albert said, we both worked for Latitude Communications Inc. this past summer. They were just starting to formalize their internship recruiting program/plans and whatnot so we were prototypes of sorts. We were treated, work-wise, basically like normal employees. Latitude was extremely hospitable to us throughout our internship.

They gave us an advance on our salary to help us move out there and also resolved some basic logistical issues for us like housing. Transportation we had to find on our own and we eventually had to resort to the much-doubted public transportation system of light rails, buses, and trains.

After we arrived and marveled at our wonderfully oversized furnished apartments, we found the company, which was about a mile and a half up the street. Before we found better transportation, we just took a leisurely 20-minute stroll every morning to the company. Of course, being the new interns and just arriving at the company, we would go in at 8 every morning. That didn't last terribly long. Our bosses, coworkers, administration, and the company in general really didn't mind what sort of schedule we kept. We

l to r: Luke Ma '03, Melissa Cheng '01, Albert Huang '03

would arrive at work anywhere from 8 to 11 and work until quite late. It was definitely more than eight hours of work per day, but most of the time it was because we wanted to work. The freedom in schedule also allowed us to consolidate work and play time. We would work crazy days (on the order of 16 hours) for a while and then take a three-day weekend and learn to sail in Santa Cruz. That particular trip started on a Friday, and the night before I had basically not slept and pulled an all-nighter to get a demo up and running. Wonderfully tiring but fun nonetheless.

Our coworkers were also a blast to work with and be around. Albert, Melissa and I were all part of the "Engineering Interns" group. We ended up doing lots of Web development work using ASP and SQL, also some work on win32 applications and NT services. We basically went in every day and wrote lots of code. The great thing about it all was that the projects we were working on had definite scheduled shipping dates, so we knew we were working on a viable product—that does wonders for motivation. Interestingly enough, most of what we ended up doing over the summer (so far as required skills are concerned) we did not know before we got to Latitude. Our boss just tossed a project at us and told us what he thought about it and how he'd like to approach it. We would then go and research all the technologies needed or call upon the tremendous knowledge base present in our coworkers. Once we knew what we needed to know, we got to work. All in all it was a tremendously educational process made that much better because we had wonderful motivation. While we did our work (say from 10am to 11pm), we'd play Starcraft for a breather in between and go downstairs to the conference room and watch a movie on a projector afterwards.

Eventually, we all had to say our goodbyes and run away, not before our boss tried to get us to skip college and just stay there though! Apparently, he was prepared for the worst when he heard about us, and we turned out to do some decent work. They even had a big party for us toward the end of the summer (took us to see an IMAX film, got some food, and gave us some presents....Albert and I got a scooter!). A good many times we felt almost guilty about how nice they were being to us, our being lowly interns and all, which of course was another great motivation to work. But seriously, the Latitude attitude is one of the best I've seen for workplaces. It's never slow, but manages to have a sense of relaxed purposefulness at the same time.

This past winter break I went back to Latitude to work for a short time. For about three weeks in January, I was back in California with my old boss working on a project to add some load-balancing/ server-farming features to an existing data-conference engine. Again, I learned more in those three weeks than I would have ever expected (lots of C++ code, T120 conferences, server farming schemes, funky stuff). And again, everyone was very friendly."

Melissa Cheng '01

"I worked on the web interface to Latitude MeetingPlace hardware, fixing bugs in the Mac/Solaris versions and adding a main feature. I had a great time working at Latitude. The project was fun, I got involved right away, and my code went into the final product, which shipped a few months after I left. Moreover, the company culture is ideal—especially for interns who are new to living in the area—with laid-back, smart, and friendly people. So if you're willing to ask questions and either have a car or lots of patience with public transportation, you'll love working at Latitude."

COMMUNICATIONS FROM ALUMNI AND FRIENDS

DILIP BARMAN, ScM '92

I just read on the department home page about Jak Kirman's death, and wanted to convey my condolences and share my sorrow with the Brown CS community. Before I came to Brown, I was considering several graduate programs, and Jak (and Kathleen, whom he had just started dating) and his roommate Moi helped to make me feel at home and hosted me in their welcoming apartment for several visits. Jak was very friendly and a great help to me always, whenever I needed help with getting acclimated to the department or Providence, or with emacs wizard questions. I'm sorry I didn't stay in touch with Jak in recent years, and the news saddens me as we've lost a kindly gentleman scholar.
barman@cs.unc.edu

CASSIDY CURTIS, Math ScB '92

To introduce a particularly fascinating website, math major Cassidy Curtis '92 will henceforth be considered an honorary CS major—besides, his stint as a UTA for CS123 and a slew of graphics work confirm that he has paid his dues.

We learned of the telestereoscope ("Eyestilts") and Cassidy's installation at Burning Man via graphics faculty and couldn't resist including a squib about it.

The Telestereoscope is an optical device that increases the distance between your eyes. This has the effect of enhancing your perception of depth. For some people, it makes the world seem miniaturized: cars become toys, and landscapes look like model train sets. For others, the environment deepens and splits into many distinct planes. One viewer described it as "like looking at the real world through a Viewmaster."

The device itself is very simple. It consists of two periscopes, one for each eye. Each periscope is made of a pair of mirrors. The periscopes shift the eyes upwards and out to either side. The principle behind it is based on human visual perception. In order to make sense of your visual surroundings, part of what your brain must do is estimate how far away things are. One of the ways your brain does this is by using the relative disparity between the images projected onto the retinas of your two eyes. Each object in your field of view will project to a slightly different location on each retina. Essentially, the closer the object, the greater that difference will be. Your brain already knows how far apart your eyes are (about two or three inches), and using that information it can make a good estimate of exactly how far away each object is. This is also the principle behind 3-D movies and stereograms. Now, if you artificially exaggerate this relative disparity by placing your eyes several feet apart, your brain, still believing that your eyes are only two inches apart, will come to the wrong conclusion about how far away things are. This incorrect information about depth then interacts with everything else you know about what you're seeing (that is a tree, that is a car, etc.) and you begin to draw strange conclusions about the size of things. In short, because it's hard to believe that your head is

Participants at the Burning Man Festival line up to try the Eye stilts. *Photo, Cassidy Curtis*

really ten feet wide, you are forced to conclude that the world around you is really small.

Cassidy Curtis and Chris Whitney took the Eyestilts to last summer's Burning Man Festival in the Black Rock Desert, Nevada. Burning Man is an annual experiment in temporary community dedicated to radical self-expression and self-reliance held on the 400-square-mile playa, terrain that looks like another planet. Now that we've piqued your interest, you can enjoy the website (and read up on Burning Man too) at *http://eyestilts.com*. If you'd like to check out some of Cassidy's graphics research and other projects, go to *http://www.cs.washington.edu/homes/cassidy.*

NIYU GE, PhD '00

I am now working in the conversational machine group of IBM natural language processing research. The following is a brief summary of what we do (this information is also available on the Watson research website):

Our research in this area spans an entire spectrum of conversational systems technologies, including multimodal dialogue management, natural language generation, and statistical natural language understanding. Our goal is a flexible, free-flow conversational system that places the user in the driver's seat. Simply by speaking naturally in their own words, users will control the application in their own personal style. We are exploring several techniques to construct both universal and application-specific dialogue engines. The emphasis is on scalable dialogue systems capable of simultaneously handling many users, tasks and input modalities (voice, keyboard, gesture and mouse). We have developed prototype systems for many applications, ranging from stock and mutual fund trading systems to phone banking, air travel reservations, and Web-based shopping, in many languages, including English, French, German, Mandarin and Spanish. Regards!
niyuge@us.ibm.com

KEN HERNDON, ScM '96

I've got my own consulting company, a one-man shop at present, called Arpali. This has been active for a little more than two years since I left Sony in NYC. I've been working on various projects, generally with a strong user-interface component. Donna (Miele, AB AmCiv '92) graduated from Law School last May, and passed the NY and NJ bar exams last fall. She's going to be in private practice working with me and with the family real estate business. We've got three children, Armand (6), Paul (4) and Lionel (3), and are expecting a fourth in August.
ken@yellahouse.com

BENOÎT HUDSON, ScB '99

I've been working at NASA Ames since graduating almost two years ago. I've learned plenty here about AI, symbolic arithmetic, software engineering, spacecraft, and the national budget, to name a few. In the fall I'll be going to CMU to start a Ph.D. in the algorithms program. To get in touch, send me email: *bh@techhouse.brown.edu*.

HAGIT SHATKAY, PhD '99

After two years as a post-doc in the National Center for Biotechnology Information at the NIH, where I worked on using information retrieval for gene analysis, I joined the Informatics Research group at Celera Genomics. For those who have followed the human genome sequencing race, Celera is also known as "the private effort." It is a very exciting place right now. The genome sequencing itself amounts to converting a large book from one media format (human cell) to another (computer disk), without looking closely at the actual contents. Given this sequence, the current challenge is to find the genes and their products and, more importantly, to understand what they are doing. This is analogous to actually understanding the semantic contents of the book; needless to say, it is a hard problem. We are working on developing computational techniques, utilizing a multitude of data sources, to find out more about the discovered genes and their effects. There's a lot of room for interesting machine-learning research, and I am looking forward to working on these projects.

Suzi has specifically asked for an update about Eadoh and Ruth, so here it is. Eadoh, our seven-year-old son, enjoys life as a second grader. He has started playing the piano, and he and I have already performed a duet in his first recital. He also swims a lot and, defying my genes, enjoys playing baseball and basketball. He does still miss the New England snow; today was another day in which we were promised a foot of snow but only a disappointing drizzle materialized.

Ruth is now three years old and growing fast. She attends preschool and, like Eadoh, enjoys swimming and playing with all of HIS friends. We are living in a very pleasant neighborhood and Eadoh has made lots of new friends, but still likes visiting his RI classmates. This provides us with a good excuse to take I-95 North and come back for ice cream at Maximilian's. It was a real pleasure to combine our RI trip this last October with a visit and a talk at the department! Will be looking forward to future ***conduit!***s and good departmental news. Best wishes!

hagit.shatkay@celera.com

We were delighted to have Ken Herndon ScM '96 and his family visit the department at the end of February. l to r: Paul, Ken, Donna '92, Armand and Lionel

JACK STANKOVIC, PhD '79

Andy van Dam suggested that I send some info about an award I won for inclusion in the next ***conduit!*** issue.

Professor Jack Stankovic, the BP America Professor and Chair of the CS department at the University of Virginia, has received the second-ever IEEE Real-Time Systems award for "Outstanding Technical Contributions and Leadership in Real-Time Computing." This award was presented at the Real-Time Systems main conference in Orlando in December 2000 and comes with a $1500 cash award. Jack has also been re-elected to a three-year term on the Computing Research Association Board of Directors.

stankovic@cs.virginia.edu

fac.activities@cs.brown.edu

Michael Black. Michael received a gift from the Xerox Foundation's University Affairs Committee to support research on human motion estimation in video sequences. He was also a Program Committee member for the IEEE Workshop on Human Motion held in Austin in December. Also in December he had a paper at Neural Information Processing Systems (NIPS'2000) with colleagues from Stanford and Sweden on learning and tracking cyclic human motions. In November he gave an invited talk at the Robotics Institute at Carnegie Mellon University on the stochastic tracking of humans.

John Hughes. Spike has been chosen as the papers chair for SIGGRAPH 2002 in San Antonio. He went to the first of many planning meetings in mid-January; if you try to reach him in the next 18 months and he's not around...it's probably because of SIGGRAPH. In the meantime, start buying cowboy gear and booking your tickets to San Antonio...

Shriram Krishnamurthi. Shriram's textbook, *How to Design Programs*, co-authored with Matthias Felleisen, Robert Bruce Findler and Matthew Flatt, was published by the MIT Press in February. The book continues to be distributed on-line at *http://www.htdp.org/*. He organized the New England Programming Languages and Systems symposium series by hosting its inaugural event in the department on December 7, 2000 (p. 14.), and he was on the program committee for the Third International Symposium on Practical Aspects of Declarative Languages, which took place in March.

Franco Preparata. As a member of the Scientific Advisory Committee for Mathematics and Computing at Argonne National Laboratory, Franco participated in the review of their technical program. He is also a continuing member of the Gödel Prize Committee. In December he was invited by the Academia Sinica to visit their institutes in Taipei. He has been invited to deliver a plenary lecture on his recent work in Computational Biology by the Institute of Mathematical Statistics at their annual meeting in Charlotte, N.C.

John Savage. John is in his first year as an officer of the Faculty. He serves as Vice Chair this year and will move on to Chair and Past Chair in subsequent years. The officers, who are also members of the Faculty Executive Committee (a central steering committee for Faculty business), are very busy people. Each has a dozen or more scheduled meetings on faculty business per month. John traveled to Japan in late February to give talks at Kyoto and Sendai Universities.

Eli Upfal. Eli was appointed associate editor of the *SIAM Journal on Computing*. He served as the chair of the ACM Doctoral Dissertation Award committee and had a paper on modeling the Web in the 41st Annual Symposium on Foundations of Computer Science in California in November 2000.

Andries van Dam. Andy gave the keynote at a workshop sponsored by the Learning Federation, a non-profit foundation that is exploring the requirements for a grand challenge project in e-learning research for post-secondary science, mathematics, engineering and technology. The forty attendees at the workshop included some of the most prominent leaders in the field of educational research, such as Hal Abelson, John Seely Brown, Ed Lazowska, Don Norman, and Raj Reddy.

CHARNIAK UNPLUGGED

Eugene Charniak

You may remember my mentioning the five-foot-tall versions of Mr. Potato Head that popped up around town a while ago. Each one, it turns out, was designed and sponsored by a private organization. So Fleet Bank did the Mr. Potato ATM that we showed in the last ***conduit!*** Well, if you were not able to come to Providence to see them, you can now have a potato head come to you. The potato heads are coming down and at least some of the sponsoring companies are selling them on e-Bay. So if you hurry you can have one of your very own. The 'Spud of Steel' (aka Mr. Potato Construction Worker), created for the Gilbane Construction Company, was just sold for $1500. Ms. Potato Bishop is on the auction block even as I write. I saw the Episcopal Bishop plugging her potato incarnation on a TV newscast a few days ago. She was quite charming in the role.

Just about the time our last ***conduit!*** was being printed our esteemed chair, Tom Dean, sent around the following e-mail (here slightly abridged):

From: Tom Dean tld@cs.brown.edu>Date: Tue, 14 Nov 2000 13:24:02--0500 (EST)Subject: just for fun: missing pants--a story for a drab day

As some of you may know, Jo and I are in midst of renovation purgatory and have been since late summer. We just completed the carpentry on the new stairs this past weekend—-mahogany newel posts and balusters, oak treads, lots of trim work—-and we were using the evenings to sand, stain and finish the woodwork.

OK, we remembered to bring our clothes downstairs before applying the varnish to the stair treads-—our bedroom is on the top level of the house and the new stairs are the only access. We all remember the lesson of the cartoon character who paints himself into a corner. I did carry a hastily chosen pile of clothes downstairs but somehow my pants slipped out of the pile and when it was time to leave there were no pants available on the accessible levels of the house. Fine, I had been wearing some ragged paint-stained sweat pants and I had some "emergency" pants at the office.

I decided to take a swim before heading to the office. I felt a little funny walking past Charley at the Smith Swim Center wearing a suit coat and sweat pants, but Charley's used to such things. I take my swim, return to the locker room and find... my sweat pants are gone.

I look everywhere but no luck; I look in the lost-and-found for some pants... still no luck. Maybe someone took them by mistake or the custodian threw them away thinking that they were rags. Charley manages to find me a pair of baggy swimming shorts in the custodial rag bag, however, and since they are less revealing than my Speedo swim suit I put them on.

That's it; end of story. Coincidence? I don't think so. As I walked down Thayer St. in the rain with my baggy swimming suit flapping, I felt a little strange until I noticed a lot of students in baggy shorts and sandals. The adults looked a little crooked at me but the students took my look in stride. It dawned on me that perhaps if I dressed like this all the time I wouldn't be asked to serve on any more committees or invited to fancy dos on Power Street. Perhaps the CS faculty would ask me to step down as chair so as to avoid further embarrassing the department by my tacky dressing habits.Perhaps this is a way out of another form of hell...

After reading this, Shriram Krishnamurthi came to ask me if I thought it would be "appropriate" for the faculty to buy Tom the most ridiculous pair of pants

Virtual paint can be dripped out of a real bucket that has a tracker attached to it. This paint can be thrown or dripped onto the walls and floor of the Cave.

A French film crew for Electralis 2001 visited Brown CS recently. Electralis 2001 (*www.electralis.com*), a major exhibit that just opened in Liege, Belgium, on the history and future of electricity, features multimedia presentations on future electrical research topics such as innovations in medicine, physics, and of course the future of the Web and the Internet. Looking for a glimpse of cutting-edge research in computer graphics and human-computer interaction, they interviewed graphics groupies and Andy van Dam. Andy talked at length about the philosophy and social implications of virtual reality (e.g. what is "truth" in a psycho-physical computer-induced experience) in addition to the more general theme of how the technology will evolve and mature. The crew interviewed the following staff/grad students/post-docs—Takeo Igarashi, Cagatay Demiralp, Jean Laleuf, Tim Miller, Anne Spalter. They also saw cave demos from Dan Keefe, Andrew Forsberg, Daniel Acevedo, Cagatay Demiralp, and Song Zhang. They were very impressed by the research and were particularly awed by Dan Keefe's Cave Painting program, seeing it as a major milestone in the evolution of artistic expression through technology.

Much more film was shot than could possibly be used in Electralis 2001 and the producer and director expressed interest in producing a second documentary for French television. There may also be interest in a government-funded cultural project further exploring the new frontiers of technology in artistic expression.

Don't miss the Cave Painting website: *www.cs.brown.edu/research/graphics/research/sciviz/cavepainting/cavepainting.html*

An artist at work

To appreciate the true vileness of these pants, check out the color version of **conduit!** *on our website:* http://www.cs.brown.edu/publications/

we could find. It was clear that Shriram came to me because he wanted to do it and figured if there was any faculty member with no sense of decorum, it was me. Living up to my reputation, I said it was a great idea, and immediately went to our Editor-in-Chief to make sure we would capture the event on film (Suzi still uses analog media). I was not able to attend the faculty meeting at which they were presented, but a while afterward I asked Shriram how it went. He said fine, but that Tom had told him subsequently that he was using the pants quite a bit. Shriram expressed his considerable disappointment at this development. On the other hand, Tom has not, as he implicitly threatened at the end of his e-mail, worn them to work. Not, mind you, that it would have done him any good. As Milo Minderbinder could tell him, people who wear clown suits to work in order to be declared so mentally incapacitated as to be no longer fit to be chair—those folks are clearly sane. That Catch 22 is really some catch!

Congratulations to Department Manager Trina Avery on her 20+ years at Brown (see over)

Recently Steve Feiner (BA Music '73, PhD CS '87) was featured in a *New Yorker* article on global positioning systems. Steve's name recently came up at a faculty meeting here. We were reviewing PhD students' progress and one student was taking so long we wondered if he was going to break Steve's record for number of years to graduate. (He isn't even close yet.) After finishing Steve went to Columbia and is now a world expert in "assisted reality," which is sort of like virtual reality except you overlay the "virtual" on top of the "real." To do this, however, the computer needs to know what you are looking at, and to do this outdoors it needs global positioning.

I was amused by the author's description of Steve's office: 'His office looked as if it had been constructed as a set for a film about an absent-minded professor: laptops—whole and disassembled—digital cameras, special optics, and antique computer mice were everywhere. There were reprints of articles from the days when transistor radios were making news and several bottles of Taittinger champagne sat on a table in the center of the room.' My amusement stems from the fact that Steve was Andy van Dam's student, and this could very well serve as a description of Andy's office (see ***conduit!*** vol. 2, no. 2). However, the most interesting quote from the article is the description of Steve as 'Walter Mitty with a government grant.' I sent Steve e-mail asking his reaction to this description. Steve's reply:

`"Well, it's not exactly my favorite quote from the article. However, the writer seemed to think it was a compliment. I guess they don't assign Thurber in high school anymore. :-)"`

I received a nice letter from the University a few weeks ago inviting me to a reception in honor of staff members who had been at Brown for 20 years or more. In particular, the reception would recognize Trina Avery's "20.33 years of service." That .33 mightily impressed me—Human Resources was really up on things! Of course, the effect was spoiled slightly by their having sent the letter to the wrong person. It should have gone to Tom Dean, our chair. I have not been the chair of this department for nigh onto 3.66 years.

conduit!

A publication of
The Computer Science Department
Brown University

Inquiries to: ***conduit!***
Department of Computer Science
Box 1910, Brown University
Providence, RI 02912
FAX: 401-863-7657
PHONE: 401-863-7610
EMAIL: sjh@cs.brown.edu
WWW: http://www.cs.brown.edu/publications/conduit/

Suzi Howe
Editor-in-Chief

Katrina Avery
Editor

Jeff Coady
Technical Support

Department of Computer Science
Brown University
Box 1910, Providence, RI 02912, USA

conduit!

NON-PROFIT
U.S. Postage
PAID
Providence, RI
Permit #202

Printed on recyled paper

Address changes welcomed