the WINNOWER

REDDIT

I am here to talk about the science behind visualization. I am Prof. Tamara Munzner from the University of British Columbia. Ask Me Anything!

TAMARAMUNZNER R/SCIENCE

ABSTRACT

Hello world! Tamara Munzner here.
I've been doing computer-based visualization for almost 25 years, starting as technical staff at the NSF-funded Geometry Center, continuing as a grad student in the Stanford graphics group with Pat Hanrahan, and then as a professor at UBC since 2002. I have worked in a broad range of application domains including genomics, evolutionary biology, fisheries management, energy and sustainability, geometric topology, large-scale system administration, web log analysis, computer networking, computational linguistics, data mining, and journalism. Yet more details on my web site in general or my bio page in particular.
Let's talk about the science behind visualization!
I'm particularly excited to talk about the ideas covered my book,Visualization Analysis and Design. Since it's done at long last.
Or any of the visualization research papers, videos, or software at on my lab web site.
Or anything about the visual representation of data, broadly construed.
And hey, it's an AMA, so anything else is fair game too.
Including books, especially science fiction and fantasy, since reading too much is a vice of mine. As you can see from my reading lists: books read in reverse chronological order and books read ordered by author, with commentary.
Proof: https://twitter.com/tamaramunzner/status/636466649541902336
Update 1: forgot to say that the official start time for me answering is noon Pacific time which is 3pm Eastern. That's soon!
Update 2: Answers have started. Typety-type-type.
Update 3: 3pm Pacific, taking the teeniest of breaks for a snack and cup of tea. Must hold body and soul and neurons together. I'll be back!
Update 4: 3:15pm Pacific, back to the keyboard. A runny Brie on rosemary bread toast and an acceptable Cream Early Grey have saved the day. Might need to move on to the big guns of Lapsang Suchong or a hefty Assam soon if the questions continue at this rate!
Update 5: 6:30pm Pacific. Not dead yet - still answering! Although admittedly my posting rate slowing down, despite my fresh cup of Halmari Assam...
Update 6: 10pm Pacific. Declaring victory, or at least throwing in the towel. I've completely run out of time, I've mostly run out of neurons, and I think dinner sounds like a fine idea right about now. Wow, this has been an amazingly fun day! Many thanks to everybody below for your thoughtful questions, and also thanks to @frostickle in particular for both talking me into this and for shepherding me through it.

READ REVIEWS

WRITE A REVIEW

CORRESPONDENCE:

DATE RECEIVED:
August 27, 2015

Are there any instances you know of where alternative data visualization methods have led to breakthroughs in fundamental science?

zu7iv

Let me start at the top - it's both a good question and a hard one!

DOI:
10.15200/winn.144060.05806

ARCHIVED:
August 26, 2015

CITATION:
TamaraMunzner , r/Science , I am here to talk about the science behind visualization. I am Prof. Tamara Munzner from the University of British Columbia. Ask Me Anything!, *The Winnower* 2:e144060.05806 , 2015 , DOI: 10.15200/winn.144060.05806

I have multiple answers along a continuum of speculation (which is arguably a big word for weaselling) depending on just how high the bar is for 'breakthrough'.

The most rock-solid answer is the least satisfying: no, I don't.

The most honest answer is that I've seen multiple efforts to collect vis success stories and they never quite seem to hit the ball out of the park in the way that I'd like. I happen to be personally familiar with the sidebars in our 2004 VRC Challenge report. There are some nice stories there with cool pictures. Yet they don't quite feel like breakthrough at the level of the Archimedes running naked down the street screaming Eureka.

The most advantageous-to-me answer is why yes, I regularly publish papers where I explicitly need to include success stories in the Results section in order for them to get accepted! The flavor of paper that we call design studies in the vis literature requires them.

The most speculative answer is that perhaps the very best visual representations do such a good job of cognitive scaffolding that they melt into the background, and the scientist thinks "ooh I just figured out something crucial" rather than "hey, what a great vis tool I have here". So they're going to attribute breakthroughs to their science not to their tools, and at some level all tools enable science and that's how it should be - from prosaic things like paper to whatever new measurement instrument comes along like the telescope. I once called vis a "fooscope" because it gives you a new scope for all manner of things...

hello! What techniques do you like for reducing high-dimensional data to 2 or 3 spatial dimensions (or ~5 visual variables) that we can show?

whoInvited
Depends on just how high we're talking about.

If it's very-low-high-dim, like just 6 or 7, then I'd probably try to visually encode everything directly in a single view.

If it's low-high-dim, like a dozen or so, then I might try a scatterplot matrix (SPLOM) first, and then follow up with a parallel coordinates view if I'm feeling completist.

If it's a bit higher than that, with a few dozen dims, then I might go with scagnostics - scatterplot diagnostics, proposed by Lee Wilkinson, Anushka Anand, and Robert Grossman - where we go meta and look at a scatterplot of scatterplots.

When we get into medium-high-dim territory, with hundreds of dimensions, then it's time to start full-fledged dimensionality reduction techniques. I give up on explicitly showing all the original dims, and instead use a bunch of math to synthesize new dimensions that try to capture a bunch of the structure. Turns out you can get good results with that approach in many cases where the way you measured the data in the first place is sprawling and verbose, since you can't directly measure what you really care about, and it's legitimate to throw away a huge amount of data because there's massive redundancy. I'm a fan of multidimensional scaling (MDS), in part since we've done a whole bunch of work on that in my lab. The Glimmer algorithm exploits the parallelism of the GPU for speed, and the Glint algorithm handles the interesting case where calculating the distance between two points is really computationally expensive.

If you're looking for clusters specifically, then t-SNE from van der Maaten and Hinton is a great place to start. That's one of the things we found in a quantitative empirical study on how people perceive the results of dimensionality reduction. We also did some qualitative field work to try to figure out just what tasks people are doing when they analyze high-dimensional data. In short, we think it's helpful to think in terms of whether they care about the dimensions themselves, the clusters, or both of these.

And once we're in the realm of high-high dim territory, with thousands or even tens of thousands of dimensions - then typically the data is sparse, where there are roughly the same number of dimensions and data points. That's often the case for text data, where you've got the "bag-of-words" model of a very high-dimensional vector where each word is a dimension, each document is a point, and there are a lot of zeros for the words that don't appear in a particular document. We've proposed the Q-SNE algorithm and gotten nice results for this case, where we can handle millions of documents and get high-quality layouts.

Aaaand... that answer is arguably far too long. But it is a topic near and dear to my heart!

Update: This answer ended up getting a lot of traction, so I'll add one more thing - a link to an hour-long talk where I tell the story of several of the projects that my group has done in this area: Dimensionality Reduction From Several Angles. I've given it in several places, and a few of them taped a version that's available as a video (in addition to the usual talk slides that I always post).

What do are the most glaring or common data vis mistakes that you see in mass media?

tungs
There's always the classics: 3D bar charts will certainly make me wince. Ditto for off-axis projections of 3D pie charts too.

To spread the hate into 2D as well, a more recent trend is bad donut/arc charts. There seem to be a remarkable number of them floating around these days.

One of my particular pet peeves is when people do periodic tables of X where the structure of the original periodic table is used for the spatial layout, and of course that has absolutely nothing to do with the structure of the underlying space of X. Grrrr.

I do have the so-terrible-it's-glorious site WTF Visualizations in my RSS feed. There are so very very many ways to do it wrong it's hard to keep track. And I admit that my not-so-hidden inner curmudgeon almost enjoys the horror.

It would be nice if there was more activity on places likeThumbsUp Vis to counterbalance the schadenfreude...

Hello, thanks for doing this AMA!

What are your thoughts about using Virtual Reality (e.g., Occulus Rift, Microsoft Hololens) for the visualization of data in a research or teaching setting. I can see equipping students with a means to interact with 3D structures or data sets being an invaluable tool for allowing students to more intuitively understand complex systems. Do you have any plans to experiment with these devices in your teaching/research fields?

Clever-Username789
In short: no.

Medium version:

I mostly focus on nonspatial data in my own work these days, and in for this kind of data I find that it's very rare that viewers are engaged in a task where they need 3D spatial representations. I think VR makes sense if and only if you really do need 3D, and it's even more rare that the benefits of VR displays (immersion and presence) would outweigh their drawbacks (resolution limits, physical fatigue, latency, difficulty in switching to tools that support all the standard tasks of web browsing, email, text editing, spreadsheets, etc).

(Ridiculously) long version:

My take on VR is that so far the hardware technology forces you to pay a nontrivial price for immersion and a sense of presence.

First, let's talk about the price.

What continues to be true is that you're penalized with respect to pixels: for the same amount of money, you get a lot fewer pixels for a VR setup -- whether it's headmounted or projected onto screens as in a CAVE -- compared to a standard desktop display. I argue in my book that running out of pixels in the display is one of the most stringent limitations that we have, a wall we often hit before computational limits with CPU/rendering power andy maybe even human limits like memory and attention. A significant amount of the energy that has been devoted to inventing interaction techniques is a workaround for the fact that we don't have enough pixels at our disposal, so we can't just move physically closer to the display to see more as we do in the real world - we need to change what we draw. (High-res display walls are just barely starting to change that, but there's a lot of complexity to that question as well.) Getting back to immersive VR, even as the technology has improved dramatically in some ways, that tradeoff resolution versus immersion remains.

What's also true is that using immersive VR is hard on the human body in a way that sitting on a chair in front of a table with a desktop (or sitting on the couch with a laptop) is not. It's tiring both for your legs/back and your arms to stand and move around for a long period of time in a room with nothing in it. Moreover, headmounted devices do weigh enough that you feel it in your neck quite soon. (On the plus side, arguably typical typing-oriented repetitive strain injuries are less of a problem in this context.)

But the biggest issue is that it's hard to do other stuff with standard tools. For people doing exploratory data analysis, they're almost never doing that in a vacuum. Even when you're fully engaged with doing real work you've got your web browser open to look stuff up, you're frequently needing to check something relevant from your email archives, typically you're writing down notes in some flavor of text editor, maybe you're crunching some numbers in a spreadsheet. And the extremely common case is that you're also task switching by answering urgent email about completely different stuff, not to mention crucial activities like browsing cat pictures and catching up on Reddit. If you have to take off your headmount or walk over to another room, then that frequent task switching is an utter pain.

The good news is that huge strides have been made with latency, both on the rendering side and the sensor side. I first started playing with VR back in the early 1990s, when we adapted some of our desktop software from the Geometry Center to work in the CAVE at UIUC (thanks to George Francis for being the connector there). At that point it was SGIs Reality Engines for the rendering (oh how I *still* miss Irix), magnetic Polhemous sensors for 3D position sensing, a big physical space with tall ceilings for the high-end rear-mounted wall projectors plus the one up high projecting for the floor. The whole setup cost many hundreds of thousands of dollars at the time. Latency was a nontrivial problem for both fronts. Of course, doing stereo means rendering twice as much, one frame for each eye, and that was on top of getting the sensor data processed and incorporated into the graphics pipeline. I tried out various VR setups over the years on the Siggraph exhibit floor. While the money cost kept going down, I never felt like the sense of presence was all that compelling.

Many thanks to Gordon Stoll of Valve Software for giving me a demo of their utterly kickass new VR system this year just as it was being unveiled at the Game Developers Conference. (I think it's currently branded as SteamVR.) They've nailed it in a way that I believe far outstrips the Oculus Rift setup, with extremely low-latency sensing and quite nice optics, that gave me a far more compelling sense of presence than I've ever had. There was one particularly notable demo. In one, the same Halo cyclops guy is shown at three scales. The little one that looks knee-high looked cute and I wanted to rush up and pat it on the head. The medium sized one that seemed to be about my height was one where I was happy to give it enough space to avoid colliding with it. And the huge one that was apparently 12 feet tall was intimidating and made me physically jerk back. And despite the fact that I understood intellectually that these were *identical* 3D geometry models, the difference in the spatial scale alone is what led to dramatically different emotional responses.

(The closest analog to that kind of scale-driven response that I've had in the real world is the summer I lived in Berlin when Christo wrapped the Reichstag. I must admit that when I'd just seen photos of his artwork before that, I thought he was a frivolous idiot. After experiencing my own emotional response to a huge building wrapped up like a birthday present, I think he's amazing.)

And now back once again to VR. I never had that kind of emotional engagement in previous VR setups, I think it was not only about the artistry of the game designers but also about the fidelity of the technology. But... it's not clear to me that this kind of emotional response actually helps with any of the tasks of visual data analysis. So while my first response is that while I had a lot of fun and maybe would want a setup like that for entertainment, I didn't immediately feel the need to use it in my actual work.

VR is very interwoven with the use of 3D. As I said above, I think immersive VR is justified if and only if 3D spatial layouts are justified. And that question is an even bigger can of worms! (My book has one page on the "Resolution over Immersion" rule of thumb, and 15 pages on the "No Unjustified 3D" one.) This one question has taken me an hour already to answer, so I'll wait and see if somebody wants to ask about 3D in a separate question!

Hi, I'm studying a bachelor in CS, and visualization is a topic that attracts me, How can I start in this wonderful world?

iaveiga
Good start on an answer from meem1029.

Re courses worth taking, I'd add human-computer interaction to the list, at a priority at least as high as computer graphics.

Re whether undergrads can get involved in research, it depends. In some cases professors welcome them with open arms, and have set up their labs so that undergrads can immediately hit the ground running. Ron Rensink's Visual Cognition Lab is a great example.

In other cases professors focus much more on grad students. I'm more of the latter in general, as my a personal preference. (I find that it's hard to get undergrads up to speed quickly enough to be effective in a research setting before they graduate, since we have no undergrad course in vis at UBC yet. But even so I still occasionally end up with great ones!)

I also very much like the answer from yelper. But ow, don't grab me by the ear, that hurts :-)

In addition, there are all kinds of great resources out there even if you don't find a human near you to connect with: videos, books, and lists of resources.

Re videos, at the top of my short list to watch soon isthe videos from the OpenVis Conference, I've heard great things about them from multiple sources.

Re books, I'm oh so biased on this front but unsurprisingly I do like my own book - I did write it exactly with the idea of trying to bootstrap people into thinking about vis the way I do in a way that scales beyond the one-on-one mentoring I do with the students in my research group over the course of several years...

Re lists, the goto place that I check first is Matt Brehmer's lovely and comprehensive resource list.

Hi Tamara! Happy to have you on /r/DataIsBeautiful today - thank you for holding this AMA!

What are some non-visual data communication methods that you think are promising? Sound,

smell, taste, and touch are all rich senses that I think are relatively unexplored in the data communication world. Do you think there's any promise exploring data communication via those senses?

rhiever
Thanks for having me!

In short: no.

Medium version:

I think that the focus on visual perception rather than the non-visual modalities isn't just that nobody has gotten around to it, there are deeper reasons. With sound, it's about the human brain: we process sound sequentially. With the rest, it's about the state of the art in technology: we have impoverished technology for recording and playing back touch, and that we have barely anything to deal with smell/taste at all.

At length:

For sound, the problem is that there's a fundamental difference from vision. Sound is something we perceive sequentially, so it's very hard to get an overview. Overviews to summarize things are one of the key things people use vis for.

There's massive processing happening in our brain that makes us feel like we see everything all at once. As it turns out that's an illusion that breaks down at the edges sometimes, since in fact you only see things at high resolution in your fovea. (Hold out your arm straight in front of you, stick your thumb up, and look at your thumbnail. That's size of your fovea, what you can see crisply. Everything else is blurry. But you hardly ever notice that because of all this behind-the-scenes crunching your visual processing system is doing!) You think you're seeing the whole room around you, at least the part that's in front of you.

But we don't have such processing for sound: you don't think you hear a whole song all at once. You can hear multiple notes simultaneously in the form of chords, but when it gets down to it we process sound as a sequence whereas visual processing results in a stitched-together collage.

For touch, there's some medium-expensive support for haptics feedback like the Phantom, but that only gives you the experience of a single pointing finger. There's some cheaper stuff that's vibrotactile, but mostly feels like faint buzzing at your fingertips. We'll need a lot more to approximate the experience of an embodied human. I frankly don't know the answer about what will happen then, I'm curious.

And for taste/smell, no idea whether we'll manage to do something interesting once the technology gets there. Now we're still in the wild west territory of weird demos at Siggraph from Japanese labs where machines squirt things into your mouth and you make strange faces. At least in my personal experience :-)

Hi Tamara! Thanks for doing this AMA! Occasionally I get asked how to start learning data visualization, and I usually find myself asking what a person's background is, considering people usually enter the field from a couple different paths. I was wondering what your thoughts on the matter: is there a really good starting point for everyone, or should advice be specific to a person's existing skillset?

tungs
Thanks for asking nice chewy questions!

Specific to the person, absolutely.

There's not just a couple of paths, there's a zillion. Everybody has a pretty idiosyncratic story - one great way to hear a lot of them is the super fun Data Stories podcasts.

Some people come in from computer graphics, I was one of them. Others come from human-computer interaction (HCI). Some are from statistics. Some are from cognitive psychology. Some are from math. Some were in a domain and needed vis tools that didn't exist yet, started building them, and then blinked and realized that few decades had gone by and they had become vis people.

And there's all kinds of cross-pollination stories, where people have an academic background in X that ends up doing them a lot of good when they later find themselves in vis, in a way that they didn't plan in advance. In my research group alone I've had folks with many values of X: astronomy for Miriah Meyer and Michelle Borkin; fine arts for Jessica Dawson; biology for Joel Ferstay.

Even questions like "what math do I need" are totally nontrivial to answer. If you do a lot with spatial data then the mathematics of continuous spaces are key, and to get what's happening with signal processing and reconstruction, you do want a lot of calculus under your belt. If you care about network layouts, well then graph theory might be your cup of tea. For some things you want discrete math like combinatorics. There are all kinds of nooks and crannies like hyperbolic geometry, which happens to be the route that got me personally into vis, but it would be insane to tell everybody they needed to take that path. If you do anything with controlled experiments in a lab setting, then by God you need stats. And then there's the math behind whatever domain you're targeting.

What technologies do you prefer for data visualization creation?

I (try to) teach data visualization to activists and journalists and I've been frustrated by the data vis community's reliance on R, which is all but inaccessible for beginners with little coding experience. Instead, I've leaned toward Processing (and ProcessingJS) and d3.js for the sake of visualization accessibility.

matthiasshapiro
Oof, no wonder you're frustrated, I bet if I tried to teach vis with R to people who are not already familiar with coding I would also be tearing out my hair. But huh, I was actually surprised to see 'R' in that sentence, I thought you were going to say 'D3'.

It's not so clear to me that Processing (Java or JS version) or D3 are any better solutions to the problem, they both require coding too. D3 is quite a different way to program, with its functional mark-based view of the world that is very different than traditional imperative languages like Processing. But they're both still coding!

The underlying question is whether vis creators need to code? Or, to put it a bit differently, what are the limits of expressivity and power on the tools available to non-coders? I think we as a field are at least making progress. For instance, tools like Lyra that attempt to bridge this gap. When I say "we", to a first approximation I mean Jeff Heer, who has been the driving force behind an enormous fraction of the academic vis world's output of tools/toolkits/systems - Prefuse, Protovis, D3, Vega, Lyra, and more. Of course, that's in collaboration with many folks including the stupendous Mike Bostock, late of the NYTimes - I note @mbostock will be doing an AMA himself here on DataIsBeautiful next month!

I'm not the best person to answer this, since thus far I've been teaching visualization creation to people who can code in my graduate vis class, and certainly all of my grad students know how to program since I'm in a CS department. (My class is open to non-coders, but typically they do analysis projects where they use existing tools rather than trying to create something new.) What I've noticed is after years of students using a very wide variety of tools, there's been a massive shift where nearly everybody used D3 last year. And a few used R. So the coders at least are voting with their feet (well, ok, with their fingers) for D3.

I might have better answers soonish. I am grappling with this very question myself since in exactly 20 days I will roll out pilot version of a data vis module for journalism students at UBC who are not coders. I just talked with @eagereyes this morning by Skype about whether/how to use Tableau, even as y'all were posting questions on Reddit.

Hi Tamara,

I found hyperbolic projection very useful for graph visualization. I tried to find some recent real world applications employing that technique but I failed. I feel like people suddenly stopped exploring that technique 10 years ago. Could you explain this as you are an expert in hyperbolic-based graph explorers.

Thanks!

frank7v
Hi frank7v! (Using my ninja-like sleuthing skills I'm guessing you might be Frank van Ham.)

Take one:

It's not completely dead, I've seen a bit of stuff here and there, like that paper onNon-Euclidean Spring Embedders at InfoVis from Kobourov. Ah, I see, that was 2004. That would indeed be 10 years ago! Dang, I'm getting old. Never mind.

Take two:

So the question is why it stopped. You've finally made me sit down and try to articulate a conjecture that's been nibbling around the back of my mind for a while. Maybe it a bit of a dead end because while it's extremely elegant mathematically, there's a deep challenge on the usability front.

For those readers who are not hyperbolic geometry geeks but are CS geeks, here's the super-quick version of why it's so elegant: there's an exponential amount of room available in hyperbolic space, in contrast to the polynomial amount of room in Euclidean space. And thus it's a very appealing way to lay out trees, where the number of children can grow exponentially in the depth from the root to the leaves. My PhD thesis chapter on the hyperbolic graph explorers has a more relaxed discussion about all this with more explanatory diagrams than the H3 paper, but graphics people who really want to follow the math of hyperbolic geometry should check out this great article from Jeff Weeks on Real-time Rendering in Curved Spaces.

But here's the tricky bit: there's a deep issue about why hyperbolic geometry is confusing, and it's not *just* about unfamiliarity at the surface level. It's relatively straightforward to walk people through the idea that you've got this projection that's a disc (or a sphere in the 3D case), stuff at the center is big, and but it's projected to a smaller and smaller region as you get out towards the periphery - that circle (or sphere surface) is infinitely far away from the center. If they're into that kind of thing :-).

The messy part is that there's a way that hyperbolic geometry is just plain more complicated than Euclidean geometry. We're used to rotations having a center, but we're *not* used to translations having a center. They do in the hyperbolic case, and so stuff goes swinging/swooshing around in a way that people find very surprising and hard to follow.

This PoincareDraw explanation page has a reasonably understandable explanation of the implications of this fact. Here's the key bit:

A hyperbolic translation has a "center," which is a line. Points on the center are all moved the same distance along the center. Points off the center are moved farther, and not along lines. Instead they are moved along curves that are equidistant from the center.

I conjecture that's why people end up backing away from it. It could be that somebody who really knows hyperbolic geometry at a deep level and also cares about interaction design could figure out a solution. That's not me - I'm not actually a hyperbolic geometer, I just used to hang out with a bunch of them. (And in that spirit let me add the caveat is that I might not even be right about this point at all! Not being a hyperbolic geometer and all...)

What's your favorite book?

imthatguy25
Ahhhh, books. It's been six straight hours of vis ranting, I declare that I deserve a break to switch gears and rant about non-vis books!

I definitely can't answer that question with a single book, and I can't even answer with a single author. I can answer with a single genre if I'm allowed to smoosh science fiction and fantasy together into 'SF' for Speculative Fiction.

I can answer with a whole bunch of favorite authors. Here's the short list of the absolute top ones, it's filtered down a lot from my longer list of the authors I like enough to recommend

Kage Baker, Iain M. Banks, John Barnes, Greg Bear, David Brin, Steven Brust, Lois McMaster Bujold, Emma Bull, C.J. Cherryh, Susan Cooper, Samuel R. Delany, Cory Doctorow, Lynn Flewelling, Randall Garrett and Vicki Ann Heydron, Edward Gorey, Steven Gould, Mira Grant, Nicola Griffith, Lian Hearn, Robin Hobb, P.C. Hodgell, Tanya Huff, N.K. Jemsin, Rosemary Kirstein, Ursula LeGuin, Jonathan Lethem, R.A MacAvoy, Madeline L'Engle, Ken MacLeod, George R.R. Martin, Julian May, Ian McDonald, Seanan McGuire, Maureen McHugh, Juliet McKenna, Patricia A. McKillip, China Mieville, Steve Miller and Sharon Lee, Pat Murphy, Linda Nagata, Paul Park, Kim Stanley Robinson, Geoff Ryman, Brandon Sanderson, Robert J. Sawyer, John Scalzi, Karl Schroeder, John Slonczewski, Wen Spencer, Neal Stephenson, Bruce Sterling, S.M. Stirling, Charles Stross, James Tiptree Jr, Karen Traviss, John Varley, Joan Vinge, Vernor Vinge, Martha Wells, Scott Westerfeld, Walter Jon Williams, Robert Charles Wilson, Gene Wolfe, Roger Zelazny, Charles de Lint

Do you have a favorite map projection? If so, what is it?

In_Shambles
The safe answer:

I don't work with geospatial data enough to have a truly informed opinion about map projections, I'm a dabbler in comparison to the London giCentre cartographer/geographer pack of Dykes, Wood, Slingsby, and the Adrienkos.

The real answer:

My favorite general projection is probably stereographic projection. My formative early years in vis were at the The Geometry Center (now dearly departed, with the archive living on thanks to UIUC). Stereographic projections are quite useful in "low-high" dimensional geometric topology, where we'd look at 4D and 5D mathematical objects projected into 3D. Stereographic projections make hypercubes look like soap bubbles. And they're also handy for thinking about the non-Euclidean geometries -- hyperbolic and spherical. I have often showed people stereographic projections of a wireframe Earth model, but it was intended to help people get an intuitive sense of how the projection works before moving on to more exotic mathematical structures. In the context of what I do these days, it's hard to imagine a circumstance where that would be the right thing to do if my goal was actually helping somebody understand geographic data.

All this backstory explains in part why my favorite map projection is an exotic one - I can't help but to be enchanted by Jack van Wijk's paper Unfolding the Earth: Myriahedral Projections. I'd probably said "you have to pick which property to preserve - angles or distances - you can't have both" at least 100 times while wearing the hat of tour czar at the Center. (A job I sort of resented at the time, but it taught me how to soldier on when public speaking come hell or high water, after giving live demos to groups ranging from research mathematicians to fourth graders.) This paper points out that there's a third option - you can preserve both angles *and* distances if you open up the space of possibilities to allow interruptions! In addition to being elegant on the mathematical front, it's also sophisticated on the computer science front - with an algorithm for recursive subdivision that's motivated by the graphics literature on meshing. And then it's *also* sophisticated on the visualization front, where he applies streamline algorithms that come out of vector/flow field visualization. Swoon.

Hello Dr. Munzner, I recently graduated from a Bioinformatics PhD program. I didn't get to study data visualization, but around the end of my term, I got extremely fascinated by it through a smaller project. What would you (or another vis researcher) look for in a potential post-doc candidate like me? I see myself doing more data visualization than bioinformatics in the future. I really really enjoyed it, and it's unfortunate that I didn't get to study it at length. Is there any hope for me in a vis post-doc?

i_shall_pass
Yes, I think there's considerable hope.

In the past few years I've seen that there is a lot more demand for vis postdocs than supply. So it's quite conceivable that somebody might be willing to invest in helping you jump from bioinformatics to vis as a postdoc. What you'd be bringing to the table in addition to a bit of experience on the vis side is a whole lot of experience on the bioinformatics side. You might be able to find a labs where that could be a win for both the PI and for you.

Conveniently for you, biological/bioinformatics visualization is a very hot area these days.

One place to start looking is the set of author and of organizers of theBioVis symposium series.

You might want to join the ieee_vis_open_positions mailing list, see https://listserv.uni-tuebingen.de/mailman/listinfo/ieee_vis_open_positions

How do you like to visualize samples from a larger dataset?

I can see a focus + context (minimap) approach being useful when all of the values in your sample are adjacent. How do you like to show a user a random sample of non-adjacent values?

whoInvited
There's no single right answer. But what comes to mind is two examples from my own work that I think were good answers, in contrast to the all-too-many bad answers from the enormous space of possible vis designs.

In general, I find it useful think about four main alternatives for handling complexity: transforming the data based on the user's task into a new data abstraction (that can be visually encoded to address that task in a way that's more effective than the original data abstraction that you were given); manipulating the view through interaction in general including navigation in particular; partitioning the data into multiple views (which are often but not always juxtaposed side by side); reducing what you show within a single view with (possibly sophisticated) combinations of filtering and aggregation.

In specific, let me separate out the question of handling non-adjacent values in general from the special case of random samples in particular.

In a recent design study aimed at genomics researchers, we argue that a pitfall of standard genome browsers is that they do a poor job of handling non-adjacent values that are small in size and scattered across a large region of interest. If you're zoomed in far enough to see one of them, you'll need to zoom way back out and translate over to get to the next one, and this extensive navigation makes it hard to understand the big picture all at once.

We designed the Variant View system for researchers studying the genetic basis of human diseases like leukemia, to support them in understanding which genetic sequence variants across different people were predictive of disease states, and these were scattered throughout the enormous genome. We proposed a transformation of the dataset from the standard data abstraction of genome coordinates to a different data abstraction: transform coordinates (essentially cutting out the stuff that's not active, which is certainly not always legitimate to do but was warranted for this particular kind of investigation). We also did some transformations so that instead of a variant-centric view (a big table with one row for each variant), we combined all of the known variants into a gene-centric view (showing all the variants together in terms of where they fell within a gene, to make hotspots where many different people all had variation jump out visually). And then we did some careful visual encoding of that. While that visual encoding was important -- and it's something that's also easy to get wrong if you don't have a clear sense of the perceptual properties of human vision -- I think the main intellectual contribution of this system was really about the data transformations at what I call the data abstraction level (what to draw), rather than the visual encoding level (how to draw it).

And now for random sampling. That brings me to a design study for a very different target audience: investigative journalists faced with huge heap of documents. One of the use cases we originally had in mind for the Overview system was a journalist who wanted to do random sampling of that set, in circumstances where it's infeasible to just power through and read them all. I'm not going to try to summarize all of the task abstractions, data abstractions, and visual encoding decisions here, lest my fingers fall off before I get down to the end of the AMA questions :-), but there's a whole paper about this that you might find interesting! See Overview: The Design, Adoption, and Analysis of a Visual Document Mining Tool For Investigative Journalists for an in-depth discussion.

Hi Tamara,

Im starting at UBC for Computer Science in a few daysand I have always been fascinated by data visualizations. I am wondering if there are any interesting projects in this area being conducted at UBC right now?

Hopefully I will sit in one of your lectures in the near future!

SomeNerdyDude
Well, pretty much by definition I think all of the projects we're working on inmy research group are interesting, or I wouldn't bother doing them!

There's certainly new and ongoing stuff that we haven't posted about publicly. I don't typically announce things to the outside world until they're done (or done enough that I want people to use the tools to give feedback), so by the same logic I'm not willing to say much about those in this forum either...

Sadly we don't yet have any vis courses at the undergrad level. Mygrad class starts week after next, but it might be hard to find a seat the first few sessions since it's full up with a waitlist, there are more people signed up than chairs in the room (and I can't get a bigger room, I already checked). I'm hoping the space crunch will ease after people switch from shopping-around-for-courses mode to committing-to-final-choices mode.

In any case, do come up to say hello afterwards if you do end up at one of my lectures.

Also, as I mentioned in this other reply, the other major infovis / visual analytics presence in UBC is

Ron Rensink - check out his UBC Visual Cognition lab.

Is development of software to facilitate the production of visualisations as important as the theory behind good presentation?

I ask as I'm interested in the software used to produce visualisation and was wondering if you could expand on the workflow and software you use. I'm familiar with ggplot2 in R thanks to /u/hadley and find it far more flexible than the graphics in Stata, but many of the custom plots appear to be done with different software, are they levering tools like D3.js?

enilkcals
The short answer:

Yup, these days we use D3 a lot. Only a tiny bit of R thus far, but that's increasing. Before that was Processing and Prefuse. Before that was OpenGL with GL4Java bindings. Before that, back in grad school when I got to code myself (alas no more), a lot of OpenGL with C++ and C. And before that, when I was techstaff at the Geometry Center, IrisGL with C, with forms/xforms for the GUI. And perl. And Mathematica.

The long answer:

Yes, building software is crucial to multiple angles of attack in my view of research. My view of the world aligns quite closely with the paper types that we're using at most of the major vis conferences: technique-driven work (techniques and algorithms), problem-driven work (design studies), evaluation work (controlled experiments with human subjects and also field studies before and after deployment), and theoretical foundations. It's an (upward) spiral: we rely on the existing theories as we build the software, and then issues that come up as we build the software that both fuel the need for and inform subsequent theory.

Building software is a non-negotiable requirement for technique-driven and problem-driven work, it's frequent even in evaluation work, and it's sometimes even part of the theoretical founcations work

Some of the evaluation work involving controlled human-subjects experiments uses existing software rather than doing new software development, but often even then we end up doing at least some hacking to make the experimental platform. And sometimes it's a significant amount of coding.

With problem-driven design study work we address the real-world problems with real-world users, and this work always involves incremental refinement, so we're iterating by building many prototypes. Rapid/agile prototyping is very helpful in these cases. Our paper on Design Study Methodology talks about this philosophy in detail.

And with technique-driven design, we've *also* got to build software to see if our ideas work. Usually there's some amount of refinement there too, although that's not as front and center as with design studies. Often scalability and robustness are more central considerations in this case. And then to publish, we need to validate that it works through methods like computational benchmarks, not to mention typically we'd submit a video of the system in action to show reviewers the look and feel of the techniques.

I wrote up the initial idea of paper types when I was chair of InfoVis in 2003/2004 as a way to try to help both authors and reviewers find common ground. Folks have found that useful enough that is has percolated to the other conferences, and the call for papers has been kept pretty much intact since then. I've continued with two "meta-papers" about the visualization design and validation process that expand further on some of these ideas: Process and Pitfalls in Writing Information Visualization Research Papers, and A Nested Model for Visualization Design and Validation.

Hello my northern neighbor. How is the smoke up there in BC? Bellingham has been quite hazy. Ever visit Bellingham? Thanks for the AMA!

PorchMonkeyBreeze
Hiya!

No problems breathing up here today, but it was hazy a few months ago when the wildfires were at their worst.

Yup, I've been to Bellingham many times, I usually stop at the Trader Joe's if I'm driving back from Seattle.

I was just there four days ago, on the way home from Friday's visit to Microsoft Research!

You're most welcome.

Is your research based more on aesthetics or function? How do you quantify either?

GTMonk
While I do want both, I'm far more focused on function.

Another slogan from the book: "Function First, Form Later". My rationale is that if you've got function, you can incrementally improve the form. If you've got something beautiful but ineffective, you'll typically need to start over from scratch.

I've got a lot of thoughts for how to quantify effectiveness, I'd say that's one of the central themes of my book.

I don't have good answers for how to quantify aesthetics. I think that's a pretty open problem...