Editor: Theresa-Marie Rhyne

Bridging the Gaps

Jarke J. van Wijk
Technische Universiteit Eindhoven

Users play a central role in visualization. The ultimate aim of visualization is to provide insight to users, not just to produce images. Since the late 1980s, our field has spent much effort on developing new methods to help users obtain insight, and we've made a lot of progress. Many researchers nowadays use visualization routinely to understand the results of their measurements and simulations.[1]

However, many problems still exist, and not every method reaches its intended audience. In recent years, discussion has focused on the position of our field and which goals to pursue. These discussions have led to two important reports, which give clear overviews of the field and recommendations for future research: *NIH/NSF Visualization Research Challenges* by Chris Johnson et al.[1] and *Illuminating the Path*, edited by Jim Thomas and Kristin Cook.[2]

In both reports, close cooperation with domain experts is a central issue. If we, as visualization researchers, want to make a step forward, we should make sure that we aim for useful results for domain experts that fit into their processes. This doesn't always happen, and often a gap exists between visualization research and application domains.

These reports take a high-level view and give recommendations for policies to stimulate cooperation. In contrast, in this article, I consider the issue from the perspective of the day-to-day practice of academic visualization research.

The gaps

Let's consider Vivian, a visualization researcher, and Douglas, a domain expert (see Figure 1), who work together on a visualization research project. This is a simplification, but in fact most graduate students complete their visualization research by cooperating with a small group of domain experts. I assume that Vivian has the typical background of most visualization researchers. By that, I mean that she studied computer science, followed courses on graphics and human– computer interaction (HCI), and she's a skilled programmer. Vivian sets out to solve Douglas's problem using a standard design process with analysis, design, implementation, evaluation, and publication as the main stages. In this situation there are two gaps that must be bridged: a knowledge gap and an interest gap.

We first consider the knowledge gap, both persons have different backgrounds and expertise. The domain expert—for instance, a researcher from a different department— has a deep knowledge on some esoteric topic. Douglas talks about concepts Vivian is unaware of, and even when he uses familiar words, they can have a different meaning. Once a visualization colleague told me that he had a good discussion with a biologist. They agreed that the interface was important, but only after some time he discovered that the biologist meant the interphase, which is a phase of the cell cycle, and not the human–computer interface.

The visualization researcher has to spend effort to understand at least the basics of the other field, but to support the expert in his search for new knowledge, just the basics is often not enough. If the expert is interested in explorative visualization, then he's probably aiming at advancing the state of the art in his domain. This also means that he's not exactly sure and cannot express what he's looking for, except that he's aiming for new insights. Insight is a difficult issue, as Chris North recently pointed out very clearly while noting the main characteristics of insight.[3] Insight is complex, deep, qualitative, unexpected, and relevant. He uses these characteristics to discuss proper methods for the validation of visualization methods. But, these characteristics also have an impact on the earlier stages of the research and development process.

Vivian also uses concepts that are unfamiliar to Douglas. A domain expert can't be expected to know the ins and outs of (for instance) mesh-reduction algorithms, color spaces, and transfer functions. In an ideal world, he wouldn't need to know; in practice, often he's unwilling to take an interest in such things. In terms of my model for assessing the value of a visualization,[4] it's best to avoid these costs to increase the overall benefit of the visualization.

Next, we consider the interest gap. The visualization researcher aims at publishing in journals and at leading conferences on visualization, and therefore she focuses on developing new and interesting methods and techniques—that is, interesting in the eyes of her visualization colleagues.

Our field does have a tradition of accepting and publishing application papers. Also, usefulness is an important criterion for acceptance of research papers.

 Published by the IEEE Computer Society

Reviewers usually require that new methods be used by real users, and expect at least anecdotal evidence of the value of such methods. But still, the focus is mostly on novelty, and not primarily on usability, including down-to-earth issues such as the kind of data sources that can be handled, availability on various platforms, ease of installation, and ease of use.

These issues, however, are crucial for the domain expert, who's primarily interested in tools that will help him do his work faster and better, here and now. Application papers that compare and evaluate systems on these issues as well as functionality offered and scalability are rare—but would be useful—both for domain and visualization experts.

We can argue that there's not a real problem here. If Vivian just produces new methods on her own, they will be picked up and integrated in time if they're interesting. Possibly, but how do we know which methods are interesting? Also, it's unclear who's willing to spend the effort in trying this out, and it seems much more effective to cooperate from the start and directly target methods and tools that are both new and useful.

Let's consider different models for cooperation. Vivian could either

- take a user-centered design approach,
- just work for Douglas as a toolsmith,
- just do what she's good at (as a computer scientist), or
- she could follow her own interest (curiosity driven).

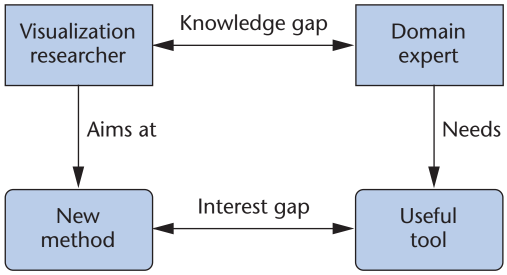
This list isn't exhaustive. One other route, for instance, is to educate domain experts to define visualizations themselves. I assume here that the visualization expert has to adapt, but we could also consider that the domain expert could make steps to cross the bridge. Also, the models sketched are extremes, and don't exclude each other. The practical challenge is to find an optimal mix.

User-centered design

The royal road is to use a user-centered design approach. In this approach, the needs, wants, and limitations of the end user are given extensive attention at each stage of the design process, and usability is crucial. Typical usability considerations include who the users are, what they know, and what they want. In the context of explorative visualization these are tough questions. For instance, in contrast to designing user interfaces for consumer devices, where the tasks of users can be defined clearly, acquiring insight—the main task here—is much harder to pinpoint.[3]

Taking this road, Vivian therefore has to spend much time to understand what Douglas is doing and what he needs. She has to take classes and read books on his subject, observe him while doing his work, and ideally participate in his research activities. This will almost certainly lead to new insights for Vivian. One example I once heard was that the observed researchers used a ruler to measure things shown on the screen. Indeed, for visualization people, it's an eye-opener that domain experts often want quantitative results, not just pictures.

Close collaboration is important at every stage, and prototypes play an important role. Users are good at stating what they do and don't like, whereas the task to come with new solutions is primarily the realm of the visualization researcher. Evaluation is important for the interface as well as for the visualization itself. It's hard for visualization researchers to see their images through the domain experts' eyes. I've given an example where things went wrong elsewhere.[4] Another example concerns a project I once did in molecular dynamics. I was disappointed in the resulting animations—all the ions just seemed to move randomly without any visible pattern. Fortunately, the physical chemist involved was glad. He had modeled a steady-state situation, and hence only a random thermal motion should be visible.

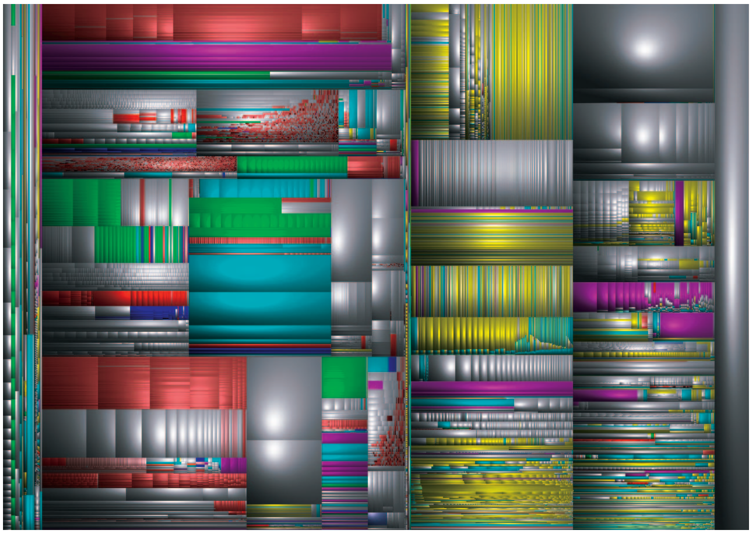


1 Gaps between the visualization researcher and domain expert.

A user-centered approach will probably lead to useful results for Douglas, but unfortunately, there are some pitfalls. Vivian and Douglas have to invest much time and energy. Note that we assume that Vivian has a computer science background. If she also has some education in Douglas's domain, then bridging the knowledge gap will be easier. New multidisciplinary curricula—for instance, on bioinformatics—are promising in this respect. Another pitfall is that we aren't certain we really need new visualization methods to solve Douglas's problem; possibly a combination of more traditional approaches will do, supplemented with an easy-to-use interface tailored to Douglas's domain. If so, this could be considered a useful research result in its own right, but often it isn't really satisfying for Vivian.

It's interesting to note here that novelty is relative. Tim Peeters, then a master's student at Eindhoven, cooperated with a biology group to visualize annotated DNA sequences. The biologist asked for static images, but we brought in a number of standard information visualization concepts, such as multiple linked windows, real-time interaction, customizable views, and perspective walls, all of which we considered more or less straightforward.[5,6] The biologists were enthusiastic and claimed that this was highly innovative and the most effective tool for this purpose they knew.

Again, there's no guarantee that strong visualization results will occur, such as generic insights, surprising new visual representations, or innovative user interface concepts. An important factor here is whether Vivian is capable of keeping these issues on the agenda and coming up with fresh ideas. But Douglas also plays an important role. Finding a "good" user (or, in some respects, a good collaborator) is important. Such a user should be open to new ideas, have time to spend, and also, his problems should be instances of more generic situations.



2 **Visualization of a hard disk with SequoiaView. Rectangles denote files, the area indicates their sizes, and colors denote file types.**

Toolsmith

Another approach is to swallow ambitions with respect to novelty, and just steer directly toward a useful tool. This can (and maybe should) be done in combination with a user-centered approach, but we can also accomplish this just by letting the users specify what they want and implementing the tool. Domain experts already use tools and are aware of their limitations. A typical request is something like "Make something that takes database X as input, produces images like tool Y, and is accessible through the Web." Such a request can be completely valid; however, to satisfy it you do not need a PhD student—you need a skilled programmer who acts as a toolsmith, and not a visualization researcher.

At meetings where plans are made to begin multidisciplinary cooperation, computer scientists and especially visualization experts are often warmly welcomed. However, domain experts sometimes consider computer scientists just as a bunch of programmers who can resolve their practical problems. From a computer science point of view, it's important to make it clear in an early stage which problems require an advancement of the state of the art and which are primarily a matter of engineering.

On the other hand, it can pay off to play the role of toolsmith for some time. Each visualization project requires some basic operations to be handled (such as reading data and showing summaries), and it helps to gain trust from domain experts. Also, basic visualization solutions can be useful during the development of more innovative approaches, for debugging as well as for evaluation and comparison purposes.

Computer scientist

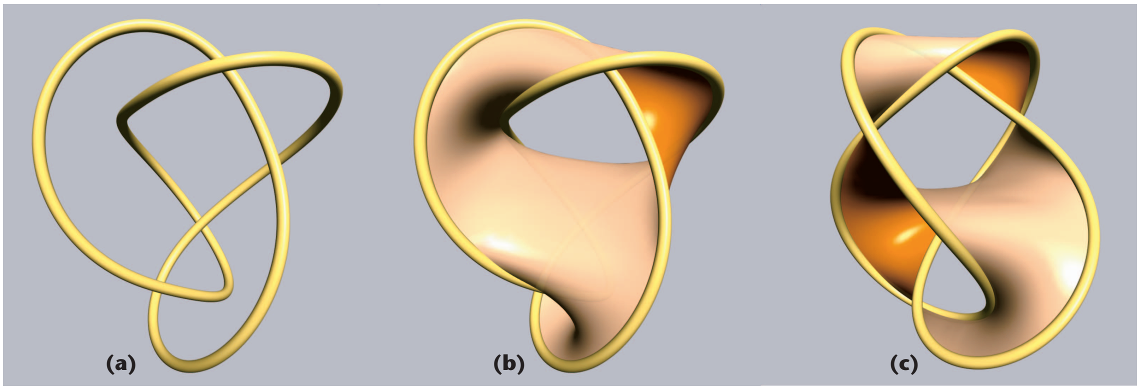
For effective cooperation, we can also defend that each cobbler involved should stick to his last. Vivian has a solid training in computer science—so she's familiar with topics such as algorithms, data structures, and graphics. She can focus on some established visualization method, ideally implemented in a widely used tool, and try to improve this.

A typical example is the reduction of the use of resources (memory and CPU) for the computation of, for instance, isosurfaces, volume renderings, and graph layouts. If successful, larger data sets can be shown in shorter time, and nobody will object. But this also could lead to some pitfalls. Many have already pursued this path, and nowadays it's not easy to achieve significant improvements. If she performs this optimization in close cooperation with a commercial vendor, then publishing the new methods can be problematic. Also, this attitude generally doesn't lead to visually new methods. A good performance is often required—if not vital—but there's much more to visualization than improving algorithms.

Curiosity driven

I conclude with another approach to bridge the gap. I've used it successfully in the past, but you might consider it cheating. Academic freedom allows many of us to select users with interesting problems. Hence, Vivian can also choose herself as the domain expert, and attack a visualization problem which she already knows a lot about and where she has a direct interest in solving it. In other words, she follows her own curiosity and views the project both as a hobby and a personal challenge.

Our work on visualizing large hierarchies[7] has been driven by the question Why is my disk full? I'm poor at



3 **(a) A figure-8 knot. A Seifert surface is an orientable surface bounded by a knot, and in (b) and (c) we see two views of such a Seifert surface.**

managing resources such as disk space, desktop space, and cabinet space, and only when they are used up do I reorganize. Hence, I have a strong need for tools that support me and that give me a good overview of how I'm using my storage (see Figure 2). An additional benefit here is that a file structure is a good example in general of a large and complex hierarchical data set.

A more recent example concerns knot theory (see Figure 3). My colleague Arjeh Cohen asked me two years ago if I could visualize a Seifert surface. Upon my obvious first question, he explained to me that a Seifert surface is an orientable surface, bounded by a knot. A knot here is a mathematical knot—that is, a closed curve. Surprisingly, such surfaces exist for any knot or link. I was puzzled and intrigued. What do these surfaces look like? I spent some time trying to understand these, using hand-drawn drawings, paper, and clay mock-ups. Finally, I developed methods to generate them automatically, and implemented them in a tool.[8] With this tool I could satisfy my curiosity.

Note that my role of user here was that of the naive outsider, similar to a high-school student interested in higher math. Knot theory is a sophisticated topic, and the average paper in this field isn't understandable for me. Also, this tool doesn't help knot theorists develop more theory. Fortunately, knot theorists are also interested in teaching their work and showing interesting images, hence they positively received my work.

This approach might seem outrageous. Focusing on your own interests seems narrow minded, and there's no feedback from the real world about whether it makes sense. Vivian might get to know enough to satisfy her own personal curiosity, but not enough to make any difference otherwise, and this doesn't address problems that require deep domain knowledge. Nevertheless, I think we can learn a lesson from this, if we combine it with the user-centered approach. In the ideal case, Vivian should take Douglas's problem as her own, personal challenge, and cooperate side-by-side with him to solve it. This gives Vivian the proper motivation and direction, which she'll need if she really wants to come up with both exciting new results as well as useful tools.

Finally

Cooperation between domain experts and visualization experts is important, but not without problems. I've tried to identify causes for these, and enumerated a number of models for cooperation. None of these is clearly superior; I'm afraid there's no easy way to bridge the gaps. A user-centered design approach requires much effort to bridge the knowledge gap, without a guarantee that the interest gap is bridged. Just operating as a toolsmith will often not lead to publishing papers, and hence the interest gap is certainly not bridged. Acting just as a computer scientist reduces the knowledge gap but has a limited scope. And finally, the curiosity-driven approach doesn't help solve problems requiring deep domain expertise.

Nevertheless, I hope that the gaps described and the models for cooperation are helpful for Vivian and Douglas to understand each other's interests and positions better, to discuss various modes for cooperation, and to obtain useful results for both of them. ■

Acknowledgment

I thank the reviewers for their thorough and inspiring comments.

References

1. C.R. Johnson et al., eds., *NIH-NSF Visualization Research Challenges Report*, IEEE Press, 2006.
2. J.J. Thomas and K.A. Cook, eds., *Illuminating the Path: Research and Development Agenda for Visual Analytics*, IEEE Press, 2005.
3. C. North, "Towards Measuring Visualization Insight," *IEEE Computer Graphics and Applications*, vol. 26, no. 3, 2005, pp. 6-9.
4. J.J. van Wijk, "The Value of Visualization," *Proc. IEEE Visualization*, IEEE CS Press, 2005, pp. 79-86.
5. T. Peeters et al., "Case Study: Visualization of Annotated DNA Sequences," *Proc. Symp. Visualization* (VisSym), Eurographics Press, 2004, pp. 109-114.
6. S.K. Card, J. Mackinlay, and B. Shneiderman, eds., *Readings in Information Visualization: Using Vision to Think*, Morgan Kaufmann, 1999.
7. J.J. van Wijk, F. van Ham, and H.M.M. van de Wetering, "Rendering Hierarchical Data," *Comm. ACM*, vol. 46, no. 9, 2003, pp. 257-263.
8. J.J. van Wijk and A.M. Cohen, "Visualization of Seifert Surfaces," *IEEE Trans. Visualization and Computer Graphics*, vol. 12, no. 4, 2006, pp. 485-496.

Contact Jarke J. van Wijk at vanwijk@win.tue.nl.

Contact editor Theresa-Marie Rhyne at tmrhyne@ncsu.edu.