EDITORS: Manish Parashar, manish.parashar@utah.edu
David Abramson, david.abramson@uq.edu.au

DEPARTMENT: CASE STUDIES IN TRANSLATIONAL COMPUTER SCIENCE

Reflecting on the Scalable Adaptive Graphics Environment Team's 20-Year Translational Research Endeavor in Digital Collaboration Tools

Mahdi Belcaid, Jason Leigh, Ryan Theriot, Nurit Kirshenbaum, Roderick Tabalba, and Michael Rogers, *University of Hawai'i at Mānoa, Honolulu, HI, 96822, USA*

Andrew Johnson, Maxine Brown, Luc Renambot, Lance Long, and Arthur Nishimoto, *University of Illinois Chicago, Chicago, IL, 60607, USA*

Chris North and Jesse Harden, *Virginia Tech, Blacksburg, VA, 24061, USA*

Translational software research bridges the gap between scientific innovations and practical applications, driving impactful societal advancements. However, developing such software is challenging due to interdisciplinary collaboration, technology adoption, and postfunding sustainability. This article presents the experiences and insights of the Scalable Adaptive Graphics Environment (SAGE) team, which has spent two decades developing translational, cross-disciplinary, collaboration tools to benefit computational science research. With a focus on SAGE and its next-generation iterations, we explore the inherent challenges in translational research, such as fostering cross-disciplinary collaboration, motivating technology adoption, and ensuring postfunding product sustainability. We also discuss the roles of funding agencies, policymakers, and academic institutions in promoting translational research. Although the journey is fraught with challenges, the societal impact and satisfaction derived from translational research underscore its significance in the broader scientific landscape. This article aims to encourage further conversation and the development of effective models for translational software projects.

Translational research transforms basic science discoveries into practical tools and methods that ultimately benefit society.[2] It addresses cross-disciplinary challenges, stimulates innovation, and maximizes the impact of research findings. However, unlike applied research, translational science involves creating and maintaining a reliable end product beyond the initial funding period, posing unique challenges and offering substantial benefits. Figure 1 depicts the process that defines domain science or computer science and the process that defines sustainable translational research.

Computer science plays a crucial role in translational research, yet we identified the following three obstacles to success:

1) *Facilitating efficient collaboration*: Collaborations typically involve teams with different perspectives, backgrounds, and occasionally, expectations. Good communication is necessary, or else people from different disciplines tend to disagree on what to develop, which

1521-9615 © 2023 IEEE
Digital Object Identifier 10.1109/MCSE.2023.3297753
Date of current version 23 August 2023.

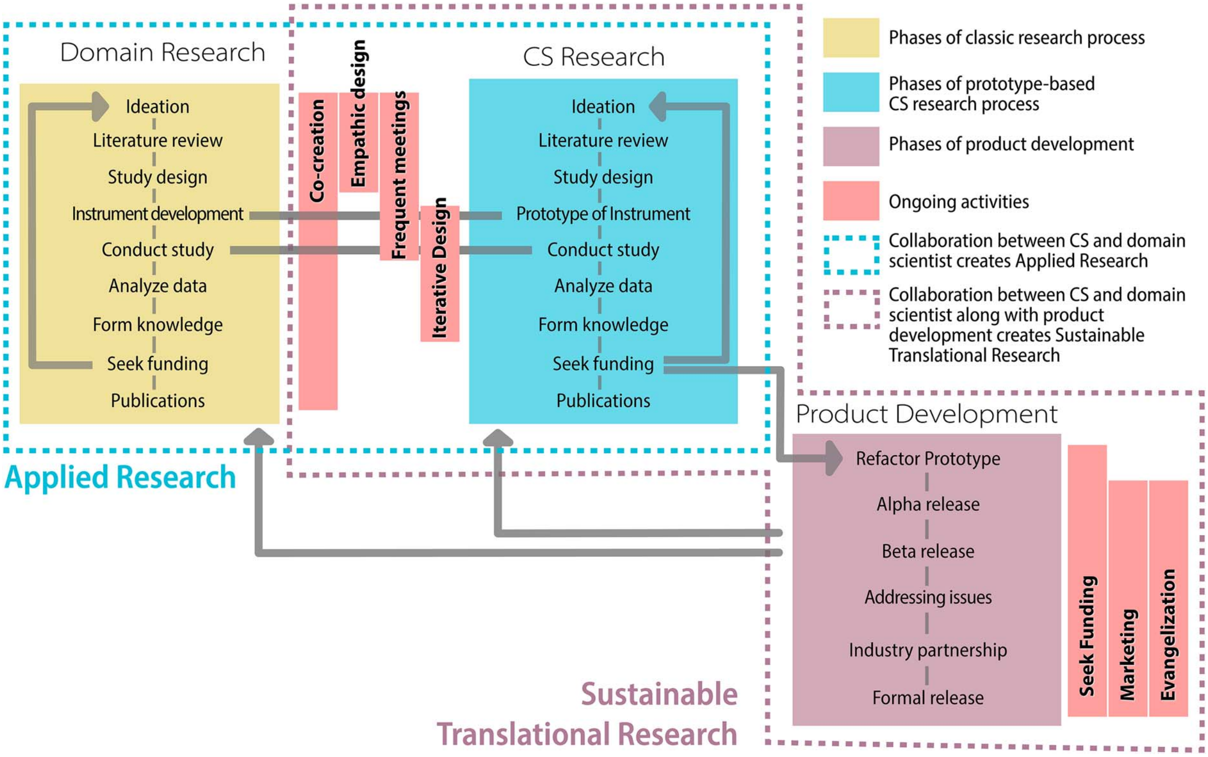

FIGURE 1. The process that defines domain science, computer science (CS), and sustainable translational research.

technologies to deploy, or other critical considerations. For instance, computer scientists may develop new tools using the latest methods while domain scientists prefer familiar tools or updates to existing ones.

2) *Motivating the adoption of new technologies*: More so than developing innovative or updating tried-and-true tools, widespread adoption is critical. A product's longevity is at risk if it relies solely on a small cohort of collaborators.
3) *Ensuring product maintenance postfunding*: Domain scientists strive for sustainable solutions, whereas computer scientists must conduct novel research and publish papers to secure tenure. This dichotomy poses a challenge for maintaining translational software postfunding.

By strategically addressing these obstacles early on, researchers can more effectively navigate the intricate landscape of translational research, unlock its potential to drive innovation, and yield tangible benefits across numerous domains. Given our team's 20-year history developing the Scalable Adaptive Graphics Environment (SAGE) (seven years working on another research project where it was conceived, followed by 13 years as a funded project), we provide insights and propose strategic initiatives to help researchers and organizations overcome the difficulties of translational research and maximize the impact of their work.

WHAT IS SAGE?

SAGE1–3[a] are open source projects developed with US$12 million in funding from the National Science Foundation (NSF). SAGE1, created in 2004 and subsequently funded by the NSF in 2009, was middleware developed in C++ that enabled scientists, educators, and students to collaborate over high-speed networks, sharing visualization data streamed to large-scale high-resolution display walls.[9] As the technical landscape shifted, SAGE2 (Scalable *Amplified Group* Environment), funded in 2013, leveraged emerging JavaScript-based web technologies, increasing its accessibility to broader audiences. By 2018, SAGE2 had more 5000 users across 800 institutions in 18 countries, spanning 22 disciplines.[12,13] In response to the upsurge of remote work during the COVID-19 pandemic, we proposed SAGE3 (*Smart* Amplified Group Environment), a web-based spatial collaborative workspace that integrates

[a]https://sagecommons.org/

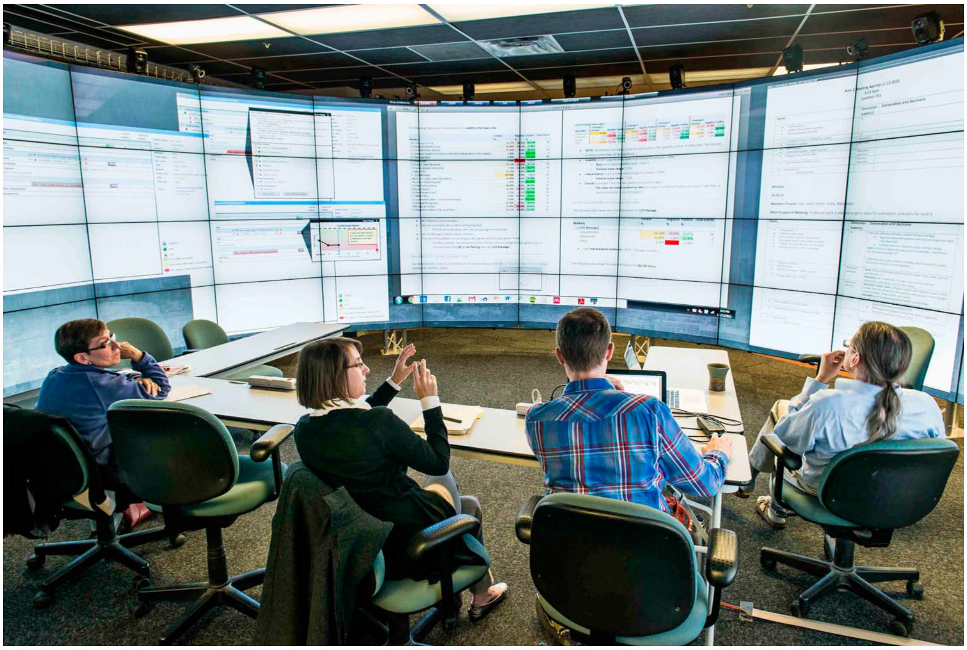

FIGURE 2. Nurses collaborate with computer scientists to develop a user interface to help nurses monitor hospital patients' conditions.

artificial intelligence (AI) and a wide array of applications in a "space to think."[3] SAGE3 expands on collaborative features from various platforms, such as multiscreen sharing, peer programming and code execution with 2D computational notebooks,[11] freehand drawings, sticky notes, and document-based interaction[b] as well as task assistance[c] and cobrowsing.[d] Although SAGE3 is open source, it requires a commercial license for business applications. This licensing model facilitates technology transfer and industry partnerships and helps ensure sustainable funding. SAGE3 combines multiple technologies and frameworks, providing unique features such as natural language processing-based interaction and comprehensive representational state transfer and Python application programming interfaces. However, the lack of standardized architectures brings about several challenges, including the need for data consistency, stringent security, and reliable scalability. Three teams lead the SAGE3 project: Human–Computer Interaction, Front-End and Server-Side Development, and an AI team. Among the contributors are two full-time developers, along with six graduate students and principal investigators who contribute intermittently. Currently, approximately 1400 users are testing SAGE3 beta on a variety of platforms, from high-resolution display walls to personal laptops.

[b]https://miro.com/
[c]https://chat.openai.com/
[d]https://www.surfly.com/

SAGE(LY) ADVICE FOR A SUCCESSFUL TRANSLATIONAL PROJECT

Translational software development faces collaboration, innovation, and maintenance challenges.[7,10] We share effective strategies from our experiences with SAGE and other projects to show that successful translational research can be rewarding.

Fostering Successful Cross-Disciplinary Collaborations

The importance of cross-disciplinary collaboration in translational research has been well documented,[17,18] but only a few articles focus specifically on computer science. Rather than delve into the characteristics of successful cross-disciplinary collaborations, we discuss our experiences building translational software and its unique challenges and opportunities.

Translational software often requires computer scientists and domain experts to collaborate to ensure validity, reliability, and trustworthiness (see Figure 2). These collaborations can be challenging due to differing perspectives, expertise, backgrounds, and objectives.

Miscommunication may arise from different terminologies, and disagreements can occur regarding product features. Computer scientists prioritize cutting-edge tools, while domain scientists prefer familiar ones. These differences hinder coordination in cross-disciplinary teams. We have identified strategies for successful collaboration.

Frequent Formal or Informal Meetings to Further Mutual Learning

Computer scientists should familiarize themselves with discipline scientists' approaches, methods, and terminology and become proficient in communicating their research to them. The SAGE team has engaged in activities like geological surveys, DNA sampling, fish surveys, and reef tagging to gain a deeper understanding of colleagues' disciplines and pain points:

- *Begin with empathy and maintain receptiveness*: Immersive empathic design,[5] a method pioneered at the University of Illinois Chicago's Electronic Visualization Laboratory in the late 1970s and later promoted in design thinking,[4] enables us to improve communication, build trust and engagement, address concerns, and identify mutually beneficial opportunities.
- *Cocreate the product*: Cocreation goes beyond collecting feedback on iterations. It requires active stakeholder involvement, from ideation and prototyping to testing. Involvement in this way, which aligns with empathic design principles, leads to user-centered solutions and enhances understanding between computer scientists and domain scientists.

Designing SAGE with pathologists revealed the need for flexibility in customizing and organizing microscopy images. Cocreation led to a comprehensive set of features applicable to various communities.

- *Release, review, reflect, and refine*: Iterations and refinements should not be segmented by milestones but rather through continual collaboration among the parties involved. Although users cannot always articulate what they want, they are rarely shy to point out what they dislike upon seeing it. Based on empathic design principles, prototypes and iterations can be high-quality mockups.

To maintain institutional knowledge and achieve financially viable projects, a contemporary software development approach is crucial. Before merging student-produced code with SAGE, we hold a rigorous review, ensuring consistency and knowledge retention. SAGE undergoes testing in pair- or team-based demonstration sessions involving noncomputer science participants as well as during classes and demonstrations. Early releases of alpha and beta versions enable public feedback, and critical issues are promptly addressed.

Driving Adoption and Building User Engagement in Translational Software Projects

The Science Gateways Community Institute[8] and other organizations that provide guidance on engagement and adoption strategies are essential to translational projects' success.[15] Another beneficial approach is having a credible and trustworthy domain-science product champion to advocate for the software, highlight its benefits, foster engagement, and demonstrate its value within its community.[16] It is crucial to involve champions in cocreation to identify potential issues, prevent critical flaws, and mitigate obstacles to product acceptance.[16] Other methods of promoting a software product include software demos at conferences, social media use, and user engagement. To avoid abandonment due to bugs or complex or missing features, clear documentation, instructional resources, and online user-developer dialogue, such as on Slack, are essential.

COMPUTER SCIENTISTS SHOULD FAMILIARIZE THEMSELVES WITH DISCIPLINE SCIENTISTS' APPROACHES, METHODS, AND TERMINOLOGY AND BECOME PROFICIENT IN COMMUNICATING THEIR RESEARCH TO THEM.

Marketing accounts for only 5% of SAGE's funding budget, compared to 11% in commercial enterprises.[1] Free or minimal-cost promotional opportunities include mailing lists, conference research booths, presentations at interdisciplinary meetings, university tours, magazine articles, peer-reviewed papers, and social media videos.

Long-Term Funding for Continuity

There is no systematic process for long-term NSF funding for software evolution and maintenance. To sustain translational software postfunding, creative monetization strategies are necessary. Specific funding mechanisms may be better suited to different software types. Here are three strategies that we have either employed or investigated as revenue mechanisms for SAGE's long-term maintenance.

Start-Ups and Industry Partnerships

There are two effective methods for postfunding translational software. The first leverages start-up resources and support from accelerators or incubators, which provide funding, mentorship, and access to industry connections. Universities often offer these programs, providing assistance with funding, marketing, and legal matters. The second approach involves industry partnerships on software customization or training. The SAGE team has worked with Disney, Lucasfilm, Nippon Telephone and Telegraph, Sharp Labs of America, Monsanto Corporation, Caterpillar, Royal Dutch Airlines, Hawai'i State Energy Office, Argonne National Laboratory, National Oceanographic and Atmospheric Administration, NASA, and the U.S. Army Research Laboratory to help integrate SAGE into their work practices.

FUNDING AGENCIES AND POLICYMAKERS RECOGNIZE THE VALUE OF TRANSLATIONAL SCIENCE, BUT CHALLENGES REMAIN.

Technology Transfer

Licensing can offset innovation costs and risks while funding software upkeep. It involves a research lab as a licensor, a licensee integrating the technology into its product, and end users. The licensee assesses market demand, incorporates the technology into its product, and usually rebrands the product. SAGE2, for instance, was repackaged as Multivis and paired with display walls, never marketed as a standalone commercial software. Although this funding model holds promise, it also bears risks as revenues hinge on the licensee's success in marketing and supporting the product.

DESIRED ACTIONS FROM FUNDING AGENCIES AND POLICYMAKERS

Funding agencies and policymakers recognize the value of translational science, but challenges remain. Here are recommendations to promote the growth and success of translational software.

Increased funding for translational research is needed, as federal grant programs have not adjusted for inflation since 2004 and do not adjust for increased institutional overhead costs. This makes it difficult for academia to retain skilled staff due to stiff competition from tech companies. Additionally, dedicated funds are needed for scientists to integrate translational tools into their research pipelines, enhancing product adoption and dissemination.

Journals and funding agencies should educate readers and reviewers on the value of translational research,[6] and promotion of translational research should not be restricted to specialty journals.[14]

Also, tenure decisions should recognize cross-disciplinary collaborations. There must be congruity in messaging between funding agencies—encouraging interdisciplinary work—and universities, wary of junior professors straying from their core disciplines. For translational science to thrive, both theoretical and applied researchers should receive equal recognition.

CONCLUSION

Translational research benefits both domain scientists and computer scientists. Results from effective translational research can inspire new research, provide creative insights, and inform better practices. In our experience, introducing new capabilities into SAGE3 led to improved system responsiveness, reduced interactions, enhanced user experiences, and facilitated discoveries of emergent usage patterns, contributing to a better understanding of scientists' fundamental thought processes. Incorporating the advice shared here early in product development or proposals can be a valuable investment as developing collaboration, technology adoption, and postfunding sustainability models is challenging but essential.

ACKNOWLEDGMENTS

The SAGE project receives major funding from the NSF (Grant 0225642, Grant 0943559, Grant 1339772, Grant 1441963, Grant 2004014, Grant 2003800, Grant 2003387, and Grant 2149133) as well as support from the U.S. Office of Naval Research.

REFERENCES

1. "What marketing budgets look like in 2022," *Gartner*, Jun. 2022. [Online]. Available: https://www.gartner.com/en/articles/what-marketing-budgets-look-like-in-2022
2. D. Abramson and M. Parashar, "Translational research in computer science," *Computer*, vol. 52, no. 9, pp. 16–23, Sep. 2019, doi: 10.1109/MC.2019.2925650.
3. C. Andrews, A. Endert, and C. North, "Space to think: Large high-resolution displays for sensemaking," in *Proc. SIGCHI Conf. Hum. Factors Comput. Syst.*, New York, NY, USA: ACM, 2010, pp. 55–64, doi: 10.1145/1753326.1753336.
4. T. Brown, "Design thinking," *Harvard Bus. Rev.*, vol. 86, no. 6, pp. 84–92, Jun. 2008.

5. Y.-C. Chen, S. Lee, H. Hur, J. Leigh, A. Johnson, and L. Renambot, "Design an interactive visualization system for core drilling expeditions using immersive empathic method," in *Proc. Extended Abstr. Hum. Factors Comput. Syst. (CHI)*, New York, NY, USA: ACM, 2009, pp. 2671–2674, doi: 10.1145/1520340.1520382.
6. J. A. Curran, J. M. Grimshaw, J. A. Hayden, and B. Campbell, "Knowledge translation research: The science of moving research into policy and practice," *J. Continuing Educ. Health Professions*, vol. 31, no. 3, pp. 174–180, Summer 2011, doi: 10.1002/chp.20124.
7. E. Deelman et al., "The Pegasus workflow management system: Translational computer science in practice," *J. Comput. Sci.*, vol. 52, May 2021, Art. no. 101200, doi: 10.1016/j.jocs.2020.101200.
8. S. Gesing et al., "Science gateways: The long road to the birth of an institute," in *Proc. Annu. Hawaii Int. Conf. Syst. Sci.*, 2017, pp. 6243–6252, doi: 10.24251/HICSS.2017.755.
9. B. Jeong et al., "High-performance dynamic graphics streaming for scalable adaptive graphics environment," in *Proc. ACM/IEEE Conf. Supercomputing (SC)*, 2006, pp. 24–24, doi: 10.1109/SC.2006.35.
10. C. Johnson, "Translational computer science at the scientific computing and imaging institute," *J. Comput. Sci.*, vol. 52, May 2021, Art. no. 101217, doi: 10.1016/j.jocs.2020.101217.
11. J. Harden, E. Christman, N. Kirshenbaum, M. Belcaid, J. Leigh, and C. North, "There is no reason anybody should be using 1D anymore": Design and evaluation of 2D Jupyter notebooks, in *Graphics Interface 2023-second deadline*, 2023. [Online]. Available: https://openreview.net/forum?id=Gkogn48LeI
12. J. Leigh et al., "Usage patterns of wideband display environments in e-science research, development and training," in *Proc. 15th Int. Conf. eScience (eScience)*, 2019, pp. 301–310, doi: 10.1109/eScience.2019.00041.
13. T. Marrinan et al., "SAGE2: A new approach for data intensive collaboration using scalable resolution shared displays," in *Proc. 10th IEEE Int. Conf. Collaborative Comput., Netw., Appl. Worksharing*, Nov. 2014, pp. 177–186, doi: 10.4108/icst.collaboratecom.2014.257337.
14. M. Parashar and D. Abramson, "Translational computer science for science and engineering," *Comput. Sci. Eng.*, vol. 23, no. 5, pp. 5–6, Sep./Oct. 2021, doi: 10.1109/MCSE.2021.3109962.
15. M. Parashar and D. Abramson, "Accidental translationists: A perspective from the trenches," *Comput. Sci. Eng.*, vol. 24, no. 4, pp. 70–75, Jul./Aug. 2022, doi: 10.1109/MCSE.2022.3212621.
16. J. E. Perry-Smith and P. Vittorio Mannucci, "From creativity to innovation: The social network drivers of the four phases of the idea journey," *Acad. Manage. Rev.*, vol. 42, no. 1, pp. 53–79, Jan. 2017, doi: 10.5465/amr.2014.0462.
17. D. Rhoten and A. Parker, "Risks and rewards of an interdisciplinary research path," *Science*, vol. 306, no. 5704, pp. 2046–2046, Dec. 2004, doi: 10.1126/science.1103628.
18. D. Stokols, S. Misra, R. P. Moser, K. L. Hall, and B. K. Taylor, "The ecology of team science: Understanding contextual influences on transdisciplinary collaboration," *Amer. J. Preventive Med.*, vol. 35, no. 2, pp. S96–S115, Aug. 2008, doi: 10.1016/j.amepre.2008.05.003.

MAHDI BELCAID is an assistant professor with the Department of Information and Computer Sciences, the University of Hawaiʻi at Mānoa, Honolulu, HI, 96822, USA, and the Hawaiʻi Institute of Marine Biology. Contact him at mahdi@hawaii.edu.

JASON LEIGH is the director of the Laboratory for Advanced Visualization and Applications and the Hawaiʻi Data Science Institute, the University of Hawaiʻi at Mānoa, Honolulu, HI, 96822, USA, and director emeritus of the Electronic Visualization Laboratory, the University of Illinois Chicago, Chicago, IL, 60607, USA. Contact him at leighj@hawaii.edu.

RYAN THERIOT is a software engineer at the Laboratory for Advanced Visualization and Applications, the University of Hawaiʻi at Mānoa, Honolulu, HI, 96822, USA. Contact him at rtheriot@hawaii.edu.

NURIT KIRSHENBAUM is a postdoctoral researcher at the Laboratory for Advanced Visualization and Applications, the University of Hawaiʻi at Mānoa, Honolulu, HI, 96822, USA. Contact her at nuritk@hawaii.edu.

RODERICK TABALBA is a Ph.D. student in computer science at the University of Hawaiʻi at Mānoa, Honolulu, HI, 96822, USA, and a research assistant at the Laboratory for Advanced Visualization and Applications. Contact him at tabalbar@hawaii.edu.

MICHAEL ROGERS is a master's student in computer science at the University of Hawaiʻi at Mānoa, Honolulu, HI, 96822, USA, and a research assistant in the Department of Information and Computer Sciences. Contact him at mlr2010@hawaii.edu.

ANDREW JOHNSON is the director of research at the Electronic Visualization Laboratory, the University of Illinois Chicago, Chicago, IL, 60607, USA. Contact him at ajohnson@uic.edu.

MAXINE BROWN has been a senior research scientist at the Electronic Visualization Laboratory (EVL), the University of Illinois Chicago, Chicago, IL, 60607, USA, since her retirement in 2020. Prior to that, she was an EVL associate director and then director. Contact her at maxine@uic.edu.

LUC RENAMBOT is an associate research professor of computer science at the Electronic Visualization Laboratory, the University of Illinois Chicago, Chicago, IL, 60607, USA. Contact him at renambot@uic.edu.

LANCE LONG is a senior research programmer at the Electronic Visualization Laboratory, the University of Illinois Chicago, Chicago, IL, 60607, USA. Contact him at llong4@uic.edu.

ARTHUR NISHIMOTO was a research assistant at the Electronic Visualization Laboratory, the University of Illinois Chicago, Chicago, IL, 60607, USA. Contact him at anishi2@uic.edu.

CHRIS NORTH is a professor of computer science and associate director of the Sanghani Center for AI and Data Analytics, Virginia Tech, Blacksburg, VA, 24061, USA. Contact him at north@cs.vt.edu.

JESSE HARDEN is a Ph.D. student in computer science at Virginia Tech, Blacksburg, VA, 24061, USA. Contact him at jessemh@vt.edu.