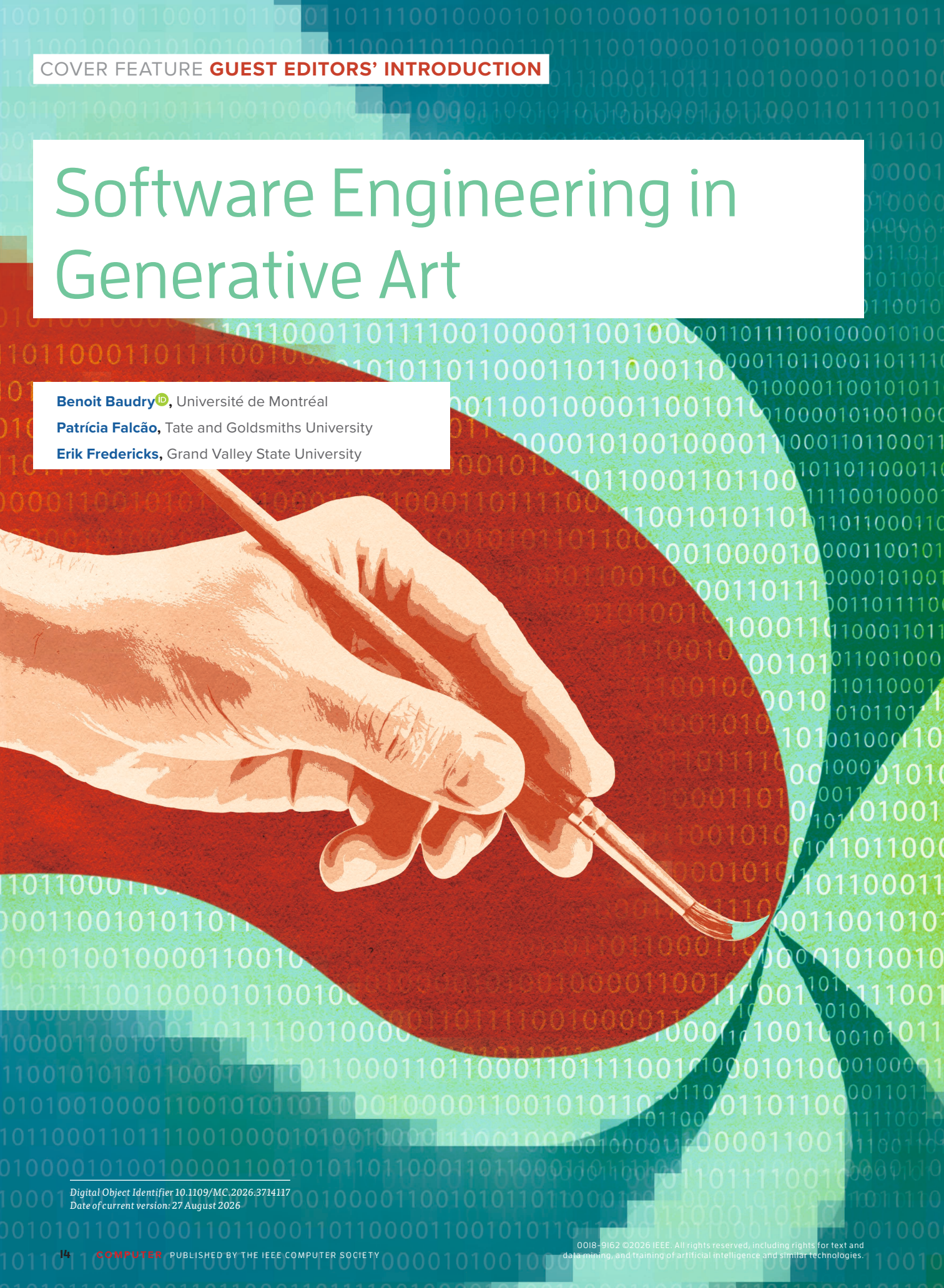
Software Engineering in Generative Art

Benoit Baudry, Université de Montréal
Patrícia Falcão, Tate and Goldsmiths University
Erik Fredericks, Grand Valley State University



Digital Object Identifier 10.1109/MC.2026.3714117
Date of current version: 27 August 2026

This Special Issue on Software Engineering in Generative Art gathers some of the latest research at the intersection of software and art, from international research groups, art institutions, and open source communities.

Software-based art spans a variety of practices such as net.art, generative art, live coding, audiovisual installations, browser art, blockchain-based art. In these contexts, artists develop software to drive any number of processes, from generating visuals and/or sound, to responding to user interactions or choreographing movements of robotic elements, among others. The code also includes some nondeterministic elements such as random numbers, live data processing, or inputs from the audience.

There exists a large number of software engineering concerns in relation to this artistic practice, spanning maintenance, evolution, programming abstractions, and performance. For example, the "correctness" of the artistic output is nontrivial to measure as output esthetics are difficult to formally quantify. The maintenance, preservation and restoration of algorithmic artworks is challenging, as they may rely on outdated runtimes or libraries, on online resources that quickly become unavailable, or on technology that is no longer available. It is challenging to balance performance for real-time interactions and domain-specific abstractions that support artistic expression with code. A large number of these artworks depend on open source libraries, many of them maintained by artists themselves, posing challenges regarding the sustainability of these projects.

This special issue was intended to open a dialogue between artists, researchers, conservators, and software developers in order to capture the latest works that address these challenges. The call was broadly disseminated within all those diverse communities. This special issue gathers novel contributions from these various perspectives.

We present six novel research works at the intersection of software engineering and generative art. Authors' backgrounds are diverse, from artists, art conservators to software developers, researchers and consultants, and

APPENDIX: RELATED ARTICLES

A1. M. Schütze, "When the internet breaks art: A self-healing virtualization framework," *Computer*, vol. 59, no. 9, pp. 18–26, Sep. 2026, doi: 10.1109/MC.2026.3691454.

A2. C. Coleman, "Cultivating the code: Community, care, and the clinic for open source arts," *Computer*, vol. 59, no. 9, pp. 27–36, Sep. 2026, doi: 10.1109/MC.2026.3689040.

A3. K. Kuksenok and L. Tusman, "Processing/p5 defined through practice and learning," *Computer*, vol. 59, no. 9, pp. 37–46, Sep. 2026, doi: 10.1109/MC.2026.3697659.

A4. Y. Westphal, "Hacking art: Software analysis as artistic practice," *Computer*, vol. 59, no. 9, pp. 47–56, Sep. 2026, doi: 10.1109/MC.2026.3686059.

A5. Y. Wang, "Maintenance, evolution, and preservation of generative artworks," *Computer*, vol. 59, no. 9, pp. 57–67, Sep. 2026, doi: 10.1109/MC.2026.3685625.

A6. R. Sacadura, L. Gonçalo, T. Martins, and P. Machado, "Addressing the combinatorial explosion of code-based visual artifacts," *Computer*, vol. 59, no. 9, pp. 69–76, Sep. 2026, doi: 10.1109/MC.2026.3709588.

these contributions reflect their experience in universities, museums, the Processing foundation, or as an independent researcher. They contribute new methods and tools for the preservation and restoration of generative art, as well as for sustaining software libraries for the arts, and for developing the generative art practice further. This unique collection of papers provides an inclusive landscape of the vibrant research area where software engineering and software-based art meet.

Marc Schütze[A1] summarizes his experience with the restoration of two software-based artworks from the early 2000s that relied on software and hardware that was no longer available in 2025. He discusses the restoration process, showing that virtualization and self-healing mechanisms provide a robust foundation for exhibiting generative Web art, and provides a practical method for future restorations.

Christopher D. Coleman[A2] presents the Clinic for Open Source Arts (COSA) and the role it plays to address the fragility of generative art's software supply chain. He discusses COSA's support model, in particular the role of the Open Source Arts Contributors Conference. Through this conference, the lay communities of the supply chain, such as the Processing foundation, engage with more niche libraries, nurturing mutual support for a sustainable future of software-based art.

Kit Kuksenok and Lee Tusman[A3] focus on the Processing/p5 family of libraries across different programming languages. They discuss the fundamental similarities between the libraries across these languages and capture the essentials of these libraries, irrespective of the host language. They analyze other generative art libraries through the lens of these essential properties and discuss how they can guide a community-driven development of Processing/p5 in other languages.

Yannick Westphal[A4] uses the web-based artwork *WebTracer2* by Tom Betts as the starting point to an exploration of "hacking as an artistic practice." He uses various reverse engineering methods to "reveal the nature of algorithmic systems." Westphal starts by altering the *WebTracer2* software's context and environment, then proceeds to hack into its internal interfaces, decompiles the software, and finally reassembles it to create a new logic that only becomes apparent using the View Source function in the resulting webpage.

Yifei Wang[A5] proposes a new aesthetic correctness framework—a quality metric that supports a structured evaluation or aesthetic aspects of an artwork through testable software specifications. The proposal is put in the context of existing challenges and practices of software-based art conservation, providing a useful overview of the field and bringing together the perspectives of software engineering and conservation ethics.

Ricardo Sacadura and colleagues[A6] propose a novel solution to address the explosion of parameter combinations in generative artworks. They adapt a generic algorithm to automatically explore the combinatorial space, and they use two per-trained models to assess the fitness of the generated visual outcomes. They implement this technique on top of the Processing library, in a tool called EvoProteus. They evaluate the approach with four sketches inspired by Hofmann compositions and demonstrate the technique's ability to generate diverse sets of parameters, which produce visually appealing outcomes.

In this special issue researchers and practitioners look into the software engineering challenges related to the use of code as an artistic medium. The readers can discover the latest research works that provide artists with the appropriate tools to develop software for the arts, or provide museums with novel tools and methods to preserve and restore artworks that are based on software systems.

ABOUT THE AUTHORS

BENOIT BAUDRY is a professor in software engineering at the Université de Montréal, Montreal, QC H3T, 1N8 Canada, and Canada Research Chair on Secure and Reliable Software Supply Chains. Contact him at benoit.baudry@umontreal.ca.

PATRÍCIA FALCÃO is a time-based media conservator and doctoral researcher at Tate and Goldsmiths University, SE14 6NW London, U.K. Contact her at patricia.falcao@tate.org.uk.

ERIK FREDERICKS is an associate professor in the College of Computing, Grand Valley State University, Allendale, MI 49401 USA. Contact him at frederer@gvsu.edu.