The Journal of Media Art Study and Theory
Volume 4, Issue 2, 2023
The Art of Media Research
www.mast-journal.org
MAST

Exhibiting Computational Language Art

Nick Montfort
Massachusetts Institute of Technology, USA
University of Bergen, Norway

Introduction

I use the phrase “computational language art” to describe work created for aesthetic purposes where computation and language are the primary mediums. I take the perspectives of literature and poetics in developing such work, so I could just as easily use the term “literary” rather than “language.” The choice is a matter of emphasis and suggests different connections rather than precluding any. For my purposes here, I will focus on how the artworks I discuss engage and are made out of language, rather than on all of their many literary dimensions. While I mention aspects of these works that are common to literature and art, such as intertextuality, reference, and allusion, my approach here is not grounded in poetics, as it is in some of my other discussions.

Christiane Paul makes a division in her *Digital Art* between practices and works in which digital technologies are used “as a Tool” (chapter 1) and those in which such technologies are used “as a Medium” (chapter 2). What I refer to as computational art is mainly treated by Paul as software art, on a single page of the book (Paul 124). Although software art seems to have little representation there, it’s true that there are a huge variety of other digital art practices. Paul’s framing of this practice helps make clear that in computational art, computation is not used instrumentally, but is the stuff out of which artworks are made. And, finally, it shows that computational language art, while a vibrant category from my perspective, is in a subcategory of a subcategory.

At the same time, computational language art connects to rich historical traditions of story generation research that have spanned sixty years, recent generative AI systems based on large language models, and concise, evocative computer programs written by a wide range of artists and programmers. On the last count, these range from well-known art world figures such as Alison Knowles (who programmed *The House of Dust* with James Tenney in the late 1960s) to people who promoted popular engagement with programming, including David Ahl (who edited *101 BASIC Computer Games* and the magazine *Creative Computing,* which reached millions in the early days of home computing). This type of media art, and computational art, has historically been resonant and has welcomed many into the potential of the computer.

Computational language art can manifest itself variously. It has circulated on floppy discs (Kirschenbaum) and via bulletin board systems (Driscoll), and of course is now often accessed via the internet, the Web, and app stores. Leaving aside for the moment that computational language art can be programmed in performance (livecoded) (Blackwell et al.) it can also be run live as it is screened at an art festival or a demoparty (Reunanen). One might imagine that a computer has to be used to present the work, but no: The output and code can appear in a printed book. I edit a series of these, Using Electricity (published by the nonprofit press Counterpath, based in Denver and New York). *Travesty Generator* (Bertram) provides one of several excellent examples of this sort of work, in book form, beyond my series.

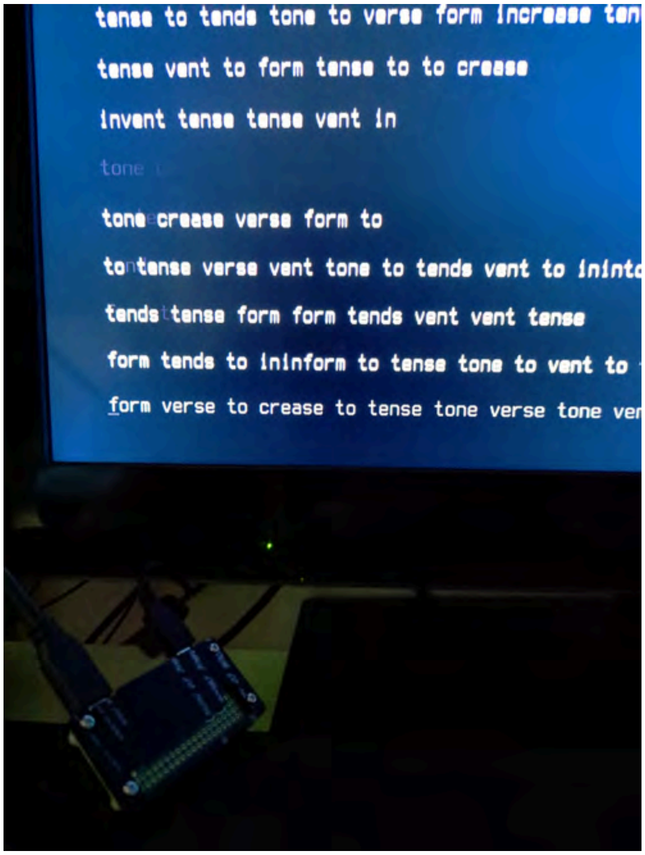
In addition, computational language art can be shared in exhibitions, which has unique advantages—along with challenges. To make my discussion of these concrete, I will focus on four of my own artworks. Before discussing *Round*, *Autofolio Babel*, *Process Pages*, and *Tech Section*, however, I offer one reason these text-forward projects do not have the straightforward appeal of text art.

Computational Language Art and Text Art

Artists have incorporated text (numerals, letters, words, phrases, etc.) in many ways in recent decades, and some of this work has contributed to the development of text art, a.k.a. word art. Contributions have been made by artists working in traditional media, such as Robert Indiana, Jasper Johns, and Kay Rosen, as well as conceptual artists—e.g., Jenny Holzer, On Karawa, Barbara Kreuger, Glenn Ligon, and Lawrence Weiner. Many of Holzer's projects are digital media works in that they use LED displays and projection; they are generally not well understood, however, as *software* artworks or *computational* artworks.

Text art is varied and has important cultural engagements. It can quote from significant texts to keep them alive in memory and bring them into contemporary contexts—Glenn Ligon's *Untitled (I Am A Man)* and his *Give Us a Poem* (which quotes Muhammed Ali's spontaneous "Me—We!") are two examples. Note, however, that language is not needed to achieve this sort of reference and memorialization; it can and has been done with image in other cases. Artist Michael Winker, whose practice involves a visual tracing of spelled-out words and who questions some of the assumptions of Saussurean semiotics, believes that, because of text art, "visitors to galleries were being conditioned to assume that a simple reading of the text would instantly convey the message in language-based works" (Winkler 110). Text art, of course, is also not computational. Whether it involves language or not, computational artwork can invite the viewer to imagine what underlying process is generating that which is visible. This is not alien to the experience of art, since people often imagine the artist's process. But computational work also prompts us to think about non-human, computational processes.

Round



```
tense to tends tone to verse form increase ten
tense vent to form tense to to crease
invent tense tense vent in
tone t
tone crease verse form to
to tense verse vent tone to tends vent to ininto
tends tense form form tends vent vent tense
form tends to ininform to tense tone to vent to
form verse to crease to tense tone verse tone ver
```

Fig 1. Detail of *Round* (2014). Live computational artwork. Landscape-orientation flat-panel FHD display, cables, single-board computer, free software OS, custom free software program.

Round was first shown June 18–21, 2014 in the *Electronic Literature Organization Media Arts Show* in Milwaukee. The work has other manifestations: An online HTML page with CSS and JavaScript, a Python program people are welcome to download and run, and part of the print book *#!* (pronounced "Shebang"). *Round* has an algorithmic basis, with two platform-specific programs exploring how Python and JavaScript implementations differ. It is not abstractly algorithmic, however, but deeply engaged with the material specifics of software and hardware platforms. *Round* displays the digits of π. Instead of using the numerals 0 through 9, it presents short strings of text that correspond to each digit. Eight are words that can stand alone, one is the prefix "in," and one is a line break, shaping the poem into lines of irregular length. Since 3 is represented by "form," 1 by "in", 4 by "tends," 1 by "in", 5 by "tense," and 9 by "verse," the poem begins (3.14159) "form intends intense verse."

Round is a computational poem (also an artwork) that is both non-interactive and deterministic (the text produced is the same each time, as are the digits of π). The poem is infinite, as is π, without a final line or internally specified condition to cause it to stop. Its output serves as a score to be read aloud, allowing sound to suggest sense. I have often presented the beginning of the work in readings and performances, where listeners can appreciate how the elements of language fit together in particular, evocative ways.

There is no list of the digits of π in *Round,* so the program can never come to the end of such a list. The program computes each digit using my implementation of a spigot algorithm. As it runs, the production of text slows down as more and more steps are necessary to determine the next digit. If a multitasking computer is used, it will run other processes more slowly. The computer can physically heat up and its fan can speed up, cooling the processor as it labors to complete computations.

The appearance of the artwork when it is exhibited is simple: A single-board computer, rather than being hidden, is shown as driving the small flat-panel monitor. *Round* serves to remind us that despite all the talk about the immateriality of computing—"the cloud" and the evanescent "Web"—computers are material artifacts, literal machines that run. When *Round* is exhibited, for instance at a festival, it can be revisited as a running process at different points, over time, and visitors can notice the slower rate at which output proceeds.

Autofolio Babel

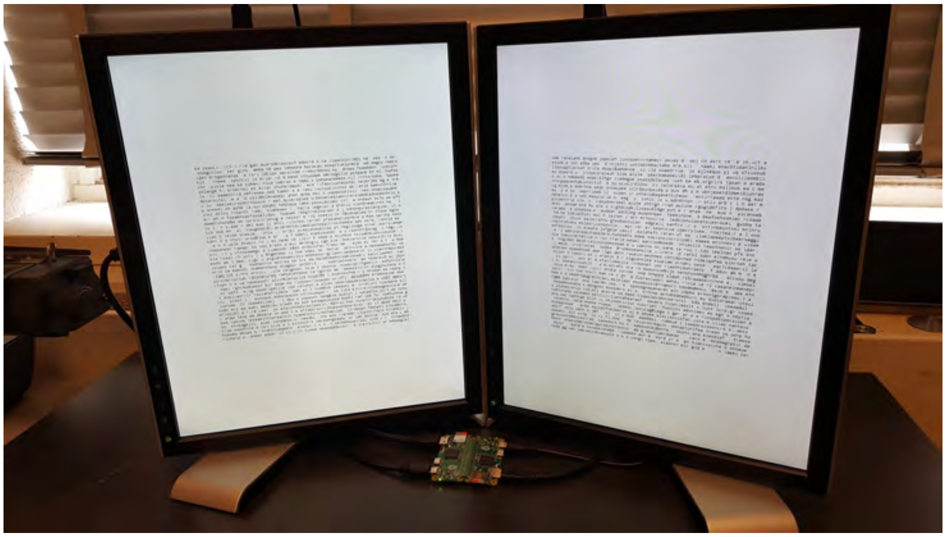


Fig 2. *Autofolio Babel* (2018). Two-channel live computational artwork. Salvaged flat-panel displays with stands, two single-board computers, twist ties, USB cables, power cables, video cables, free software OS, custom free software program.

This piece runs a rotated version of a free, open-source artwork of mine called *Una página de Babel*, first shown at *Translations—Translating, Transducing, Transcoding* in Porto, July 18–22, 2017. However, *Autofolio Babel* constitutes its own work, a hardware and software assemblage incorporating salvaged monitors and single-board computers in a particular self-connected configuration. It was first presented in the group show *History of the Future* at the Boston Cyberarts Gallery, May 26–July 1, 2018.

Autofolio Babel is a folio of flat-panel monitors, recto and verso, providing power via USB to a folio of two single-board computers, bound by wire—which in turn provides the monitors with video signals. Running on each computer is a program that is based on Jorge Luis Borges's description of the pages in the immense but not infinite Library of Babel, a building that houses every possible 410-page book, that is, one book for each possible sequence of a fixed set of characters. The on-screen "pages" are actually Web pages, although the self-contained system runs without a network connection. They are of the same format as the pages of Borges's books but populated by the glyphs in his story, the original Spanish-language text, “La biblioteca de Babel,” instead of the special ones he specified.

“Reading,” of course, must mean something different here, even in comparison to a reading of the hard-to-interpret *Round*. This output is unpronounceable and illegible and would be even if we had a screenshot rather than a rapid progression of pages. The reading has to involve an understanding of the work’s interconnections, a contemplation of how it relates to Borges’s story, and perhaps a willingness to study and even modify the code of *Una página de Babel*. “Reading” can also involve seeing this configuration of hardware and software as a sort of book and considering its relationship to book history and book art.

Process Pages

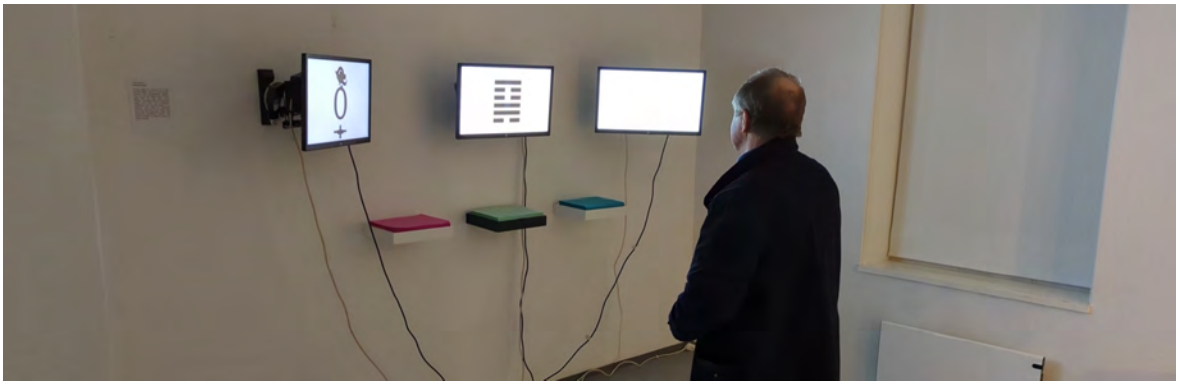


Fig 3. *Process Pages* (2022). Three-channel live computational artwork. Three landscape-orientation flat-panel FHD displays, power cables, video cables, three mini PCs, free software OS, custom free software programs, toner on colored letter-sized paper.

Process Pages is a collection of twenty-one very tiny Web pages with JavaScript, each running live. Each of the three channels presents seven pages in a fixed sequence, synchronized. The work was first installed and shown at PikselXX in Bergen, Norway, November 17–20, 2022. The individual pages are visual poems, artworks, and computational systems—but not the typical sorts of Web pages found online. If anything, these non-interactive pages are more like demoscene productions; their development was informed by sizecoding practices in which programmers aim for extremely small programs. No page exceeds 180 bytes.

Unlike most demoscene productions, these pages explore Unicode and the nature of writing and poetry. *Process Pages* raises the question of why we find certain characters in strange corners of the Unicode standard. Many code blocks do not really represent writing systems and sometimes have elements that are not used in direct connection with writing. These pages explore how rather obvious computational techniques can compellingly manipulate characters, using default fonts and the standard black-on-white presentation of text. As part of this installation, visitors are invited to take one or more sheets of paper from three stacks, one under each monitor. The sheets contain the complete source code of the 21 pages, seven on each sheet. While the gallery setting may not facilitate the examination of this code, it can be brought into another context—home, café, bar—and serve as fodder for individual study or discussion. Visitors are also welcome, of course, to type in and try out any or all of the pages themselves.

Tech Section

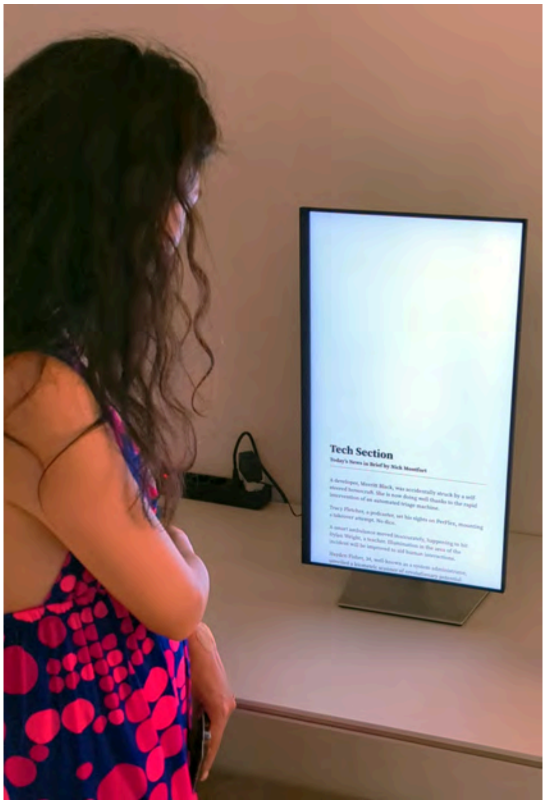


Fig 4. *Tech Section* (detail, 2022). Live computational artwork. Portrait-orientation flat-panel FHD display, cables, single-board computer, free software OS, custom free software program.

A flat panel presents an ever-scrolling ticker of very short, computer-generated news items in *Tech Section,* first shown July 12–15, 2023 in *Resistance* in Coimbra, Portugal. The work is inspired by Félix Fénéon's *Nouvelles en trois lignes* (translated to English as *Novels in Three Lines* by Lucy Sante) and an accident report by Franz Kafka. The texts are generated by concise grammar, although the program at the core of the work is still much longer than in the other three cases.

Today, when people are killed by self-driving vehicles and other autonomous systems, this is news, and definitely worth an article! How remarkable will such incidents be in a few decades? When Fénéon wrote his brief items (initially filler copy for a 1906 newspaper), he mentioned even non-fatal automobile accidents, which certainly would not be newsworthy today. *Tech Section* projects a world in which a failed corporate acquisition and an intriguing invention may be worth a few sentences, but at the same time, a serious collision between robot and human has become barely fit to print.

Surprises can arise, amusing events ensue. Even after the individual incidents begin to seem repetitive, there is variation in how they are expressed that can draw visitors in and encourage them to continue reading. Eventually, visitors may notice that there are set categories into which each item falls and that there is journalistic rhetoric at play. The near-future feed represents human relationships with computing technology—particularly with autonomous robots—using only a few dramatic situations, as Georges Polti called them. It also models some of the classic tropes that technology journalists use in writing stories. The project's compact, self-contained nature invites others to inspect its workings—they can even base their own projects on this system. As with the other three artworks, all source code is made available as free (libre) software.

The gallery is not a reading room, but it is a place where very short narratives of the sort generated here can amass meaning. Visitors who see the slow-scrolling *Tech Section* can't get it going faster; they are invited, instead, to await the deliberate, deadpan emergence of the next item from the bottom of the screen. Exhibition also encourages those who pass by an artwork several times to look at it again, perhaps to read further and in different ways. While some may associate exhibitions with casual glances (at least until a work really attracts the eye), these settings can invite a more relaxed consideration than the fast-paced Web, which is often purposefully, even doggedly browsed. Web reading is often done in isolation, while visitors to a gallery may comment to each other on interesting aspects of a computational language artwork. Of course, such artworks also benefit from being juxtaposed with others that address similar themes or work in similar ways.

New Viewing and Reading Angles

Exhibition has highlighted different things about the relationship been computing and language in each of these four cases. The material nature of the exhibited work is especially important in the first two. *Round* is about how computation is a mechanical process that takes energy and becomes more effortful over time. A visitor who happens to see the work at two different times may notice the pace of its output changing significantly. Unlike the three other artworks here, *Autofolio Babel* is a particular art object incorporating the software artwork *Una página de Babel* twice; It shows how parts of both software and hardware systems are commonly set up to take the outputs of other parts and send their outputs onward. A monitor faces a monitor and a computer faces a computer the way a page faces a page, in a peer-to-peer way. While *Round* can be read aloud (and I present it this way), *Autofolio Babel* is a confusing and incredibly rapid grid of characters, far beyond any reasonable sort of pronunciation.

Process Pages benefits from the gallery setting because its rapid display of Unicode characters can be enticing and because the offer of printed code welcomes those who wouldn't "View Page Source" online. When "Changes" (a page that rapidly displays large hexagrams of the I Ching) is running, it can be visually appealing even to those who know nothing about these glyphs. Those who do might find the extremely fast procession of these symbols strange — they are meant for contemplation, after all. Beyond that, those willing to look at the code and see that these hexagrams are represented in Unicode could begin to ask questions about the nature of modern-day computing: How are they ordered? Why are they placed where they are? Why are these even built into our predominant format for encoding writing systems?

Finally, *Tech Section* takes inspiration from writing that really was not considered literary to begin with, but has been appropriated into literature. The texts produced read, in some ways, like the antiquated filler of a 1906 newspaper, while the gallery presentation also recalls news tickers that can still be seen in places, for instance at 1211 Avenue of the Americas in New York City. The deliberate pace will not drive everyone away: Those who can't bear to wait for the next news item can skip around the screen reading what has already been generated.

These are works that are based on language and computation—they of course have abstract algorithms behind them, but all four artworks include specific programs running on particular platforms. Some (such as *Round*) have several manifestations, including one for the gallery or museum; *Autofolio Babel,* on the other hand, is a unique object to be apprehended only at an exhibition. The presentation of these four artworks in festivals and at galleries was able to invite people in certain ways to the complexities of language—even to its literary dimensions—as well as to engagement with computational art. Computational language art can have the effect of military shock and awe or the powerful theatrical effect that Richard Foreman termed a "disorientation massage," but the works I have discussed here do not tend that way. Many works in this category call for a presentation that will invite viewers into computational and linguistic cognition—to read, to imagine what program is running, and to run that program in their minds.

This is not just a step toward *computational thinking* in a STEM learning sense, or, to expand Science, Technology, Engineering, and Mathematics to include the arts, a STEAM learning mode. It is a mode for the reception of artwork that brings in new types of perception and cognition while also allowing for the serious provocations and significant interventions that art can accomplish. The challenges of making, presenting, and approaching computational language art are notable, but they are worth facing, as the potential of this sort of art is also great.

Acknowledgments:
This research was partially financed by the Research Council of Norway through its Centers of Excellence Scheme, Project No. 332643.

Works Cited

Bertram, Lillian-Yvonne. *Travesty Generator.* Noemi Press, 2019.

Blackwell, Alan F., et al. *Live Coding: A User's Manual.* Cambridge, MIT Press, 2022.

Borges, Jorge Luis. "La biblioteca de Babel." *El jardín de senderos ue se bifurcan.* Buenos Aires: SUR, 1941.

Driscoll, Kevin. *The Modem World: A Prehistory of Social Media.* Yale University Press, 2022.

Fénéon, Felix. *Novels in Three Lines.* New York Review of Books, 2007.

Kirschenbaum, Matthew. "'Poor black squares': Afterimages of the Floppy Disk." *The Routledge Companion to Media Technology and Obsolescence.* Routledge, 2018, pp. 296–310.

Kostopoulou, Ioanna. “Bureaucracy and Desire: Franz Kafka’s Accident Report.” *Open Library of Humanities*, vol. 6, no. 1, 2020, pp. 1-28. doi.org/10.16995/olh.480.

Ligon, Glenn. *Give Us a Poem.* Black PVC and white neon, 75 5/8 × 74 1/4 in. The Studio Museum in Harlem. 2007.

---. *Untitled (I Am a Man).* Oil and enamel on canvas, 40 × 25 in. National Gallery of Art. 1988.

Montfort, Nick. *Autofolio Babel.* Computational book art object. Incorporates *Una página de Babel,* nickm.com, August 25, 2015.; *Cream City Review I/O,* vol. 40, no. .1, Spring/Summer 2016; nickm.com/poems/babel.html.

---. *Process Pages.* Installation, PikselXX: AI AI AI, Bergen, Norway, November 17–20, 2022.

---. *Round.* Multiple manifestations. nickm.com/code/round.py. Python version in print in *#!,* Denver: Counterpath, 2014, pp. 2–9. JavaScript version published by New Binary Press, August 14, 2013, round.newbinarypress.com.

---. *Tech Section.* Installation, ELO 2023, Overcoming Divides: Electronic Literature and Social Change, Coimbra, Portugal, July 12–15, 2023. nickm.com/tech_section/index.html.

---. *Una página de Babel.* nickm.com/poems/babel.html.

Paul, Christiane. *Digital Art.* 3rd ed. Thames & Hudson. 2015.

Reunanen, Markku. “Times of Change in the Demoscene: A Creative Community and Its Relationship with Technology.” PhD thesis, Turun yliopisto, University of Turku, 2017. Winkler, Michael. *The Image of Language: An Artist’s Memoir.* Artists Books Editions, 2021.

As a poet and artist, **Nick Montfort** uses computation as his medium. His computer-generated books range from *#!* to *Golem.* His digital projects include the collaborations *The Deletionist* and *Sea and Spar Between.* Montfort is a scholar, researcher, and educator. His MIT Press

publications include *The New Media Reader* (which he co-edited) and *Twisty Little Passages, The Future,* and *Exploratory Programming for the Arts and Humanities*. He is professor of digital media at MIT and principal investigator in the Center for Digital Narrative at the University of Bergen. He directs a lab/studio, The Trope Tank, and lives in New York City. Email: nickm@nickm.com.