Proceedings of the 7th International Conference on Live Coding (ICLC 2023)
April 19th - 23rd – Utrecht, Netherlands

Addressing Accessibility for Blind and Visually Impaired Live Coders

Matthew Kaney[*]
Livecode.NYC
matthew.s.kaney@gmail.com

William Payne[*]
New York University
william.payne@nyu.edu

Amy Hurst
New York University
amyhurst@nyu.edu

ABSTRACT

Live coding environments that use text input and audio output have the potential to be accessible and expressive for Blind and Visually Impaired (BVI) people. However, little is known about their compatibility with the assistive technologies BVI people use, including screen readers and braille displays, or the experiences of BVI live coders. To address this gap, we formed FiLOrk, a live coding ensemble made up of five BVI high school students at the Filomen M. D'Agostino Greenberg Music School (aka the Fil'). We introduce the goals and open questions guiding FiLOrk and report initial findings from our experiences designing a learning environment and building a custom, collaborative text editor, text.management.

1 Introduction

Most music software uses visual interfaces that pose accessibility barriers to Blind and Visually Impaired (BVI) musicians (Saha and Piper 2020; W. C. Payne et al. 2020). Similarly, many coding environments designed for non-coders—for example, Scratch (Resnick et al. 2009) and Blocky-Talky (Shapiro et al. 2016) for novices, or Max (Puckette 2002) for musicians—represent program logic through interactive visual diagrams. Text-based music environments can be more accessible for BVI people (W. Payne, Ahmed, Gardell, et al. 2022) because they interact well with existing assistive technologies, including text-to-speech screen readers, refreshable braille displays, and high-visibility color schemes (Bhowmick and Hazarika 2017). In addition to being a text-based music practice, live coding has attracted a community that values inclusion along multiple dimensions—access to open-source software, attention to the diversity of the community, and promotion of novel performance practices (A. Skuse 2020). Our goal is to build on this strong foundation, supporting BVI live coders by addressing accessibility barriers and ocularcentrism (the privileging of vision) in tools, teaching, collaboration, and performance.

In this report on a nascent stage of our work, we introduce FiLOrk, a laptop ensemble consisting of five BVI high school learners who write Tidal Cycles (McLean 2014) code in text.management[1], our experimental collaborative editor. We also describe our observations of the challenges and workarounds that emerged as we adapted the environment to learners' diverse vision abilities and equipment, and we discuss the possibilities for greater accessibility in the design of live coding tools and performances.

2 Related Work

FiLOrk draws on prior efforts to make live coding more inclusive and collaborative and to make music systems and coding environments more accessible to BVI people.

[*] These authors contributed equally to this work.
[1] https://text.management/, the project page and online version of our editor.

2.1 Live Coding, Inclusion, and Collaboration

A. Skuse (2020) explores live coding as a practice and community for disabled musicians, identifying particular challenges and opportunities. To inform her work, she interviewed a group of disabled musicians, investigating their preferences and requirements in the context of live coded music. Many findings, such as the need for configurability in software tools, are relevant to FiLOrk even though her participants do not specifically identify as BVI. Like Skuse, our team is engaging disabled people in participatory design of new systems, focusing on BVI learners in order to develop further technological, pedagogical, and social insights.

We are also informed by prior work in collaborative live coding. Researchers and performers have developed a wide variety of tools and strategies for synchronizing code and musical signals, both for in-person and online performances (Roberts et al. 2022). Ensuring that these systems are accessible to disabled users presents additional challenges, and work along these lines is still in the early stages (A. H. C. Skuse and Knotts 2020). Furthermore, electronic music ensembles, including laptop orchestras, tend to use graphic scores and visual signalling (Edwards and Sutherland 2012; Hewitt and Tremblay 2012). Little work has been done so far to address how these performance aspects can be adapted for and by BVI musicians.

2.2 BVI Accessibility in Music and Code Tools

To address the accessibility of live coding for BVI learners, we can look to existing accessibility research on tools for music or coding. Recent interview studies describe how BVI users of commercial music tools address accessibility barriers through custom software and informal knowledge-sharing in spaces like online forums (Green and Baker 2017; W. C. Payne et al. 2020; Saha and Piper 2020). Music companies, including Avid (Smith 2019) and Native Instruments, have begun engaging BVI user testers to ensure accessibility in future development, while community efforts like Flo-Tools[2] and the DAISY Music Braille project[3] have contributed free and/or open source enhancements. Researchers have developed innovative non-visual music systems that use haptic feedback (Chu 2002; Karp and Pardo 2017; Tanaka and Parkinson 2016) or perform well with screen readers and braille displays (W. Payne, Ahmed, Gardell, et al. 2022).

Similarly, BVI programmers can have difficulties navigating code and determining its structure (Mealin and Murphy-Hill 2012; Stefik, Hundhausen, and Patterson 2011). Integrated development environments use complicated designs with visual signifiers to aid sighted developers. This can lead BVI developers to prefer basic text editors and command line interfaces (Sampath, Merrick, and Macvean 2021). Researchers have prototyped non-visual interfaces that use audio output to aid navigation and debugging (Baker, Milne, and Ladner 2015; Armaly, Rodeghero, and McMillan 2018; Stefik et al. 2007; Potluri et al. 2018). While most of these projects are intended for solo developers, a recent VSCode library targets mixed-vision pair programming, featuring additional navigation and audio notifications, such as when two users enter the same line (Potluri et al. 2022). Although these developments can apply to coding in a live performance context, collaborative live coding adds unique constraints, such as timed input, a noisy environment, and performers editing and executing the same code in parallel.

3 FiLOrk (Fil' Laptop Orchestra)

We formed FiLOrk in fall 2022 at the Filomen M. D'Agostino Greenberg Music School (the Fil'), a community music school in New York City that trains BVI musicians of all ages and skill levels. To our knowledge, FiLOrk is the first live coding ensemble made up entirely of participants with vision loss. We aim to understand how collaborative live coding may be made accessible to BVI novices. We view FiLOrk to be a laboratory for the development of pedagogical interventions, technical solutions, and collaboration strategies. Our work is guided by the following Research Questions:

- RQ1: In what ways are current teaching methods and resources ill-suited to BVI learners, and how can the learning environment (covered topics, instructional materials, and classroom setup) be adapted to better support them?
- RQ2: What accessibility barriers exist in live coding tools, and how can new technologies or adaptations to existing technologies incorporate the needs of BVI users?
- RQ3: What strategies do BVI musicians use to communicate amongst themselves and with the audience during performance, and how can new technologies augment that communication?

[2] http://flotools.org, a set of scripts that improve the accessibility of Pro Tools.
[3] https://daisy.org/activities/projects/music-braille/, a project to improve the tools and technical standards surrounding braille music notation.

3.1 Ensemble Overview

FiLOrk consists of five high school learners between the ages of 14 and 18 who enrolled out of interest in the ensemble. We had an existing relationship with these learners through three years of teaching and research activities at the Fil', including the co-design and classroom deployment of SoundCells, a web-based music notation technology (W. Payne, Ahmed, Gardell, et al. 2022; W. Payne, Ahmed, Zachor, et al. 2022). FiLOrk originated out of informal conversations between teachers who wanted an electronic music ensemble, learners who wanted more technology offerings, and members of our research team who practice live coding. In realizing this project, our team has adopted an approach which can be categorized as Design Based Research (Collective 2003).

3.1.1 Technology Used in FiLOrk

Two learners use Mac laptops with the built-in VoiceOver screen reader[4] and braille displays. Two use iPads with magnification and Bluetooth keyboards. One uses an iPad with a keyboard, reading from both the iPad held close to her face and a braille display. Currently, all use Safari because it is perceived to pair best with VoiceOver. The ensemble meets Saturdays for 45 minutes in a traditional classroom. To date, we have explored multiple styles of teaching, including lectures and pre-written code prompts with challenges that learners complete in pairs or as a group.

The ensemble writes Tidal code within text.management, an editor for collaborative live coding we are developing. Using our own editor in the classroom allows us to rapidly address accessibility issues and shape development priorities around classroom needs. The editor is web-based, built on the latest version of the open-source CodeMirror editor[5].

As this is a community music school and is meant to be a safe space for learners, we do not record classroom interactions. The descriptions below reflect our experiences as educators and developers.

3.2 Initial Challenges and Mitigation

3.2.1 Learning Environment

The range of vision abilities informed our approach to teaching and classroom collaboration. Because most learners could not see projected code or each others' screens, the shared editor ensured that all could access content on their own devices. The editors sent code to be evaluated on a single Tidal instance connected to classroom speakers so that everyone shared a clock and audio mix, and screen reader users could hear the contents of their own screens through headphones. Even with a single audio stream, the learning environment was noisy, and learners quickly developed the habit of silencing their own code and asking permission to silence others'. We often split the ensemble into two groups at different tables, each writing patterns with audio panned to the speaker closest to them. Differentiating sample sets, e.g. giving one group melodic content and another rhythmic content, further allowed learners to hear themselves in the combined composition.

Some Tidal concepts are often conveyed alongside imagery[6]. We made tactile graphics of topics (such as waveforms and Euclidean rhythms) using swell form (raised ink) and embossed braille. We observed confusion with these graphics (e.g. distinguishing waveforms from surrounding grid lines) and continue to adjust formatting and information density in response to learner feedback. Additionally, we created an online introductory Tidal curriculum for learner reference outside of rehearsal, and we prepared large print and braille cheat sheets with code snippets and descriptions for each week's topics.

3.2.2 Technical Resources

The classroom and combination of devices resulted in connection and accessibility inconsistencies. Initially, text.management required local network access for collaboration, which was blocked on the school's Wi-Fi. We overcame this by running a local network from our computer until we were able to develop a fully-remote version of the collaboration server. At times, student braille displays would disconnect from their devices meaning they had to switch to VoiceOver text-to-speech on the fly.

Our text editor also imposed technical barriers and challenges. The CodeMirror 6 default setup lacked some VoiceOver support that we solved through identifying and removing incompatible extensions. For example, line highlighting caused VoiceOver to lose focus when the return key was pressed, while default keybindings prevented the delete key

[4] https://www.apple.com/accessibility/vision/, documentation of Apple accessibility features.
[5] https://codemirror.net/, documentation of the CodeMirror library.
[6] E.g. https://tidalcycles.org/docs/reference/cycles/, which visually depicts subdivisions of a cycle.

from announcing the character removed. Other technical challenges remain. For example, VoiceOver in Safari only announces changes when a user has pressed a key, so it was not always clear to learners if someone else had edited shared code.

Finally, we note that the Tidal automatic installation script is insufficient for FiLOrk. The three iPad users cannot download Tidal, so we showed them Estuary (Ogborn et al. 2017) to use outside of class instead. The Atom editor was not accessible with VoiceOver, so the two Mac users installed VSCode for outside use.

4 Discussion

Differing resources and accessibility needs meant that our learners required a variety of hardware and software setups. Existing tools (such as the Tidal installation script) do not fully address these needs and popular components of the live code toolkit, such as the Atom editor, have poor screen reader support. Addressing this requires tools that can be customized to fit a wide variety of circumstances, a requirement that informs our own tool development. Our use of web technologies, particularly our research around accessibility in CodeMirror, means that the accessibility features we develop and document will be useful for the broader world of web-based code editing, within the live coding community and beyond. By designing for this specific population, we can open up new capabilities for all users to adapt their tools, both allowing for greater accessibility, but also enabling further artistic exploration and experimentation.

Going forward, we intend to explore the ways in which live coding performances can be designed around the experience of both BVI performers and audience members. A signature component of Algoraves and other live coded performances is the inclusion of visuals and projected code, offering the audience increased access to the performer's process. Whether projected code is accessible (in the sense of meaningful to a non-technical audience) has been widely explored and theorized (McLean et al. 2010; zmölnig 2016; Lawson and Smith 2019). Its accessibility by a BVI audience is as yet unexplored. Performers like Miyuki Tanaki, Molly Joyce, and Jerron Herman (Joyce 2021, 2020) have integrated accessible forms of communication, such as audio description and sign language, into live performances. This work suggests the potential benefits of modalities beyond projection to live coded performance, and presents accessibility itself as further material for aesthetic exploration.

5 Conclusion and Future Work

We introduced FiLOrk, a live code ensemble made up of five BVI high school learners, and we described how our environment and technology support the ensemble's learning. We then identified opportunities for the live code community to increase the accessibility of tools and performances and to learn from our BVI learners. FiLOrk will finish the Fall 2022 semester with its first performance, after which we will interview learners to gather ideas for future improvements. The class will continue through Spring 2023 with public performances planned for general and BVI audiences. Ahead, we intend to extend FiLOrk to include BVI adult participants, to co-design new teaching materials and text.management accessibility features, and to explore networked performances with remote participants.

References

Armaly, Ameer, Paige Rodeghero, and Collin McMillan. 2018. "AudioHighlight: Code Skimming for Blind Programmers." In *2018 IEEE International Conference on Software Maintenance and Evolution (ICSME)*, 206–16. Madrid, Spain: IEEE. https://doi.org/10.1109/ICSME.2018.00030.

Baker, Catherine M., Lauren R. Milne, and Richard E. Ladner. 2015. "StructJumper: A Tool to Help Blind Programmers Navigate and Understand the Structure of Code." In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, 3043–52. New York, NY, USA: Association for Computing Machinery. https://doi.org/10.1145/2702123.2702589.

Bhowmick, Alexy, and Shyamanta M. Hazarika. 2017. "An Insight into Assistive Technology for the Visually Impaired and Blind People: State-of-the-Art and Future Trends" 11 (2): 149–72. https://doi.org/10.1007/s12193-016-0235-6.

Chu, Lonny L. 2002. "Haptic Design for Digital Audio." In *Proceedings of the IEEE International Conference on Multimedia and Expo*, 2:441–44. IEEE. https://doi.org/10.1109/ICME.2002.1035634.

Collective, Design-Based Research. 2003. "Design-Based Research: An Emerging Paradigm for Educational Inquiry." *Educational Researcher* 32 (1): 5–8.

Edwards, Timothy, and R Benjamin Sutherland. 2012. "Eyes Off the Screen! Techniques for Restoring Visual Freedom in LEO Performance." In *Proceedings of the 1st Symposium on Laptop Ensembles & Orchestras (SLEO)*, 33–40. Baton Rouge, Louisiana. https://ccrma.stanford.edu/~ruviaro/texts/SLEO_2012_Proceedings.pdf.

Green, Lucy, and David Baker. 2017. "Digital Music Technologies: The Changing Landscape." In *Insights in Sound: Visually Impaired Musicians' Lives and Learning*, 158–73. Routledge. https://doi.org/10.4324/9781315266060.

Hewitt, Scott, and Pierre Alexandre Tremblay. 2012. "Notational Approaches for Laptop Ensembles." In *Proceedings of the Symposium on Laptop Ensembles and Orchestras*, 72–75. Baton Rouge, Louisiana. https://ccrma.stanford.edu/~ruviaro/texts/SLEO_2012_Proceedings.pdf.

Joyce, Molly. 2020. "Moving and Thriving Alongside the Physicality of Disability: The Work of Jerron Herman." *Disability Arts Online*. https://disabilityarts.online/magazine/opinion/moving-and-thriving-alongside-the-physicality-of-disability-the-work-of-jerron-herman/.

———. 2021. "Left and Right." https://www.mollyjoyce.com/projects/left-and-right.

Karp, Aaron, and Bryan Pardo. 2017. "HaptEQ." In *Proceedings of the 12th International Audio Mostly Conference on Augmented and Participatory Sound and Music Experiences*, 1–4. New York, NY, USA: ACM. https://doi.org/10.1145/3123514.3123531.

Lawson, Shawn, and Ryan Ross Smith. 2019. "What Am I Looking At?: An Approach to Describing the Projected Image in Live-Coding Performance." In *Proceedings of the Fourth International Conference on Live Coding*, 353. Madrid, Spain: Medialab Prado / Madrid Destino. https://doi.org/10.5281/zenodo.3946290.

McLean, Alex. 2014. "Making Programming Languages to Dance to: Live Coding with Tidal." In *Proceedings of the 2nd ACM SIGPLAN International Workshop on Functional Art, Music, Modeling & Design*, 63–70. FARM '14. New York, NY, USA: Association for Computing Machinery. https://doi.org/10.1145/2633638.2633647.

McLean, Alex, Dave Griffiths, Nick Collins, and Geraint Wiggins. 2010. "Visualisation of Live Code." In *Proceedings of the 2010 International Conference on Electronic Visualisation and the Arts*, 26–30. EVA'10. Swindon, GBR: BCS Learning & Development Ltd.

Mealin, Sean, and Emerson Murphy-Hill. 2012. "An Exploratory Study of Blind Software Developers." In *2012 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC)*, 71–74. https://doi.org/10.1109/VLHCC.2012.6344485.

Ogborn, David, Jamie Beverley, Luis Navarro del Angel, Eldad Tsabary, Alex McLean, and Esteban Betancur. 2017. "Estuary: Browser-Based Collaborative Projectional Live Coding of Musical Patterns." In *Proceedings of the Second International Conference on Live Coding*.

Payne, William Christopher, Alex Yixuan Xu, Fabiha Ahmed, Lisa Ye, and Amy Hurst. 2020. "How Blind and Visually Impaired Composers, Producers, and Songwriters Leverage and Adapt Music Technology." In *The 22nd International ACM SIGACCESS Conference on Computers and Accessibility*, 1–12.

Payne, William, Fabiha Ahmed, Michael Gardell, R. Luke DuBois, and Amy Hurst. 2022. "SoundCells: Designing a Browser-Based Music Technology for Braille and Print Notation." In *19th Web for All Conference*. W4A '22. New York, NY, USA: Association for Computing Machinery. https://doi.org/10.1145/3493612.3520462.

Payne, William, Fabiha Ahmed, Michael Zachor, Michael Gardell, Isabel Huey, R. Luke DuBois, and Amy Hurst. 2022. "Empowering Blind Musicians to Compose and Notate Music with SoundCells." In *The 24th International ACM SIGACCESS Conference on Computers and Accessibility*. ASSETS '22. New York, NY, USA: Association for Computing Machinery. https://doi.org/10.1145/3517428.3544825.

Potluri, Venkatesh, Maulishree Pandey, Andrew Begel, Michael Barnett, and Scott Reitherman. 2022. "CodeWalk: Facilitating Shared Awareness in Mixed-Ability Collaborative Software Development." In *Proceedings of the 24th International ACM SIGACCESS Conference on Computers and Accessibility*. ASSETS '22. New York, NY, USA: Association for Computing Machinery. https://doi.org/10.1145/3517428.3544812.

Potluri, Venkatesh, Priyan Vaithilingam, Suresh Iyengar, Y. Vidya, Manohar Swaminathan, and Gopal Srinivasa. 2018. "CodeTalk: Improving Programming Environment Accessibility for Visually Impaired Developers." In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, 1–11. CHI '18. New York, NY, USA: Association for Computing Machinery. https://doi.org/10.1145/3173574.3174192.

Puckette, Miller. 2002. "Max at Seventeen." *Computer Music Journal* 26 (4): 31–43. http://www.jstor.org/stable/3681767.

Resnick, Mitchel, John Maloney, Andrés Monroy-Hernández, Natalie Rusk, Evelyn Eastmond, Karen Brennan, Amon Millner, et al. 2009. "Scratch: Programming for All." *Communications of the ACM* 52 (11): 60–67.

Roberts, Charlie, Ian Hattwick, Eric Sheffield, and Gillian Smith. 2022. "Rethinking Networked Collaboration in the Live Coding Environment Gibber." In *NIME 2022*.

Saha, Abir, and Anne Marie Piper. 2020. "Understanding Audio Production Practices of People with Vision Impairments." In *The 22nd International ACM SIGACCESS Conference on Computers and Accessibility*, 1–13.

Sampath, Harini, Alice Merrick, and Andrew Macvean. 2021. "Accessibility of Command Line Interfaces." In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. New York, NY, USA: Association for Computing Machinery. https://doi.org/10.1145/3411764.3445544.

Shapiro, R Benjamin, Annie Kelly, Matthew Ahrens, and Rebecca Fiebrink. 2016. "BlockyTalky: A Physical and Distributed Computer Music Toolkit for Kids." In. NIME.

Skuse, Amble. 2020. "Disabled Approaches to LiveCoding, Cripping the Code." In *Proceedings of the 2020 International Conference on Live Coding (ICLC2020)*, 69–77. Limerick, Ireland: University of Limerick. https://doi.org/10.5281/zenodo.3939238.

Skuse, Amble H C, and Shelly Knotts. 2020. "Creating an Online Ensemble for Home Based Disabled Musicians: Disabled Access and Universal Design - Why Disabled People Must Be at the Heart of Developing Technology." In *Proceedings of the International Conference on New Interfaces for Musical Expression*, edited by Romain Michon and Franziska Schroeder, 115–20. Birmingham, UK: Birmingham City University. https://doi.org/10.5281/zenodo.4813266.

Smith, Dave. 2019. "Avid and Berklee College of Music Team Up to Deliver Innovative Accessibility Features for Visually Impaired Music Creators." https://www.avid.com/press-room/2019/12/avid-and-berklee-team-up-for-accessibility.

Stefik, Andreas, Roger Alexander, Robert Patterson, and Jonathan Brown. 2007. "WAD: A Feasibility Study Using the Wicked Audio Debugger." In *15th IEEE International Conference on Program Comprehension (ICPC '07)*, 69–80. https://doi.org/10.1109/ICPC.2007.42.

Stefik, Andreas, Christopher Hundhausen, and Robert Patterson. 2011. "An Empirical Investigation into the Design of Auditory Cues to Enhance Computer Program Comprehension." *International Journal of Human-Computer Studies* 69 (12): 820–38. https://doi.org/https://doi.org/10.1016/j.ijhcs.2011.07.002.

Tanaka, Atau, and Adam Parkinson. 2016. "Haptic Wave," 2150–61. https://doi.org/10.1145/2858036.2858304.

zmölnig, IOhannes m. 2016. "Audience Perception of Code." *International Journal of Performance Arts and Digital Media* 12 (2): 207–12. https://doi.org/10.1080/14794713.2016.1227604.