



PDF Download
3748336.3748340.pdf
19 March 2026
Total Citations: 2
Total Downloads: 358

Latest updates: <https://dl.acm.org/doi/10.1145/3748336.3748340>

RESEARCH-ARTICLE

Drafting the Landscape of Computational Musicology Tools: a Survey-Based Approach

JORGE JUNIOR MORGADO VEGA, Gran Sasso Science Institute, L'Aquila, AQ, Italy

SACHIN SHARMA, Gran Sasso Science Institute, L'Aquila, AQ, Italy

FEDERICO SIMONETTA, Gran Sasso Science Institute, L'Aquila, AQ, Italy

Open Access Support provided by:

Gran Sasso Science Institute

Published: 25 September 2025

Citation in BibTeX format

DLM '25: 12th International
Conference on Digital Libraries for
Musicology
September 26, 2025
Seoul, Republic of Korea

Drafting the Landscape of Computational Musicology Tools: a Survey-Based Approach

Jorge Junior Morgado Vega
Gran Sasso Science Institute
L'Aquila, L'Aquila, Italy
jorge.morgado@gssi.it

Sachin Sharma
Gran Sasso Science Institute
L'Aquila, L'Aquila, Italy
sachin.sharma@gssi.it

Federico Simonetta
Gran Sasso Science Institute
L'Aquila, L'Aquila, Italy
federico.simonetta@gssi.it

Abstract

Since the 60s, musicology has been increasingly impacted by computational tools in various ways, from systematic analysis approaches to modeling of creativity. This article presents a comprehensive assessment of the current state of Computational Musicology tools based on survey data collected from practitioners in the field. We gathered information on tool usage patterns, common analytical tasks, user satisfaction levels, data characteristics, and prioritized features across four distinct domains: symbolic music, music-related imagery, audio, and text. Our findings reveal significant gaps between current tooling capabilities and user needs, highlighting some limitations of these tools across all domains. This assessment contributes to the ongoing dialogue between tool developers and music scholars, aiming to enhance the effectiveness and accessibility of computational methods in musicological research.

CCS Concepts

• **Applied computing** → **Sound and music computing**; • **Software and its engineering** → **Collaboration in software development**.

Keywords

Computational Musicology, Musicological Tools, Digital Humanities

ACM Reference Format:

Jorge Junior Morgado Vega, Sachin Sharma, and Federico Simonetta. 2025. Drafting the Landscape of Computational Musicology Tools: a Survey-Based Approach. In *12th International Conference on Digital Libraries for Musicology (DLfM 2025)*, September 26, 2025, Seoul, Republic of Korea. ACM, New York, NY, USA, 8 pages. <https://doi.org/10.1145/3748336.3748340>

1 Introduction

Computational approaches in musicological study have been documented since the 1960s [3, 17]. These pioneering efforts occurred during the mainframe period, when computing capabilities were limited and highly specialized. The subsequent proliferation of hardware, expansion of programming language options, joint to the advent of personal computing technology, facilitated the development of increasingly specialized analytical processes, enabling individual researchers to conduct Computational Musicology (CM)

projects autonomously [10] and leading CM to gain importance in the whole context of musicology [7].

Today's landscape offers a diverse array of tools spanning multiple domains of computational music analysis research. These tools extend beyond basic statistical score analysis to encompass audio manipulation, algorithmic composition, data retrieval, and numerous other applications.

The multimodal nature of music [31] has contributed to the creation of heterogeneous manipulation approaches and software tools. These divergent implementation paradigms create substantial interoperability challenges, often requiring researchers to convert between multiple data formats or master several different interfaces depending on the task.

This technical fragmentation exacerbates the communication gap between musicologists and software developers, which has been well documented since the first years of development [15]. Despite decades of CM development, these communities frequently operate in parallel with limited mutual understanding [6]. Musicologists may struggle to articulate technical requirements in terms familiar to developers, while developers may create powerful tools that fail to address musicologists' actual research questions or workflow patterns.

In this article, we use a survey-based approach to investigate the main needs and concerns of CM practitioners, thus helping strengthen the connections between user requirements and available technologies in the future. The main contributions of this work are:

- (1) An attempt to understand the state of CM tools from a perspective as objective as possible, in order to inform the future development of tools.
- (2) Quantitative results supporting the identification of strengths and weaknesses of the existing CM tools.
- (3) A discussion of potential approaches to limit the problems that emerged from this work.

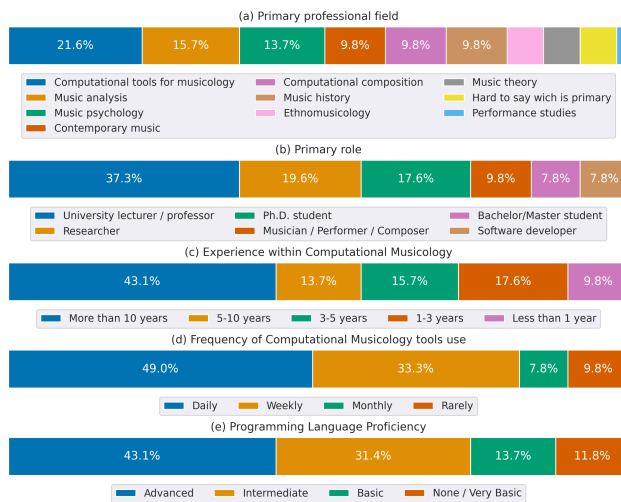
The paper is organized as follows: Section 2 summarizes previous research in this area; Section 3 details the survey methodology applied to musicology practitioners; Section 4 presents the findings from the survey implementation; Section 5 offers our interpretation of the data, highlights the main issues that exist in the CM field, and proposes approaches to tackle them; Section 6 offers concluding observations.

2 Related works

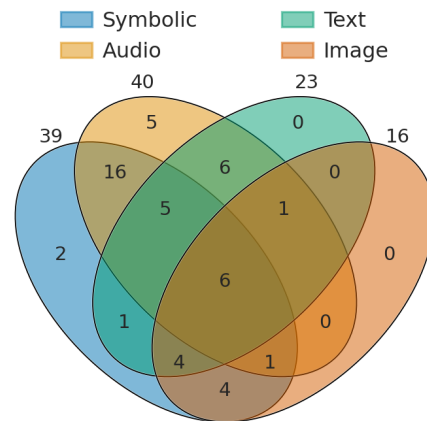
The use of technology in musicology has been widely studied in recent decades. Inskip and Wiering [12] conducted a survey of musicologists, examining their primary specializations, digital skills,



This work is licensed under a Creative Commons Attribution 4.0 International License. *DLfM 2025, Seoul, Republic of Korea*
© 2025 Copyright held by the owner/author(s).
ACM ISBN /25/09
<https://doi.org/10.1145/3748336.3748340>



(a) Professional experience of the participants



(b) Amount of respondents who reported to work with each combination of modalities.

Figure 1: Description of the survey participants

and views on the risks and limitations of using technology in musicological research. Additional studies [23, 33] have explored both the application of computational tools in musicology and the challenges that arise when combining different disciplines such as mathematics, computer science, and musicology. Specifically, Volk and Honingh [33] highlighted the lack of a comprehensive approach due to the development of many different models for specific aspects of music.

Reflecting this inherently multimodal nature of musical phenomena, CM utilizes a diverse range of tools tailored to different representations, or modalities, of music [31]. As computing resources expanded, a broader array of tools emerged, each addressing specific modalities and research objectives.

For symbolic music data, such as scores encoded in formats like Kern, MusicXML, or MIDI, a variety of tools facilitate computational analysis and feature extraction. Early tools like the command-line interface (CLI) based Humdrum Toolkit [11] focused on processing structured text representations of scores. Python libraries such as music21 [8] provide extensive capabilities for parsing, manipulating, and analyzing symbolic structures. Tools like jSymbolic [21] and the more recent Musif [18, 30] specialize in extracting high-level features from symbolic formats, commonly used for tasks like style, genre, or composer identification, a subfield that has been systematically surveyed by Simonetta [28]. Emerging approaches also explore graph-based representations and tools like graphmuse [14] for complex structural analysis.

Complementing symbolic analysis, tools for processing audio data address the acoustic dimension of music. Libraries such as librosa [20], Essentia [5], and Torchaudio [34] provide foundational capabilities for signal processing, feature extraction, and deep learning on audio signals, essential for tasks in Music Information Retrieval (MIR). To support research reproducibility, mirdata [4] standardizes access to commonly used MIR datasets.

Beyond programmatic libraries, graphical user interfaces (GUIs) and interactive environments enhance accessibility and support visual or real-time workflows. Applications like Sonic Visualiser [26] offer visual analysis of audio features (e.g., spectrograms), while OpenMusic [1] provides a visual environment for algorithmic composition. PureData [25] and Max/MSP are widely used for interactive and real-time music generation and performance in experimental and artistic contexts. These tools also exemplify different interface paradigms: some offer graphical user interfaces (e.g., OpenMusic [1], Sonic Visualiser [26], Puredata [25]), prioritizing visual feedback, while others employ command-line interfaces (e.g., Humdrum Toolkit [11]) optimized for automation. Still others exist primarily as programming libraries (e.g., music21 [8]), requiring integration into codebases. Some tools even offer multiple interfaces (e.g., jSymbolic [21]).

Despite this rich ecosystem of tools spanning various modalities, the field faces persistent challenges in usability, interoperability, and alignment with musicological perspectives. Usability remains a significant barrier, particularly for musicologists without extensive programming backgrounds, as noted by Ariza [2], while developers may lack deep musicological insight. Furthermore, evaluating computational models solely based on technical metrics like classification accuracy may overlook musicologically meaningful interpretations, a concern raised by Sturm [32]. Bridging the gap between computational features and human perception is also vital [29]. Pearce and Wiggins [24] emphasized the necessity of incorporating cognitive and perceptual models to support theoretically grounded research in music cognition and analysis.

This non-exhaustive summary of CM tools highlights a heterogeneous landscape that sets problematic aspects in interoperability, usability, and alignment with musicological perspectives. However, how these issues relate to the single tools, modalities, and musicological tasks remains obscure. In this article, we try to shed light on the limits of the existing CM tools.

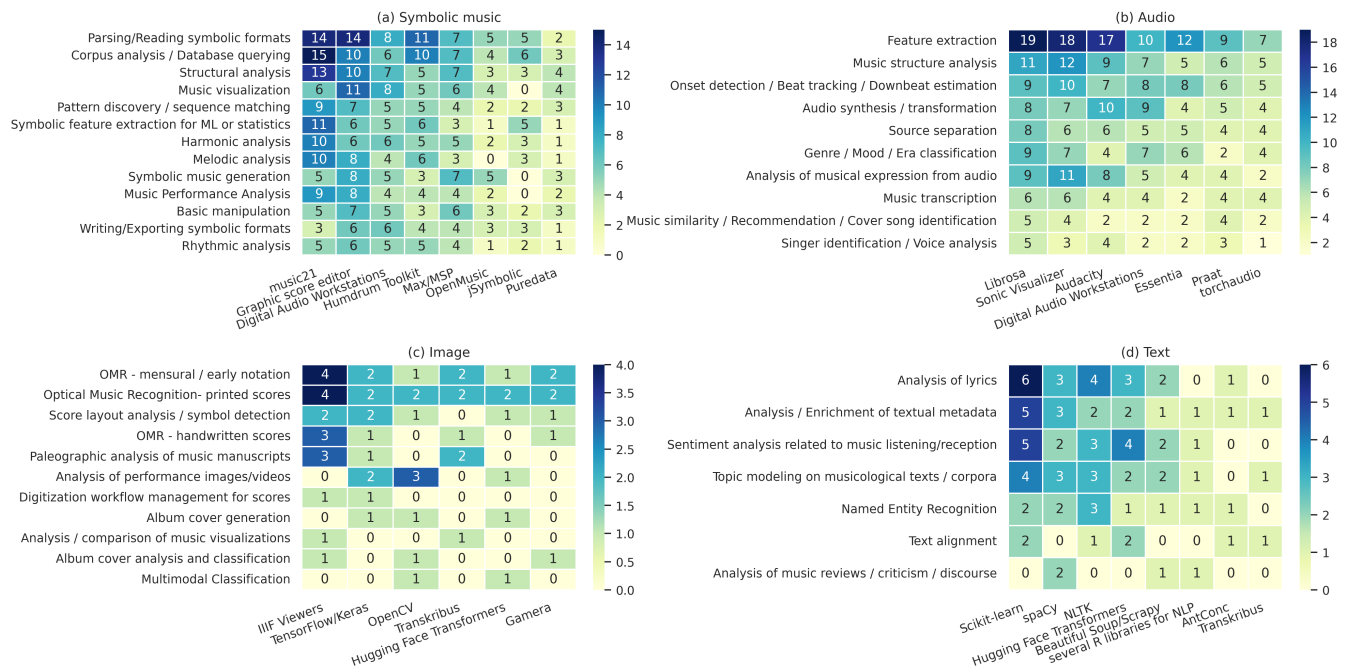


Figure 2: Tasks vs. tools reported. Each value represents the number of participants who reported performing a task X (among all that they perform) and also reported using a tool Y (among all they use). Values are filtered to show the top 80% common matches within each domain.

3 Methodology

To assess the current state of CM tools, we conducted a comprehensive online survey targeting practitioners in the field (e.g., Professors, Researchers, PhD. Students, Software Developers). The survey was designed to collect quantitative and qualitative data on tool usage patterns, user satisfaction, and unmet needs in four primary modalities of music data: (1) symbolic music, (2) audio music, (3) music-related images, and (4) lyrics and music-related texts.

The questionnaire was distributed across authors' direct contacts and public academic mailing lists, including international communities (ISMIR), Italian-based music computing associations (AIMI), and UK/US-based mailing lists about musicology (JISCMail). It must be noted that although these communities are mainly Europe and North America-centric, they still include a range of international professionals.

The subsequent sections provide a detailed overview of each part of the survey.

3.1 Participant demographics

The first questions were dedicated to knowing the background of each interviewee.

What is your primary role in relation to CM? This question helps us understand the professional distribution of the interviewees and enables analysis of how different roles might influence tool preferences and research approaches. Possible values were: Student (Bachelor/Master); PhD Student; Post-doctoral researcher;

Researcher; University Lecturer/Professor; Software Developer; Music Archivist / Librarian / Digital Curator; Musician / Performer / Composer using computational tools; Other (with the possibility of writing them). Upon analysis, in order to avoid confusion, we decided to merge the classes "Researcher" and "Post-doctoral researcher".

What is your primary field of research within musicology? By identifying participants' specific research domains, we can assess how computational methods are being applied across different musicological subdisciplines. Possible values were: Music History; Contemporary music; Ethnomusicology; Music theory; Performance studies; Algorithmic/computational composition; Music psychology; Music neurology; Computational tools for musicology (R&D); Music analysis; Popular music; Other (With the possibility of writing them).

For how many years have you been actively working or studying in the field of CM? Understanding experience levels allows us to distinguish between early-career and more established participants, which can impact perspectives on computational methods. Possible values were: Less than 1 year; 1-3 years; 3-5 years; 5-10 years; More than 10 years.

What is your level of proficiency in programming and using software libraries? This assesses technical fluency, which is crucial for interpreting how users engage with computational musicology tools. Possible values were: None / Very Basic (e.g., only using GUIs); Basic: Using existing scripts/tools, perhaps with minor modifications; Intermediate: Writing/modifying complex scripts,

using library APIs, combining tools; Advanced: Developing complex libraries/tools, contributing to open-source projects.

How often do you use computational tools for your research? The frequency of use gives insight into how integral computational tools are to the respondent's research practice. Possible values were: Daily, Weekly, Monthly, and Rarely.

3.2 Modality-specific inquiries

We then asked participants whether they have experience working in each of the four modalities. Based on their responses, for each modality in which they indicated involvement, we presented a series of modality-specific questions.

Which specific tasks are the most common in your workflow? This question helps us understand the practical activities carried out within each modality and identify common workflows. We provided a list of common tasks tailored to each modality. Participants could select as many tasks as they regularly perform, and also had the option to specify additional tasks not listed.

Name the libraries, software tools, or platforms that you use the most frequently for the mentioned tasks? (at most 5) This allows us to identify which tools are most widely adopted and assess their prevalence across different research contexts. A curated list of tools was presented for each modality. Participants could also add tools not included in the list.

How satisfied are you, overall, with the tools/libraries that you mentioned? Understanding satisfaction levels helps us evaluate whether existing tools meet users' needs and expectations. Participants rated their satisfaction on a scale from 1 to 5, with 5 indicating "Completely satisfied" and 1 meaning "Not satisfied at all".

What do you not like about them? This question aims to highlight common usability issues and barriers to effective tool adoption. Possible values were: Difficult installation; Hard to learn; Has major bugs; Poor documentation; Slow performance; Difficulty in integration with other tools; Limited functionality; Other (with the possibility of writing it).

How large are the corpora you typically analyze? Corpora size informs us about the scale of computational work and potential performance or scalability requirements. Possible values were: Tiny collections (1-10 pieces); Small collections (10-100 pieces); Medium collection (100-1000 pieces); Large collections (1000-1M pieces); Massive collections (1M+ pieces).

List tasks for which you lack adequate tools. What specific features are missing or need improvement? This helps identify gaps in current toolsets and user needs that are not being met by available solutions. Participants could freely describe tasks for which they feel existing tools are insufficient, and detail the features they believe are missing or need enhancement.

3.3 Final questions

We also aimed to understand participants' broader practices, preferences, and challenges when working within the field of Computational Musicology. These questions address how users find resources, choose tools, and manage data integration across modalities.

Where do you usually look for support, new libraries, or information within the field of Computational Musicology?

This question aims to map the main channels through which users access support and stay informed, highlighting community practices and resource visibility. Possible values were: Official documentation of libraries/tools; GitHub (Issues, Discussions, Code search); Stack Overflow; Specific mailing lists (e.g., ISMIR community, music-ir, audiodsp, specific library lists); Academic conferences (e.g., ISMIR, ICMC, CMMR, DLfM, TENOR, SMC) & proceedings; Specific forums (e.g., KVR, ModWiggler, specific software forums); Academic papers / Theses; Blogs / Online tutorials; Colleagues / Personal contacts; Other (with the possibility to write them).

How important are the following factors when choosing a computational musicology/MIR library? Participants were asked to rate the importance on a scale from 1 to 5 (with 5 being the highest importance) of the following factors: Ease of integration with other software; Availability of well-documented API and examples; Speed and efficiency of processing; Cross-platform compatibility (e.g., Windows, MacOS, Linux); Active community support; Availability of pre-trained models; Easy-to-use graphical interface; Programming Language. This helps identify the criteria that guide tool selection, offering insight into user priorities and expectations when adopting new libraries.

Do you encounter difficulties integrating data, methods, or tools from the different data types you worked with (e.g., aligning audio and symbolic data, linking text analysis to musical events, combining OMR output with audio analysis)?

This question assesses the technical and methodological challenges of multimodal integration, a critical aspect in many computational musicology projects. Possible values were: Yes, No, or Not Applicable. Participants who reported difficulties were invited to describe their experiences.

4 Results

We collected a total of 59 responses from participants representing diverse research fields and professional roles. From these, we removed answers that lacked at least one modality and one tool, resulting in 51 final answers. The following sections present the significant findings that emerged from the survey analysis.

Figure 1a details respondent demographics and expertise. Primary research areas (a) included mainly musicological fields, with computational-related topics summing to 35.9%. It must be noted that, during the collection of responses, the category "Hard to say" was named "Other" and had space for users' custom description; it has been successively renamed upon analysis of the answers. As expected, most users were academics, with 37.3% being lecturers or professors, making up nearly 75% of those involved in research (b). Regarding CM tool experience, most were highly experienced (>70% with ≥ 3 years, and >40% with ≥ 10 years) and used tools daily or weekly (c, d). Programming proficiency varied (e); a majority reported advanced or intermediate skills, with others having basic or no experience.

As shown in Figure 1b, symbolic music analysis and audio analysis stood as the most common modalities among practitioners.

Figure 2a reveals how both graphical user interface tools (e.g., graphic score editors, Digital Audio Workstations) and non-GUI

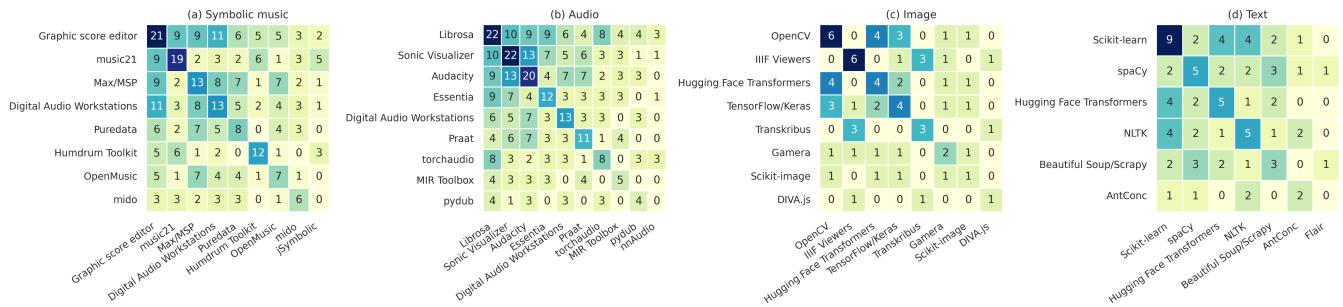


Figure 3: Tools vs. tools reported. Each value represents the number of participants who reported using a pair of tools X and Y together (among all they use). Values are filtered to show the top 80% common matches within each domain.

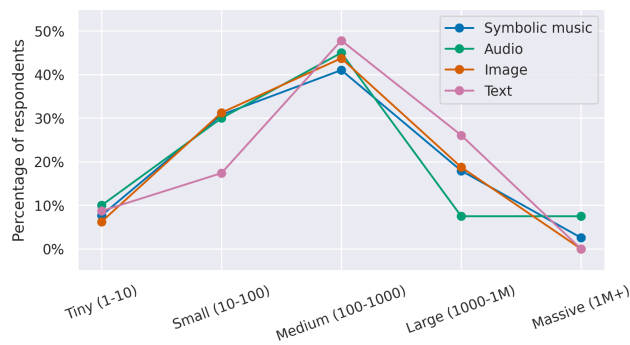


Figure 4: Corpora size

tools (e.g., music21, Humdrum toolkit) are notably employed for tasks dealing with the symbolic music modality. Moreover, visual programming language tools (Max/MSP, PureData, OpenMusic) are also used remarkably frequently, presumably for music generation.

On the audio modality side, Figure 2b shows feature extraction as the most common task, followed by onset detection / beat tracking / downbeat estimation, and music structure analysis. The most used tools were reported to be Sonic Visualiser and the Python library *librosa*, further reinforcing how both graphical user interface and non-GUI tools are widely used across different domains.

Unsurprisingly, the most common tasks within the image modality are related to musical scores and Optical Music Recognition (OMR) – see Figure 2c. Related tools are IIIF viewers for Digital Humanities collections and Deep Learning tools (Tensorflow/Keras). Regarding texts, Figure 2d shows the prevalence of Python libraries for analyzing lyrics and textual metadata.

Analyzing the co-occurrences of the tools, it is clear that GUIs are often used in conjunction by the same users using non-GUIs software – see Figure 3. For instance, graphic score editors were used by 9 people together with music21, while *librosa* was used by 10 people aside from Sonic Visualizer and by 9 people in conjunction with Audacity. This suggests that in the audio and symbolic modalities GUI software represents a complementary solution and not an alternative to non-GUI frameworks. This is partially confirmed by Figure 2.

When asked about the typical data size, medium-sized collections (100-1,000 items) dominate research practices, representing more than 40% of the cases on each modality – see Figure 4.

Across all four modalities, satisfaction levels were predominantly in the neutral to satisfied range – see Figure 5. The average satisfaction level was 3.33 for symbolic music, 3.45 for audio, 3.31 for images, and 3.43 for text.

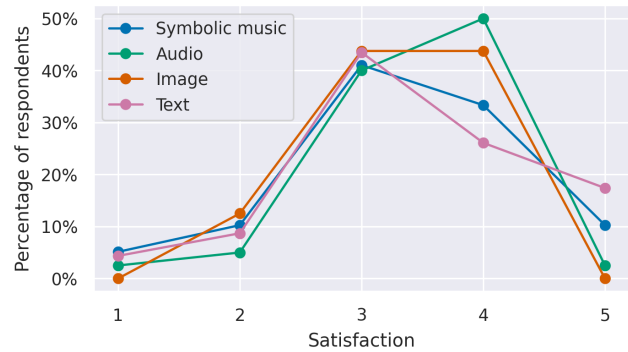


Figure 5: Percentage of answers across satisfaction levels per modality. 1="Not satisfied at all", 5="Completely satisfied"

As Figures 6 and 7 show, limited functionality emerges as the consistent primary concern across all modalities, followed by integration difficulties and poor documentation. Notably, audio tools present a more significant learning curve challenge compared to other domains.

Regarding the importance of different factors when choosing a CM tool, the availability of well-documented APIs and examples emerged as one of the most critical considerations. This was followed by cross-platform compatibility and ease of integration with other software tools.

When asked about integration difficulties of data, methods, or tools, nearly half of the participants reported encountering such issues. Commonly mentioned challenges included the use of different standards across tools, limited support for various data formats, and insufficient capabilities for data translation.

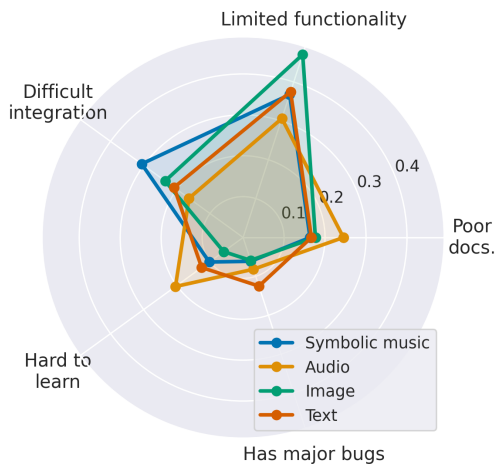


Figure 6: Percentage of reports per unmet need across modalities.

5 Discussion

While our survey provides valuable insights into the CM community’s practices and needs, certain limitations should be acknowledged. The sample size of 51 responses may not fully capture the global perspectives within the CM community. Additionally, the distribution through academic mailing lists and professional networks may introduce a bias toward more established researchers and those already engaged in computational approaches, potentially underrepresenting emerging practitioners. Despite these limitations, the consistent patterns observed across different modalities suggest that our findings can offer guidance for future tool development and community practices.

Next, we will discuss the results presented in the previous section and try to extrapolate useful insights that can positively contribute to the CM community.

Aligning Tools with Practice. This study represents, to our knowledge, the first systematic investigation of the practical needs of CM practitioners. The results provide a basis for computer scientists to align their development efforts with the requirements of the musical community. Research of this nature is vital for effectively guiding multidisciplinary fields like CM towards successful outcomes. Priorities for future tool development should be informed by, or directly geared towards, addressing the most common tasks undertaken by musicologists, thus reducing the heavily reported limited functionality across tools. Historically, reciprocal needs and challenges between computer scientists and musicologists have often been addressed only within the confines of specific projects or research teams. Adopting a more participatory approach to tool development would optimize resources and reduce the redundant development of similar features across independent tools. **Take-away:** The CM community should attempt to establish tighter connections and tools for monitoring the overall mismatch among the different sides of the community.

Towards Integrated Music Analysis. Difficulties in integrating tools and data consistently ranked among the top concerns reported by researchers. This finding is particularly significant given that over 85% of surveyed users engage with multiple modalities in their research workflows. The necessity for cross-modal analysis exposes a key weakness in the current CM landscape: powerful tools often exist in isolation, impeding integrated analysis of music in its entirety. Future tool development should prioritize interoperability to facilitate seamless data sharing and synchronization between different modalities [16, 19]. While this topic has been explored within MIR (e.g., audio-to-score alignment, lyrics alignment), practical tools for integrating information and analysis derived from disparate modalities remain uncommon [9], especially within user-friendly GUI environments. **Take-away:** A larger effort towards the standardization of CM tools and their interoperability and integration would help towards a more holistic analysis of music.

The Data Scale Challenge. The analysis highlights the prevalence of small to medium-sized datasets across various musical modalities (Figure 4). This presents a notable challenge for CM, particularly in the context of leveraging Deep Learning (DL) methods, which have become standard in other areas of MIR but are less frequently applied in CM. This limitation is not solely technological; it is often intrinsically linked to the nature of the data itself, such as the scarcity or incompleteness of historical musical sources. Consequently, researchers face fundamental hurdles in developing reproducible, state-of-the-art DL models for musicological analysis. This necessitates the exploration of innovative approaches capable of extracting meaningful insights from limited data. For instance, strategies like Active Learning are being explored to minimize the manual annotation effort required for Optical Music Recognition (OMR) in historical manuscripts, directly tackling the data scarcity problem [27]. The infrequent adoption of advanced NLP methods, such as transformer-based architectures, among musicologists further underscores this point. **Take-away:** developing tools for CM using LLM-based approaches may be geared towards different purposes than what musicologists seek in the short term.

Bridging the Interface Gap. A clear preference for GUIs among musicologists was observed, contrasting with the tendency of computer scientists to develop libraries and command-line tools. This divergence highlights a critical misalignment in development priorities. While the limited technical background of some humanists surveyed likely contributes to this preference, the increasing integration of LLMs into everyday tools may be perceived by some as diminishing the urgency for widespread technical training in humanities curricula [13]. However, rather than assuming musicologists will acquire extensive coding skills, a more pragmatic approach involves computer scientists developing software that facilitates interaction for non-technical users. Conversely, a valid concern exists that complex visual interfaces could potentially introduce bias into the analyses [22]. Providing modular building blocks in libraries and command-line tools offers developers a defensive position against this. The challenge lies in fostering both approaches: enabling technically proficient individuals to investigate potential biases and develop novel analysis methods while simultaneously providing accessible visual and interactive tools for musicologists with less technical expertise. **Take-away:** By

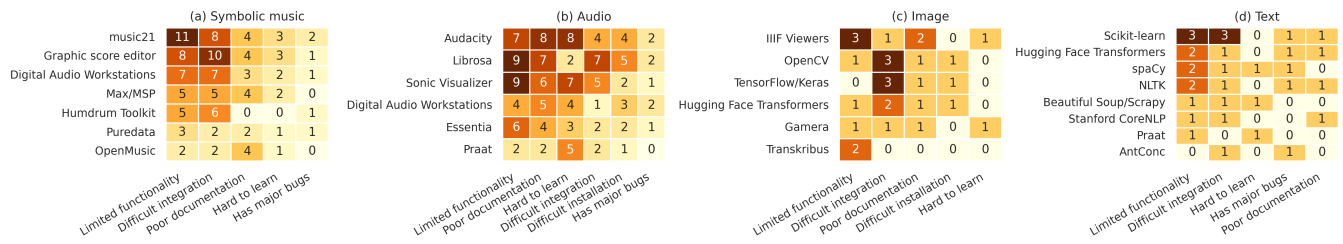


Figure 7: Tools vs. main concerns reported. Each value represents the number of participants who reported using a tool X (among all that they use) and also reported a concern Y (among all they reported). Values are filtered to show the top 75% common matches within each domain.

creating or contributing to visual-based tools, or integrating them into command-line interfaces and libraries, developers would better serve the practical needs of the musicological community while reducing the technical barriers to adoption.

6 Conclusions

The results reveal not only the primary concerns of practitioners in the field regarding the use of CM tools but also offer valuable insights into their workflows, tasks, tool preferences, and data practices. These findings shed light on how CM tools are integrated into daily research activities and point to specific areas where usability, functionality, and support can be improved.

Our survey highlights the significant gap that still exists between the music and developer communities, emphasizing that deeper collaboration is necessary to create tools that better serve the needs of computational musicologists. However, such collaboration is inherently challenging, primarily due to the mutual lack of domain-specific knowledge that prevents practitioners from articulating their precise requirements effectively.

Consequently, targeted surveys and similar investigative strategies become crucial in bridging this communication divide and gaining a comprehensive understanding of the nuanced needs across these interdisciplinary domains.

Acknowledgments

This work has been funded by the European Union (Horizon Programme for Research and Innovation 2021-2027, ERC Advanced Grant “The Italian Lauda: Disseminating Poetry and Concepts Through Melody (12th-16th century)”, acronym LAUDARE, project no. 101054750). The views and opinions expressed are, however, only those of the author and do not necessarily reflect those of the European Union or the European Research Council. Neither the European Union nor the awarding authority can be held responsible for such matters.



References

[1] Carlos Agon. 1998. *OpenMusic: Un langage visuel pour la composition musicale assistée par ordinateur*. Ph.D. Dissertation. Paris 6.

[2] Christopher Ariza. 2008. The Interdisciplinary Future of Music Informatics. *Computer Music Journal* 32, 3 (2008), 66–70.

[3] Gary Berling, Barry S Brook, Lejaren A Hiller, Jan P LaRue, and George W Logemann. 1966. Addendum: Writings on the Use of Computers in Music. In *College Music Symposium*. JSTOR, 143–157.

[4] Rachel M Bittner, Magdalena Fuentes, David Rubinstein, Andreas Jansson, Kenwoo Choi, and Thor Kell. 2019. mirdata: Software for Reproducible Usage of Datasets. In *ISMIR*. 99–106.

[5] Dmitry Bogdanov, Nicolas Wack, Emilia Gómez, Sankalp Gulati, Perfecto Herrera, Oscar Mayor, Gerard Roma, Justin Salamon, José Zapata, and Xavier Serra. 2013. Essentia: an open-source library for sound and music analysis. In *Proceedings of the 21st ACM international conference on Multimedia*. 855–858.

[6] Lee Cheng and Samuel Leong. 2017. Educational affordances and learning design in music software development. *Technology, Pedagogy and Education* 26, 4 (2017), 395–407.

[7] Nicholas Cook. 2004. *Computational and Comparative Musicology*. Oxford University Press, New York, Chapter 6. doi:10.1093/acprof:oso/9780195167498.003.0006 online edn, Oxford Academic, 1 May 2008, accessed 16 May 2025.

[8] Michael Scott Cuthbert and Christopher Ariza. 2010. music21: A toolkit for computer-aided musicology and symbolic music data. (2010).

[9] Johanna Devaney, Daniel McKemie, and Alex Morgan. 2024. pyAMPACT: A Score-Audio Alignment Toolkit for Performance Data Estimation and Multimodal Processing. In *International Society for Music Information Retrieval, Late Breaking Demo*.

[10] Walter B Hewlett and Eleanor Selfridge-Field. 1991. Computing in musicology, 1966–91. *Computers and the Humanities* 25 (1991), 381–392.

[11] David Huron. 2002. Music information processing using the Humdrum toolkit: Concepts, examples, and lessons. *Computer Music Journal* 26, 2 (2002), 11–26.

[12] Charles Inskip and Frans Wiering. 2015. In their own words: using text analysis to identify musicologists’ attitudes towards technology. In *16th International Society for Music Information Retrieval Conference*.

[13] Andres Karjus. 2025. Machine-assisted quantizing designs: augmenting humanities and social sciences with artificial intelligence. *Humanities and Social Sciences Communications* 12, 1 (2025), 1–18.

[14] Emmanouil Karystinaios and Gerhard Widmer. 2024. Graphmuse: A library for symbolic music graph processing. *arXiv preprint arXiv:2407.12671* (2024).

[15] Jan LaRue. 1970. *Some Computer Aids to Musicology*. (1970).

[16] David Lewis, David M. Weigl, Joanna Bullivant, and Kevin R. Page. 2018. Publishing musicology using multimedia digital libraries: creating interactive articles through a framework for linked data and MEI (DLfM ’18). Association for Computing Machinery, New York, NY, USA, 21–25. doi:10.1145/3273024.3273038

[17] Harry B Lincoln. 1970. The current state of music research and the computer. *Computers and the Humanities* 5, 1 (1970), 29–36.

[18] Ana Llorens, Federico Simonetta, Martín Serrano, and Álvaro Torrente. 2023. Musif: A Python Package for Symbolic Music Feature Extraction. In *Proceedings of the Sound and Music Computing Conference*. Stockholm, Sweden. doi:10.5281/zenodo.8341211

[19] Luca Andrea Ludovico, Adriano Baratè, Federico Simonetta, and Davide Andrea Mauro. 2019. On the Adoption of Standard Encoding Formats to Ensure Interoperability of Music Digital Archives: The IEEE 1599 Format. In *6th International Conference on Digital Libraries for Musicology*. ACM, 20–24. doi:10.1145/3358664.3358665

[20] Brian McFee, Colin Raffel, Dawen Liang, Daniel PW Ellis, Matt McVicar, Eric Battenberg, and Oriol Nieto. 2015. librosa: Audio and music signal analysis in python. *SciPy 2015* (2015), 18–24.

[21] Cory McKay, Julie Cumming, and Ichiro Fujinaga. 2018. JSYMBOLIC 2.2: Extracting Features from Symbolic Music for use in Musicological and MIR Research. In *ISMIR*. 348–354.

- [22] Aliaksei Miniukovich, Simone Sulpizio, and Antonella De Angeli. 2018. Visual complexity of graphical user interfaces. In *Proceedings of the 2018 international conference on advanced visual interfaces*. 1–9.
- [23] Maria Panteli, Emmanouil Benetos, and Simon Dixon. 2018. A review of manual and computational approaches for the study of world music corpora. *Journal of New Music Research* 47, 2 (2018), 176–189.
- [24] Marcus T. Pearce and Geraint A. Wiggins. 2012. Auditory Expectation: The Information Dynamics of Music Perception and Cognition. *Topics in Cognitive Science* 4, 4 (2012), 625–652.
- [25] Miller Puckette et al. 1996. Pure data: another integrated computer music environment. *Proceedings of the second intercollege computer music concerts* (1996), 37–41.
- [26] Michael Rector. 2018. Sonic Visualizer: A Tool For Closer Listening. *American Music Teacher* 67, 5 (2018), 21–23.
- [27] Sachin Sharma and Federico Simonetta. 2025. Experimenting Active and Sequential Learning in a Medieval Music Manuscript. In *2025 IEEE 35th International Workshop on Machine Learning for Signal Processing (MLSP)*.
- [28] Federico Simonetta. 2025. Style-Based Composer Identification and Attribution of Symbolic Music Scores: A Systematic Survey. *Transactions of the International Society for Music Information Retrieval* (2025). doi:10.48550/arXiv.2506.12440
- [29] Federico Simonetta, Federico Avanzini, and Stavros Ntalampiras. 2022. A Perceptual Measure for Evaluating the Resynthesis of Automatic Music Transcriptions. *Multimedia Tools and Applications* (2022). doi:10.1007/s11042-022-12476-0
- [30] Federico Simonetta, Ana Llorens, Martín Serrano, Eduardo García-Portugués, and Álvaro Torrente. 2023. Optimizing Feature Extraction for Symbolic Music. In *Proceedings of the 24th International Society for Music Information Retrieval Conference*. Milan. doi:10.5281/zenodo.10265409
- [31] Federico Simonetta, Stavros Ntalampiras, and Federico Avanzini. 2019. Multimodal Music Information Processing and Retrieval: Survey and Future Challenges. In *Proceedings of 2019 International Workshop on Multilayer Music Representation and Processing*. IEEE Conference Publishing Services, Milan, Italy, 10–18. doi:10.1109/mmpr.2019.00012
- [32] Bob L. T. Sturm. 2014. A Simple Method to Determine if a Music Information Retrieval System is a “Horse”. *IEEE Transactions on Multimedia* 16, 6 (2014), 1636–1644.
- [33] Anja Volk and Aline Honingh. 2012. Mathematical and computational approaches to music: challenges in an interdisciplinary enterprise. *Journal of Mathematics and Music* 6, 2 (2012), 73–81.
- [34] Yao-Yuan Yang, Moto Hira, Zhaoheng Ni, Artyom Astafurov, Caroline Chen, Christian Puhersch, David Pollack, Dmitriy Genzel, Donny Greenberg, Edward Z Yang, et al. 2022. Torchaudio: Building blocks for audio and speech processing. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 6982–6986.