The Impact of Salient Musical Features in a Hybrid Recommendation System for a Sound Library

Jason Brent Smith[1], Ashvala Vinay[1] and Jason Freeman[1]

[1]*Georgia Tech Center for Music Technology*
840 McMillan Street NW, Atlanta, Georgia, USA, 30308

Abstract

EarSketch is an online learning environment that teaches coding and music concepts through the computational manipulation of sounds selected from a large sound library. It features sound recommendations based on acoustic similarity and co-usage with a user's current sound selection in order to encourage exploration of the library. However, students have reported that the recommended sounds do not complement their current projects in terms of two areas: musical key and rhythm. We aim to improve the relevance of these recommendations through the inclusion of these two musically related features. This paper describes the addition of key signature and beat extraction to the EarSketch sound recommendation model in order to improve the musical compatibility of the recommendations with the sounds in a user's project. Additionally, we present an analysis of the effects of these new recommendation strategies on user exploration and usage of the recommended sounds. The results of this analysis suggest that the addition of explicitly musically-relevant attributes increases the coverage of the sound library among sound recommendations as well as the sounds selected by users. It reflects the importance of including multiple musical attributes when building recommendation systems for creative and open-ended musical systems.

1. Introduction

EarSketch [1] is a computational music remixing environment designed to teach music and computing concepts through the process of writing code to creatively manipulate audio loops. It is a web application that contains a code editor for students to write Python or JavaScript code using a custom API, and a Digital Audio Workstation for them to view and listen to the musical output produced by their code.

Previous analysis of EarSketch users revealed that a sense of creative ownership and expression for their work has been linked to intentions to persist in computer science education [2]. To this end, EarSketch was designed with the goal of being authentic to industry tools in terms of music production, musical content, and computing languages. It achieves this with the design of its interface and API as well as with the inclusion of a large sound library for students to explore and find sounds that are personally expressive and meaningful to them.

EarSketch contains a library of over 4,500 sounds produced by professional artists such as sound designer Richard Devine, hip-hop producer/DJ Young Guru, and additional stems from popular musicians such as Alicia Keys, Ciara, Common, Dakota Bear, Irizzary y Caraballo, Jayli Wolf, Khalid, Milk + Sizz, Pharrell Williams, and Samian. Users are able to search for sounds by name, filter by artist, genre, instrument, or key signature, and mark them as favorites for future use (see Fig 1) and can preview or copy these sounds into their code as constants.

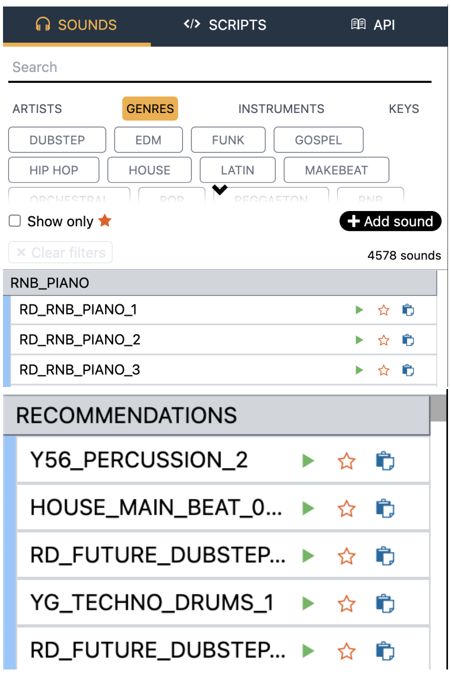


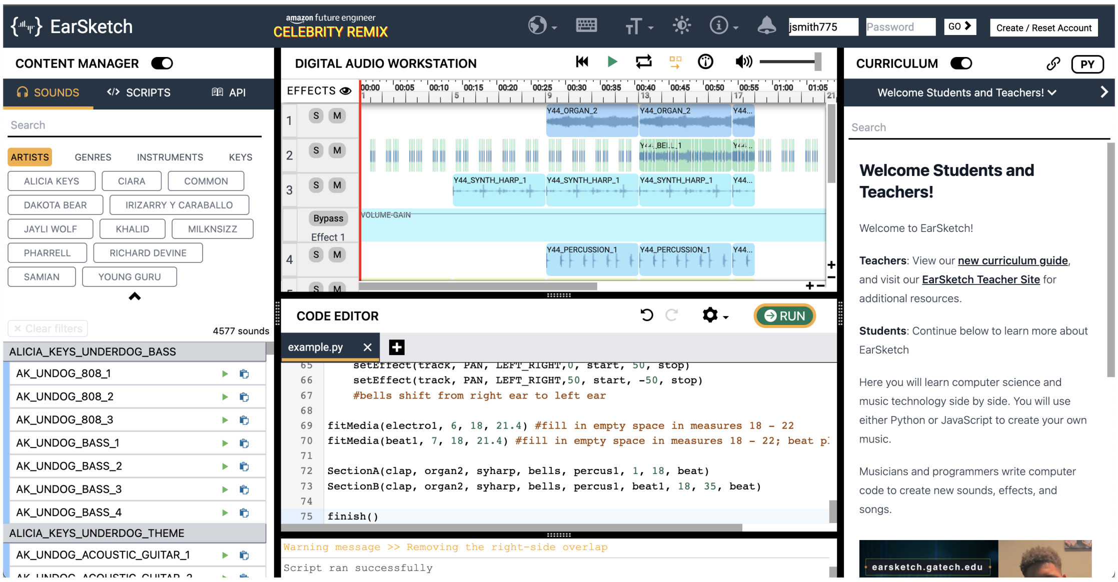
Figure 1: View of EarSketch Sound Browser interface (top), with example recommendations (bottom).

A previous analysis of 20,000 user-created scripts showed that fewer than 200 library sounds were used in over 1% of scripts and under 20 sounds were used in over 10% of

Joint Proceedings of the ACM IUI Workshops 2023, March 2023, Sydney, Australia

jsmith775@gatech.edu (J. B. Smith); ashvala@gatech.edu (A. Vinay); jason.freeman@gatech.edu (J. Freeman)
0000-0002-7075-6132 (J. B. Smith); 0000-0002-2487-2052 (A. Vinay); 0000-0003-3827-1060 (J. Freeman)

CEUR Workshop Proceedings (CEUR-WS.org)



```
    setEffect(track, PAN, LEFT_RIGHT,0, start, 50, stop)
    setEffect(track, PAN, LEFT_RIGHT,50, start, -50, stop)
    #bells shift from right ear to left ear

fitMedia(electro1, 6, 18, 21.4) #fill in empty space in measures 18 - 22
fitMedia(beat1, 7, 18, 21.4) #fill in empty space in measures 18 - 22; beat p

SectionA(clap, organ2, syharp, bells, percus1, 1, 18, beat)
SectionB(clap, organ2, syharp, bells, percus1, beat1, 18, 35, beat)

finish()
```

```
Warning message >> Removing the right-side overlap
Script ran successfully
```

Welcome Students and Teachers!

Welcome to EarSketch!

Teachers: View our new curriculum guide, and visit our EarSketch Teacher Site for additional resources.

Students: Continue below to learn more about EarSketch

Here you will learn computer science and music technology side by side. You will use either Python or JavaScript to create your own music.

Musicians and programmers write computer code to create new sounds, effects, and songs.

Figure 2: View of the EarSketch interface, with Sound Browser (left), Digital Audio Workstation (top), and Code Editor (bottom), and Curriculum (right).

scripts. It was hypothesized that this was due to difficulty in navigating the sound browser, as users reported that it was hard to discover groups of sounds relevant to their current work. In order to address this under-utilization of the sound library and to promote compositional diversity among its users' projects, a recommendation system was added to EarSketch [3].

Diversity and coverage, measures of how different a set of recommendations are from each other and how much of a set of available options is being recommended, are common design goal of recommendation systems [4]. Recommendation systems to present diverse compositional material are prevalent in music production platforms with which EarSketch aligns its design goals. The EarSketch sound recommendation system was designed to assist in the process of navigating the sound library by presenting relevant, novel sounds for users to include in their code. By giving users more easily-accessible sound options that match the content of their in-progress compositions, the system aims to improve the variety of sounds that users preview and copy into their scripts. It uses collaborative filtering [5] and acoustic similarity metrics to minimize or maximize co-usage and similarity scores in various combinations to generate recommendation scores, which can be used for different recommendations such as "Songs that Fit Your Script" or "Others Like You Used These". Combining multiple recommendation strategies allowed for increased user exploration and sound usage and that users preferred different types of recommendations when freely creating a unique project than when matching sounds to others out of the context of EarSketch [6].

While the initial recommendation system, a hybrid model using collaborative filtering and content-based similarity metrics, improved the number of sounds explored by users, users have reported a lack of musical cohesion between sounds after they have already included contrasting elements in a project, as well as a lack of sound suggestions that facilitated specific compositional ideas such as creating a new section of a song. This work aims to improve the recommendation system's impact on sound exploration and usage by adding two additional musical features as inputs: key signature and beat similarity. These features are musically motivated in that, unlike the existing system's use of Short-time Frequency Transform, they use explicit human-understandable labels grounded in music theory. Although EarSketch does not include western music notation by design, each tonal sound was originally composed with a major or minor key signature in mind. As such, by adding explicit key labels [7] to sounds, the overall key of a user's current project can be estimated and sounds with that key can have their recommendation scores increased. In addition to tonal similarity, the system can prioritize recommendations that are rhythmically consistent with a user's project [8]. Beat detection is performed by generating a numerical vector representing the rhythm of each sound in the sound library, then computing the distance between two sounds' vectors and factoring it into their pairwise recommendation scores.

By adding the above features, we aim to answer the following question:

- How does the addition of salient musical features in the EarSketch sound recommendation system impact the diversity of sounds recommended and used in student projects?

The contributions of this work include the augmentation of a hybrid recommendation system, combining collaborative filtering with multiple aspects of feature-based audio similarity, and the evaluation of sound recommendations in a creative, open-ended task. The rest of this paper will detail the process of adding the musically motivated features of key signature and beat similarity to the EarSketch recommendation system (the dataset, the key signature and beat similarity extraction, and how they were incorporated into the recommender), followed by the methodology and analysis of an evaluation of these recommendations on aggregate statistics of users on the EarSketch website.

2. Implementation

The EarSketch web client continuously monitors the sounds included in a user's project as they edit their code. Once a change is detected, the recommendation system generates a set of recommendations using the newly stored list of sounds as input [6]. The output is presented to users as a list of recommendations in the sound browser (Figure 1). This section will discuss the implementations of key signature estimation and beat similarity calculation, as well as their addition to the existing EarSketch recommendation algorithm at the time of generation.

2.1. Key signature and beat extraction

In order to extract key signatures for the clips in the sound library, we used Essentia [9], a popular software package for music information retrieval. It implements several key-profiles to estimate the key signature for a given sound such as "edmm" [10], a profile that is generally suited to estimating key signatures from electronic music and "braw" [11], a more general key signature estimation profile. In addition to the key signature, Essentia's key signature estimator also produced a strength score indicating how strong the presence of an annotated key signature is in the sample.

Identifying the best key signature profile in *Essentia* was done using the annotated subset of the library described in the section above. For each profile, we compared the predicted key signatures for the subset against the ground truth annotations. The "edmm" profile stood out as the best profile since it predicted the largest number of correct annotations. Therefore, it was used to compute the key signatures for the dataset where key signatures were appropriate.[1]

Beats were extracted using `librosa`'s [12] beat track prediction method. The method takes an audio signal and predicts its tempo and beat track. Details of the method can be found in Daniel Ellis' paper [13], which is the implementation used by `librosa`. The beat track prediction provided by librosa is a series of timestamps indicating where a beat might be. We take these time stamps and construct an audio signal that is a click at those time stamps. For the sake of computational and space efficiency when computing scores, we downsampled the signal from 44100 Hz to 100 Hz.

In the paper detailing the implementation of the beat tracker, it is shown that the dynamic programming approach is capable of achieving 93.4% accuracy on the MIREX beat tracking dataset [13]. Since we did not have ground truth annotations, we manually verified the beat predictions internally by using generated click tracks on a random subset of the sound library with an informal subjective evaluation. Using 5 sets of 16 sounds at a time, testers from the EarSketch development team rated the implementation as appropriately matching their perception for each example.

2.2. Recommendations

The previous algorithm to recommend sounds to users was described in [3]. In short, sounds in the library were assigned a score S

$$S = \mathcal{D}_{STFT}^{-1} + \mathcal{D}_{MFCC}^{-1} + \mathcal{U} \tag{1}$$

where $\mathcal{D}_{STFT}$ and $\mathcal{D}_{MFCC}$ are acoustic feature distances between a given sound and every sound in the library and $\mathcal{U}$ is the co-usage score, i.e, a score indicating how often two sounds were used together.

In order to add key signatures into our algorithm, we compute the key signature of the project $\mathcal{K}_{\text{proj}}$ as the most frequent key signature label across all sound clips in a project. For a given sound clip $\mathbb{S}$ and its corresponding key signature $\mathcal{K}_{\mathbb{S}}$ we compute it's key signature score $\mathcal{K}$ as

$$\mathcal{K} = \begin{cases} 1, & \text{if } \mathcal{K}_{\mathbb{S}} \in \{(\mathcal{K}_{\text{proj}} \mid \text{relative}(\mathcal{K}_{\text{proj}})\} \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

where $\mathcal{K}$ is set to 1 *if* the clip's key signature matches the project or has a relative major/minor key relationship with the project's key signature.

To add a beat similarity score, we compute the Hamming distance [14] between two given beat tracks. This is denoted as $\mathcal{D}_{hamm}^{-1}$. We assume that users might select

[1] We excluded purely percussive sounds and short, single shot examples - for e.g, snare samples

a set of samples with varying attributes, for example, genre or instrumentation, but happen to have a consistent rhythmic structure. Hamming distances have been shown by Toussiant et al[**?**] as a good measure of rhythmic similarity. Given that EarSketch time-stretches samples to match a specified tempo, we wanted to choose a similarity measure that was tempo invariant and primarily focused on the difference in how the rhythms are performed directly.

Adding key and beat information to the system was done as an addition to the score S described in Equation 1:

$$S = \mathcal{D}^{-1}_{STFT} + \mathcal{D}^{-1}_{MFCC} + \mathcal{U} + \mathcal{K} + \mathcal{D}^{-1}_{\text{hamm}} \tag{3}$$

Like the co-usage and acoustic similarity scores in the initial version of the recommendation system [6], the key signature estimation and beat extraction processes are performed offline for the whole sound library. Their results are deployed to the EarSketch web client to be retrieved for individual sound-sound pairs and used in real-time recommendations. This is done to allow for faster recommendations without the requirement for heavy audio processing while users are editing a project.

3. Results

As with a previous evaluation of the recommendation system [6], the impact of this recommender was measured through statistical analysis of the sounds recommended and added to projects by EarSketch users. The key signature and beat similarity recommendation changes were added to the EarSketch website in October 2022. Using an analytics engine, the actions of 103,828 users before the update were recorded between July and September 2022, and the actions of 133,349 users after the update were recorded between October and December 2022.

During each session for a given user, each unique recommendation made is stored as a separate event *recommendation*. A separate event, *recommendationUsed*, is stored when the student uses a recommended sound in their project by copying it directly from the sound browser interface or by writing the name of the sound into their code. The density of sounds being recommended is a function of *recommendationUsed* / *recommendation* for each individual sound constant. We determine coverage of the sound library by the distribution of unique recommendations made, as well as the rate at which these recommendations are used in student projects.

The analysis of this data suggests that the inclusion of musically driven features further improves the diversity of sounds suggested by our hybrid recommendation system. Fig 3 depicts the frequency density of each sound based on the number of times that sound was recommended. The higher average and lower skew of the distribution after the addition of key signature and beat similarity indicates that more sounds from the EarSketch library are more likely to be recommended. In the period prior to the update, a given sound would be recommended an average of 1790 times (1.72% of sessions). Comparatively, following the update a given sound would be recommended 3240 times (2.42% of sessions). Using a two-sample t-test, we note that the difference in recommendation frequency across the entire library was statistically significant ($p < 0.05$).

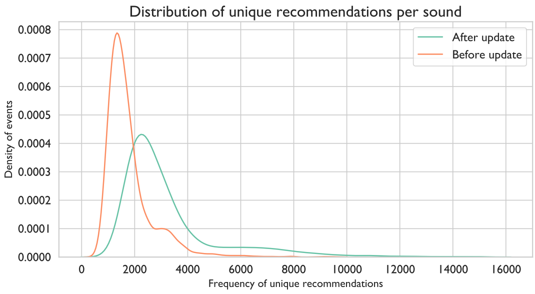


Figure 3: The distribution of how frequently a sound was recommended across the entire library before and after the addition of key signature and beat similarity to the recommendation system. The figure is scaled by the number of unique user sessions in the two time periods.

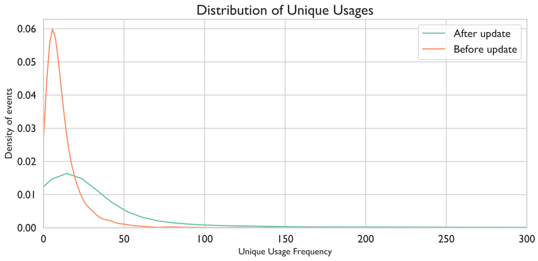


Figure 4: The distribution of how **recommended** sounds were added to projects before and after the addition of key signature and beat similarity to the recommendation system. The figure is scaled by the number of unique user sessions in the two time periods.

When comparing the usage of recommendations across both periods, we measured the frequency distribution of *recommendationUsed* events, or the unique instances of previously recommended sounds being used in projects (see Fig 4). A two sample t-test shows that there is a statistically significant increase in recommendation usage frequency following the update ($p < 0.05$). On average, a recommended sound was used *0.94%* of times it was recommended following the update compared to the average of *0.73%* prior to the update.

We also observed that the inclusion of rhythmic features coincides with a noticeable uptick in the usage of percussive loops with a larger proportion of used sounds from a recommendation being percussive. We investigated the top 10 sounds being recommended and used during both periods. We found that 6 of the 10 most frequently used recommendations after the update are categorized as purely percussive sounds. In the period prior to the update, there was no majority among the instruments in the most used recommendations.

4. Discussion and Future Work

We implemented key signature and beat extraction in the EarSketch sound recommendation system, to improve the diversity and coverage of sounds that are being recommended to users and to make more musically relevant suggestions for a student's project. We analyzed two periods of data above to identify trends in usage before and after the addition of these two musical features.

In our results, we were able to successfully demonstrate that the inclusion of these features improves the diversity and coverage of recommended sounds. By comparing the distributions of unique recommendations per sound before and after the change, we found that the number of recommended sounds was more evenly distributed across the sound library after the change. This may be because the algorithm is able to pick up on more sounds that are pertinent to a given user's project more frequently. Additionally, there was a statistically significant increase in how often students elected to use a recommendation. This could be attributed to the prominence of beat similarity in the recommendation algorithm providing sounds that stylistically match a user's current sounds and as such present more viable options to try in a given project.

We noticed that there has been a shift in the types of recommended sounds that are more frequently used across both periods. Following the introduction of our updated algorithm, we found that a majority of the most *used* recommendations were percussive or primarily rhythmic. We believe that this is an artifact of how the key signatures and rhythmic similarities of sounds are weighted in the recommendation process. We speculate that students are largely seeking rhythmic sounds at the beginning stages of the song-creating process. Given that the weighting for pitched sounds necessitates the existence of a key signature, the recommendation algorithm skews heavily towards rhythmic sounds at the start of a new project. Additionally, users with developed projects may prefer recommendations that do not clash with their current selections, such as the percussion samples without a key signature. In order to understand this behavior better, we need a more in-depth user study to understand how recommendation behavior influences the song creation process for students. By rating the understanding of and satisfaction with recommendations by users with and without the musical features in a controlled setting, we can determine how effective these features are and what visual design changes are necessary to enhance the effectiveness of musically-informed recommendations.

In conclusion, this analysis of the impact that salient musical features have on EarSketch users reveals multiple insights for the design of recommendation systems and other creative systems. The use of recommendation density to compare groups shows how artifact analysis can represent trends in user interaction with a creative musical assistant, even at its most simple form. The significant change in the density of unique sound recommendations shows the effectiveness of multimodal domain knowledge on recommendation generation. As the EarSketch recommendation system either minimizes or maximizes co-usage scores as well as acoustic similarity [3], the addition of features to multiple types of recommendations shows the importance of understanding task specifications when discussing recommendations for a creative system.

5. Acknowledgments

This material is based upon work supported by the National Science Foundation Award No. 1814083. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation. EarSketch is available online at https://earsketch.gatech.edu

References

[1] B. Magerko, J. Freeman, T. Mcklin, M. Reilly, E. Livingston, S. Mccoid, A. Crews-Brown, Earsketch: A steam-based approach for underrepresented populations in high school computer science education 16 (2016) 1–25. doi:10.1145/2886418.

[2] T. McKlin, B. Magerko, T. Lee, D. Wanzer, D. Edwards, J. Freeman, Authenticity and personal creativity: How EarSketch affects student persistence, in: Proceedings of the 49th ACM Technical Symposium on Computer Science Education, 2018, pp. 987–992. doi:10.1145/3159450.3159523.

[3] J. Smith, D. Weeks, M. Jacob, J. Freeman, B. Magerko, Towards a Hybrid Recommendation System for a Sound Library, in: Joint Proceedings of the ACM IUI 2019 Workshops, CEUR-WS, 2019.

[4] C. C. Aggarwal, Recommender Systems, Springer International Publishing, 2016. URL: http://link.

springer.com/10.1007/978-3-319-29659-3. doi:10.1007/978-3-319-29659-3.

[5] J. B. Schafer, D. Frankowski, J. Herlocker, S. Sen, Collaborative Filtering Recommender Systems, in: P. Brusilovsky, A. Kobsa, W. Nejdl (Eds.), The Adaptive Web: Methods and Strategies of Web Personalization, Lecture Notes in Computer Science, Springer, 2007, pp. 291–324. URL: https://doi.org/10.1007/978-3-540-72079-9_9. doi:10.1007/978-3-540-72079-9_9.

[6] J. Smith, M. Jacob, J. Freeman, B. Magerko, T. Mcklin, Combining collaborative and content filtering in a recommendation system for a web-based daw, in: A. Xambó, S. R. Martín, G. Roma (Eds.), Proceedings of the International Web Audio Conference, WAC '19, NTNU, Trondheim, Norway, 2019, pp. 53–58.

[7] M.-K. Shan, F.-F. Kuo, M.-F. Chiang, S.-Y. Lee, Emotion-based music recommendation by affinity discovery from film music 36 (2009) 7666–7674. doi:10.1016/j.eswa.2008.09.042.

[8] X. Wang, Y. Wang, Improving content-based and hybrid music recommendation using deep learning, in: Proceedings of the 22nd ACM International Conference on Multimedia, 2014, pp. 627–636. doi:10.1145/2647868.2654940.

[9] D. Bogdanov, N. Wack, E. Gómez, S. Gulati, P. Herrera, O. Mayor, G. Roma, J. Salamon, J. Zapata, X. Serra, ESSENTIA: An open-source library for sound and music analysis, in: Proceedings of the 21st ACM International Conference on Multimedia, MM '13, Association for Computing Machinery, 2013, pp. 855–858. URL: https://doi.org/10.1145/2502081.2502229. doi:10.1145/2502081.2502229.

[10] Á. Faraldo, E. Gómez, S. Jordà, P. Herrera, Key Estimation in Electronic Dance Music, in: N. Ferro, F. Crestani, M.-F. Moens, J. Mothe, F. Silvestri, G. M. Di Nunzio, C. Hauff, G. Silvello (Eds.), Advances in Information Retrieval, volume 9626 of *Lecture Notes in Computer Science*, Springer International Publishing, 2016, pp. 335–347. URL: http://link.springer.com/10.1007/978-3-319-30671-1_25. doi:10.1007/978-3-319-30671-1_25.

[11] Á. Faraldo, S. Jordà, P. Herrera, A Multi-Profile Method for Key Estimation in EDM, in: Proceedings of Conference on Semantic Audio, 2017, p. 7. URL: https://doi.org/10.5281/zenodo.3855499. doi:10.5281/zenodo.3855499.

[12] B. McFee, A. Metsai, M. McVicar, S. Balke, C. Thomé, C. Raffel, F. Zalkow, A. Malek, Dana, K. Lee, O. Nieto, D. Ellis, J. Mason, E. Battenberg, S. Seyfarth, R. Yamamoto, viktorandreevichmorozov, K. Choi, J. Moore, R. Bittner, S. Hidaka, Z. Wei, nullmightybofo, A. Weiss, D. Hereñú, F.-R. Stöter, L. Nickel, P. Friesch, M. Vollrath, T. Kim, Librosa/Librosa: 0.9.2, 2022. URL: https://doi.org/10.5281/zenodo.6759664. doi:10.5281/zenodo.6759664.

[13] D. P. W. Ellis, Beat tracking by dynamic programming, Journal of New Music Research 36 (2007) 51–60. URL: https://doi.org/10.1080/09298210701653344. doi:10.1080/09298210701653344.

[14] G. T. Toussaint, A comparison of rhythmic similarity measures., in: Proceedings of the 5th International Conference on Music Information Retrieval, ISMIR, Barcelona, Spain, 2004. URL: https://doi.org/10.5281/zenodo.1416812. doi:10.5281/zenodo.1416812.