Digital Music Lab:
A Framework for Analysing Big Music Data

Samer Abdallah*, Emmanouil Benetos†, Nicolas Gold*, Steven Hargreaves†, Tillman Weyde‡, and Daniel Wolff ‡

*Department of Computer Science, University College London, UK
†Centre for Digital Music, Queen Mary University of London, UK
‡Department of Computer Science, City University London, UK
Email: {s.abdallah,n.gold}@ucl.ac.uk, {emmanouil.benetos,s.hargreaves}@qmul.ac.uk, {t.e.weyde,daniel.wolff.2}@city.ac.uk

Abstract—In the transition from traditional to digital musicology, large scale music data are increasingly becoming available which require research methods that work on the collection level and at scale. In the Digital Music Lab (DML) project, a software system has been developed that provides large-scale analysis of music audio with an interactive interface. The DML system includes distributed processing of audio and other music data, remote analysis of copyright-restricted data, logical inference on the extracted information and metadata, and visual web-based interfaces for exploring and querying music collections. A system prototype has been set up in collaboration with the British Library and I Like Music Ltd, which has been used to analyse a diverse corpus of over 250,000 music recordings. In this paper we describe the system requirements, architecture, components, and data sources, explaining their interaction. Use cases and applications with initial evaluations of the proposed system are also reported.

I. INTRODUCTION

Musicology has traditionally relied on data of many kinds, such as scores and recordings as well as representations of other aspects of music, e.g. lyrics and metadata. Within musicology, the field of *Digital Musicology* addresses both the computational tools for analysing digital audio, scores and metadata but also the methods for musicological research in this context. This development has attracted increasing attention in recent years, e.g. from the European Science Foundation [1], the IMS study group on Digital Musicology[1] as well as publications, e.g. [2], [3], [4].

Digital datasets in music are smaller than in some other domains, and according to [5], of the openly accessible music datasets only the Million Song Dataset[2] qualifies as "truly 'big'" with 280GB of feature data extracted from 1 million audio tracks. However, the quantity of music data is growing and even the smaller data sets available now are big in the sense that the traditional musicological method, where the

researcher closely inspects every work, is no longer applicable. Since scholars in the Humanities are typically not trained in the development or even the use of technology, there is a gap to bridge in order to make systems accessible to music researchers and to enable scholars to develop questions and seek answers that can be approached with the growing digital datasets and computational tools.

In this paper we present the Digital Music Lab (DML) system, which addresses this gap by providing an environment for musicologists to explore and analyse large-scale music data collections, offering a range of tools and visualisations. We bring computation to the data, in order to enable the remote analysis of copyright-restricted material, and enable scalable interactive processing with large-scale parallelisation. The DML system is available as open source software[3], so that additional installations can be set up and connected via Semantic Web interfaces to create a distributed musicological research environment across institutions. Our first installation has access to a collection of over 1.2 million audio recordings from multiple sources (cf. Section II for details) across a wide range of musical cultures and styles, of which over 250,000 have been analysed so far, producing over 3TB of audio features and aggregated data. This paper presents requirements, design and technical architecture, as well as an implementation and initial evaluation of the DML system, showing how it addresses the needs of musicologists.

On related work, the field of Music Information Retrieval (MIR) has mostly focused on commercial use cases, and there has been little interaction between musicology and MIR [6]. The use of MIR at different scales for musicology has been addressed by [7] in the context of an experimental system which is no longer available. More recently, several web-based systems have been developed for presenting collections of ethnomusicological recordings. The Telemeta system [8] provides a function-rich framework for presenting music audio archives on the web. A similar approach is followed by the Dunya system, which supports browsing one of several collections in specific music cultures, each with a specific interface [9]. These systems are focused on searching and inspecting audio

[1] http://ims-international.ch/content/index.php/study-groups/digital-musicology
[2] http://labrosa.ee.columbia.edu/millionsong/

[3] See: http://dml.city.ac.uk/code

recordings individually rather then analysing collections of recordings. For analysing data, the AcousticBrainz project [10] uses a crowdsourcing approach, collecting feature data from audio that private contributors have on their computers; the extracted features are accessible via an online API.

The remainder of this paper is organised as follows: Section II introduces the main concepts and components of the DML system. Section III provides information on the track-level and collection-level feature processing. Section IV describes the middle-tier information management system. Section V introduces the front-end interface and Section VI describes the user evaluation and applications. Section VII concludes this paper.

II. THE DML SYSTEM

A. Concepts

An initial workshop was held within the DML project on 19th March 2014, in which 48 musicologists and music researchers with varying degrees of computational expertise participated, discussing research questions and requirements for Big Data systems in musicology. Based on the user input we developed our approach for the analysis of large music collections. In the following we describe the main concepts used throughout the paper.

The content accessible through the DML system is organised into *libraries*, *works* and *recordings*. The *library* information identifies the provider of the data, e.g. as the British Library (BL). A *work* reflects a composition, which may have a digital score or other information associated with it. Works can be associated with one or more digital audio *recordings*, which are currently the main objects of analysis in the system. The DML manages internal files as needed and provides, if available, URLs for public download or streaming of audio. Recordings are grouped by the user based on metadata into *collections*, which form the basis for analysis and inspection.

In the DML system we distinguish *recording-level analysis* and *collection-level analysis*. An analysis on either level is defined as a triple of *perspective*, *parameters* and *target*. The target to be analysed can be either a recording or a collection. The perspective specifies a transformation of this data which may be based on multiple sub-transformations. The triple of perspective, parameters and target identifies each analysis for storage and retrieval of previously computed results.

B. Architectural Overview

As displayed in Figure 1, the DML system consists of three main components: the *Analytical Compute Servers (CS)* and the *Information and Computation Management Systems (ICMS)* in the back-end and the *Visualisation Interface (VIS)* in the front-end. The computation of audio feature data is done as the CS servers, which are placed at the content providers for in-place processing. The CS instances extract features from audio and pre-compute aggregate statistics as far as possible. This model reduces network load and addresses copyright restrictions so that an analysis of copyrighted audio material can be conducted. An ICMS organises the available

media and related information and addresses data diversity with the use of Semantic Web technology. Extracted features and aggregate data become part of this information graph. The ICMS schedules the computation and makes efficient use of existing information. The CS performs the recording-level and collection-level analysis in parallelised processes. The result is returned to the ICMS which saves it to a Resource Description Framework (RDF) triple-store and forwards it to the requesting client interface. The VIS visualisation provides an end-user interface to define musicological queries. It focuses on collection-level analysis and provides the user with individual and comparative perspectives of distinct collections.

C. Datasets and Music Collections

Four collections of audio recordings have been integrated to our DML installation, spanning many music cultures and genres, with scope to include more collections as the project grows; over 250,000 recordings are currently available.

We have currently imported over 49,000 recordings from the British Library, which originate from the Classical ($\sim$19k recordings) and World and Traditional Music ($\sim$29k recordings) Collections. Secondly, the CHARM database [11] contains digitised versions of 4,882 copyright-free historical recordings of classical music transferred from 78rpm discs, dated between 1902-1962, as well as metadata describing both the provenance of the recordings and the digitisation process. Thirdly, the Mazurka database[4] contains 2,732 recorded performances for 49 Mazurkas by Frédéric Chopin, ranging from 1902 until recent years. Finally, I Like Music Ltd[5] (ILM) has a repertoire of over 1 million commercial music recordings, of which we have so far analysed a selection of 6 music genres with 216,523 audio recordings: jazz, rock & roll, reggae, classical, blues, and folk. Recording dates span from 1927, with the vast majority from the last two decades.

III. BACK-END PROCESSING

A prerequisite for performing collection-level analysis is the extraction of low and mid-level audio features for the audio recordings under consideration. This process is carried out using the batch tool *Sonic Annotator*[6]; in order to speed up the process for large collection, we have parallelised this back-end processing. We used one server physically located at the premises of I Like Music (ILM, 24 cores @2.4GHz) and a second at the BL (20 cores @3 GHz). This in-place access to data is at the core of the DML system design, enabling analysis on datasets that cannot be copied off-site due to copyright.

A. Feature Extraction

We selected a number of audio features that are frequently used in MIR research. These features were extracted in the DML by means of Vamp[7] plugins within *Sonic Annotator*:

---

[4]http://www.mazurka.org.uk/
[5]http://www.ilikemusic.com
[6]http://vamp-plugins.org/sonic-annotator/
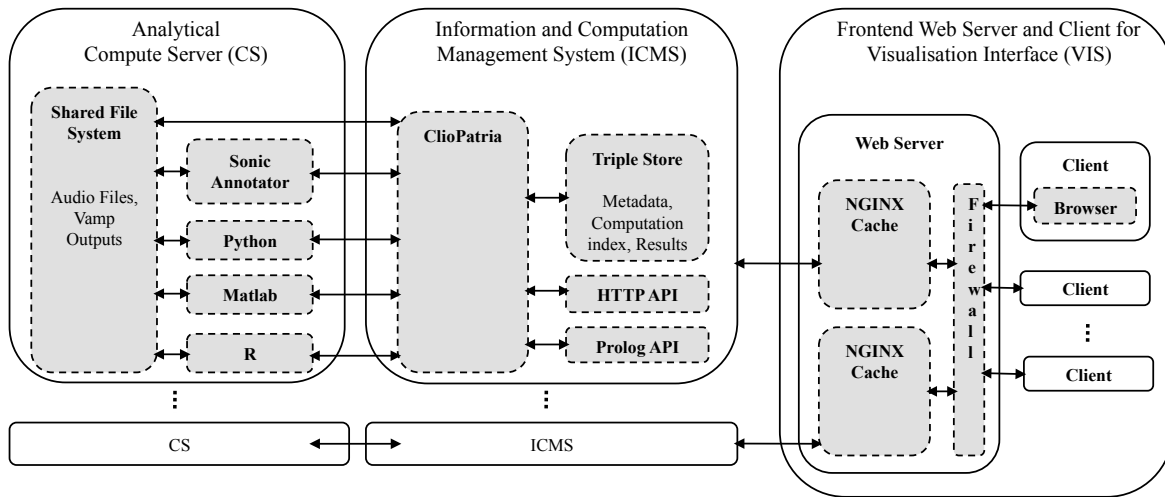[7]http://vamp-plugins.org

Fig. 1. Overview of the DML system. The web front-end (VIS) is connected to one or more information and computation management systems (ICMS) which request and store metadata as well as analysis results that have been computed by the back-end compute servers.

1) *Spectrograms* provide time-frequency content of the recordings, using the short-time Fourier transform or the constant-Q transform [12].
2) *Mel-frequency Cepstral Coefficients* (MFCCs) offer a compact representation of the frequency content of an audio signal; 20 MFCCs per time frame were extracted using the QM Vamp Plugin Set.
3) *Chroma* projects the entire spectrum onto 12 semitone bins. Two implementations were used: QM Chromagram and NNLS Chroma Vamp [13].
4) *Onsets* represent the beginning of a musical note in an audio signal using the QM Onset plugin [14].
5) *Speech/music segmentation* on ethnographic and radio recordings was done using the BBC Speech/Music Segmentation[8].
6) *Chords* provide a concise description of musical harmony; we used the Chordino Vamp Plugin [13].
7) *Beats* were extracted using: Beatroot [15], Marsyas [16], and Tempotracker [17].
8) *Tempo* following is strongly related to beats. We use the Tempotracker [17] and Tempogram [18].
9) *Keys* are detected (in a Western tonal music context) with the QM Key plugin [19].
10) *Melody* is estimated by the MELODIA Vamp plugin [20]
11) *Note transcription* from audio to music notation uses the Silvet Vamp plugin [21] with two different settings: 12-tone equal temperament (for Western tonal music), and 20 cent resolution (for World & Traditional music).

B. Collection-Level Analysis

Based on the low and mid-level features listed above, the DML system computes *collection-level* features across groups of recordings for large-scale musicological analysis as shown below:

1) *Key-relative chord sequences* combine information from chord and key extraction, and are based on sequential pattern mining (CM-SPADE) [22].
2) *Mean tempo curve* summarises tempo changes over the duration of a recording. The curve displays average normalised tempo vs. the normalised track length.
3) *Pitch class histogram* summarises detected pitches from the (semitone-scale) *Note Transcription* in a histogram with octave-equivalent pitch classes 0-11 (C-B). The individual histograms are averaged across the collection.
4) *Pitch histogram* aggregates all detected pitches over a collection of recordings, (without the octave-wrapping of the *pitch class histogram*). Information from *Note Transcription* is used in two versions: *semitone resolution* histogram and *20 cent resolution*.
5) *Similarity matrix* contains the pairwise feature similarity of the recordings in a collection, using a distance metric (Euclidean as in [23] or normalised compression distance [24]). The user can select any combinations of the following features: chords, chromagram, MFCCs.
6) *Similarity plane* arranges recordings on a two-dimensional pane according to their similarity. The spatial arrangement is determined using Multidimensional Scaling [25] on the basis of the Similarity Matrix.
7) *Tempo histogram* summarise all tempi using the QM tempo tracker plugin across the entire collection.
8) *Tonic histogram* shows the tonic (i.e. key for Western tonal music) over all recordings as estimated by the QM key detector. In tonal music the last tonic detected is considered a good estimate for the entire piece.
9) *Tuning stats* summarise the reference pitch distribution based on the 20 cent resolution *Note Transcription* feature in a histogram plus average and standard deviation. The tuning frequency is estimated per recording based on the precise F0 for all detected A, E and D notes.

IV. THE INFORMATION AND COMPUTATION MANAGEMENT SYSTEM

The information and computation management (ICMS) sub-system of the DML has the job of organising and keeping track of the recordings, their metadata, and the details of any computations done on them. In addition, it is responsible for triggering new computations when required, and so must keep information about the functions available for application in new computations. These requirements are realised using a relational data model based on Semantic Web technologies [26] combined with a system for managing computations based on memoisation.

On data representation, the ICMS has at its core the RDF: a data model built from sets of *triples*, which are simple logical statements of the form $(\langle Subject \rangle, \langle Predicate \rangle, \langle Object \rangle)$, such as $(\langle J.S.Bach \rangle, \langle composed \rangle, \langle TheGoldbergVariations \rangle)$. The ICMS is implemented in SWI Prolog [27], providing a substantial set of libraries for managing an RDF database. The DML ICMS is written as a *ClioPatria* add-on package that provides many facilities for managing, exploring, and presenting music-related data.

The first step in making a music library available to the DML system is the translation of its metadata and audio files to a set of triples in the RDF database, achieved by writing a set of importers to handle the various metadata supplied by the collections (cf. section II). Computation management in the ICMS is based on the idea that when computation results are requested, the database is checked to see if the result is already known (*memoisation*). Computations done using the VAMP system of audio analysis plugins [28] and are managed using the RDF database, as VAMP plugins are already described by RDF documents. The bulk of the VAMP analysis results currently in DML were pre-computed off-line using general purpose parallelisation frameworks (cf. section III). Collection-level analysis functions, relying on the results of the primary VAMP-based analysis, were written in several languages: Prolog, Matlab, R, or Python; the results are memoised in a persistent Prolog database.

V. FRONT-END INTERFACES

Two interfaces are provided in the DML system: a database-oriented data management web interface and a user-oriented visual interface (VIS). The data management web interface enables to browse and manage the RDF database. The core RDF concepts of triples, resources, predicates and classes are exposed, so that users can see all the triples for a given subject, or for a given predicate, and traverse the RDF graph by following links associated with resources, predicates, or literal values. For example, a recording of 'Blackthorn Stick' is described in a page[9], which lists the predicate-object pairs for that subject as shown in figure. It also provides several hooks by which the display of information can be customised. For example, the page representing the results of an automatic
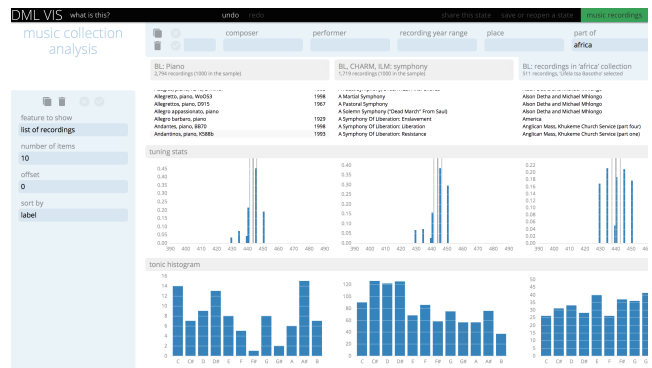
[9]http://bit.ly/1OqoQ1z

Fig. 2. Screenshot of the VIS interface: Analysis on user-defined collections (specified through the menu on the top). Three perspectives (list view, tuning statistics and tonic histogram) are shown as rows.

transcription is augmented with a sonification interface and a piano-roll view of the transcription[10].

Our VIS interface supports musicologists in exploratory data analysis on the collection level. The VIS interface is publicly available[11] and a screenshot is shown in Figure 2. The VIS design is based on a grid of collections in the columns and analyses in the rows. In each cell of the grid, a visual representation, such as a bar chart, graph or histogram, is shown, including tool tips for individual values at mouse point. The user defines collections by selecting libraries and providing search terms in metadata fields at the top of the screen. Analyses are selected from a list and parametrised on the left. Further discussion of the specifics of the visualisation technology are out of the scope of this paper.

VI. USER EVALUATION AND CASE STUDIES

A. User Evaluation

A user-based evaluation took place during a second DML workshop on 13 March 2015; in attendance were 40 participants with interests in digital music and musicology. Participants were asked to carry out two tasks from the following list in pairs using the VIS interface:

- *Tuning Frequency*: identify trends in orchestral pitch
- *Pitch Profile*: identify, compare, and explore pitch class sets and pitch hierarchies
- *Tempo Curves*: identify historical trends in classical music through tempo summaries

We asked participants to rate their level of agreement with the statement *these kinds of tools could help with the task in hand*. In total, we received 28 partial and 16 complete responses from participants. Out of the 16 complete responses, 9 participants strongly agreed that the developed tools can help with the task in hand, while 6 participants agreed. Most suggestions for improvements were about usability of the UI (e.g. providing information about the server state), many of which have been addressed in the current version. Apart from

[10]http://bit.ly/1RXfBfX
[11]http://dml.city.ac.uk/vis

those, the most frequent requests were the addition of more music genres (e.g. electronica), integration with other services (e.g. Echo Nest), and of additional metadata (e.g. duration information) to support more powerful search and selection.

B. Case studies

In addition to the main use case of aiding digital musicology research, two specific case studies demonstrating use of the DML system are outlined. First, tuning trends over time and between instrumentations can be detected with the DML. We performed exploratory experiments on identifying trends in tuning frequencies over time, as shown in Figure 2, which shows a comparison of symphonic recordings vs. piano recordings, with the latter showing lower spread. Regarding pitch levels, we found that already in the 1950s an elevated pitch level of 444Hz was frequently used, despite the then recent standardisation to 440Hz. These initial findings justify further musicological studies.[12] Secondly, the DML system was used in a study commissioned by *Sky Arts* regarding characteristics of successful musical theatre music, as compared to less successful musicals (commercially or by critics' judgement). Harmonic, dynamic and tempo-related patterns were found and used by a team that then created a computer-generated musical [29] that was performed in London's West End[13].

VII. CONCLUSIONS

We have proposed the DML system as an approach to bridge the gap between musicology, music information retrieval and big data technology, enabling music researchers to explore and analyse substantial music audio collections and datasets. The system allows for effective analysis approaches on large music collections, based in several locations. Our evaluations showed that the combination of different facets of music, audio analysis, musical structure, and metadata is of value to musicology, as it enables researchers to understand music in its context and conduct comparative analyses between music collections. In the future, we will address issues relating to the system design that have not fully been resolved during the DML project or have emerged as new problems. This includes parallelising collection-level analyses which follow the map-reduce paradigm, enabling users to import audio and scores into the framework, and creating a distributed system over several ICMS to improve scalability.

REFERENCES

[1] E. Dahlig-Turek, S. Klotz, R. Parncutt, and F. Wiering, Eds., *Musicology (Re-) Mapped: Discussion Paper*. European Sci. Foundation, 2012.

[2] L. Pugin, "The challenge of data in digital musicology," *Frontiers in Digital Humanities*, vol. 2, p. 4, 2015.

[3] E. Duval, M. van Berchum, A. Lentzsch, G. Parra, and A. Drakos, "Musicology of early music with europeana tools and services," in *ISMIR*, 2015, pp. 632–638.

[4] K. Ng, A. McLean, and A. Marsden, "Big data optical music recognition with multi images and multi recognisers," in *Proc. of the EVA London 2014 on Electron. Visualisation and the Arts*, ser. EVA London 2014. UK: BCS, 2014, pp. 215–218.

[5] J. Burgoyne, I. Fujinaga, and J. Downie, "Music inform. retrieval," in *A New Companion to Digital Humanities*, S. Schreibman and R. Siemens, Eds. Wiley, 2016, pp. 213–228.

[6] K. Neubarth, M. Bergeron, and D. Conklin, "Associations between musicology and music information retrieval." in *ISMIR*, 2011, pp. 429–434.

[7] C. Rhodes, T. Crawford, M. Casey, and M. d'Inverno, "Investigating music collections at different scales with audiodb," *J. of New Music Res.*, vol. 39, no. 4, pp. 337–348, 2010.

[8] T. Fillon, J. Simonnot, M. Mifune, S. Khoury, G. Pellerin, and M. Le Coz, "Telemeta: An open-source web framework for ethnomusicological audio archives management and automatic analysis," in *DLfM*, 2014.

[9] A. Porter, M. Sordo, and X. Serra, "Dunya: A system for browsing audio music collections exploiting cultural context," in *ISMIR*, 2013.

[10] A. Porter, D. Bogdanov, R. Kaye, R. Tsukanov, and X. Serra, "Acousticbrainz: a community platform for gathering music information obtained from audio," in *ISMIR*, 2015.

[11] R. Beardsley and D. Leech-Wilkinson, "A brief history of recording," http://www.charm.kcl.ac.uk, 2009, centre for the History and Analysis of Recorded Music.

[12] C. Schörkhuber and A. Klapuri, "Constant-Q transform toolbox for music processing," in *7th Sound and Music Comput. Conf.*, Barcelona, Spain, Jul. 2010.

[13] M. Mauch and S. Dixon, "Approximate note transcription for the improved identification of difficult chords," in *Proc. of the 11th Internat. Soc. for Music Inform. Retrieval Conf. (ISMIR 2010)*, 2010.

[14] J. Bello, L. Daudet, S. Abdallah, C. Duxbury, M. Davies, and M. Sandler, "A tutorial on onset detection of music signals," *IEEE Trans. on Audio, Speech, and Language Process.*, vol. 13, no. 5, pp. 1035–1047, 2005.

[15] S. Dixon, "Evaluation of the audio beat tracking system beatroot," *J. of New Music Res.*, vol. 36, no. 1, pp. 39–50, 2007.

[16] G. Tzanetakis and P. Cook, "MARSYAS: a framework for audio analysis," *Organised Sound*, vol. 4, pp. 169–175, 12 2000.

[17] M. Davies and M. Plumbley, "Context-dependent beat tracking of musical audio," *IEEE Trans. on Audio, Speech, and Language Process.*, vol. 15, no. 3, pp. 1009–1020, March 2007.

[18] P. Grosche, M. Mller, and F. Kurth, "Cyclic tempogram - a mid-level tempo representation for music signals," in *ICASSP*, 2010.

[19] K. Noland and M. Sandler, "Signal processing parameters for tonality estimation," in *Audio Engrg. Soc. Convention 122*, May 2007.

[20] J. Salamon and E. Gomez, "Melody extraction from polyphonic music signals using pitch contour characteristics," *IEEE Trans. on Audio, Speech, and Language Process.*, vol. 20, no. 6, pp. 1759–1770, 2012.

[21] E. Benetos and S. Dixon, "A shift-invariant latent variable model for automatic music transcription," *Comput. Music J.*, vol. 36, no. 4, pp. 81–94, 2012.

[22] M. Barthet, M. Plumbley, A. Kachkaev, J. Dykes, D. Wolff, and T. Weyde, "Big chord data extraction and mining," in *Conf. on Interdisciplinary Musicology*, Dec 2014.

[23] E. Pampalk, "Computational models of music similarity and their application in music information retrieval," Ph.D. dissertation, Vienna University of Technology, Vienna, Austria, March 2006.

[24] M. Li, X. Chen, X. Li, B. Ma, and P. Vitanyi, "The similarity metric," *IEEE Trans. on Inform. Theory*, vol. 50, no. 12, pp. 3250–3264, Dec. 2004.

[25] I. Borg and P. Groenen, *Modern multidimensional scaling*, 2nd ed. New York, USA: Springer, 2005.

[26] T. Berners-Lee, J. Hendler, O. Lassila *et al.*, "The semantic web," *Sci. Amer.*, vol. 284, no. 5, pp. 28–37, 2001.

[27] J. Wielemaker, T. Schrijvers, M. Triska, and T. Lager, "SWI-Prolog," *Theory and Practice of Logic Programming*, vol. 12, no. 1-2, pp. 67–96, 2012.

[28] C. Cannam, C. Landone, M. Sandler, and J. Bello, "The Sonic Visualiser: A visualisation platform for semantic descriptors from musical signals." in *ISMIR*, 2006, pp. 324–327.

[29] S. Colton, M. Llano, R. Hepworth, J. Charnley, C. Gale, A. Baron, F. Pachet, P. Roy, P. Gervs, N. Collins, B. Sturm, T. Weyde, D. Wolff, and J. Lloyd, "The beyond the fence musical and computer says show documentary," in *Proceedings of The Seventh International Conference on Computational Creativity*, Paris, France, June 2016.

[12]We would like to thank Prof Stephen Cottrell (City University London) for musicological contribution and guidance.

[13]http://beyondthefencemusical.com