IMPROVING THE UNDERSTANDING OF TURKISH MAKAM MUSIC THROUGH THE MEDIACYCLE FRAMEWORK

Onur Babacan, Christian Frisson, Thierry Dutoit
TCTS Lab, numediart Institute, University of Mons, Boulevard Dolez 31, B-7000 Mons, Belgium
onur.babacan@umons.ac.be christian.frisson@umons.ac.be thierry.dutoit@umons.ac.be

ABSTRACT

The goal of this work is to investigate the challenges of creating a tool to aid people of diverse profiles, from musicology experts and music information retrieval (MIR) specialists, to the interested non-technical users outside these fields in understanding traditional makam music of Turkey. We aim at providing a playground approach, with which MIR specialists can easily validate algorithms for feature extraction, clustering and visualization, and non-technical users can navigate by easily varying parameters and triggering audiovisual previews. We adapted the MediaCycle framework for organization of media files by similarity. AudioCycle, its audio application, allows users to cluster a large number of audio files against a subset of extracted audio features, visualized in a 2D space through positions, distances, colors. Transitions between parametric changes are animated, which helps the user create and retain a mental model of the sounds and their relationships. For our proof-of-concept, we defined our use case as detecting *makamlar* (plural) from makam music. We integrated the pitch histogram technique proposed by Bozkurt et. al as a feature extraction plugin in AudioCycle to meet this goal.

1. INTRODUCTION

Fitting both profiles of researchers in the field of audio signal processing and audio information retrieval, and passionate about Turkish music, we attempted to make both ends meet: how can we better understand Turkish music using the audio analysis tools we manipulate or create?

A fundamental question is "How can tools enhance the process of understanding music?". The expression "process of understanding music" encompasses all the stages that are passed through to obtain a systematic understanding of a music genre by diverse profiles of musically-inclined people:

- (music theorists and musicologists) theorizing categories and rules that define music genres and practices
- (MIR researchers) defining, refining, testing and validating algorithms to observe through computer-aided analysis whether the theory matches the practice
- (non-technical users) navigating in music collections with several pathways of understanding of the music offered by verified computer analysis

Throughout this paper, we'll use the term "music understanding tools" as a subset of music information retrieval (MIR)-based tools that go beyond the stage of extracting information out of music by also allowing the user to navigate a representative space based on this extracted information. Some examples include audio feature extraction and classification toolkits, music recommendation engines, score editors with score digitization, score following, score reconstruction, score annotation.

This paper is divided into two sections: Section 2 discusses the requirements for building a "music understanding tool" and Section 3 provides feedback from a use case tested while trying to create such a tool: analyzing makam music of Turkey. More precisely, Section 2.1 gives an overview of the necessary and existing components of a "music understanding tool" and Section 2.2 presents a recent solution, the MediaCycle framework and the AudioCycle application. In the second part, Section 3.1 briefly summarizes makam music and its definitions, Section 3.2 recalls the related existing MIR research on the topic, Section 3.3 describes the integration of these algorithms into the MediaCycle framework. Section 4 discusses the results of this work.

2. HOW TO BUILD A MUSIC UNDERSTANDING TOOL?

How can music understanding tools efficiently assist MIR researchers, music theorists and musically-avid people?

2.1 Brief Overview of Existing Components of a Music Understanding Tool

Figure 1 groups some existing music understanding components by the expected profile(s) of users of these applications.

Stéphanie Weisser discusses in [1] (in French) the usability of such of these applications for ethnomusicologists: the major drawback of most of these tools is the lack of user-friendliness (learning to use Matlab is not straightforward for anybody but people with technical training); and there's no all-in-one integrated solution that fulfills all steps of the global analysis: from recording to manual annotation and analysis over multiple time scales (from the measure if there is to the whole composition or compositions from the same genre).

As already pointed out by CompMusic contributors, music recommenders such as EchoNest and LastFM offer the power to process and relate massive databases, but these are mainly constituted of popular western music.

2.2 A Recent Solution in Progress: The MediaCycle Framework

The MediaCycle framework, developed within the numediart Institute of the University of Mons since 2009, is less mature than many of the aforementioned solutions, and not yet released for distribution. It allows to organize media files by content-based similarity, particularly audio files using its AudioCycle application [2]. The framework attempts to be modular, supporting diverse media types (not only audio, but images, videos, text), allowing different clustering methods (k-means has been chosen as the default one) and visual representation algorithms.

Here follow highlights of the MediaCycle framework that benefit the "music understanding" research cycle:

- MediaCycle proposes a plugin API (feature extraction, clustering, visualization) and a collection of plugins, including a wrapper of the Yaafe (Yet Another Audio Feature Extractor) toolbox [3] [1] and VAMP plugins [2] support through a wrapper of the VAMP SDK initiated for Sonic Visualiser (from the same authors of VAMP). For the work described in this paper, we added a wrapper of the GNU Octave environment [3] to allow feature extraction with Matlab/Octave algorithms
- Every time plugins are set or their parameters are modified, changes in the view are animated, making sure the user maintains an understanding of the representation of sounds in the representative space
- The user can choose to display a visual representation of each music piece (a waveform witth various scales) and the metadata of piece, or open the related file directly into the standard operating system file browser or the default application associated to its file type.

We believe that such a setup allows to test algorithms and improve them in a cyclic manner.

3. USE CASE: CLASSIFICATION OF MAKAM MUSIC OF TURKEY

3.1 Makam Music of Turkey

As explained in [4], makam music of Turkey is primarily classified by *makamlar* (plural) and *usul* (rhythmic patterns). A *makam* provides a complex set of rules for composition and improvisation. These rules include both the type of scale and melodic development. Another major category for classification is form, which could be any one of fixed forms (e.g. *beste*, *peşrev*) or improvisational forms (e.g. *taksim*, *gazel*). Compositions are named by following the makam name with the form name (e.g. *hicaz taksim*, *saba peşrev*). An *usul* name, which defines the rhythmic structure (e.g. *aksak* (9/8), *semai* (3/4)), is also added. Improvisational forms are considered to be free-rhythmic.

Although there is a definite consensus about the names of *makamlar* at least in practice, the rules that define them are both the primary and the most problematic issues in theory and practice. Pitches do not correspond to fixed frequencies in makam music as in western music. There are several dimensions of this issue [4]:

- the concept *ahenk* (the tuning system)
- the performance of each pitch within a frequency band rather than a fixed frequency and freedom of musicians in performance of a specific makam by varying the pitches especially for the certain pitches of the scale.
- the small variations of pitches performed depending on the direction of melodic progression being descending or ascending.

There is more than one school of thought in theorizing the practice-based tradition of makam music. For the scope of this work, we limit our understanding of makam music to the Arel theory [5], as was the choice in [6].

3.2 Automatic Classification of Makamlar

Due to the issues discussed in section 3.1 existing western-oriented MIR techniques are not suitable for use on Turkish traditional art music. A new method was proposed by Bozkurt [7] [4] for automatic classification of *makamlar*. In this method, classification is done by using template matching with the pitch distributions generated from samples against previously-trained distributions for each *makam* (*makam* templates). YIN [8] is used as a pitch extractor combined with a novel post-filtering process [7] designed to make corrections using information specific to Turkish traditional art music. Then, in order to obtain consistent comparison, tonic of the sample is detected with a novel tonic detection algorithm, and pitch frequencies are converted to intervals with respect to the tonic in Holdrian comma (Hc), as defined in Arel theory [5]. Distributions are computed using the interval data, with the bin size chosen as 1/3 Hc, which is reported to provide a good sensitivity while avoiding erroneous peaks. The algorithm works best on the *taksim* form, which is monophonic, and especially well on instruments without strong time-domain transients due to attacks (e.g. plucking). This is caused by the performance of pitch extraction being dependent on auto-correlation. Further details of the algorithm is beyond the scope of this work and we refer the reader to [7] and [6].

3.3 Implementation, integration into AudioCycle

Matlab is widely used in the MIR community. For integration into MediaCycle, we chose wrappers of an open

[1] http://yaafe.sourceforge.net
[2] http://www.vamp-plugins.org
[3] http://www.octave.org

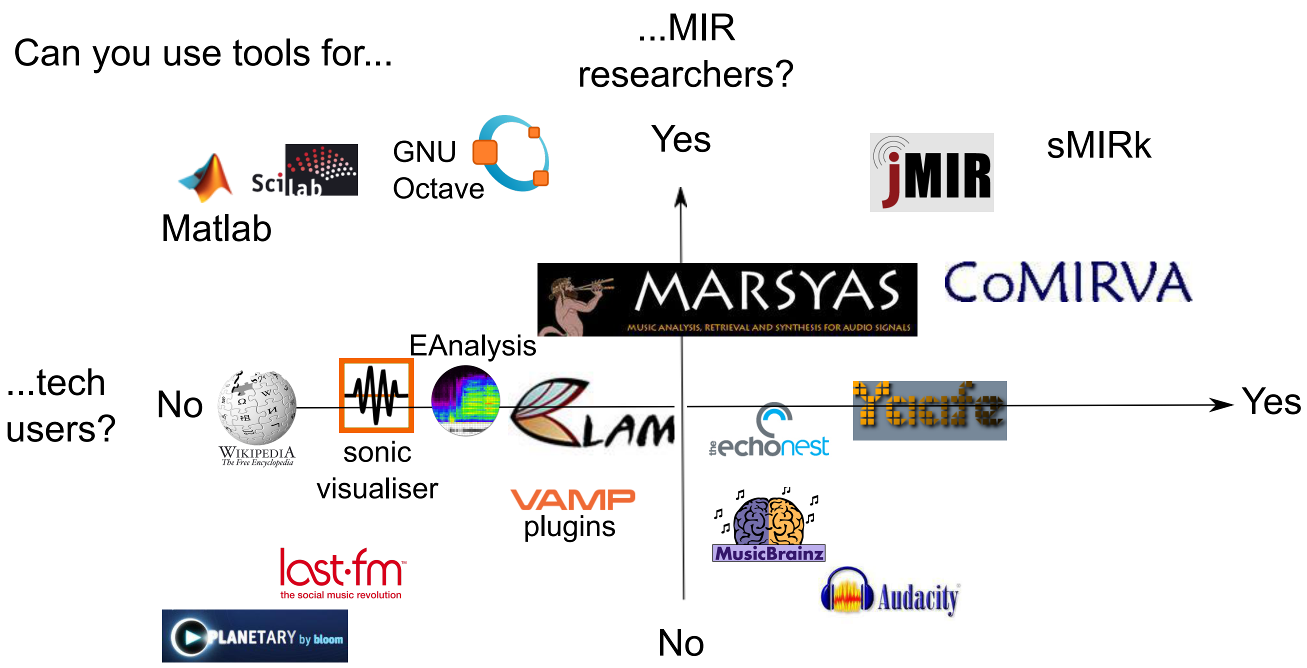

Figure 1. Exisiting components of a music understanding tool grouped by types of user profile(s)

source alternative, Octave. This way, Matlab implementations of algorithms can be run without the expensive Matlab itself, nor the heavy Matlab runtimes.

For the most part, porting code from Matlab to Octave is not difficult. Some problems that might be encountered are missing functions, and minor syntax changes (for instance, stricter requirements in Octave against the confusion between the logical comparison (e.g. `&&`) and the bitwise operator (e.g. `&`) symbols). These are easily overcome, but it may take some manual effort to verify that the ported code works correctly.

We provide an interface for integrating Octave code into AudioCycle and executing multiple feature extraction algorithms at run-time. Figure 2 shows an example of usage.

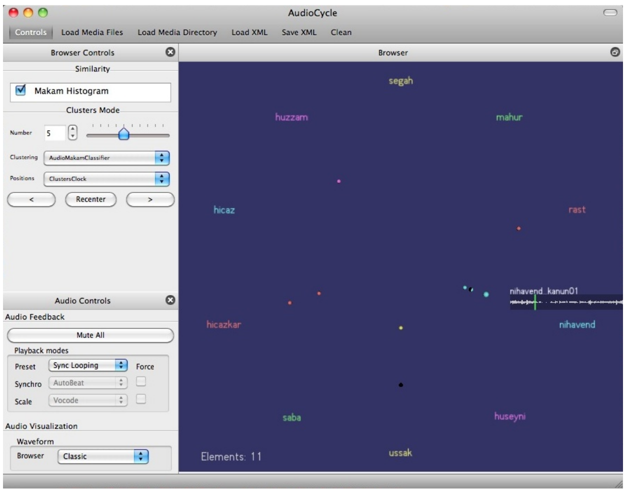

Figure 2. Screenshot of the AudioCycle application with the makam histogram as feature, radial position plugin with file duration as radius and expected makam as angle.

3.4 AudioCycle Experiments with Makamlar

One of our primary goals is to illustrate the similarity of samples to each other in an intuitive way. AudioCycle accomplishes this goal by going through three stages: extraction of audio features, clustering based on extracted features and distance-to-position mapping. For each stage, a variety of algorithm choices are provided, with the option of using multiple features simultaneously in the first stage.

Using the discrete *makam* classifications as features would not have provided a meaningful clustering, because this method discards the fine-grained information required by the mapping algorithms. We chose to use the pitch histograms directly as audio features instead. To establish consistency in mapping, all the pitch histogram vectors are pre-processed to be centered on their tonic, and normalized so their sum is 1.

We have also provided the functionality to change the weights of audio features to be used in clustering, letting the user emphasize or de-emphasize certain features as desired.

We obtained recordings of *taksimler* from various albums. For purposes of visualization, we narrowed the dataset down to the six best classified *makamlar* as reported in [7] (*hicaz*, *hüzzam*, *nihavend*, *rast*, *saba* and *segah*), which resulted in 57 samples. A limiting factor in this process was the unavailability of metadata regarding the makam of the sample.

Initially, we tried using manually-segmented shorter phrases (30-45 seconds) from *taksimler*. To verify whether this was a meaningful input for clustering, we ran these samples through automatic *makam* classification. The recall rate was zero. Since *taksimler* are improvisational, it seems plausible that the pitch histogram method, which relies on statistical accumulation, requires longer samples to work as intended. We decided that pitch histograms generated from shorter samples were not reliable as audio features. In empirical trials, we used only the pitch histogram as a feature for clustering. We observed that the *saba*, *hüzzam* and *rast makamlar* were clustered well together distinctly, albeit with other *makamlar* mixed into the clusters they were dominant in as well.

4. RESULTS AND DISCUSSION

We pushed the boundaries of an existing MIR tool by trying to use it for analyzing makam music. While the tool focuses on user-friendliness by providing a visual representation of the relations between the analyzed music pieces, we adapted it so as to easily integrate algorithms created by MIR researchers using an Octave wrapper, as part of the family of the Matlab interpreted language.

We couldn't reproduce results as satisfactorily as expected from the algorithms we borrowed. However, we believe AudioCycle integration is potentially valuable for researchers to reach insights that are harder with other tools.

We aim to add more clustering and visualization algorithms, as well as new audio features that are developed by the MIR community in order to make AudioCycle a full-fledged research tool.

Since we wrapped the Octave interpreter into AudioCycle, the GPL license of GNU Octave currently doesn't allow us to distribute AudioCycle whose release license is not yet determined. We could solve this issue by replacing the Octave wrapper by a system that calls user-defined scripts (themselves calling Octave or other environments) expected to output a file for instance in the .mat (Matlab binary) format (that we can open with the MAT File I/O Library [4] available under a BSD license) or CSV formats.

We previously proposed an installation based on MediaCycle that allows visitors to create an instant music composition by moving on the floor to activate audio loops organized by similarity of timbre and synchronized by tempo: LoopJam [9]. We believe that creating a database of short sounds within the music genres analyzed by CompMusic consortium, imported into LoopJam, could have an educational impact, especially towards children.

Acknowledgments

The authors would like to thank Barış Bozkurt for his comments and generous permission for the usage of his code.

Onur Babacan's work is supported by a PhD grant funded by UMONS and ACAPELA-GROUP SA.

Christian Frisson works for the numediart long-term research program centered on Digital Media Arts, funded by Région Wallonne, Belgium (grant N°716631), which supports his PhD studies.

5. REFERENCES

[1] S. Weisser, "L'ethnomusicologie et l'informatique musicale : une rencontre nécessaire," in *Actes des Journées d'Informatique Musicale (JIM 2012)*, 2012. 1

[2] S. Dupont, C. Frisson, X. Siebert, and D. Tardieu, "Browsing sound and music libraries by similarity," in *128th Audio Engineering Society (AES) Convention*, 2010. [Online]. Available: http://www.aes.org 2

[3] B. Mathieu, S. Essid, T. Fillon, J. Prado, and G. Richard, "Yaafe, an easy to use and efficient audio feature extraction software," in *Proceedings of the 11th ISMIR conference*, 2010. 2

[4] A. C. Gedik and B. Bozkurt, "Automatic classification of turkish traditional art music recordings by arel theory," in *Proceedings of the fourth Conference on Interdisciplinary Musicology (CIM08)*, 2008. 2

[5] H. S. Arel, *Türk Musikisi Nazariyatı Dersleri*. Kültür Bakanlı ğı Yay., 1993. 2

[6] B. Bozkurt, *Klasik Türk müziği kayı tları nı n otomatik olarak notaya dökülmesi ve otomatik makam tanı ma (Automatic Transcription of Classical Turkish Music Recordings and Automatic Classification of Makam)*. Izmir Institute of Technology, Izmir, Turkey, 2010. 2

[7] B. Bozkurt, "An automatic pitch analysis method for turkish maqam music," *Journal of New Music Research*, vol. 37, pp. 1–13, 2008. 2, 3

[8] A. de Cheveigne and H. Kawahara, "Yin, a fundamental frequency estimator for speech and music," *Acoustical Society of America*, vol. 111, no. 4, pp. 1917–1930, 2002. 2

[9] C. Frisson, S. Dupont, J. Leroy, A. Moinet, T. Ravet, X. Siebert, and T. Dutoit, "Loopjam: turning the dance floor into a collaborative instrumental map," in *Proceedings of the International Conference on New Interfaces for Musical Expression (NIME)*, 2012. 4

[4] http://sourceforge.net/projects/matio/