Musical Instrument Taxonomy Classification

Deniz Alpay
1650403
deniza@uw.edu

Aidan Johnson
1431797
johnsj96@uw.edu

02 May 2018

EE 443, Spring 2018
Capstone Design Project Proposal
Design and Application of Digital Signal Processors
Department of Electrical Engineering
University of Washington

Motivation

We intend to design a system which will automatically filter music based on the instrument being played. The system will detect and classify the instrument being played in realtime, and filter the audio based on the frequency bands of the instrument being played. Only music consisting of a single instrument will be considered. This device would serve the purpose of a sort of audio recording and playing enhancement. For example, during a musical performance one could use the device to filter the frequency bands containing noise and interference while recording the music. The device would also serve as a sort of music information retrieval system that would sift through audio and categorize it into the instrument being played. Broader applications of audio classification include implementing a compact version of the method in smartphone-sized devices for medical clinicians to classify respiratory audio features, in addition to consumer and professional audio engineering as a whole.

Problem Formulation

The task can be broken down into three major components: feature extraction, classification, and filtering. The following section addresses the methods that'll be used for these purposes and the data that'll be used to train and evaluate the classifier.

Feature Extraction

We will begin by extracting relevant features for the classification stage. To ensure that the features that are used to train the classification model can also be computed on the LCDK we will start off by using some of the features provided by Yet Another Audio Feature Extractor (Yaafe) package. Yaafe is a software packaged with bindings in MATLAB and an API for C++ that will allow us to integrate the training in MATLAB with feature acquisition on the LCDK [5]. Some features we will start off with are the mel-frequency cepstrum coefficients (MFCCs) and linear predictive coding coefficients. We will look into other spectral features such as the spectral centroid and variance, spectral shape (like the flatness, slope, rolloff), and the octave band signal intensities (OBSI). Some temporal features that we may consider are the derivative and zero crossing rates. We have identified these features as the most common the literature we've reviewed that are also available in Yaafe [1], [3], [4]. Some experimentation with the feature selection will be necessary to avoid features that increase the dimensionality without improving the classification accuracy.

Classification

The taxonomy of musical instruments can be defined by how they produce sound. In general, each taxon instrument belongs to five families. The family of instruments that generate sound by vibrating the air: (1) directly are called aerophones; (2) with an oscillating a string are called chordophones; (3) with its body are called idiophones; (4) with a membrane are called membranophones; (5) with electrical signals are called electrophones [6]. Moreover, humans are able to perceive a difference in the sound produced by different instruments at a same pitch and loudness. In music, this distinguishing characteristic is referred to as timbre. The definition of timbre, in measurable terms, nonetheless is uncertain. Timbre is multidimensional, with spectral and temporal features playing a major role [1]. To address this, features must first be extracted (see above). The MFCCs serve as the foremost measurement for timbre and the distinct sonic properties of different instruments because it contains the spectral patterns of recorded sound [6]. After feature extraction, the audio can then be classified.

Classification takes form in multiple techniques. The most popular classifiers are K-Means Clustering, Gaussian Mixture Models (GMM), and Support Vector Machines (SVM). The literature reports conflicting differential performance. For example [1] found that SVM performed best in discriminating between instruments. In 2012, [2] achieved 90% classification accuracy with SVM. While [3] reported only 5% error with GMM. We plan to experiment with different methods, and then judge which is more suitable for our particular application. After training, which will form our model to compare input audio against, we can evaluate the performance of the different classification algorithms by computing the accuracy and precision of the model's predictions. The model will be tested with a validation data set in order to calculate these metrics for classification performance.

Filtering

Digital filters will be used in first stages and the final stage of the system. The first filter will be used after the silence detection, so that the when audio is detected a wide bandpass filter removes any extraneous noise. The final filter will be a bandpass filter for the frequency range of the instrument detected.

Data Collection

The data for the classification will come from solo performances consisting of a single instrument without vocals. Although we haven't found a single distinct dataset that would serve this purpose, finding appropriate data shouldn't be a problem. There are a number of datasets publicly available (e.g., IRMAS from Universitat Pompeu Fabra) [6]. If we're unable to find a single dataset, we will use collections of music without vocals for the instruments we decide to include. An equitable number of instruments from each instrument family is also necessary. While we haven't selected what instruments, we wish to include at least one string, wind, and percussion instrument. These instruments are each taxa of the chordophone, aerophone, and idiophone families. One potential issue that we must consider in the data used is that the style of genre of music be diverse enough. If the styles of music for an instrument are too specific the classifier may not be able to generalize to identify the instrument in another style of music. For example, the experimental results [1] showed SVM performed measurably worse when discriminating instruments belonging to the same family.

Constraints

As is true in all engineering projects, realistic constraints must be considered. The performance of our system will be dependent on the computing resources available on the LCDK. While processing power isn't much of a concern, the memory restrictions of the LCDK will limit the complexity of the classification. This is because the number of values that the LCDK can store will limit the number of features our classification model can use, limiting the complexity and effectiveness of the classification. For the classification to be real time, the time duration of computation must be within reasonable limits (i.e., no longer than a few seconds). The training datasets will also be limited by the diversity, both in terms of completeness and culture, of musical instruments. Technical constraints aside, we would like to minimize the economic cost of the project (e.g., out-of-pocket expenses).

Responsibilities

Each of us will have the same project roles and responsibilities in order to ensure a fair division of labor. We will strive to not unfairly and unduly burden the other by shirking our collective responsibilities. We will both be responsible for the:

- Selection and preparation of the training dataset.
- Implementation of Yaafe C API.
- Development and implementation of feature extractor and classifier (GMM, SVM, etc.) in C.
- Training, validation, and refinement of classifier model.
- Aiding each other in troubleshooting and debugging.
- Drafting, editing, and revising of the report and presentation.
- Communicating with each other in a professional and academic manner.
- Managing the code and literature in the shared Git repository.
- Any and all unplanned duties that become necessary.

Timeline

The following is the expected weekly progress schedule for this project:

Submittal of project proposal	02 May 2018 (Week 0)
Completion of background research	07 May 2018 (Week 1)
Preparation of dataset and resources	11 May 2018 (Week 1)
Start of initial design & development	14 May 2018 (Week 2)
Start of model training	18 May 2018 (Week 2)
Completion of design & development	21 May 2018 (Week 3)
Refinement and tuning	21 May 2018 (Week 3)
Testing (reliability, performance, etc.)	24 May 2018 (Week 3)
Start to draft project report	25 May 2018 (Week 3)
Final project report & presentation	30 May 2018 (Week 4)
End of Spring Quarter	01 June 2018 (Week 4)

References

1) G. Agostini, M. Longari, E. Pollastri, “Musical Instrument Timbres Classification with Spectral Features,” *EURASIP Journal of Applied Signal Processing*, 2003, pp. 5-14
2) E. Guven, A.M. Ozbayoglu, “Note and Feature Classification by Local Features of a Spectogram,” *Complex Adaptive Systems*, 2012, pp. 182-187
3) J. Marques, P.J. Moreno, *A Study of Musical Instrument Classification Using Gaussian Mixture Models and Support Vector Machines*, Cambridge Technical Reports, 1999
4) C. Joder, S. Essid, G. Richard, “Temporal Integration for Audio Classification With Application to Musical Instrument Classification,” *IEEE Trans. On Audio, Speech, and Language Processing*, 2009, Vol. 17, No. 1, pp. 174-186
5) Yet Another Audio Feature Extractor. https://yaafe.github.io/Yaafe/
6) F. Fuhrmann, “Automatic musical instrument recognition from polyphonic music audio signals,” Ph.D dissertation, Universitat Pompeu Fabra, Barcelona, Spain, 2012.