

Hierarchical Tree for Dissemination of Polyphonic Noise

Rory A. Lewis, Wenxin Jiang, Amanda Cohen and Zbigniew W. Raś

University of North Carolina, 9201 University City Blvd. Charlotte, NC 28223, USA

Abstract. In the continuing investigation of identifying musical instruments in a polyphonic domain, we present a system that can identify an instrument in a polyphonic domain with added noise of numerous interacting and conflicting instruments in an orchestra. We present a strictly classified hierarchical tree where a classification model is trained to correctly estimate an unknown polyphonic sound. A hierarchical tree specifically designed for the breakdown of polyphonic sounds is used to enhance training. This paper shows how goals to determine what hierarchical levels and what combination of mix levels is most effective has been achieved. Learning the correct instrument classification for creating noise together with what levels and mixed the noise optimizes training sets is crucial in the quest to discover instruments in noise. Herein we present a novel system that disseminates instruments in a polyphonic domain

1 Introduction

The challenge for automatic indexing of instruments and Music Instrument Retrieval has moved from the monophonic domain to the polyphonic domain [15, 7]. Previously we presented the rationale and need for creating a categorization system more conducive for music information retrieval, see [6]. Essentially, the Dewey-based, Hornbostel-Sachs classification system, [2, 9] which classified all instruments into the four categories of Idiophones (vibrating bodies), Membranophones (vibrating membranes), Chordophones (vibrating strings), and Aerophones (vibrating air) firstly, permits instruments to fall into a more than one category and secondly, humanistic conventions of categorization of certain instruments such as a piano or tamborine are alien to machine recognition. After fine tuning our categorization, see figure 1 we focused on solving an issue that was prevalent in MIR of polyphonic sounds: In the past, if the training data did not work, one did not know if the bug was in the classification tree or if it was in the levels used to mix noise. With the classification issue resolved we could focus on learning the optimal choice of mixes to create training noise and the optimal levels of mix ratios for noise. Knowing the aforementioned allows discovery of optimal conditions for machine learning how distortions caused by noise can be eliminated in finding an instrument in a polyphonic domain.

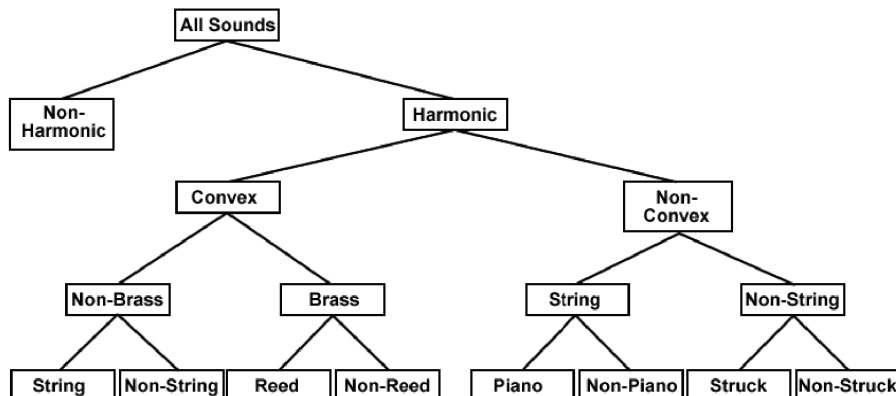


Fig. 1. Instrument Hierarchy Tree: Categorized by the MPEG-7 audio descriptor LogAttackTime. We split the Convex and Non-Convex categories to smaller, more specific groupings of instruments. We select instruments from the smaller categories and combined them to make the polyphonic sounds.

2 Creating a Controlled Noise

With the issue of hierarchical categorization of music instruments solved we decided to create the best controlled environment for training. To do this we decided to use similarly pitched notes. We randomly chose Middle-C because it was, simply put, in the middle of the spectrum. Even when certain instruments could not reach Middle-C we still used the closest C, be it up or down one octave. We also decided to create training sets consisting of both polyphonic and single instrument tuples. The test set comprised all the mixed sounds with different noise-ratios. Also, it was clear that the environment of this polyphonic domain would have to be controlled. The issue would be how one controls noise in a manner that empirical calculations can be run and tested upon the polyphonic domain? Considering our database is MPEG-7 based the authors decided to use 1) MPEG-7 descriptors [5] and five non-MPEG-7 based descriptors based upon the following rationale:

3 Descriptors

3.1 MPEG-7 Descriptors

SpectrumSpread To differentiate between tone-like and noise-like sounds the authors used SpectrumSpread because it is an economical descriptor that indicates whether the power is concentrated in the vicinity of its centroid, or else spread out over the spectrum.

$$S = \sqrt{\frac{\sum_n \log_2(f(n)/1000) - C)^2 P'_x(n)}{\sum_n P'_x(n)}} \quad (1)$$

where $P'_x(n)$ is the power spectrum, $f(n)$ is corresponding frequencies. C is spectrum centroid and S is the spectrum spread, in the form of RMS deviation with respect to the centroid,

SpectrumCentroid To identify instruments with a strong or weak center of gravity of the log-frequency power spectrum we used the SpectrumCentroid descriptor which is defined as the power weighted log-frequency centroid. Here, frequencies of all coefficients are scaled to an octave scale anchored at 1 kHz

$$C = \frac{\sum_n \log_2(f(n)/1000) P'_x(n)}{\sum_n P'_x(n)} \quad (2)$$

where $P'_x(n)$ is the power associated with the frequency $f(n)$.

HarmonicSpectral Variation and Deviation (HSV) and (HSD To realize shifts within a running window of the harmonic peaks we used the HSV and HSD descriptors. Where the HarmonicSpectralVariation is the mean over the sound segment duration of the instantaneous HarmonicSpectralVariation. The HarmonicSpectralDeviation is the sound segment duration of the instantaneous HarmonicSpectralDeviation within a running window computed as the spectral deviation of log-amplitude components from a global spectral envelope.

$$HSV = \frac{\sum_{frame=1}^{nbframes} IHSV(frame)}{nbframes} \quad (3)$$

$$HSD = \frac{\sum_{frame=1}^{nbframes} IHSD(frame)}{nbframes} \quad (4)$$

where nbframes is the number of frames in the sound segment.

HarmonicPeaks To differentiate instruments based on peaks of the spectrum located around the multiple of the fundamental frequency of the signal we used the HarmonicPeaks descriptor. The descriptor here looks for the maxima of the amplitude of the Short Time Fourier Transform (STFT) close to the multiples of the fundamental frequency. The frequencies are then estimated by the positions of these maxima while the amplitudes of these maxima determine their amplitudes.

LogAttackTime (LAT) The motivation for using the MPEG-7 temporal descriptor, LogAttackTime (*LAT*), is because segments containing short LAT periods cut generic percussive (and also sounds of plucked or hammered string) and harmonic (sustained) signals into two separate groups [4, 5]. The LAT is the logarithm of the time duration between the point where the signal starts to the point it reaches its stable part.[10] The range of the LAT is defined as $\log_{10}(\frac{1}{\text{samplingrate}})$ and is determined by the length of the signal. Struck instruments, such as most percussive instruments have a short LAT whereas blown or vibrated instruments contain LATs of a longer duration.

$$LAT = \log_{10}(T1 - T0), \quad (5)$$

where $T0$ is the time the signal starts; and $T1$ is reaches its sustained part (harmonic space) or maximum part (percussive space).

TemporalCentroid To sort instruments based upon the time averaged over the energy envelope we used the TemporalCentroid descriptor which is extracted as follows:

$$TC = \frac{\sum_{n=1}^{\text{length}(SEnv)} n/sr \cdot SEnv(n)}{\sum_{n=1}^{\text{length}(SEnv)} SEnv(n)} \quad (6)$$

3.2 Non-MPEG-7 Descriptors

Energy MFCC The MPEG=7 work in the frequency domain but what about differentiating instruments in the time-domain. In other words, like we hear instruments? We know that the ears perception to the frequency components of sound do not follow the linear scale but the mel-frequency scale [3], which in the linear frequency domain below 1,000 Hz and a logarithmic spacing above 1,000 Hz [13]. To do this, filters have in the past, been spaced linearly at low frequencies and logarithmically at high frequencies [12]. We chose MFCC because it can key in on the known variation of the ears critical band-widths with frequency[11]:

$$M(f) = 2595 \log_{10}(1 + f/700) \quad (7)$$

where f is frequency in hertz. Based on this assumption, the mel-frequency cepstrum coefficient, once known, opens the door to computing MFCC [1].

ZeroCrossingDensity When a pure sound, a monophonic harmonic sound is affected by noise, such as we are doing, the average number of positive and negative traces of the sound wave that cross over zero (zero-crossings) per second is affected. Using the ZeroCrossing Density descriptor allows us to consider this dimension of the experiment.

RollOff To differentiate frequencies that fall below an experimentally chosen percentage of the accumulated magnitudes of the spectrum [14] We chose to include the RollOff descriptor.

Flux When non-linear sound waves are disturbed, such as being bombarded by noise, the measurement of the variations of the energy through the frequency scales is flux. Having a means to measure these changes as various levels of noise are imposed upon a sound is crucial in differentiating the noise.

4 Experiments

In our experiments we used Weka 3.5.7 and chose the J48 decision tree as our classification algorithm to build models for each training data. The goal is re-creating rules of how modification of the mix levels of various combinations of instruments influences the quality of the trained classifiers. We used the McGill University CDs, used worldwide in research on music instrument sounds [8]. To test how the accuracy of a classifier improves the estimation of the dominant instrument in a polyphonic sound, we built a training dataset comprising mixtures of single instrument sounds and polyphonic sounds. The polyphonic sounds comprised one dominant sound and a specific mix of instruments located in the leaves at the same level of the hierarchical tree with decreased amplitude which we observed as "noise." Continuing with this strategy we combined more instruments according to our hierarchical tree. However, before making the polyphonic sounds for the entire hierarchical tree, we ran experiments to determine what levels of noise would be optimal for each instrument in order to ensure a trained robust estimation of the classification model's unknown polyphonic sound. To make the size of training data reliable we used the pitch of a single tone of 4C containing 10 different dominant instruments.

instrument	category
ElectricGuitar	string
Oboe	reed
B-flatclarinet	string
CTrumpet	brass
TenorTrombone	brass
Violin	string
Accordion	reed
TenorSaxophone	reed
DoubleBass	string
Piano	string

Table 1. List of instruments making the basis for the noise

As shown in Table 2, we observed that the 75% mixture got the best performance in terms of dominant instrument estimation. In order to decide whether this conclusion holds when we observe each single node of the hierarchy tree, we divided this instrument sound into 2 groups which belong to the different category (reed, string), then run the same test, we got the following results training set string accuracy reed accuracy

Training Set	Accuracy
mix100+single	69.90%
mix75+single	79.56%
mix50+single	73.82%
mix30+single	73.48%
mix25+single	70.26%
mix65+single	73.06%
mix80+single	73.58%

Table 2. The results showing classification confidence from the 10-folder cross validation process

Training Set	Accuracy
mmix100+single	68.14% 71.70%
mix80+single	73.76% 82.94%
mix75+single	79.50% 83.98%
mix65+single	70.63% 71.72%
mix50+single	77.66% 76.53%
mix30+single	68.05% 53.19%
mix25+single	74.34% 78.47%

Table 3. Results after we divided the instrument sounds into 2 groups according to each category

As shown in Table 3, the 75% noise ratio is still the best choice for each single group of instruments, regardless the distribution of the whole tests changed a little bit when different path of hierarchy tree is followed.

Figure 2 shows a graphic representation of the performance of the polyphonic sounds with the non-dominant instrument at 80% volume, 75% volume, 65%, 50%, 30%, and 25%, and the accuracy with which the database identified the dominant instrument. It also shows the performance of the specific groups of sounds, the string group and the reed group. In all three data sets, a non-dominant instrument volume of 75% yielded the most accurate results. However, in the case of the reed group a non-dominant of 80% gave only slightly less accurate results in comparison and more accurate results than any other group at that volume. A non-dominant instrument volume of 30% yielded the least accurate results in the case of the reed group and the string group, with the reed group showing the least accurate results of any instrument group at any volume. The accuracy of the whole data set was lowest at 100%. Figure 3 shows the same results with more exact values on the left-hand vertical axis. The values on the right-hand axis show the percentile that the data falls in, in order to see how the groups compare overall to each other. Overall the reed group yielded the most accurate results. Even at its lowest accuracy, the results of the reed group didn't drop below 72%, which cannot be said for either the string group or the entire set of sounds. In fact the accuracy of the string group drops to nearly 68% at its lowest point. The string group also displayed the greatest changes in accuracy,

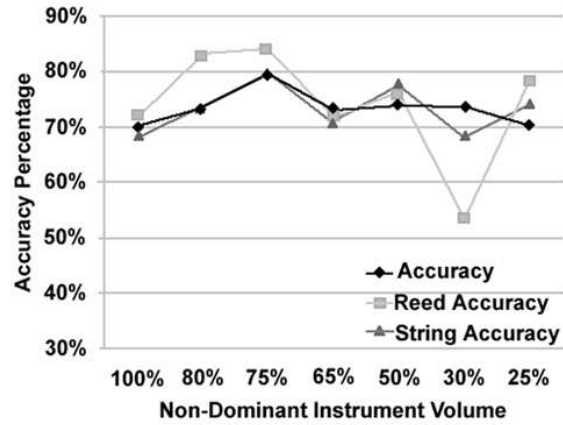


Fig. 2. Graphic Representation of the Results: This graph shows the performance of the polyphonic sounds and the accuracy with which the database identified the dominant instrument. It also shows the performance of the specific groups of sounds, the string group and the reed group.

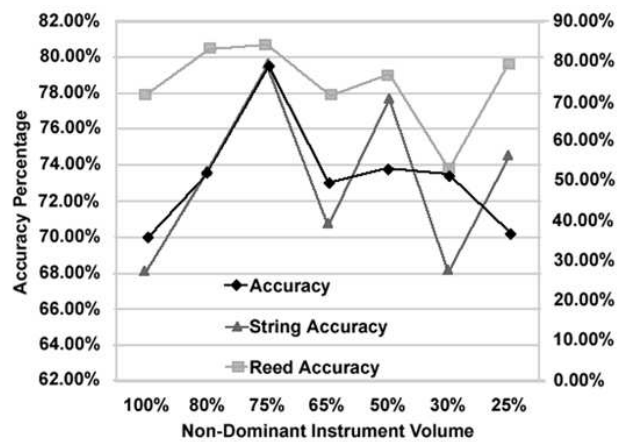


Fig. 3. Graphic Representation of the Results with Percentile Axis: This graph shows the same results with more exact values on the left-hand vertical axis. The values on the right-hand axis show the percentile that the data falls in, in order to see how the groups compare overall to each other.

with the largest difference between two non-dominant instrument volumes being roughly 9%.

5 Conclusion and Future Work

Using the new hierarchical tree in our closed domain show that a 75% volume is optimal. Knowing this we now have the tools to know that future errors in retrieving instruments in a polyphonic sound if wrong, will be because of a property inherent in the expanded domain. This is good news as it directs us to the fault. Our next domain will bear instruments playing various sets of harmonics in tex training set not lal on one similar pitch. Once this is achieved, the ultimate goal of identifying instruments in a non-harmonic or harmonic noise will be a step closer.

6 Acknowledgment

This work is supported by the National Science Foundation under grant IIS-0414815.

References

1. S. Davis and P. Mermelstein. Comparison of parametric representation for monosyllabic word recognition in continuously spoken sentences. *IEEE Transactions of Acoustics, Speech, and Signal Processing*, ASSP-28, No. 4, Aug.:357–366, 1980.
2. M. Doerr. Semantic problems of thesaurus mapping. *Journal of Digital Information*, 1 issue 8, Article No. 5: 2001-03-26:2001-03, 2001.
3. Z. Fang and G. Zhang. Integrating the energy information into mfcc. *International Conference on Spoken Language Processing*, 1. Oct 16-20:389–292, 2000.
4. E. Gomez, F. Gouyon, P. Herrera, and X. Amatriain. Using and enhancing the current mpeg-7 standard for a music content processing tool. *Proceedings of the 114th Audio Engineering Society Convention, Amsterdam, The Netherlands*, March, 2003.
5. J.M.Martinez and F. P. R. Koenen. Iso/iec jtc 1/sc 29. *Information Technology Multimedia Content Description Interface Part 4: Audio*, 2001.
6. R. Lewis and A. Wiczorkowska. Categorization of musical instrument sounds based on numerical parameters. *June 28-30 in Warsaw Poland*, 4585/2007:784–792, 2007.
7. R. Lewis, X. Zhang, and Z. Raś. Knowledge discovery based identification of musical pitches and instruments in polyphonic sounds. *International Journal of Engineering Applications of Artificial Intelligence*, Volume 20 Issue 5 August:637–645, 2007.
8. F. Opolko and J. Wapnick. Mums – mcgill university master samples. cd's. 1987.
9. M. Patel, T. Koch, M. Doerr, and C. Tsinaraki. Semantic interoperability in digital library systems. *Technology-enhanced Learning and Access to Cultural Heritage. UKOLN*, University of Bath, IST-2002-2.3.1.12, 2005.

10. G. Peeters, S. McAdams, and P. Herrera. Instrument sound description in the context of mpeg-7. *Proceedings of the International Computer Music Conference*, (ICMC'00), Berlin, Germany, 2000.
11. J. Picone. Signal modeling techniques in speech recognition. *Life Science Research Report*, T.H. Bullock, Ed., IEEE, 1993, 81(9):1215–1247, 1993.
12. M. Schroeder. Recognition of complex acoustic signals. *Life Science Research Report*, T.H. Bullock, Ed., (Abakon Verlag, Berlin) vol. 55:323–328, 1977.
13. S. Stephens and J. Volkman. The relation of pitch to frequency. *American Journal of Psychology*, 53(3):329–353, 1940.
14. A. Wierzchowska and E. Kolczynska. Quality of musical instrument sound identification for various levels of accompanying sounds. *Machine Learning: ECML 2007. 18th European Conference on Machine Learning. Warsaw, Poland*, September 17-21, 2007:28–36, 2007.
15. A. Wierzchowska, P. Synak, R. Lewis, and Z. Raś. Creating reliable database for experiments on extracting emotions from music. *Intelligent Information Processing and Web Mining, Advances in Soft Computing*, pages 395–404, 2005.