

COVER SONG DETECTION: FROM HIGH SCORES TO GENERAL CLASSIFICATION

Suman Ravuri

Intl. Computer Science Inst., UC Berkeley
Dept. of Electrical Engineering
Berkeley, CA

Daniel P.W. Ellis

LabROSA, Columbia University
Dept. of Electrical Engineering
New York, NY

ABSTRACT

Existing cover song detection systems require prior knowledge of the number of cover songs in a test set in order to identify cover(s) to a reference song. We describe a system that does not require such prior knowledge. The input to the system is a reference track and test track, and the output from the system is a binary classification of whether the reference/test pair is either from a reference/cover or reference/non-cover. The system differs from state-of-the-art detectors by calculating multiple input features, performing a novel type of test song normalization in order to combat against “impostor” tracks, and performing classification using either a SVM or multi-layer perceptron (MLP). On the covers80 test set, the system achieves an equal error rate of 10%, compared to 21.3% achieved by the 2007 LabROSA cover song detection system.

Index Terms— Cover songs, music information retrieval.

1. INTRODUCTION

Much of the focus of cover song detection over the past few years has been, given a reference track, trying to pick out its n cover songs from a test set of m songs. Implicit in the problem statement is that we have prior knowledge of the structure of the test set. While there has been a strong increase in performance in this particular task (one merely has to look at the scores of the winners of MIREX audio cover song detection competition from 2006 to 2008 - 761/3300 in 2006 vs 2422/3300 in 2008), ultimately, we expect a theoretically perfect cover song detector, when given a reference and test song, to give us a determination of whether the reference/test pair is a reference/cover. Our systems, named the “HydraSVM” and “HydraMLP” for reasons outlined below, is to our knowledge, is the first system to perform this sort of general classification. We build upon the work of previous systems to build the general classification system.

There are, however, two major problems in traditional cover song detectors that preclude such systems from being strong general cover song classifiers. The first is that most structured detection systems use only one feature and try to

tune this feature to identify all types of cover songs. The problem with this approach is that cover songs have many different types of changes from the original track. Possible changes include genre, tempo, instrumentation, singer gender, key, and sometimes even melody. Using a single feature to capture all these differences adequately may be impossible, so we have opted for a multiple feature approach for our system. We have found the multistream systems significantly outperformed its single-stream counterparts.

The second, and possibly more difficult, problem is that all current features are not scaled well enough to identify covers without prior knowledge. For instance, a score of 100 may indicate a cover song for one reference track, but this same score may indicate a non-cover for a different reference track. This does not cause problems when prior knowledge of the test set is given, because with prior knowledge, the aim is to rank the test set, but this is a significant problem for general classification because the correct threshold for a given reference track is not known. We have found a normalization based on “impostor” test tracks that allows us to perform proper score normalization.

2. SYSTEM

Figure 1 shows a block diagram of the system structure. First, beat-synchronous chromagrams are calculated for both the reference and test tracks. This calculation is well-described in [1] and [2]. For each track, the chroma are calculated with preference windows set at 3 different tempo means: 240 beats/minute, 120 beats/minute, and 60 beats/minute. Then, the system calculates 3 different features at each of the 3 tempo means, giving 9 separate scores¹ on how “cover-like” a test song is to a given reference track. Then, each score undergoes a normalization to a) properly scale all the values and b) reduce the effect of “impostor” test tracks. Finally, these normalized scores are placed in a 9-dimensional feature vector and the vector is classified as either from a reference/cover or reference/non-cover by either a support vector machine or multi-layer perceptron (MLP). The SVM-based

¹The 9 separate scores is why the system is named “Hydra”

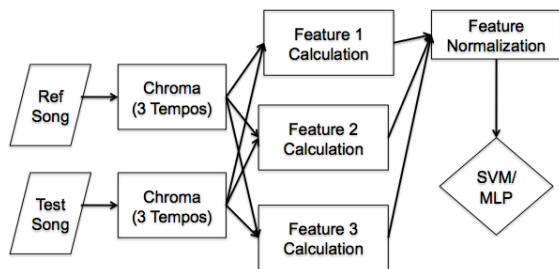


Fig. 1. Block Diagram of Hydra Systems.

system is called “HydraSVM” while the MLP-based counterpart is named “HydraMLP”

The next few subsections outline the system in more depth.

2.1. Feature Calculation

2.1.1. Feature 1

This feature is the one used in the 2006 LabROSA submission to the MIREX audio cover song competition and details of its calculation can be found in [1]. The system square-root compresses the reference and test chroma and cross-correlates the two resulting chroma. The utility of the cross-correlation arises from the observation that if fragments of the reference and test track match - as often happens in a cover - the cross-correlation will exhibit rapidly-changing peaks at different time lags. This cross-correlation is performed for all twelve circular shifts of the test chroma and the shift for which the highest cross-correlation peak score occurs is selected. Then, that cross-correlation is high-pass filtered to remove the general triangular structure, leaving only the peaks. Finally, the score of the maximum peak is outputted as a feature.

2.1.2. Feature 2

This cross-correlation feature, used in the 2007 LabROSA and described in [3], is a minor variant of that described in Section 2.1.1. For this feature, the chroma energy of each beat of the chromagram is normalized to sum to one after square-root compression. Then, the chromagram itself is high-pass filtered to de-emphasize a same note being played for multiple beats. Then, cross-correlation is performed as above, and the maximum value of all twelve cross-correlations is outputted as the feature. [3] cited a performance improvement of this feature as a reason to switch features, but we have since found that keeping both features leads to better performance.

2.1.3. Feature 3

This is a feature re-implemented from [4] and modified for use with beat-synchronous, 12-dimensional chroma instead of

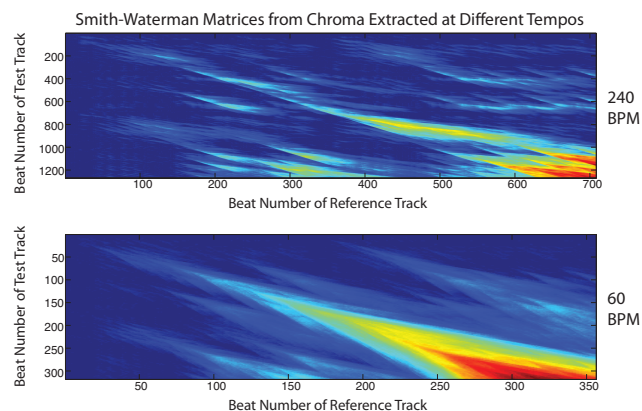


Fig. 2. Top Pane: Smith-Waterman matrix at 240bpm. Bottom Pane: SW matrix at 60 bpm.

93ms-windowed, 36-dimensional Harmonic Pitch Class Profiles.

The dynamic programming feature is a two stage process. First, the cover song detection system calculates a “binary similarity matrix” of the reference/test pair. Then, the Smith-Waterman algorithm is run on the binary similarity matrix, and the highest value of the dynamic program is returned as a feature. Details of the calculation can be found in [5].

2.2. Multiple Tempo Levels

Sometimes, an incorrect tempo level in the beat-tracking algorithm will lead to poor representation of the musical progression of the reference or cover track. This will invariably lead to bad feature scores, even if the features themselves are somewhat robust to changes in melody. In order to circumvent this problem, we calculate the three above features from chroma beat-tracked at 240 beats/minute, 120 beats/minute, and 60 beats/minute. We also experimented with mixing tempo levels (i.e. using 240 beats/minute for the reference track and 120 beats/minutes for the test track), but including these cross-tempos resulted in no performance improvement.

Figure 2 shows Smith-Waterman matrices for a reference/cover pair at 240 beats/minute (top pane) and 60 beats/minute (bottom pane). At 240 beats/minute, there is a very small matching region and this region is indistinguishable from reference/non-cover matrices, but at 60 beats/minute, the matrix exhibits very strong matching characteristics and looks very much like that of a reference/cover.

2.3. Feature Normalization

In order to introduce the idea of feature normalization, consider a chromagram of a test song such that all the semitones

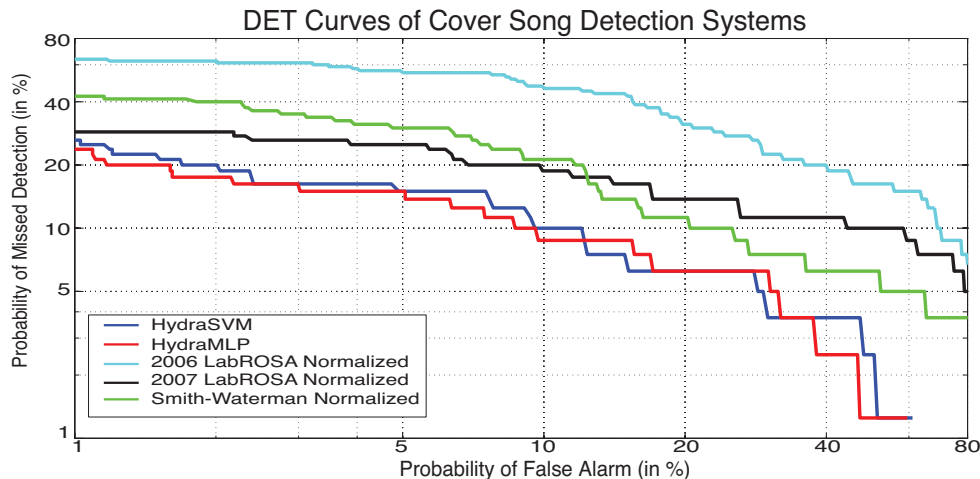


Fig. 3. Detection Error Tradeoff Curves for various systems on covers80 test set.

were of equal value and the beats had equal energy to each other (this would be a “white noisy” track). Such an “impostor” track would score highly on all three aforementioned feature calculations and the test track would be classified a cover for every reference song.

In order to combat this problem, for every test track, we calculate features with random reference tracks and take a mean and standard deviation of these features. Since the prior probability of a reference/test pair being a cover is much less than 1%, we can consider this feature normalization to be a form of crude modeling on how the test track performs with random non-cover reference tracks. We then mean/variance normalize the features to obtain a z-score for use during classification.

2.4. Classification

We train a support vector machine and a multilayer perceptron to classify cover songs. Training is done on [hypeful.com](http://www.hypeful.com)’s 25 best covers of 2008.² The set consists of 34 original tracks and 39 covers. Most songs are pop music and between 1 and 3 cover songs exist for each original track.

For training, one has to be careful not to train the classifier on too many reference/non-cover pairs. If, for example, one trains the classifier on all possible combinations of the 34 reference tracks and 39 covers, there would be 39 reference/cover pairs and 1287 reference/non-covers in the training set and during test the classifier would determine that every reference/test pair is a non-cover. We found that remov-

ing 75% of the reference/non-cover training examples yields weights that perform well for general classification.

For the support vector machine, we use linear outputs. For the multilayer perceptron, we structure the perceptron with 9 input units (corresponding to each of the 9 features), 75 hidden units, and 2 output units. The outputs are softmax so that the outputs correspond to the probability of the reference/test pair being that of a reference/cover or not.

3. EVALUATION

We tested HydraSVM and HydraMLP on the “covers80” test set.³ The test set is structured so that there are 80 reference tracks and 1 song per reference in the test set. Each query to the system is a reference song/test track, giving 6400 queries to the system. 80 of these queries are reference/covers, while the other 6320 queries are of reference/non-covers. We also tested these two systems against three different baselines: the 2006 LabROSA MIREX submission, feature-normalized, the 2007 LabROSA MIREX submission, feature-normalized, and one that is a feature-normalized Smith-Waterman feature system. These baselines comprise the current state-of-the-art systems.

Figure 3 shows the DET curves for all 5 systems. The HydraSVM and HydraMLP perform significantly better than the 3 baseline systems and similarly to each other. The Equal Error Rates shown in Table 1 show that the HydraSVM and HydraMLP has a 27.5% and 29.7% relative improvement in EER over the next best system.

²<http://www.hypeful.com/2008/12/23/25-best-cover-songs-of-2008/>

³available at <http://labrosa.ee.columbia.edu/projects/coversongs/covers80/>

It is also important to note that the baseline systems are feature-normalized. Figure 4 shows DET curves for the three baseline systems, each with and without feature normalization. All 3 systems show some improvement, with the most dramatic being the Smith-Waterman system.

System	Equal Error Rate
2006 LabROSA System + Normalization	27.5%
2007 LabROSA System + Normalization	16.8%
Smith-Waterman System + Normalization	13.8%
HydraSVM	10.0%
HydraMLP	9.7%

Table 1. Equal Error Rates on covers80 test set

System	unnorm EER	norm EER	Relative Improv.
2006 LabROSA	32.4%	27.5%	15.1%
2007 LabROSA	21.3%	16.8%	21.1%
Smith-Waterman	20.0%	13.8%	31.0%

Table 2. Performance on covers80 test set for unnormalized and normalized features

4. DISCUSSION

The Hydra systems show how one can create a general classifier from using existing parts and a couple new ideas. In particular, using multiple features, performing feature normalization, and using a classifier (whether a SVM or MLP), can make a robust and high-performance general cover song classifier without any prior knowledge. The penalty we pay to perform general classification is a supervised learning step, but we have found that very few training examples are needed to generate good training weights.

The possible more attractive upshot of the system is that specialized features can be plugged into the system with minimal effort. Moreover, we no longer need features that score highly if the pair is a reference/cover and low if the pair is the reference/non-cover. All we need now are features that exhibit a strong separability between reference/covers and reference/non-covers. This hopefully can allow for new features that have not been currently thought of yet.

5. ACKNOWLEDGMENTS

This work was supported by the National Defense Science and Engineering Graduate Fellowship (NDSEG). Moreover, I would like to thank Professor Nelson Morgan for discussing ideas and suggesting improvements for the algorithm.

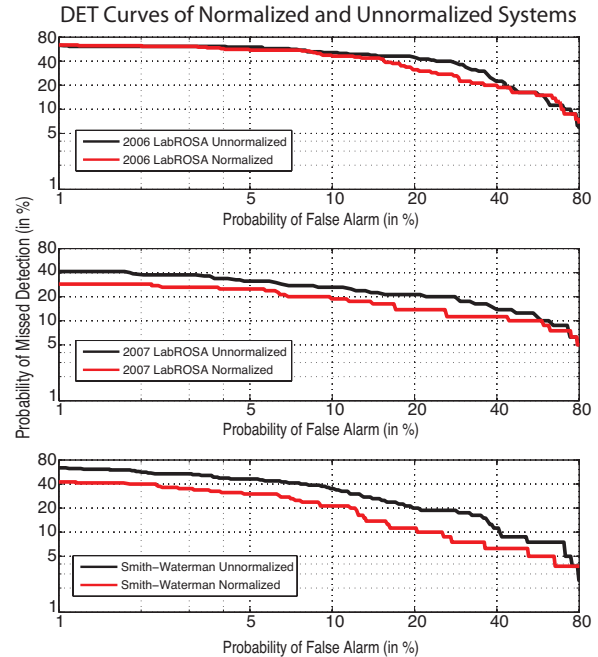


Fig. 4. Detection Error Tradeoff Curves for normalized and unnormalized baseline systems.

6. REFERENCES

- [1] D.P.W. Ellis and G. Poliner, "Identifying 'cover songs' with chroma features and dynamic programming beat tracking," *IEEE Transactions on Audio, Speech, and Language Processing*, pp. 1429–1432, April 2007.
- [2] D.P.W. Ellis, "Identifying 'cover songs' with beat-synchronous chroma features," *Music Information Retrieval Evaluation eXchange (MIREX)*, 2006.
- [3] D.P.W. Ellis and C. Cotton, "The 2007 labrosa cover song detection system," *Music Information Retrieval Evaluation eXchange (MIREX)*, 2007.
- [4] J. Serra and E. Gomez, "A cover song identification system based on sequences of tonal descriptors," *Music Information Retrieval Evaluation eXchange (MIREX)*, 2007.
- [5] J. Serra, E. Gomez, P. Herrera, and X. Serra, "Chroma binary similarity and local alignment applied to cover song identification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 16, pp. 1138–1151, August 2008.