MIREX 2009
ENHANCING AUDIO CLASSIFICATION WITH TEMPLATE FEATURES AND POSTPROCESSING EXISTING AUDIO DESCRIPTORS

A. Grecu, T. Lidy, A. Rauber
Vienna University of Technology, Austria
Department of Software Technology
and Interactive Systems

ABSTRACT

In our efforts for the actual MIREX, we introduce two new feature sets, discuss example feature sets which show the advantages of postprocessing and combining existing audio descriptors and explain a weighted one-against-one multi-class SVM. The new audio descriptor is based on simple spectral amplitude and power binning and shows promising results when postprocessed accordingly thereby expanding its dimensionality. Our second feature set is of symbolic nature and captures onsets and loudness of repetitive tones thereby covering another area of the musical piece than other audio descriptors do. These feature sets – some existing sets and some postprocessed extensions of them – are then feed to a multiclass SVM ensemble which votes using the one-against-one principle where each binary SVM classifier is additionally weighted by the output of a correction SVM which estimates whether the input falls into one of the classes serviced by the classifier.

1 INTRODUCTION

Classification of music by genre, artist or mood are important tasks for retrieval and organization of music databases. In previous works audio features were used in classifiers directly and without any postprocessing leading to suboptimal exploitation of the extracted information. Due to the missing postprocessing step simple feature-sets may have been overlooked which would develop their potential only if postprocessed accordingly. Our aim is therefore to use existing audio descriptors, postprocess them and combine the results with a new kind of symbolic audio descriptor which extracts onsets of repetitive tones in music. We will also describe a simple feature-set based on spectral binning which is believed to give very good classification results by just using adequate postprocessing.

The overall scheme of our proposed genre classification system is shown in Figure 1. It processes an audio file in two ways to predict its label (genre, mood, artist, etc.). While in the first branch, the audio feature extraction methods described in Section 2.1 are applied directly to the audio signal and then postprocessed resulting in the sets discussed in Section 2.2, the second branch uses a template extractor described in Section 2.3 to find onsets of repetitive tones and subsequently generate new audio features from the resulting data. The feature-sets extracted from the music serve as input to a multi-class SVM ensemble consisting of binary SVMs combined with a weighted one-against-one voting method which is explained in more detail in Section 3.2.

2 SYSTEM DESCRIPTION

2.1 Audio Feature Extraction

All the following descriptors are extracted from a spectral representation of 6 sec. segments in the audio signal. While in full length songs, the number of segments varies and can be controlled using a 'step_width' parameter, in a 30-second audio clip, usually 5 segments are extracted. Rhythm Patterns are summarized using the median over the 5 segments, Statistical Spectrum Descriptors are summarized computing the mean.

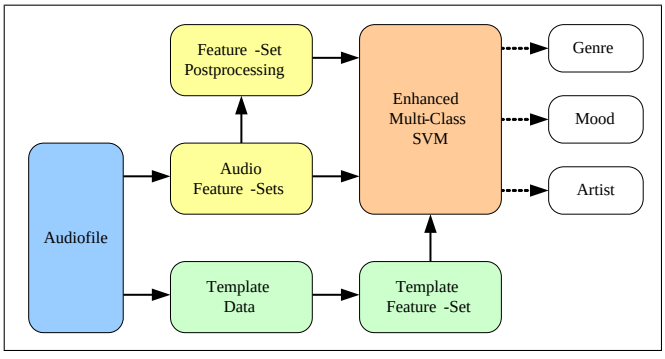

Figure 1. General framework of the system

2.1.1 Rhythm Pattern (RP)

The feature extraction process for a Rhythm Pattern [4, 2] is composed of two stages. First, the specific loudness sensation on 24 critical frequency bands is computed, by using a Short Time FFT, grouping the resulting frequency bands to the Bark scale, applying spreading functions to account for masking effects and successive transformation into the Decibel, Phon and Sone scales. This results in a psychoacoustically modified Sonogram representation that reflects human loudness sensation. In the second step, a discrete Fourier transform is applied to this Sonogram, resulting in a (time-invariant) spectrum of loudness amplitude modulation per modulation frequency for each individual critical band. After additional weighting and smoothing steps, a Rhythm Pattern exhibits magnitude of modulation for 60 modulation frequencies (between 0.17 and 10 Hz) on the 24 critical bands. Note that when using 22khz audio, the number of critical bands is reduced to 20 and the final Rhythm Pattern has 1200 dimensions. For details refer to [4, 2].

2.1.2 Statistical Spectrum Descriptor (SSD)

In the first part of the algorithm for computation of a Statistical Spectrum Descriptor (SSD) the specific loudness sensation is computed on 24 Bark-scale bands, equally as for a Rhythm Pattern. Subsequently, the mean, median, variance, skewness, kurtosis, min- and max-value are calculated for each individual critical band. These features computed for the 24 bands constitute a Statistical Spectrum Descriptor. SSDs describe fluctuations on the critical bands and are able to capture additional timbral information compared to a Rhythm Pattern, yet at a much lower dimension of the feature space, as shown in the evaluation in [2].

2.1.3 Modulation Frequency Variance Descriptor (MVD)

This descriptor measures variations over the critical frequency bands for a specific modulation frequency (derived from a Rhythm Pattern). Consider a Rhythm Pattern, i.e. a matrix representing the amplitudes of 60 modulation frequencies on 24 critical bands: The MVD vector is computed by taking statistics (mean, median, variance, skewness, kurtosis, min and max) for one modulation frequency over the 24 (resp. 20) bands. A vector is computed for each of the 60 modulation frequencies. The MVD descriptor for an audio file is computed from the mean over the multiple MVDs of its segments.

2.1.4 Average Spectral Energy (ASE)

This feature set is a very simple one, containing just a coarse binning of the amplitude and the power spectrum at $40, 120, 500, 2000, 6000, 11000$ and 22050Hz respectively, averaged over all SFFT windows.

2.2 Postprocessed Feature-Sets

2.2.1 Small Rhythm Pattern Extension (SRPE)

For enhancing the classification capability of the Rythm Pattern set we designed an additional set constructed by nonlinear postprocessing of the RPs where the first the loudness amplitudes and their squares at the 60 modulation frequencies are summed up to result into "modulation energies" pro critical band. Then two matrices are built, the first one by dividing each amplitude pro band with each squared amplitude and the second one by dividing the squared amplitudes by the simple ones. The final extension to RP has $2 \cdot (24 + 24 \cdot 24)$ or 1200 dimensions for 24 critical bands. We call this set the small extension because using the same procedure it is also possible to sum the energies of the critical bands at the modulation frequencies resulting in a much higher-dimensional set (7320).

2.2.2 Relative Spectral Energy Matrix (RSEM)

As an extension to the Average Spectral Energy set and in analogy to the Small Rhythm Pattern Extension, we calculate two matrices by dividing each of the amplitude bins by each of the power bins and vice versa, resulting in a 98 dimensional extension.

2.2.3 ASE-weighted Statistical Spectrum Descriptor (ASE×SSD)

This is actually a combination of the Average Spectral Energy set with the Statistical Spectrum Descriptor. For each of the 24 critical bands 4 matrices are calculated, with the first one being built by dividing each amplitude bin by each statistic, the second one by dividing each power bin by each statistic and the other two being the built using the reciprocals of the elements of the first two matrices.

2.3 Symbolic Feature Extraction

2.3.1 Template Descriptors

An algorithm coming from the blind source separation domain was adapted for genre classification and related tasks. The goal of the template extractor [1] in blind source separation is to separate sounds or tones from instruments by making use of the repetitive structure of music. In the original setting, each instrument sound is represented by a template which is adapted during an iterative training process to better represent its sound, suppressing the other instruments. The sum of these templates at their respective onsets will then reconstruct the song, though this is not a perfect reconstruction.

In genre classification, the sheer amount of information makes such an approach infeasible due to the high

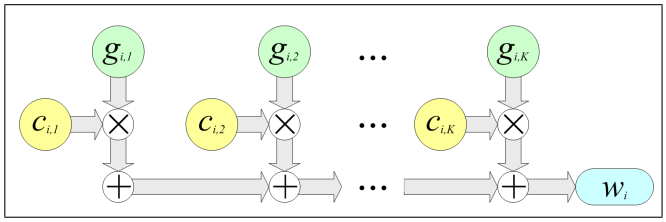

Figure 2. Workflow of the SVM ensemble

demand on computational resources, thus several simplifications were done leading to a different interpretation of the templates. In order to save time, the templates are not adapted and are initialized only by cutting a part of a track which is chosen randomly. Thus the songs are reconstructed only by small pieces of (possibly) other songs. Furthermore the length of the templates is restricted to 1024 samples or about $1/20$ of a second which due to their short duration represent the timbre or texture of the sound at a specified time rather than a tone or a mixture of tones.

These templates themselves are then further processed to result in the template feature set. The descriptors this set is composed of are for example the mean onset amplitude, mean onset distance, mean overlap with the other templates (matrix), template count, etc.

3 CLASSIFICATION

3.1 Classification Setup

For classification purposes we use a weighted one-against-one multi-class support vector machine (SVM) ensemble which is fed with all feature-sets concatenated into one high-dimensional vector. The output class is decided by adding the weighted decisions of each class i to class j binary SVM and taking the class with the highest accumulated value.

3.2 Weighted one-against-one SVM

We use an one-against-one SVM ensemble for classification, which incorporates correcting SVMs [3] as a second layer to the classifier SVMs. The correcting SVMs weight the vote of each classifier SVM c_{ij} which differentiates between classes i and j. Figure 2 shows the workflow of the weighted SVM ensemble, depicting the classifiers c_{ij}, correcting SVMs g_{ij} and the summed vote w_i for class i. The rationale is that the classifier SVMs can only competently respond to a new sample if its class label falls into one of the two classes that it was trained for. If the sample has some other class label, then the SVM is assumed to respond randomly and uncorrelated with other SVMs. In that way the votes for the incorrect classes will be distributed more uniformly while only the correct class will draw off votes from the other classes thereby reaching the highest score. As this may not be the case we use correcting SVMs which only have to estimate the probability that the sample falls into one of the two classes serviced by their corresponding classifier SVMs.

We noticed a general improvement in the prediction accuracy of the ensemble which is most noticeable when having a high number of classes with only few samples per class. As the correcting SVMs are not expected to estimate the error rate of the binary SVM but just its competency in classifying a given sample, they can be trained independently from the classifier SVMs.

4 REFERENCES

[1] Andrei Grecu. *Musical Instrument Sound Separation: Extracting Instruments from Musical Performances - Theory and Algorithms*. VDM Verlag Dr. Müller, Saarbrücken, Germany, 2008.

[2] T. Lidy and A. Rauber. Evaluation of feature extractors and psycho-acoustic transformations for music genre classification. In *Proc. ISMIR*, pages 34–41, London, UK, September 11-15 2005.

[3] M. Moreira and E. Mayoraz. Improved pairwise coupling classification with correcting classifiers. In *ECML '98: Proceedings of the 10th European Conference on Machine Learning*, pages 160–171, London, UK, 1998. Springer-Verlag.

[4] A. Rauber, E. Pampalk, and D. Merkl. The SOM-enhanced JukeBox: Organization and visualization of music collections based on perceptual models. *Journal of New Music Research*, 32(2):193–210, June 2003.