ON THE RELEVANCE OF SPECTRAL FEATURES FOR INSTRUMENT CLASSIFICATION

Andreas B. Nielsen, Sigurdur Sigurdsson, Lars K. Hansen, and Jerónimo Arenas-García*

The Technical University of Denmark
DK-2800, Kgs. Lyngby, Denmark
{abn,siggi,lkh,jag}@imm.dtu.dk

ABSTRACT

Automatic knowledge extraction from music signals is a key component for most music organization and music information retrieval systems. In this paper, we consider the problem of instrument modelling and instrument classification from the rough audio data. Existing systems for automatic instrument classification operate normally on a relatively large number of features, from which those related to the spectrum of the audio signal are particularly relevant. In this paper, we confront two different models about the spectral characterization of musical instruments. The first assumes a constant envelope of the spectrum (i.e., independent from the pitch), whereas the second assumes a constant relation among the amplitude of the harmonics. The first model is related to the Mel Frequency Cepstrum Coefficients (MFCCs), while the second leads to what we will refer to as Harmonic Representation (HR). Experiments on a large database of real instrument recordings show that the first model offers a more satisfactory characterization, and therefore MFCCs should be preferred to HR for instrument modelling/classification.

Index Terms— Musical instruments modelling, harmonics structure, feature extraction

1. INTRODUCTION

In the last years there has been an increasing interest in methods that aid music organization and music recommendation systems, mainly motivated by the large digitalization of music. For a summary of relevant advances in this exciting field, the reader is referred to the website of the series of Music Information Retrieval Conferences[1].

In this paper, we will pay attention to the problem of instrument classification from the rough audio data (see, for instance, [4]). Among the features that are normally used for this task, those related to the spectral characteristics of the instrument are particularly relevant. We can think of two different models of how the spectrum of a particular instrument changes for different pitches. The first model accepts that the envelope of the spectrum remains constant for all notes, while the second, proposed in [8], states that it is the relation among the amplitude of the harmonics which remains constant. These two models are associated to two set of features: the Mel Frequency Cepstrum Coefficients (MFCCs) and the Harmonic Representation (HR) features.

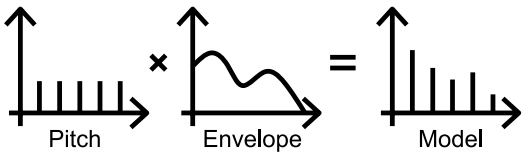

Fig. 1. Model of the spectrum of a harmonic signal. The spectrum is divided into a pitch and an envelope.

The two models above are conflicting ones, and, therefore, the main goal of this paper is to illustrate which is the one that better explains the structure of musical instruments. In order to do so, we will train different classification models using both MFCCs and HR features extracted from a rather large database of real instruments recordings [5].

The result of our analysis shows that the models built upon MFFCs outperform those relying on HR. Therefore, MFCCs should be preferred for instrument modelling/classification.

2. SPECTRAL CHARACTERIZATION OF MUSICAL INSTRUMENTS

The spectral structure of a harmonic signal can roughly be divided in two components, as illustrated in Fig. 1: the pitch and the envelope. The pitch is what is perceived as the tone, and its value is given by the fundamental frequency, i.e., the frequency of the first harmonic. The envelope is a modulation of the pitch. If two instruments are playing the same note the pitch will be the same. Under this simplified model it will therefore only be the envelope that makes the two sounds different. Obviously, the pitch changes for different notes, but how the envelope changes is a bit more subtle. Two models are suggested, one that assumes the envelope to be constant, and a second that accepts that it is the relative amplitude of the harmonics that remains constant.

2.1. Constant envelope model: MFCC features

According to this model, the envelope for the spectrum of a particular instrument does not change with the pitch. Therefore, when the pitch is changed the amplitude of each harmonic in the sound varies (see Fig. 2). This model is well

*This work was partly supported by the Danish Technical Research Council, through the framework project 'Intelligent Sound', www.intelligentsound.org (STVF No. 26-04-0092), and by the Spanish Ministry of Education and Science with a Postdoctoral Felowship to the last author.

[1] http://www.ismir.net

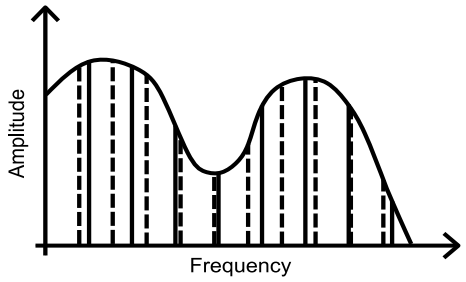

Fig. 2. Constant envelope model. Spectrums for two notes with different fundamental frequency are shown (solid and dashed). If the envelope is constant the amplitude of the harmonics must change.

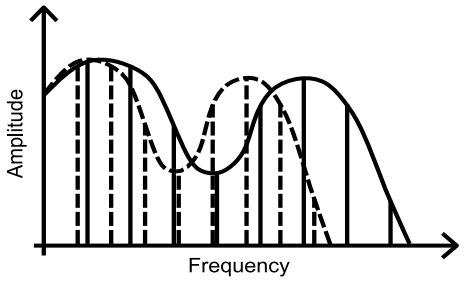

Fig. 3. Constant harmonics amplitude model. The same two notes from Figure 2 are shown (solid and dashed). The envelope is stretched under this model.

motivated for some instruments, such as string instruments, by assuming that the pitch is induced by the vibration of the string and the envelope is controlled by the casing, which is of course constant. For other instruments, like trumpets, the validity of the model is not that clear.

It is hard to directly extract the shape of the envelope, but MFCCs capture much of the same information. MFCCs were initially developed for speech, but they are also heavily used in other sound applications, see, for example, [6]. To compute the MFCCs the amplitude of the spectrogram is first found using the Discrete Fourier Transform (DFT) on a small window of the audio data. The modulus of the DFT is then filtered with a Mel filter bank and the logarithm of the outputs is taken. In this way, we obtain a series of numbers related to the energy of the input signal in different frequency bands, whose central frequencies approximate the Mel scale[2]. Finally, the Discrete Cosine Transform (DCT) is taken, and the result is the MFCCs.

Since MFCCs consist, roughly speaking, of a DCT of a mel-scaled version of the power spectrum, they contain information about the shape of the envelope of the spectrum. Then, if the envelope were constant, the MFCCs extracted from different windows of the same instrument should be similar, even if they correspond to different notes.

From our explanation, it can be seen that the first MFCC is closely related to the amplitude of the original signal. Therefore, in this paper we will leave out that coefficient, using the values of the next 10 MFCCs to construct the models.

2.2. Constant harmonics amplitude: HR features

This model was suggested in [8], and works under the assumption that it is the amplitude of the different harmonics which remains constant. This means that when the pitch is increased (decreased) the envelope of the spectrum is stretched (compressed) and, therefore, its shape changes, see Fig. 3.

If this model is valid, a good representation for instrument modelling consists simply of the estimated amplitudes of the harmonics, to which we refer in the sequel as Harmonics Representation (HR) features. As we did for the MFCCs, to remove the dependence with the amplitude of the sound signal (i.e., its volume), it is advisable to normalize the amplitude of all harmonics with that of the first one.

[2]The Mel scale is related to the perceptual capabilities of the human auditory system.

The amplitude of each harmonic is directly measurable if the pitch is known. A pitch detector from [7] is used and, together with the labels of the data set and visual inspection of discrepancies, very reliable estimates were produced. The amplitudes of the first 50 harmonics are found, what gives a total of 49 relative HR features.

3. CLASSIFICATION MODELS

In order to study the accurateness of the previous models, we will build multi-class classification models that predict, from both MFCCs and HR features, which instrument is being played. We will use two different classification technologies in order to make our conclusions as general as possible, and to validate that similar conclusions are extracted when using both approaches.

The formulation of the problem can be stated as follows: given a set of N training pairs $\{\mathbf{x}^{(i)}, \mathbf{y}^{(i)}\}_{i=1}^{N}$, where $\mathbf{x}^{(i)}$ is a vector containing the features extracted from a window of audio data (either MFCCs or HR) and $\mathbf{y}^{(i)}$ is a vector of targets containing an '1' in the position associated to the right instrument and zeros elsewhere, the task is to build a function that is able to predict the right targets of new data as accurately as possible.

It is important to remark that the data in our training data sets are strongly unevenly distributed among classes (the number of data in the most numerous class is more than 20 times larger than for the smallest one), thus our classification models should be able to compensate this effect and assume equal priors for all instruments.

3.1. Probabilistic Network

Our first classifier is a multi layer perceptron (MLP) [2] with a single layer of M hidden units and C outputs, each one corresponding to one instrument. The hyperbolic tangent function is used for activation in the hidden units and the softmax function is used in the output units. This fact, together with the use of the logarithmic cost function, makes the network estimate the *a posteriori* probabilities of class membership [3].

To compensate for unbalanced classes we use the following modified cost function:

$$E = -\sum_{i=1}^{N}\sum_{k=1}^{C} \lambda_k y_k^{(i)} \ln \hat{y}_k^{(i)}, \qquad (1)$$

where $y_k^{(i)}$ is the k-th component of $\mathbf{y}^{(i)}$, $\hat{y}_k^{(i)}$ is the k-th output of the network, and $\lambda_k = 1/N_k$, N_k being the number of samples in class k.

The minimization of (1) is carried out using an implementation of the Broyden-Fletcher-Goldfarb-Shanno (BFGS) method[3].

3.2. Kernel Orthonormalized Partial Least Squares

As a second method, we will consider a kernel based method for multi-class classification. The method consists of two different steps: first, relevant features are extracted from the input data using the Kernel Orthonormalized Partial Least Squares (KOPLS) algorithm [1]; then, a linear classifier is trained to obtain the final predictions of the network.

KOPLS is a method for kernel multivariate analysis that basically works by projecting the input data into a Reproducing Kernel Hilbert Space, where standard OPLS analysis is carried out. To present the method, let us first introduce matrices $\mathbf{\Phi} = [\phi(\mathbf{x}^{(1)}), \ldots, \phi(\mathbf{x}^{(N)})]^T$ and $\mathbf{Y} = [\mathbf{y}^{(1)}, \ldots, \mathbf{y}^{(N)}]^T$, where $\phi(\cdot)$ is the function that projects input data to some feature space $\mathcal{F}$. Let us also denote by $\mathbf{\Phi}' = \mathbf{\Phi}\mathbf{U}$ a matrix containing n_p projections of the original input data, $\mathbf{U}$ being a projection matrix of size $dim(\mathcal{F}) \times n_p$. Then, the KOPLS problem can be formulated as follows (see [1]):

$$\begin{array}{ll} \text{maximize:} & \text{Tr}\{\mathbf{U}^T\mathbf{\Phi}^T\mathbf{Y}\mathbf{Y}^T\mathbf{\Phi}\mathbf{U}\} \\ \text{subject to:} & \mathbf{U}^T\mathbf{\Phi}^T\mathbf{\Phi}\mathbf{U} = \mathbf{I} \end{array} \tag{2}$$

where the maximization is carried out with respect to $\mathbf{U}$.

The Representer Theorem states that $\mathbf{U}$ can be expressed as a linear combination of the training data, i.e., $\mathbf{U} = \mathbf{\Phi}^T\mathbf{A}$, and carry out the maximization with respect to $\mathbf{A}$ instead. However, some advantages in terms of computation and regularization are obtained if we impose a sparse representation for the projection vectors, i.e., we admit that $\mathbf{U} = \mathbf{\Phi}_R^T\mathbf{B}$, where $\mathbf{\Phi}_R$ is a subset of the training data containing only R instances, and $\mathbf{B}$ is the new projection matrix of size $R \times n_p$. Then, the maximization problem for this KOPLS with reduced complexity (rKOPLS) can be stated as:

$$\begin{array}{ll} \text{maximize:} & \text{Tr}\{\mathbf{B}^T\mathbf{K}_R\mathbf{Y}\mathbf{Y}^T\mathbf{K}_R^T\mathbf{B}\} \\ \text{subject to:} & \mathbf{B}^T\mathbf{K}_R\mathbf{K}_R^T\mathbf{B} = \mathbf{I} \end{array} \tag{3}$$

where $\mathbf{K}_R = \mathbf{\Phi}_R\mathbf{\Phi}_R^T$ involves only inner products in $\mathcal{F}$.

In order to compensate for unbalanced classes, only two modifications to the standard rKOPLS algorithm are needed:

1. All classes should be equally represented in $\mathbf{\Phi}_R$.
2. The correlation matrices in (3) should be replaced by their weighted counterparts where all classes have the same influence, i.e.,

$$\begin{array}{rcl} \mathbf{K}_R\mathbf{Y} & \leftarrow & \sum_{i=1}^{N}\sum_{k=1}^{C}\lambda_k y_k^{(i)}\mathbf{k}^{(i)}\mathbf{y}^{(i)\,T} \\ \mathbf{K}_R\mathbf{K}_R^T & \leftarrow & \sum_{i=1}^{N}\sum_{k=1}^{C}\lambda_k y_k^{(i)}\mathbf{k}^{(i)}\mathbf{k}^{(i)\,T}. \end{array}$$

   where we have defined $\mathbf{k}^{(i)} = \mathbf{\Phi}_R\ \phi(\mathbf{x}^{(i)})$.

With these simple modifications, matrix $\mathbf{B}$ can be found by standard generalized eigenvalue analysis, as in [1].

[3]We have used the matlab implementation available at http://www2.imm.dtu.dk/~hbn/immoptibox/.

Once the non-linear features have been extracted from the training data, a single layer perceptron (SLP) with C outputs and softmax activation is trained to learn the relation between these features and the target data, also by minimizing (1) using the BFGS algorithm.

4. EXPERIMENTS

4.1. Data set description and settings

For our experiments we have used a comprehensive database of real instrument recordings, which is available for research purposes at [5]. There are a total of 20 instruments in the data set, all of them recorded at 44.1 kHz and 16 bit/sample. A single note is played at a time, and notes from the complete range of each instrument are included. Moreover, three different amplitude levels are played (pianissimo, mezzoforte and fortissimo). For string instruments there are both arco and pizzicato, and the notes are also played on the different strings. For some of the wind instruments vibrato is also included. We have not included in our data set the pianissimo amplitude level because of the low SNR. Also the pizzicato of string instruments is excluded due to an extremely short duration of the notes. In order for our experiments to be as independent from pitch as possible, instruments were requested to share at least one octave. Three instruments were too far away and had to be discarded, leaving 17 instruments for the classification.

The recordings were processed to remove silence periods between notes, and MFCCs and HR features were extracted using a window size of 50 ms, which is the time frame on which we do the classifications. This process resulted in a total of 282,812 patterns for training and testing the models. Two different partitions were done for the two sets of experiments described in the next subsections.

Regarding classifier settings, cross-validation was carried out to select the free parameters. For the probabilistic MLP networks (MLP in the sequel) the number of hidden units was set to 30, for which the validation curves were already flat. We found no problems of overfitting, probably because of the large data set being used. For the rKOPLS + SLP network (simply rKOPLS in the following), the number of points from each class that are included in $\mathbf{\Phi}_R$ was set to 30, also according to the behavior of validation curves. Finally, we used a Gaussian kernel, whose width was also selected by cross-validation.

As we did for the training of the networks, the accuracy rates that we report in the next subsections are balanced so that all instruments have the same influence on them. Results are averaged over 10 runs of the algorithms.

4.2. Generalization capabilities of the models

In the first experiment, the training data consists of MFCCs/HR extracted from notes spanning the common octave: from B3 to Bb4; all other data is placed in the test data set. Note that the two models of Section 2 tend to agree if the pitch is only slightly modified, while their disagreement is more important for large variations. In this sense, this experiment, where both models are trained using a small range of notes (where they should roughly agree) and tested far away, is a good setting to test their validity.

		MFCCs	HR
MLP	Tr/Val	91.4 / 79.1	79.3 / 58.8
	Tr/Tst	91.2 / 42.8	78.5 / 12.9
rKOPLS	Tr/Val	89.5 / 80.1	78.2 / 57.7
	Tr/Tst	89.3 / 42.4	77.4 / 14.2

Table 1. Accuracy rates achieved when training the models using the octave B3-Bb3, and testing outside.

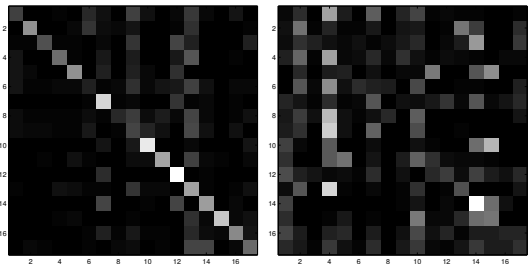

Fig. 4. Confusion matrices achieved by rKOPLS for the test set of experiment 1. MFCCs on the left and HR on the right.

Cross-validation (CV) in this setting was carried out by using 11 folds, each one consisting of one note of the training data set. Accuracy error rates are reported in Table 1, both for the 11-fold CV ('Tr/Val' rows) and for the final training and test error rates ('Tr/Tst').

We can first see that 11-fold validation accuracies are much higher than those achieved in the test data set. The fact that both the classifiers based on MFCCs and HR degrade significantly outside the training octave, indicates that both models fail when moving very far away from the training interval. Note however, that not only MFCCs based classifier always get better accuracy rates, but also their degradation with respect to validation rates is much lower (about 50 % in comparison to 25 % or even less for the classifiers working on HR). The best performance of MFCCs is also clear when looking at the test confusion matrices that are obtained when using the two sets of features (Fig. 4). Therefore, we can conclude that the constant envelope model is a useful approximation to the real behavior of the spectrum of musical instruments, and that MFCCs should be preferred to HR for instrument modelling.

Finally, it is also worth pointing out the consensus between the performance trends shown by MLP and rKOPLS networks, showing that our conclusions are indeed due to the spectral features that are used to feed the classifiers.

4.3. Complete pitch range training

For this experiment the training and test span the whole pitch range of each instrument, with every second note in each set. In this way, we will be able to study the recognition rates that can be achieved from both MFCCs and HRs, if the classifiers are provided with information covering a pitch range as wide as possible. In this case, the training set is divided into 5 folds for validation purposes, each fold taking one out of each 5 notes.

Results for this experiment are displayed in Table 2. Compared to the results of the previous setup, test recognition rates are significantly better, specially when the MFCCs are used, achieving 75.9 % recognition rate in combination with the rKOPLS classifier, whose performance is slightly better than that of the MLP network. In relation to previous published studies (see, for instance, [4]) the results in Table 2 look quite competitive, although a direct comparison is not possible given the differences in the nature of the data sets and the experimental settings.

		MFCCs	HR
MLP	Tr/Val	87.4 / 70.7	52.2 / 29.7
	Tr/Tst	86.1 / 74.7	50.2 / 38.0
rKOPLS	Tr/Val	89.4 / 73.2	63.3 / 32.4
	Tr/Tst	84.4 / 75.9	60.7 / 41.2

Table 2. Accuracy rates achieved when the training and test data sets are formed with alternating notes.

In the light of these results one can conclude that MFCCs are preferable to HR features not only for instrument modelling, but also for automatic classification systems. It also seems clear that, to obtain a classifier of high performance, the training data should include data spanning a pitch range as wide as possible.

5. CONCLUSION

In this paper we have analyzed the spectral structure of musical instruments. Two different models about the behavior of the spectrum of instruments when playing different notes and their associated feature representations, MFCCs and HR, are revised. Experiments on a rather large data base of real instruments have shown that MFCCs should be preferred to HR, both for musical instrument modelling and for automatic instrument classification.

6. REFERENCES

[1] J. Arenas-García, K.B. Petersen, L.K. Hansen, "Sparse Kernel Orthonormalized PLS for feature extraction in large data sets," to appear in NIPS, 2006.

[2] C.M. Bishop, *Neural networks for pattern recognition*, Oxford University Press, 2004.

[3] J. Cid-Sueiro, A.R. Figueiras-Vidal, "On the Structure of Strict Sense Bayesian Cost Functions and its Applications," *IEEE Trans. Neural Networks*, Vol. 12, pp. 445–455, 2001.

[4] S. Essid, G. Richard, B. David, "Hierarchical Classification of Musical Instruments on Solo Recordings," in *ICASSP'06*, vol. V, pp. 817–820, 2006.

[5] L. Fritts, "Musical Instrument Samples," http://theremin.music.uiowa.edu, The University of Iowa.

[6] K.D. Martin, "Sound-Source Recognition: A Theory and Computational Model," Ph.D. thesis, Massachusetts Institute of Technology, Cambridge, MA, 1999.

[7] A.B. Nielsen, "Pitch Based Sound Classification," M.S. thesis, IMM, The Technical University of Denmark, 2005.

[8] Y.-G. Zhang, C.-S. Zhang, "Separation of Music Signals by Harmonic Structure Modeling," in *Advances in Neural Information Processing Systems 18*, pp. 1619–1626, 2005.