KDD-based approach to musical instrument sound recognition

Dominik Ślęzak, Piotr Synak, Alicja Wieczorkowska, Jakub Wróblewski

Polish-Japanese Institute of Information Technology
Koszykowa 86, 02-008 Warsaw, Poland

Abstract. Automatic content extraction from multimedia files is a hot topic nowadays. Moving Picture Experts Group develops MPEG-7 standard, which aims to define a unified interface for multimedia content description, including audio data. Audio description in MPEG-7 comprises features that can be useful for any content-based search of sound files. In this paper, we investigate how to optimize sound representation in terms of musical instrument recognition purposes. We propose to trace trends in evolution of values of MPEG-7 descriptors in time, as well as their combinations. Described process is a typical example of KDD application, consisting of data preparation, feature extraction and decision model construction. Discussion of efficiency of applied classifiers illustrates capabilities of further progress in optimization of sound representation. We believe that further research in this area would provide background for automatic multimedia content description.

1 Introduction

Automatic extraction of multimedia information from files is recently of great interest. Usually multimedia data available for end users are labeled with some information (title, time, author, etc.), but in most cases it is insufficient for content-based searching. For instance, the user cannot find automatically all segments with his favorite tune played by the flute in the audio CD. To address the task of automatic content-based searching, descriptors need to be assigned at various levels to segments of multimedia files. Moving Picture Experts Group has recently elaborated MPEG-7 standard, named "Multimedia Content Description Interface" [8], that defines a universal mechanism for exchanging the descriptors. However, neither feature (descriptor) extraction nor searching algorithms are encompassed in MPEG-7. Therefore, automatic extraction of multimedia content, including musical information, should be a subject of study.

All descriptors used so far reflect specific features of sound, describing spectrum, time envelope, etc. In our paper, we propose a different approach: we suggest observation of feature changes in time and taking as new descriptors patterns in trends observed for particular features. We discuss how to achieve it by applying data preprocessing and mining tools developed within the theory of rough sets introduced in [13].

The analyzed database origins from audio CD's MUMS [12]. It consists of 667 samples of recordings, divided onto 18 classes, corresponding to musical instruments (flute, oboe, clarinet, violin, viola, cello, double bass, trumpet, trombone, French horn, tuba) and their articulation (vibrato, pizzicato, muted).

2 Sound descriptors

Descriptors of musical instruments should allow to recognize instruments independently on pitch and articulation. Sound features included in MPEG-7 Audio are based on research performed so far in this area and they comprise technologies for musical instrument timbre description, sound recognition, and melody description. Audio description framework in MPEG-7 includes 17 temporal and spectral descriptors divided into the following groups (cf. [8]):

- basic: instantaneous waveform, power values
- basic spectral: log-frequency power spectrum, spectral centroid, spectral spread, spectral flatness
- signal parameters: fundamental frequency, harmonicity of signals
- timbral temporal: log attack time and temporal centroid
- timbral spectral: spectral centroid, harmonic spectral centroid, spectral deviation, spectral spread, spectral variation
- spectral basis representations: spectrum basis, spectrum projection

Apart from the features included in MPEG-7, the following descriptors have been used in the research ([6], [10], [17], [18]):

- duration of the attack, quasi-steady state and ending transient of the sound in proportion to the total time
- pitch of the sound
- contents of the selected groups of harmonics in spectrum, like even/odd harmonics Ev/Od

$$Ev = \frac{\sqrt{\sum_{k=1}^{M} A_{2k}^2}}{\sqrt{\sum_{n=1}^{N} A_n^2}} \qquad Od = \frac{\sqrt{\sum_{k=2}^{L} A_{2k-1}^2}}{\sqrt{\sum_{n=1}^{N} A_n^2}} \tag{1}$$

  and lower/middle/higher harmonics $Tr_1/Tr_2/Tr_3$ (Tristimulus parameters [14], used in various versions)

$$Tr_1 = \frac{A_1^2}{\sum_{n=1}^{N} A_n^2} \quad Tr_2 = \frac{\sum_{n=2,3,4} A_n^2}{\sum_{n=1}^{N} A_n^2} \quad Tr_3 = \frac{\sum_{n=5}^{N} A_n^2}{\sum_{n=1}^{N} A_n^2} \tag{2}$$

  where A_n denotes the amplitude of the n^{th} harmonic, N – the number of harmonics available in spectrum, $M = \lfloor N/2 \rfloor$ and $L = \lfloor N/2 + 1 \rfloor$
- vibrato amplitude
- statistical properties of sound spectrum, including average amplitude and frequency deviations, average spectrum, standard deviations, autocorrelation and cross-correlation functions ([2])
- descriptors based on wavelet analysis and numerous other features

3 KDD process

One of the main goals of data analysis is to properly classify objects (described by some attributes) to some classes. Reasoning with data can be stated as a classification problem, concerning prediction of decision class basing on information provided by attributes. For this purpose, one stores data in so called decision tables, where each training case drops into one of predefined decision classes.

A decision table takes the form of $\mathbf{A} = (U, A \cup \{d\})$, where each attribute $a \in A$ is identified with a function $a : U \to V_a$ from the universe of objects U into the set V_a of all possible values on a. Values $v_d \in V_d$ correspond to mutually disjoint decision classes of objects. In case of the analysis of the musical instrument sound data [12], one deals with a decision table consisting of 667 records corresponding to samples of musical recordings. We have 18 decision classes corresponding to various kinds of musical instruments – flute, oboe, clarinet, violin, viola, cello, double bass, trumpet, trombone, French horn, tuba – and their articulation – vibrato, pizzicato, muted ([17]). These classes define decision attribute d.

Methods for construction of classifiers can be regarded as tools for data generalization. These methods include rule-based classifiers, decision trees, k-NN classifiers, neural nets, etc. However, the process of analyzing data cannot be restricted just to the classifier construction. In the particular case of the musical instrument analysis, one has to extract a decision table itself – to choose the most appropriate set of attributes-descriptors A, as well as to calculate values $a(u) \in V_a$, $a \in A$, for particular objects-samples $u \in U$. Thus, it is better to write about this task in terms of a broader methodology.

Knowledge Discovery in Databases (KDD) is a process which, according to widely accepted scheme, consists of several steps (see e.g. [5]), such as understanding application domain, determining a goal, creating/selecting a target data set, preprocessing, data reduction and transformation, selection of data mining method, algorithms and parameters, model construction (data mining), and interpretation of results. In case of musical instruments classification, the first two steps comprises of the musical domain analysis. Next, proper selection ([9]) and reduction ([13]) of the set of features is crucial for efficiency of classification algorithm. In some cases a set of attributes is worth transforming into more suitable form before it is used to model the data. For instance, when the data set is described by decision rules, one may transform attribute values to gain higher support of rules, keeping their accuracy, and increasing generality of a model. The need of such a transformation is shown for various kinds of feature domains: numeric, symbolic, as well as, e.g., for time series (see e.g. [11], [15], [16], [19]).

4 Preprocessing of musical sound data

The goal of this research is to be able to construct classifiers for the musical instrument sound recognition. Thus, we need to prepare the training data in the form of decision table $\mathbf{A} = (U, A \cup \{d\})$, where each element $u \in U$ corresponds to a sound sample, each element $a \in A$ is a numeric feature corresponding to one

of sound descriptors and decision attribute $d \notin A$ labels particular object-sound with integer codes adequate to instrument. For such a preparation we need a framework for preprocessing original data, in particular, for extracting features most relevant to the task of the sound recognition.

The main difficulty of sound analysis is that many useful attributes of sound are not concerned with the whole sample. E.g. spectrum-based attributes (tristimulus parameters, pitch, etc.) describe rather a selected time frame on which the spectrum was calculated than the whole sound (moreover, these attributes may change from one segment of time to another). One can take a frame from quasi-steady part of a sample and treat it as a representative of the whole sound but in this case we may loose too much information about the sample. Our approach is to take into account both sample based attributes (e.g. envelope-dependent as steady state or transient duration) and window based ones. Because the latter vary in time, they should be treated as time series and gathered within an additional table. Further preprocessing is then needed to transform such a family of time series into a set of attributes.

There are numerous mathematical approaches for approximation of fundamental signal frequency and thus – estimation of the length of periods in case of instrument sounds. We have used the following function (see e.g. [17]):

$$AMDF(i) = \frac{1}{N} \sum_{k=0}^{N} |A_k - A_{i+k}| \tag{3}$$

where N is the length of interval taken for estimation and A_k is the amplitude of the signal. Values of $AMDF(i)$ within the interval of admissible period lengths approximate the period for a given sound. In our experiments, we used a mixed approach to approximate periods – based both on searching for stable minima of $AMDF$ and maxima of spectrum obtained using DFT.

We propose to analyze the following descriptors:

1. Envelope descriptors: Each sample was split onto 6 intervals of equal length. Average values of amplitudes within these intervals are referred, respectively, as $Envelope1, \ldots, 6$.
2. Temporal descriptors:
   - Signal length, denoted as *Length*
   - Relative length of the attack (till reaching 75% of maximal amplitude), quasi-steady (after the end of attack, till the final fall under 75% of maximal amplitude) and decay time (the rest of the signal), denoted, respectively, by *Attack*, *Steady* and *Decay*
   - The moment of reaching maximal amplitude, denoted by $Maximum$
   - Area under the curve of envelope (approximated by means of values $Envelope1, \ldots, 6$), denoted by *EnvFill*
   - Numbers of envelope based clusters of two types (see Section 5), denoted by $Cluster6$ (number of the closest of 6 representative envelope curves, shown at Fig. 1) and $Cluster9$ (similarly, but for 9 representatives)

3. Spectral descriptors:
   - Harmonics defined by (1), denoted by $EvenHarm$ and $OddHarm$
   - Brightness and Irregularity (see e.g. [17])
   - Tristimulus parameters defined by (2), denoted by $Tristimulus1, 2, 3$
   - Fundamental frequency, denoted by $Frequency$

We consider the following structure of database: Table INSTRUMENTS (667 records, 18 columns) gathers temporal and spectral descriptors. It is linked in 1:1 manner with Table ENVELOPES (667 records, 7 columns). It has additional column $Instrument$ which states the code of musical instrument, together with its articulation (18 values). We also define table WINDOWS (190800 records, 10 columns), where each record corresponds to a small interval of the sound sample. We decided to set up the length of those intervals as 4 times the fundamental period of the sound. We decompose each musical sound sample onto such intervals and calculate value sequences and final features for each of them. For each sample we thus obtain (Length∗Frequency/4) records. Each record is labeled with spectral descriptors defined in the same way as for INSTRUMENTS but calculated locally. As a result, we obtain the relational database, where INSTRUMENTS and WINDOWS are linked in 1:n manner, by the code of the instrument sample (primary key for INSTRUMENTS and foreign key for WINDOWS).

5 Time domain features

The basis of musical sound recognition process is a properly chosen set of descriptors that potentially contains relevant features distinguishing one instrument from another. It seems to be very important to choose not only descriptors characterizing the whole sample at once, but also those describing how parameters change in time. Features described in Section 2 can be used to describe a segment with a summary value or with a series of sampled values. Descriptors can be stored as a sequence corresponding to the dynamic behavior of a given feature over the sound sample. Analysis of regularities and trends occurring within such a temporary sequence can provide the values of conditional features labeling objects-sounds in the final decision table. Especially interesting trends are supposed to be observed during the attack part of signal.

We propose to search for temporal patterns that can potentially be specific for one instrument or a group of instruments. Such patterns can be further used as new descriptors, like Cluster6 and Cluster9 in table INSTRUMENTS (see Section 4). Values of those columns were calculated in the following way:

1. The most representative sound envelopes occurring in data were extracted. For column Cluster6 we found 6 representatives shown in Fig. 1. (Similarly, we derived 9 representatives for Cluster9).
2. For each object (sound sample) we calculated the Euclidean distance (calculated with respect to 6 envelope values, for both Cluster6 and Cluster9) to the closest representative.

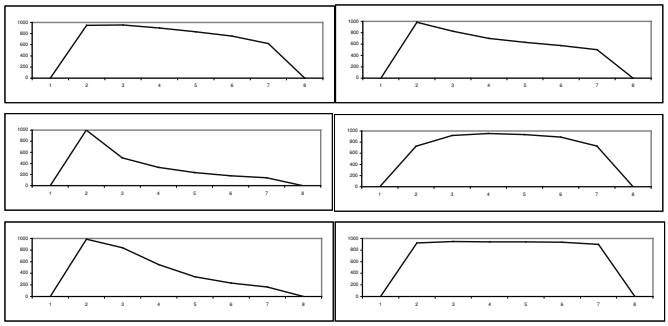

Fig. 1. Centroids (the most typical shapes) of sound envelopes, used in clustering.

The above attributes describe general trends of the amplitude values in time. Results presented in Section 7 show potential importance of such features. Similar analysis can be performed over spectral features stored in table WINDOWS (see Section 4), by searching for, e.g., *temporal patterns* (cf. [16]).

Generation of temporal patterns requires the choice of descriptors that would be used to characterize sound samples and a method to measure values of those descriptors in time. For the latter we propose to use time window based technique. We browse a sample with time windows of certain size. For a given time window we compute values of all descriptors within it, and this way generate one object of a new temporal information system $\mathbf{A} = (\{x_1, x_2, ..., x_n\}, A)$, where x_i is a measurement from the i-th window using descriptors from A (Actually, we constructed table WINDOWS by repeating this procedure for all samples of sounds). Next, we use it to determine optimal *temporal templates* that respond to temporal patterns.

Temporal templates can be of numeric or symbolic type. In the former case, one can compare them with temporal clustering methods. In the latter case they are built by using expressions $(a \in V)$, where $a \in A$ and $V \subseteq V_a$. Formally, *template* is then a set of expressions involving any subset $B \subseteq A$:

$$T = \{(a \in V) : a \in B, V \subseteq V_a\} \tag{4}$$

By *temporal template* we understand

$$\mathbf{T} = (T, t_s, t_e), \;\; 1 \leq t_s \leq t_e \leq n \tag{5}$$

Templates and temporal templates are intensively studied in literature ([1], [11], [16]). To outline the intuition, which is behind these notions, let us understand template as a strong regularity in data, whereas temporal template as strong regularity occurring in time.

In one musical sound sample we can find several temporal templates. They can be time dependent, i.e. one can occur before or after another. Though, we can treat them as sequence of events. From such a sequence we can discover frequent *episodes* – collections of templates occurring together (see e.g. [7], [16]). We expect some of such episodes to be specific only for particular instrument or group of instruments.

6 Automatic extraction of new attributes

Extraction of temporal templates or temporal clusters is an exemplary method of using 1:n connection between data tables for creating new, aggregated columns. Here, aggregation is understood in terms of deriving descriptors corresponding to trends in behavior of values of some locally defined columns (in our case: spectral columns belonging to table WINDOWS), ordered by the time column. One of the main goals of our future research is to automatize the process of defining temporal attributes, to get ability of massive search through the space of all possibilities of temporal descriptors.

Such a process has been already implemented for SQL-like aggregations in [19]. Exemplary features, found automatically as SQL-like aggregations from table WINDOWS, are the following: *average LocOdd from WINDOWS* and *sum LocTri3 from WINDOWS where LocTri3 < LocTri2*. The goal of the searching algorithm is here to extract aggregations of potential importance while distinguishing instrument decision classes. Such attributes can be added as new columns to table INSTRUMENTS. In some situations adding such new features improves and simplifies the laws of reasoning about new cases.

Automatic extraction of significantly new features is possible also for single data tables, not embedded into any relational structure. In case of numerical features, such techniques as discretization, hyperplanes, clustering, and principle component analysis (see e.g. [11]), are used to transform the original domains into more general or more descriptive ones. One can treat the analysis process over transformed data either as a modeling of a new data table (extended by new attributes given as a function of original ones) or, equivalently, as an extension of model language. The latter means, e.g., change of metric definition in k-NN algorithm or extension of language of rules or templates.

In our approach the original data set is extended by a number of new attributes defined as a linear combination of existing ones. Let $B = b_1, ..., b_m \subseteq A$ be a subset of attributes, $|B| = m$, and let $\alpha = (\alpha_1, ..., \alpha_m) \in \mathbf{R}^m$ will be a vector of coefficients. Let $h : U \to \mathbf{R}$ be a function defined as:

$$h(u) = \alpha_1 b_1(u) + ... + \alpha_m b_m(u) \tag{6}$$

Usefulness of new attribute defined as $\overline{a}(u) = h(u)$ depends on proper selection of parameters B and α. The new attribute $\overline{a}$ is useful, when the model of data (e.g. decision rules) based on discretized values of $\overline{a}$ becomes more general (without loss of accuracy). Evolution strategy algorithm optimizes $\overline{a}$ using quality function based on intuition that a model with lower number of (consistent) decision rules is better than the others (cf. [3], [13]). For further details refer to [15].

7 Results of experiments

Fig. 2 presents the results of classification of sounds with respect to the kinds of instruments and their usage. We consider 18 decision classes and 667 records. We use standard CV-5 method for evaluation of resulting decision models. Presented results correspond to two approaches to constructing classifiers:

– Best k-NN: Standard implementation with tuning parameter k
– RS-decision rules: Algorithm implemented in [3] for finding optimal ensembles of decision rules, based on the theory of rough sets [13]

Attributes	**Best k-NN**	**RS-decision rules**
Envelope	36,3%	17,6%
Envelope with linear combinations	42,1%	11,1%
Temporal	54,3%	39,4%
Spectral	34,2%	14,6%
Temporal + Spectral	68,4%	46,9%

Fig. 2. Experimental results

.

Particular rows of the table in Fig. 2 correspond to performance of the above algorithms over decision tables consisting of various sets of conditional attributes. Groups of features correspond to notation introduced in Section 4:

– Envelope: 36% of correct classification of new cases into 18 possible decision classes – a good result in case of k-NN over 6 quite naive conditional features.
– Envelope with linear combinations: Improvement of correct classification in case of k-NN after adding linear combinations over original Envelope of dimensions, found by the approach discussed in Section 6. This confirms the thesis about importance of searching for optimal linear combinations over semantically consistent original features, stated in [15].
– Temporal: Incredible result for just a few, very simple descriptors, ignoring almost the whole knowledge concerning the analysis of music instrument sounds. Still k-NN (54,3%) better than RS-decision rules (39,4%).
– Spectral: Classical descriptors related to spectrum analysis seem to be not sufficient to this type of task. From this perspective, the results obtained for Temporal features are even more surprising.
– Temporal + Spectral: Our best result, 68,4% for k-NN, still needing further improvement. Again, performance of RS-decision rules is worse (46,9%), although other rough set based methods provide better results – e.g., application of the algorithm for the RSES library (see [4]) gives 50,3%.

8 Conclusions

We focus on methodology of musical instrument sound recognition, related to KDD process of the training data analysis. We propose a novel approach, being a step towards automatic extraction of musical information within multimedia contents. We suggest to build classifiers by basing on appropriately extracted features calculated for particular sound samples – objects in a relational database. We use features similar to descriptors from MPEG-7, but also consider the time series framework, by taking as new descriptors temporal clusters and patterns observed for particular features. Experience from both signal analysis and other data mining applications suggests us to use additional techniques for automatic new feature extraction as well.

Acknowledgements

Supported by Polish National Committee for Scientific Research (KBN) in the form of PJIIT Project No. 1/2001 and KBN grant No. $8T11C02417$.

References

1. Agrawal, R., Mannila, H., Srikant, R., Toivonen, H., Verkamo, I.: Fast Discovery of Association Rules. In: Proc. of the Advances in Knowledge Discovery and Data Mining. AAAI Press / The MIT Press, CA (1996) pp. 307–328.
2. Ando, S., Yamaguchi, K.: Statistical Study of Spectral Parameters in Musical Instrument Tones. J. Acoust. Soc. of America, 94, 1, (1993) pp. 37–45.
3. Bazan, J.G., Nguyen, H.S., Nguyen, S.H, Synak, P., Wróblewski, J.: Rough Set Algorithms in Classification Problem. In: Polkowski, L., Tsumoto, S., Lin, T.Y. (eds), Rough Set Methods and Applications: New Developments in Knowledge Discovery in Information Systems. Physica-Verlag (2000) pp. 49–88.
4. Bazan, J.G., Szczuka, M.: RSES and RSESlib - A collection of tools for rough set computations. In: Ziarko, W. , Yao, Y.Y. (eds), Proc. of RSCTC'00, Banff, Canada (2000). See also: http://alfa.mimuw.edu.pl/~rses/.
5. Düntsch I., Gediga G., Nguyen H.S.: Rough set data analysis in the KDD process. In: Proc. of IPMU 2000, Madrid, Spain (2000) vol. 1, pp. 220–226.
6. Herrera, P., Amatriain, X., Batlle, E., Serra X.: Towards instrument segmentation for music content description: a critical review of instrument classification techniques. In: Proc. of ISMIR 2000, Plymouth, MA (2000).
7. Mannila, H., Toivonen, H.,Verkamo, A.I.: Discovery of frequent episodes in event sequences. Report C-1997-15, University of Helsinki, Finland (1997).
8. ISO/IEC JTC1/SC29/WG11: Overview of the MPEG-7 Standard. Doc. N4031.
9. Liu, H., Motoda, H. (eds): Feature extraction, construction and selection – a data mining perspective. Kluwer Academic Publishers, Dordrecht (1998).
10. Martin, K.D., Kim, Y.E.: 2pMU9. Musical instrument identification: A pattern-recognition approach. 136-th meeting of the Acoustical Soc. of America (1998).
11. Nguyen S.H.: Regularity Analysis And Its Applications In Data Mining. Ph.D. Dissertation, Warsaw University, Poland (2000).
12. Opolko, F., Wapnick, J.: MUMS – McGill University Master Samples. CD's (1987).
13. Pawlak, Z.: Rough sets – Theoretical aspects of reasoning about data. Kluwer Academic Publishers, Dordrecht (1991).
14. Pollard, H.F., Jansson, E.V.: A Tristimulus Method for the Specification of Musical Timbre. Acustica, Vol. 51 (1982) pp. 162–171.
15. Ślęzak, D., Wróblewski, J.: Classification algorithms based on linear combinations of features. In: Proc. of PKDD'99. Praga, Czech Republik, LNAI 1704, Springer, Heidelberg (1999) pp. 548–553.
16. Synak, P.: Temporal templates and analysis of time related data. In: Ziarko, W. , Yao, Y.Y. (eds), Proc. of RSCTC'00, Banff, Canada (2000).
17. Wieczorkowska, A.A.: The recognition efficiency of musical instrument sounds depending on parameterization and type of a classifier (in Polish), Ph.D. Dissertation, Technical University of Gdańsk, Poland (1999).
18. Wieczorkowska, A.A., Raś, Z.W.: Audio Content Description in Sound Databases. In: Zhong, N., Yao, Y., Liu, J., Ohsuga, S. (eds), Proc. of WI'01, Maebashi City, Japan, LNCS/LNAI 2198, Springer-Verlag (2001) pp. 175–183.
19. Wróblewski, J.: Analyzing relational databases using rough set based methods. In: Proc. of IPMU'00. Madrid, Spain (2000) 1, pp. 256–262.