# Chapter 18
# Time Variability-Based Hierarchic Recognition of Multiple Musical Instruments in Recordings

Elżbieta Kubera, Alicja A. Wieczorkowska, and Zbigniew W. Raś

**Abstract.** The research reported in this chapter is focused on automatic identification of musical instruments in polyphonic audio recordings. Random forests have been used as a classification tool, pre-trained as binary classifiers to indicate presence or absence of a target instrument. Feature set includes parameters describing frame-based properties of a sound. Moreover, in order to capture the patterns which emerge on the time scale, new temporal parameters are introduced to supply additional temporal information for the timbre recognition. In order to achieve higher estimation rate, we investigated a feature-driven hierarchical classification of musical instruments built using agglomerative clustering strategy. Experiments showed that the performance of classifiers based on this new classification of instruments schema is better than performance of the traditional flat classifiers, which directly estimate the instrument. Also, they outperform the classifiers based on the classical Hornbostel-Sachs schema.

**Keywords:** Music information retrieval, automatic indexing, timbre recognition, pitch tracking, Hornbostel-Sachs system, temporal data mining, random forest, agglomerative clustering.

Elżbieta Kubera
University of Life Sciences in Lublin, Akademicka 13, 20-950 Lublin, Poland
e-mail: elzbieta.kubera@up.lublin.pl

Alicja A. Wieczorkowska
Polish-Japanese Institute of Information Technology, Koszykowa 86, 02-008 Warsaw, Poland
e-mail: alicja@poljap.edu.pl

Zbigniew W. Raś
University of North Carolina, Dept. of Computer Science, Charlotte, NC 28223, USA &
Warsaw University of Technology, Institute of Computer Science, 00-665 Warsaw, Poland &
Polish Academy of Sciences, Institute of Computer Science, 01-237 Warsaw, Poland
e-mail: ras@uncc.edu

A. Skowron and Z. Suraj (Eds.): Rough Sets and Intelligent Systems, ISRL 43, pp. 347–363.
springerlink.com 

## 18.1 Introduction

In recent years, rapid advances in digital music creation, collection and storage technology have enabled various organizations to accumulate vast amounts of musical audio data. The booming of multimedia resources in the Internet brought a tremendous need to provide new, more advanced tools for querying and processing vast quantities of musical data. Many multimedia resources provide data which are manually labeled with some description information, such as title, author, company, and so on. However, in most cases those labels are insufficient for content-based searching. This problem attracted the attention of academia and industry, and initiated research in Music Information Retrieval (MIR) some years ago. As the outcome of this research, various MIR systems emerged, addressing diverse needs of the users of audio data, including audio identification (finding a title and a performer of a given excerpt, re-played or even hummed), identification of style or music genre, or audio alignment (*e.g.*, score following), etc.; examples of systems available at commercial web sites can be found at [15], [23], and systems being part of research are described in [16], [17], see also papers in [21], [22], and so forth.

Extraction of pitch, so-called pitch tracking, is performed in some of the MIR systems, and it is quite accurate in the case of melodies when only one sound is played at a time. Clearly, multi-pitch extraction (for chords) is more challenging and the problem of assigning each pitch to appropriate part of the score has to be tackled. Automatic assignment of notes to particular voices would be facilitated if instruments participating in each chord were automatically identified. The research presented in this chapter addresses automatic identification of instruments in polyphonic multi-instrumental recordings.

Timbre recognition is one of the subtasks in MIR, and it has proven to be extremely challenging especially in multi-timbre sounds, where multiple instruments are playing at the same time. Compared with this, automatic recognition of an instrument in the case of single sounds (no chords) is relatively easy, and it has been investigated, starting in the twentieth century, by many researchers. The obtained accuracy depends on the number of sounds and instruments taken into account, a feature set used, and a classifier applied, as well as the validation method utilized. Even 100% can be achieved for a small number of sounds/instruments classified with an artificial neural network, but usually is lower, and generally decreases with increasing number of instruments, even below 40% when the number of instruments approaches thirty and full range of each instrument is taken into account. We should also notice that audio data, represented as a long sequence of amplitude values (44100 samples per second per channel is a standard for CD), may vary significantly, depending on many factors, *e.g.*, recording conditions, playing method, the player and his or her particular instrument, etc. Therefore, audio data are usually parameterized before applying classifiers, and the extracted feature vector also strongly influences the obtained results. The feature set can be based on the time-domain representation describing the sound amplitude or the spectrum obtained from the sound analysis describing frequency contents derived from short audio frames and

we also believe that temporal changes of various sound features can be beneficial as the sound may undergo substantial changes in time (see Figure 18.1). Spectral features are most often extracted using Fourier transform but other analyses are applied as well, *e.g.*, wavelet transform yielding time-frequency representation.

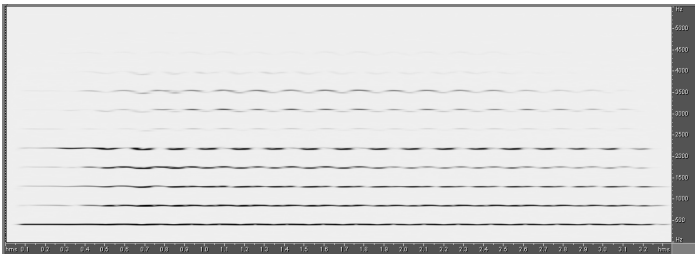

**Fig. 18.1** Spectrogram (sonogram) for A4 (440 Hz) sound of violin, played vibrato. The spectrogram shows temporal changes of the sound spectrum. Horizontal axis represents time, and vertical axis represents frequency. The darker the shade of gray, the higher the magnitude.

Feature sets vary depending on the researcher; there is no standard feature set. However, many low-level audio descriptors from the MPEG-7 standard of multimedia content description [8] are often used. Mel-Frequency Cepstral Coefficients (MFCC), originating from speech recognition, can also be applied for MIR purposes [4], including recognition of musical instruments [2]. In our research, we apply various short-time sound features describing properties of the sound in time domain and its spectrum; besides, we add temporal features to this basic set in order to capture time-variability of the sound features. Detailed description of the feature set used in this research is presented in Section 18.3.

As it was mentioned before, the accuracy of instrument identification also depends on the classifier. The algorithms applied in experiments on instrument recognition include k-nearest neighbors (k-NN), artificial neural networks (ANN), rough-set-based classifiers, support vector machines (SVM), Gaussian mixture models (GMM), decision trees and random forests, and so on. The review of the outcomes of this research is given in [6](see also [9]). Although the obtained accuracies are far from being perfect when the number of instruments to be recognized is big, simple algorithm as k-NN may still yield good results. However, algorithms successfully identifying instruments playing single and isolated sounds can be prone to errors when executed on continuous polyphonic data (multi-instrumental chords), as happens in recordings, even when tried on duets [14]. Identification of instruments in the case of chords is much more challenging, and more sophisticated algorithms are advised to be used. For instance, ANN yielded over 80% accuracy for several four-instrument sets [10]; GMM classifier yielded about 60% accuracy for duets from five-instrument set [3]; random forests produced about 75% accuracy on average [11] for 2–5 instruments from 10-instrument sets, with variable accuracy obtained for particular instruments. Since random forests are quite robust with

respect to noise [1], and already proved to be rather successful in the instrument identification task, we decided to apply this classification technique in the reported research.

### 18.1.1 Random Forests

A random forest (RF) is an ensemble of classification trees, constructed using procedure minimizing bias and correlations between individual trees. Each tree is built using different $N$-element bootstrap sample of the training $N$-element set. The elements of the sample are drawn with replacement from the original set, so roughly one-third of the training data is not used in the bootstrap sample for any given tree.

Let us assume that objects are described by a vector of $P$ attributes (features). At each stage of tree building, i.e., for each node of any particular tree in RF, $p$ attributes out of all $P$ attributes are randomly selected ($p \ll P$, often $p = \sqrt{P}$). The best split on these $p$ attributes is used to split the data in the node. It is determined as minimizing the Gini impurity criterion, which is a measure how often an element would be incorrectly labeled if labeled randomly, according to the distribution of labels in the subset.

Each tree is grown to the largest extent possible (without pruning). By repeating this randomized procedure $M$ times one obtains a collection of $M$ trees, which constitute a random forest. Classification of each object is made by simple voting of all trees [1].

### 18.1.2 Outline of the Paper

The experiments presented in this chapter concern identification of multiple instruments in polyphonic multi-instrumental recordings. Feature sets used here contain both frame-based audio parameters, as well as new parameters describing temporal variability of the frame-based features. The training audio data were taken from two repositories, commonly used in similar research worldwide. Testing data represent audio recordings of classical music, as we decided to focus our research on this music genre. The testing data were manually labeled in a careful way in order to create ground-truth data. Random forests have been applied as classifiers, also for hierarchical classification, including feature-driven hierarchy. The details of this research are presented in the next sections of our chapter; audio data are described in Section 18.2, features for sound parameterization are shown in Section 18.3, and the experiments are presented and discussed in Section 18.4. The chapter is summarized and concluded in Section 18.5.

## 18.2 Audio Data

Music we listen to can be played by numerous instruments; in various music genres, typical sets of instruments are usually used. For instance, electric guitars and drums etc. are commonly used in rock music; violins, violas etc. are commonly used in classical music; and so on. The music collections available worldwide are often labelled with these categories, so we can assume that this information is given. In the research presented in this chapter, we decided to focus on classical music, and therefore limit the set of investigated instruments to ones which are typical for this type of music. If someone would like to investigate a different music genre, the same methodology can be applied.

The audio data we decided to use in the experiments represent the following 10 instruments: B-flat clarinet, cello, double bass, flute, French horn, oboe, piano, tenor trombone, viola, and violin. Obviously, this set is not comprehensive and could be extended; still, it is sufficient for the purpose of illustrating the task we are dealing with, i.e., recognition of multiple instruments in polyphonic recordings.

Our experiments included training and testing of random forests. Therefore, we needed recordings for training RFs to be used to recognize selected instruments. We used single sounds played in various ways: *vibrato* (with vibration), *pizzicato* (plucking the strings), *forte* (loud), *piano* (soft), etc.; techniques of playing are called articulation. Also, we used all available pitches for every instrument.

The training data were taken from two commonly used repositories:

- MUMS [19]: all available articulation versions for our 10 instruments;
- IOWA [25]: *fortissimo* (very loud) for piano, and *mezzo forte* (medium loud) for other instruments;
    - cello, viola, and violin: *arco* (bowing) and *pizzicato*;
    - flute: *vibrato* and *non-vibrato* (no vibration);
    - French horn: *fortissimo* for notes within C3–B3 (MIDI notation used, i.e., A4=440 Hz) and *mezzo forte* for the remaining notes.

Some of the sounds were recorded *vibrato* (*e.g.*, strings – violin, viola, cello, and double bass from MUMS), and others with no vibration (strings in IOWA repository). Sounds of strings and tenor trombone were also chosen played muted and not muted. Flute is represented by vibrato and flutter sounds. Piano is represented by soft, plucked, and loud sounds. For each instrument, all articulation versions of sounds of this instrument represent the same class, i.e., the given instrument.

Testing data were taken from RWC Classical Music Database [5], so they were utterly different from the training data. Since we planned to evaluate temporal features, describing evolution of a sound in time (whether this would be a single sound, or a chord), we needed pieces with long sounds, i.e., long enough to observe time variability of these sounds in non-transitory parts. Such long-lasting sounds were manually selected from RWC Classical Music Database. We also wanted our test set to represent various composers and music styles. Therefore, the following pieces were used (number of test sounds selected for each piece is shown in parentheses):

- No. 4: P.I. Tchaikovsky, Symphony no. 6 in B minor, op. 74 'Pathétique', 4th movement (10 sounds);
- No. 9: R. Wagner, "Tristan und Isolde": Prelude and 'Liebestod' (9 sounds);
- No. 12: J.S. Bach, "The Musical Offering", BWV. 1079, 'Ricercare à 6' (14 sounds);
- No. 16: W.A. Mozart, Clarinet Quintet in A major, K. 581, 1st movement (15 sounds);
- No. 18: J. Brahms, Horn Trio in E♭ major, op. 40, 2nd movement (4 sounds).

Test sounds represent homogenous chords (i.e., the instruments playing and the notes played remain constant throughout the whole sound), played by two to five instruments. These sounds were manually selected in a careful way and then labelled, thus creating ground-truth data for further experiments.

Both training and testing data were recorded with 44.1 kHz sampling rate and 16-bit resolution. If the audio data were recorded stereo, then the left channel was arbitrarily chosen for processing. Also, as a preprocessing step, the silence before and after each isolated sound was removed. To do this, a smoothed version of amplitude was calculated starting from the beginning of the file, as moving average of 5 consequent amplitude values, and when this value increased by more than a threshold (experimentally set to 0.0001), this point was considered to be the end of the initial silence. Similarly, the ending silence was removed.

### *18.2.1 Hornbostel-Sachs System of Musical Instrument Classification*

Instruments we investigate in the reported research represent various families of instruments, according to Hornbostel-Sachs system of musical instrument classification [7], which is the most commonly used system describing the taxonomy of instruments. This system classifies instruments of classical music into the following groups: aerophones (wind instruments), chordophones (stringed instruments), membranophones (mostly drums), and idiophones (basically, other percussive instruments, where a solid is a source of vibration). Since Hornbostel-Sachs system provides a hierarchical classification of musical instruments, these categories are further subdivided into subcategories. According to Hornbostel-Sachs system, the investigated instruments are classified as follows:

- aerophones
  - flutes
    - ⋆ (transverse) flute,
  - reed instruments
    - ⋆ single reed: B-flat clarinet,
    - ⋆ double reed: oboe,

    - brass
        - ⋆ French horn,
        - ⋆ tenor trombone,

- chordophones
    - bowed: cello, double bass, viola, and violin; these instruments can be played *pizzicato* (and this articulation was also investigated), but bowing is a primary articulation here, this is why these instruments are classified as bowed;
    - piano.

We decided to investigate sounds of definite pitch, with harmonic spectra, as we planned to monitor harmonic structure of the spectrum, among other sound features. Therefore, percussive instruments (membranophones and idiophones) are not investigated here.

The timbre of a sound may also differ depending on articulation. However, our goal was to identify musical instruments without taking this property into account. Therefore, all sounds of each particular instrument represented the same class, i.e., this instrument, and no classification according to articulation was investigated in the reported research.

## 18.3 Feature Set

Our feature set consists of the main, basic set of features, calculated for a 40-ms Hamming-windowed frame of the analyzed sound, which is then used twofold: to calculate average values, constituting the main representation of this sound, and to observe temporal behavior of the analyzed sound. To start with, average values of the main features are calculated for a sliding analysis frame with 10 ms hop size. In order to make sure that long-term behavior is captured, 430 ms are taken for this calculation. This may not cover the entire sound, but it is sufficient to cover the onset and a good portion of the steady state, which are usually sufficient to recognize an instrument by human listeners, so we also follow this philosophy. Next, we calculate *Fits* – this proposed feature represents the type of the function which best describes the temporal behavior of the main feature set; consecutive (and overlapping) parts of the sound can be described by different functions. Finally, we calculate *Peaks*; this multidimensional feature describes relationships between three greatest temporal local maxima, representing time variability of the given feature throughout the entire sound. The obtained temporal features are then added to the feature set. The details of calculations of the above-mentioned features are described below.

The basic feature set consists of the following parameters:

- *SpectralCentroid* of the spectrum obtained through the discrete Fourier transform (DFT), calculated as Fast Fourier Transform (FFT). In this case, the frame length must equal to the power of 2. Since 40 ms equals to 1764 audio

samples in the case of 44.1 kHz sampling rate, this frame is zero-padded to 2048 samples, and next *SpectralCentroid* $C_i$ is calculated as follows:

$$C_i = \frac{\sum_{k=1}^{N/2} f(k)\,|X_i(k)|}{\sum_{k=1}^{N/2} |X_i(k)|} \tag{18.1}$$

where: $N$ - number of available elements of the (symmetrical) discrete spectrum, i.e., frame length, so $N = 2048$;
$X_i(k)$ - $k^{th}$ element of FFT for $i^{th}$ frame;
$f(k)$ - frequency corresponding to $k^{th}$ element of the spectrum;

- *SpectralSpread* $S_i$ - a deviation of the power spectrum with respect to Spectral Centroid $C_i$ in a frame, calculated as

$$S_i = \sqrt{\frac{\sum_{k=1}^{N/2} (f(k) - C_i)^2\,|X_i(k)|}{\sum_{k=1}^{N/2} |X_i(k)|}} \tag{18.2}$$

- *AudioSpectrumFlatness*, $Flat_1, \ldots, Flat_{25}$ - multidimensional parameter describing the flatness property of the power spectrum within a frequency bin for selected bins; 25 out of 32 frequency bands were used for a given frame, starting from 250 Hz, as recommended in MPEG-7. This feature is calculated as follows:

$$Flat_b = \frac{\sqrt[hi(b)-lo(b)+1]{\prod_{k=lo(b)}^{hi(b)} P_g(k)}}{\frac{1}{hi(k)-lo(k)+1} \sum_{k=lo(b)}^{hi(b)} P_g(k)} \tag{18.3}$$

where: $b$ - band number, $1 \leq b \leq 25$,
$lo(b)$ and $hi(b)$ - lower and upper limits of the band $b$, respectively,
$P_g(k)$ - grouped coefficients of the power spectrum within the band $b$; grouping speeds up the calculations;

- *RollOff* - the frequency below which an experimentally chosen percentage of the accumulated magnitudes of the spectrum is concentrated (equal to 85%, which is the most often used setting). *RollOff* is a measure of spectral shape, used in speech recognition to distinguish between voiced and unvoiced speech;
- $Flux$ - sum of squared differences between the magnitudes of the FFT points in a given frame and its preceding frame. This value is usually very small, and it was multiplied by $10^7$ in our research. For the starting frame, $Flux = 0$ by definition;
- *Energy* - energy (in logarithmic scale) of the spectrum of the parameterized sound;
- *MFCC* - multidimensional feature, consisting of 13 Mel frequency cepstral coefficients. The cepstrum was calculated as a logarithm of the magnitude of the spectral coefficients and then transformed to the mel scale. Mel scale is used instead of the Hz scale, in order to better reflect properties of the human perception of frequency. Twenty-four mel filters were applied, and the obtained results

were transformed to twelve coefficients. The thirteenth coefficient is the 0-order coefficient of MFCC, corresponding to the logarithm of the energy [12], [18];
- *ZeroCrossingRate*; zero-crossing is a point where the sign of time-domain representation of the sound wave changes;
- *FundamentalFrequency* - pitch; maximum likelihood algorithm was applied for pitch estimation [26];
- *HarmonicSpectralCentroid*, *HSC* - mean of the harmonic peaks of the spectrum, weighted by the amplitude in linear scale [8];
- *HarmonicSpectralSpread*, *HSS* - represents the standard deviation of the harmonic peaks of the spectrum with respect to *HarmonicSpectralCentroid*, weighted by the amplitude [8];
- *HarmonicSpectralVariation*, *HSV* - normalized correlation between amplitudes of harmonic peaks of each two adjacent frames, calculated as:

$$HSV = 1 - \frac{\sum_{n=1}^{N} A_n(i-1) \cdot A_n(i)}{\sqrt{\sum_{n=1}^{N} A_n^2(i-1)} \cdot \sqrt{\sum_{n=1}^{N} A_n^2(i)}}$$

where $A_n(i)$ - amplitude of $n^{th}$ harmonic partial in $i^{th}$ frame [8]. For the starting frame, $HSV = 1$ by definition.
- *HarmonicSpectralDeviation*, *HSD*, calculated as:

$$HSD = \frac{\sum_{n=1}^{N} |\log(A_n) - \log(SE_n)|}{\sum_{n=1}^{N} \log(A_n)}$$

where $SE_n$ - $n^{th}$ component from a spectral envelope,
$A_n$ - amplitude of $n^{th}$ harmonic partial.
This feature represents the spectral deviation of the log amplitude components from a global spectral envelope, where the global spectral envelope of the $n^{th}$ harmonic partial is calculated as the average value of the neighboring harmonic partials: no. $n-1$, $n$, and $n+1$, calculated as [8]:

$$SE_n = \frac{\sum_{i=-1}^{1} A_{n+i}}{3}$$

- $r_1, \ldots, r_{11}$ - various ratios of harmonic partials in spectrum: $r_1$ – energy of the fundamental to the total energy of all harmonics, $r_2$: amplitude difference [dB] between $1^{st}$ and $2^{nd}$ partial, $r_3$: ratio of the sum of partials 3-4 to all harmonics, $r_4$: partials 5-7 to all, $r_5$: partials 8-10 to all, $r_6$: remaining partials to all, $r_7$: brightness – gravity center of spectrum, $r_8$, $r_9$: contents of even/odd harmonics in the spectrum, respectively.

For these basic features, we calculated:

- *Averages* - vector representing averaged (through 430 ms) values for all features; this is our basic feature set;

- *Fits* - type of function (from 7 predefined function types) which best describes the manner of feature values' variation in time. Analysis was performed in 4 parts of the sound, each described by 10 consecutive 40 ms frames 75% overlapped (altogether 280 ms); each of these 4 parts can be assigned to any of these 7 function types. Hop size between parts was equal to 5 frames. Predefined function types were as follows: linear, quadratic, logarithmic, power, hyperbolic, exponential, and sinusoidal with linear trend. Original feature vector was treated as a function of time. Functions of each predefined type were fitted into each feature function within a given part of the sound. Linear and quadratic functions were fitted using the method of least squares. In other cases, linearization was performed before applying the least squares method. $R^2$ value was calculated for each fit, where $R$ is a Pearson's correlation coefficient. A function with the highest $R^2$ value was supposed to fit the data best. If the highest $R^2$ was lower than 0.8, then it was assumed that none of proposed functions fits data well, and "no fit" was assigned as a feature value;
- *Peaks* (new temporal features) - distances and proportions between maximal peaks in temporal evolution of feature values throughout the entire sound, defined as follows. Let us name original feature vector as $p$ and treat $p$ as a function of time. We searched for 3 maximal peaks of this function. Maximum $M_i(p)$, $i = 1,2,3$, was described by $k$ - the consecutive number of the frame where the extremum appeared, and the value of feature $p$ in the frame $k$:

  $$M_i(p) = (k_i, p[k_i]) \quad k_1 < k_2 < k_3.$$

  The temporal variation of each feature can be then represented as a vector $T = [T_1, \ldots, T_6]$ of temporal parameters, built as follows:

  $$T_1 = k_2 - k_1\ ,\ T_2 = k_3 - k_2\ ,\ T_3 = k_3 - k_1\ ,$$
  $$T_4 = p[k_2]/p[k_1]\ ,\ T_5 = p[k_3]/p[k_2]\ ,\ T_6 = p[k_3]/p[k_1]\ .$$

  These parameters reflect relative positions and changes of values representing maximal peaks in the temporal evolution of each feature [11].

## 18.4 Experiments and Results

The purpose of this chapter was to investigate automatic identification of musical instruments in polyphonic recordings, and to verify if new temporal features can be helpful to better recognize instruments in recordings. Another aim was to check if hierarchical classifiers yield better results than non-hierarchical ones.

### 18.4.1 Training and Testing of Random Forests

Training of the battery of RFs was performed on single isolated sounds of musical instruments, taken from IOWA and MUMS repositories, and on sound mixes of up to three instruments. This way we created a set of multi-instrumental audio samples, in order to train RF to identify the target instrument, even when accompanied by another instrument or instruments. Instrumental sounds added in mixes were randomly chosen in such a way that the obtained sounds constitute unisons or chords (major or minor), and the distribution of instruments in the obtained set of mixes reflects the distribution of instruments playing together in RWC Classical Music Database. One-label training of binary RFs was performed on these data, aiming at identification of a target instrument, i.e., whether it is playing in a sound, or not.

Tests of the obtained battery of RFs were performed on RWC Classical Music data. Predictions were based on the results obtained for all forests (for all instruments). Polytimbral music samples should produce multiple labels. To obtain such multi-label predictions from our classification system, we derived them in a following way. For each binary classifier we got a percentage of votes of trees in the forest on “yes" class (presence of an instrument corresponding to a given classifier), and this percentage was treated as the rate of each corresponding label (instrument name). Labels were sorted in decreasing order with respect to the corresponding rates. If the first label on a list had the rate exceeding 80% and next label had the rate below 20%, then we assumed that this sound was recognized as monotimbral and prediction contained only one label – an instrument name of the highest rate. Otherwise, the differences of rates of consecutive labels in the list were calculated, and the prediction list of labels was truncated where the highest difference was found.

In the case of hierarchical classification, binary RFs were similarly trained to recognize groups of instruments in a given node.

In this case predictions were obtained in a similar way, but rates for labels in leaves of a tree were calculated by multiplying rates from all nodes in a path from the root to a given leaf.

In this work we used the RF implementation from the *R* package *randomForest* [13], [20].

### 18.4.2 Feature-Driven Hierarchic Classifications of Musical Instruments

In our experiments, we aimed at identifying instruments playing in a given snippet of an audio recording, using several strategies of classification. To start with, we performed non-hierarchical classification using a battery of binary RFs, where each RF was trained to indicate whether a target instrument was playing in the investigated audio snippet or not. These classification results are shown in Table 18.1, together

**Table 18.1** Results of the recognition of musical instruments in RWC Classical Music Database, for the basic feature set

| Classification system | Precision | Recall | F-measure |
|---|---|---|---|
| Non-hierarchical | 71.63% | 58.43% | 64.36% |
| Hierarchical (Hornbostel-Sachs) | 70.74% | 60.06% | 64.97% |

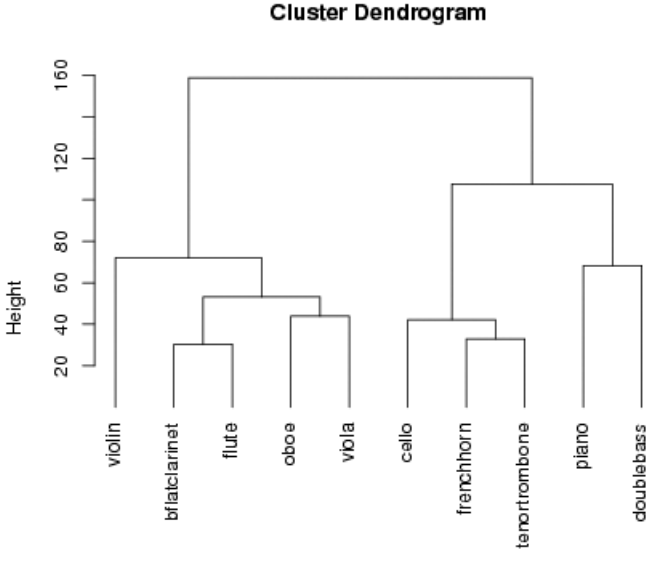


**Fig. 18.2** Cluster dendrogram for *Averages*.

with the results obtained for hierarchical classification based on Hornbostel-Sachs taxonomy of musical instruments, for the basic feature set, i.e., *Averages*.

Apart from Hornbostel-Sachs hierarchical classification, feature-driven hierarchical classification of musical instruments in recordings was performed. Hierarchies were obtained through clustering.

Hierarchical clustering was performed by means of Ward's method, appropriate for quantitative variable as ours [24]. This method uses an analysis of variance approach to evaluate the distances between clusters. Ward's method attempts to minimize the sum of squares of any two hypothetical clusters that can be formed at each step. It finds compact, spherical clusters, although it tends to create clusters of small size. This method implements an agglomerative clustering algorithm, starting at the leaves, regarded as $n$ clusters of size 1. It looks for groups of leaves, forms them into branches, and continues to the root of the resulting dendrogram. Distances between clusters were calculated using Manhattan distance, as it performed best in the conducted experiments.

Hierarchical clustering of instrument sounds was performed using *R*, an environment for statistical computing [20]. The clustering based on feature vectors rep-

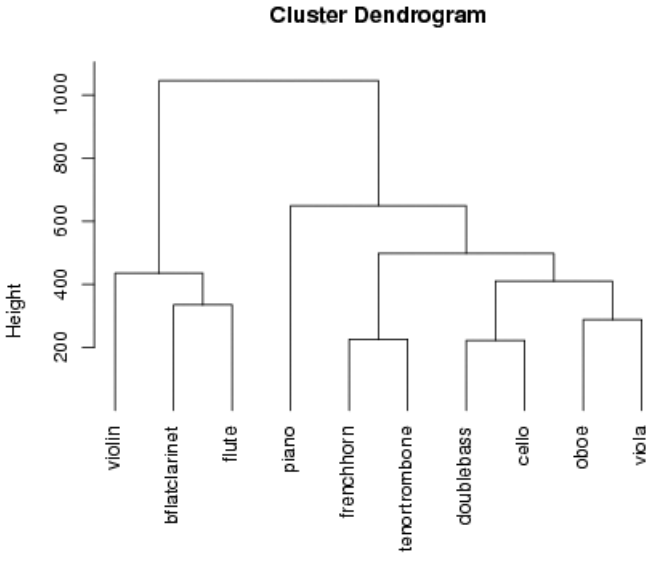


**Fig. 18.3** Cluster dendrogram for *Averages* + *Peaks*.

resenting only average values of our basic features (*Averages*), and with addition of temporal observations of these features (*Fits* and *Peaks*) are shown in Figures 18.2, 18.4, and 18.3, respectively. Each dendrogram was built on the basis of single instrumental sounds only, without mixes, thus no foreign sounds distorted representation of each target instrument. Every instrument was represented by one artificial object, calculated as averaged value of all objects, i.e., parameterized sounds of this instrument.

As we can see, the taxonomies of musical instruments obtained through clustering shown in Figures 18.2, 18.4, and 18.3, differ significantly from classic Hornbostel-Sachs system, in all cases of the feature-driven hierarchical trees.

The results obtained for hierarchical classification in various settings of hierarchies are given in Table 18.2. As we can see, precision is almost constant, around 70-72%, so it is practically independent of the hierarchy. However, the obtained recall changes significantly. For each feature set, the recall improves when feature-driven hierarchy is used as a classification basis. The best overall results (reflected in F-measure) are obtained for feature-driven classification, and for *Fits* added to the feature set. The trade-off between precision and recall can be observed in some cases, but it is rather small. In general, adding temporal features improves the obtained results, comparing to the results obtained for *Averages*; adding *Peaks* improves accuracy, and adding *Fits* improves recall.

One can be interested in seeing the details of misclassification. Since we have multiple instruments labeling both the input and output data, a regular confusion matrix cannot be produced, since we cannot show which instrument was mistaken

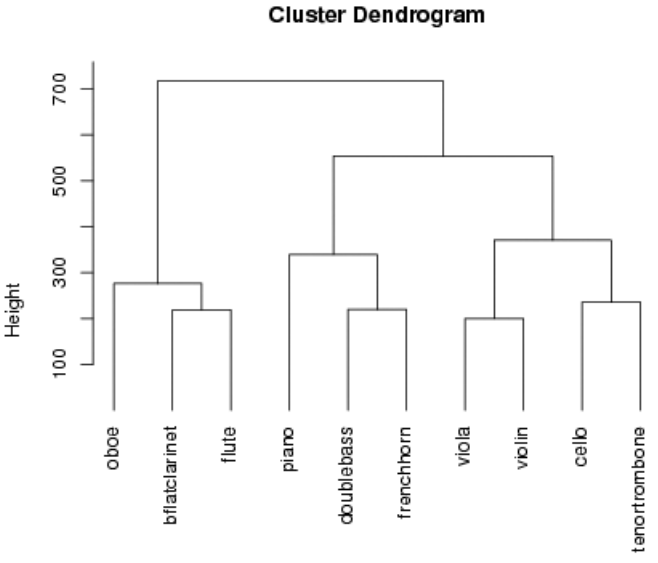


**Fig. 18.4** Cluster dendrogram for *Averages + Fits*.

**Table 18.2** Results of the recognition of musical instruments in RWC Classical Music Database for different feature sets and hierarchic classification systems

| Instruments hierarchy | Feature set | Precision | Recall | F-measure |
|---|---|---|---|---|
| Hornbostel-Sachs | Averages | 70.74% | 60.06% | 64.97% |
| Feature-driven | Averages | 70.24% | 65.74% | 67.91% |
| Hornbostel-Sachs | Avg+Peaks | 72.67% | 60.42% | 65.98% |
| Feature-driven | Avg+Peaks | 72.35% | 62.47% | 67.04% |
| Hornbostel-Sachs | Avg+Fits | 70.91% | 64.74% | 67.69% |
| Feature-driven | Avg+Fits | 71.88% | 70.67% | 71.27% |

for which one. Still, in order to illustrate the details of RF-based classification, exemplary results are presented in Figures 18.5 and 18.6, showing what types of classification errors we encountered.

Let us analyze the graphs presented in Figure 18.5. In the 1st graph, violin and cello were identified correctly, but double bass and viola were additionally indicated by the battery of RFs classifiers. Since double bass sound is similar to cello, and viola sound is similar to violin, it is not surprising that the corresponding RFs fired. In the case of the 2nd graph, the errors are more serious, since the violin and viola duo, although indicated correctly, was also accompanied by additional indication of cello and double bass. Even though cello and viola are relatively closely related instruments, the indication of double bass is considered to be a serious error here.

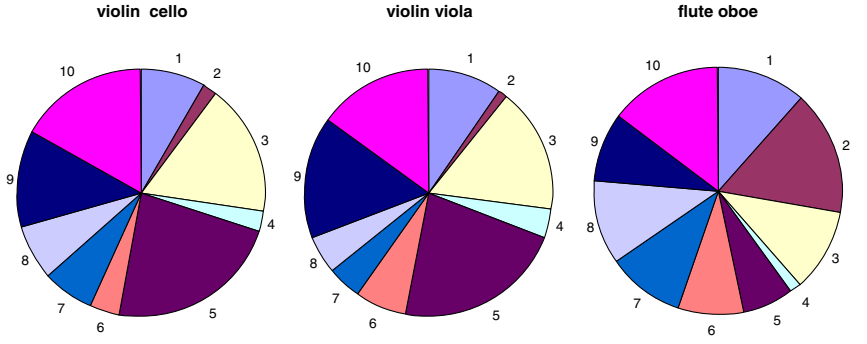


**Fig. 18.5** Exemplary results of RF-based recognition of duo sounds. The numbers correspond to the instruments in the following order: 1. piano, 2. oboe, 3. cello, 4. trombone, 5. double bass, 6. French horn, 7. clarinet, 8. flute, 9. viola, 10. violin. The values shown represent outputs for each RF representing the given instrument

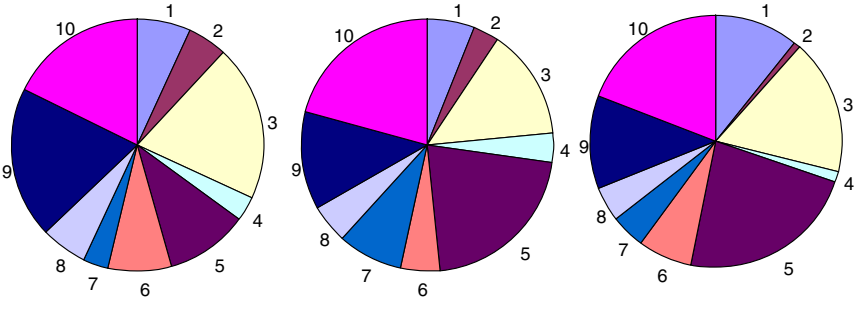

**Fig. 18.6** Exemplary results of RF-based recognition of instruments in polyphonic recordings. Each input sound represented a chord played by violin, viola, and cello.

In the case of the 3rd diagram, oboe and flute were recognized correctly, but additionally violin (higher rate than flute), piano, cello, clarinet, viola, French horn and double bass were listed by our battery of RFs. This indicates that by adjusting the way of outputting the recognition list we may improve precision, but most probably at the expense of lower recall. Since recall is generally lower than precision in this research, we believe that cutting of more instruments listed by the RFs classifiers can deteriorate the overall results.

The graphs presented in Figure 18.6 show the results for three sounds, all representing violin, viola, and cello playing together. The 1st diagram shows correct identification of these three instruments, without errors. In the case of the other two diagrams, besides of recognizing the target instruments, our battery of RFs classifiers additionally indicated double bass. Again, double bass is similar to cello, so it is not considered to be a serious mistake.

## 18.5 Summary and Conclusions

In this chapter, we presented automatic hierarchical identification of musical instruments in recordings. The Sachs-Hornbostel classification is the most common hierarchic classification of musical instruments, but feature-driven classification yields better results in automatic recognition of instruments in recordings. The audio data are described here by means of various sound features, automatically calculated for short audio frames. These features are then used to calculate the main feature vector (*Averages*), as well as two additional feature types, *Peaks* and *Fits*, describing temporal changes of the basic features. Automatic recognition of instruments in polyphonic recordings was performed using Random Forests, for ten instruments commonly found in classical music pieces. Training of RFs classifiers was based on 2 repositories of instrumental sounds. Single sounds and sound mixes were used in this training; probability of adding an instrument to the training mix reflected the distribution of instruments playing together in classical music recordings, taken from RWC Classical Music Database.

Our experiments showed that hierarchical classification yields better results than non-hierarchical one. Feature-driven hierarchic classification always improves recall, which tends to be lower than precision (since identification of all instruments in a chord is difficult even for a human), so the increase of recall is valuable, and we consider it to be a success. Also, we observed that adding *Peaks* improves accuracy of instrument recognition, and adding the proposed feature *Fits* improves recall. We plan to continue experiments, with an extended feature vector, including both *Peaks* and *Fits* added to *Averages*. We also plan to add more detailed temporal features, and conduct experiments for more instruments.

**Acknowledgments.** This project was partially supported by the Research Center of PJIIT (supported by the Polish National Committee for Scientific Research (KBN)) and also by the National Science Foundation under Grant Number IIS-0968647. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

## References

1. Breiman, L.: Random Forests. Machine Learning 45, 5–32 (2001)
2. Brown, J.C.: Computer identification of musical instruments using pattern recognition with cepstral coefficients as features. J. Acoust. Soc. Am. 105, 1933–1941 (1999)
3. Eggink, J., Brown, G.J.: Application of missing feature theory to the recognition of musical instruments in polyphonic audio. In: ISMIR 2003 (2003)
4. Foote, J.: An Overview of Audio Information Retrieval. Multimedia Systems 7, 2–11 (1999)
5. Goto, M., Hashiguchi, H., Nishimura, T., Oka, R.: RWC Music Database: Popular, Classical, and Jazz Music Databases. In: Proceedings of the 3rd International Conference on Music Information Retrieval, pp. 287–288 (2002)

6. Herrera-Boyer, P., Klapuri, A., Davy, M.: Automatic Classification of Pitched Musical Instrument Sounds. In: Klapuri, A., Davy, M. (eds.) Signal Processing Methods for Music Transcription. Springer Science & Business Media, LLC (2006)
7. Hornbostel, E.M., von Sachs, C.: Systematik der Musikinstrumente. Zeitschrift für Ethnologie 46, 553–590 (1914)
8. ISO MPEG-7 Overview, http://www.chiariglione.org/mpeg/
9. Klapuri, A., Davy, M. (eds.): Signal Processing Methods for Music Transcription. Springer, New York (2006)
10. Kostek, B.: Musical instrument classification and duet analysis employing music information retrieval techniques. Proc. IEEE 92(4), 712–729 (2004)
11. Kubera, E., Wieczorkowska, A., Raś, Z., Skrzypiec, M.: Recognition of Instrument Timbres in Real Polytimbral Audio Recordings. In: Balcázar, J.L., Bonchi, F., Gionis, A., Sebag, M. (eds.) ECML PKDD 2010. LNCS (LNAI), vol. 6322, pp. 97–110. Springer, Heidelberg (2010)
12. Kubera, E.: The role of temporal attributes in identifying instruments in polytimbral music recordings (in Polish). Ph.D. Dissertation, Polish-Japanese Institute of Information Technology (2010)
13. Liaw, A., Wiener, M.: Classification and regression by random Forest. R News 2(3), 18–22 (2002)
14. Livshin, A.A., Rodet, X.: Musical Instrument Identification in Continuous Recordings. In: Proc. of the 7th Int. Conference on Digital Audio Effects (DAFX 2004), Naples, Italy (2004)
15. MIDOMI, http://www.midomi.com/
16. Mierswa, I., Morik, K., Wurst, M.: Collaborative Use of Features in a Distributed System for the Organization of Music Collections. In: Shen, J., Shephard, J., Cui, B., Liu, L. (eds.) Intelligent Music Information Systems: Tools and Methodologies, pp. 147–176. IGI Global (2008)
17. Müller, M.: Information retrieval for music and motion. Springer, Heidelberg (2007)
18. Niewiadomy, D., Pelikant, A.: Implementation of MFCC vector generation in classification context. J. Applied Computer Science 16, 55–65 (2008)
19. Opolko, F., Wapnick, J.: MUMS – McGill University Master Samples. CD's (1987)
20. R Development Core Team A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria, http://www.R-project.org
21. Raś, Z.W., Wieczorkowska, A.A. (eds.): Advances in Music Information Retrieval. SCI, vol. 274. Springer, Heidelberg (2010)
22. Shen, J., Shepherd, J., Cui, B., Liu, L. (eds.): Intelligent Music Information Systems: Tools and Methodologies. Information Science Reference, Hershey (2008)
23. Sony Ericsson TrackID, http://www.sonyericsson.com/trackid
24. The Pennsylvania State University Cluster Analysis - Ward's Method, http://www.stat.psu.edu/online/courses/stat505/18_cluster/09_cluster_wards.html
25. The University of IOWA Electronic Music Studios Musical Instrument Samples, http://theremin.music.uiowa.edu/MIS.html
26. Zhang, X., Marasek, K., Raś, Z.W.: Maximum Likelihood Study for Sound Pattern Separation and Recognition. In: 2007 International Conference on Multimedia and Ubiquitous Engineering, MUE 2007, pp. 807–812. IEEE (2007)