Musical Instrument Classification Using Individual Partials

Jayme Garcia Arnal Barbedo and George Tzanetakis, *Member, IEEE*

***Abstract*—In a musical signals, the spectral and temporal contents of instruments often overlap. If the number of channels is at least the same as the number of instruments, it is possible to apply statistical tools to highlight the characteristics of each instrument, making their identification possible. However, in the underdetermined case, in which there are fewer channels than sources, the task becomes challenging. One possible way to solve this problem is to seek for regions in the time and/or frequency domains in which the content of a given instrument appears isolated. The strategy presented in this paper explores the spectral disjointness among instruments by identifying isolated partials, from which a number of features are extracted. The information contained in those features, in turn, is used to infer which instrument is more likely to have generated that partial. Hence, the only condition for the method to work is that at least one isolated partial exists for each instrument somewhere in the signal. If several isolated partials are available, the results are summarized into a single, more accurate classification. Experimental results using 25 instruments demonstrate the good discrimination capabilities of the method.**

***Index Terms*—Feature extraction, partialwise instrument classification, spectral disjointness, underdetermined mixtures.**

I. Introduction

THE identification of the instruments that compose a musical signal has received increasing attention in the last years. Such an interest is fed by the potential benefits that an accurate instrument classifier can bring to other digital audio applications. In particular, musical genre classification can be greatly improved if the instruments present in a given song are known, since this information can be used to narrow down the set of potential musical genres. Sound source separation algorithms can also explore such information, particularly if they deal with underdetermined signals. In this case, the knowledge about the instruments can be used to create instrument-specific rules to improve the quality of the sound source separation.

Early work in the area was mainly devoted to the identification of instruments in monophonic signals. This problem is, in general, less challenging than the polyphonic case, since the instrument to be classified is isolated from the interference of any other sound source. Most of those proposals deal with general instruments [1]–[11], while a few others deal with specific cases, like classification of woodwinds [12], [13] and discrimination between piano and guitar [14].

In the last years, a number of strategies capable of dealing with polyphonic musical signals have been proposed. Most of them have some important limitations.

- Limited number of instruments: some of the methods proposed in the literature only work and/or were only tested for a small (six or less) set of instruments (e.g., [15]–[22]).
- Low accuracy: in some cases the accuracy is below 50% even considering few instruments (e.g., [19], [23]).
- Instrument combinations set *a priori*: in this case, the strategies try to classify the signals according to predefined combinations of instruments; hence, they fail if the mixture has a combination of instruments that was not considered in the training (e.g., [24], [25]).
- Polyphony limited to duets: some strategies can only deal with two simultaneous instruments (e.g., [26], [27]).

Thus, despite the clear advancements achieved in the last years, there are still many limitations that prevent instrument identification tools to be more widely used. This paper presents a simple and reliable strategy to identify instruments in polyphonic musical signals that overcomes some of the main limitations faced by its predecessors. The identification uses a majority decision based upon pairwise comparisons of instrument likelihoods. A related but more complex approach was used by Essid *et al.* [5] to classify solo musical phrases. The method presented here is basically a system in which majority rules are successively applied, as briefly described in the following.

In real musical signals, simultaneous sources (instruments and vocals) normally have a high degree of correlation and overlap both in time and frequency, as a result of the underlying rules normally followed by western music (e.g., notes with integer ratios of pitch intervals). This can make the identification of instruments challenging. However, one can expect to find at least some unaffected partials throughout the signal, which can be explored to provide cues about the corresponding instrument. As a result of such an observation, the proposed algorithm extracts features individually for each partial that does not collide with any other partial (isolated partials). Each pair of instruments is characterized by a particular set of nine features, selected from a complete set of 34 features. Each partial is identified with one of the pair of instruments using a linear classifier. If such a feature is greater than a given threshold, it represents a certain instrument; otherwise, it represents the other one. A first majority rule is applied by summarizing the results of the nine features; as a result, each pair of instruments

Manuscript received September 04, 2009; revised December 09, 2009. Date of publication March 11, 2010; date of current version October 01, 2010. This work was supported by Foreign Affairs and International Trade Canada under a Post-Doctoral Research Fellowship Program (PDRF). The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Dan Ellis.

J. G. A. Barbedo was with the Department of Computer Science, University of Victoria, Victoria, BC V8W 3P6 , Canada . He is now with the Department of Communications, FEEC, UNICAMP C.P. 6101, CEP: 13.083-852, Campinas, SP, Brazil (e-mail: jgab@decom.fee.unicamp.br).

G. Tzanetakis is with the Department of Computer Science, University of Victoria, Victoria, BC V8W 3P6 , Canada (e-mail: gtzan@cs.uvic.ca).

Digital Object Identifier 10.1109/TASL.2010.2045186

will have a winner associated to that particular partial. The instrument that is represented more times in the previous set of winners is chosen as the label for the partial. The results are then summarized along all isolated partials related to a given note, which receives the label of the winner instrument. The summarization of the results along the entire signal is also possible because, as in a musical signal an instrument is likely to play several notes, there will be several identification instances that may provide stronger evidence of the presence of a given instrument. In this paper, an instrument is considered to be part of a musical signal if it appears in more than 5% of the duration of the entire signal. Such a value is arbitrary, as this is a parameter that can be freely set according to the intended purpose.

As a result of the above-mentioned characteristics, the strategy is able to deal with polyphonies of any order, provided that at least one partial of each instrument does not suffer interference from other instruments.

The method was tested with 25 different instruments of various types. Percussion instruments were not included because the method is currently unable to deal with nonharmonic sounds. Tests were also performed using actual recordings to determine how the method performs under real-world conditions. Experimental results reveal that the method's performance is comparable to the state-of-the-art of instrument recognition.

The paper is organized as follows. Section II presents the data preprocessing. Section III describes all steps of the algorithm. Section IV presents the experiments and corresponding results. Finally, Section V presents the conclusions and final remarks.

II. Preprocessing

The preprocessing steps described in the following are fairly standard and have been shown to be adequate for supporting the algorithm.

A. Adaptive Frame Division

In the first step, the algorithm divides the signal into frames. The best procedure here is to set the boundaries of each frame at the points where an onset [28], [29] (new note, instrument, or vocal) occurs, so the longest homogeneous frames are considered. Although the system includes the onset detector proposed by Zhou *et al.* [30], the main tests presented in Section IV were performed assuming the onsets to be known. This was done because, to accurately measure the specific performance of the novel part of the proposed instrument identification method, it was necessary to guarantee that all errors are exclusively due to it. However, since it is also important to know how the complete system performs, a study about the effects of onset misplacements on the accuracy of the algorithm is presented in Section IV-B. Additionally, the tests with real recordings presented in Section IV-F were performed using the complete system, which includes the onset detector.

B. Identification of Isolated Partials

As commented before, one of the first steps of the algorithm is to extract features from isolated partials—the ones that do not collide with any other. The identification of those partials relies on two parameters: the number of simultaneous sound sources in the frame, and the respective fundamental frequencies. In the proposed system, those parameters are estimated using the strategies proposed in [31] and [32], respectively. The observations made in Section II-A also apply here: the main tests were performed assuming those parameters to be known in order to avoid cascaded errors. A more complete picture is provided by a study about the impact of fundamental frequency (F0) misestimates (see Section IV-C), and by the tests with real recordings.

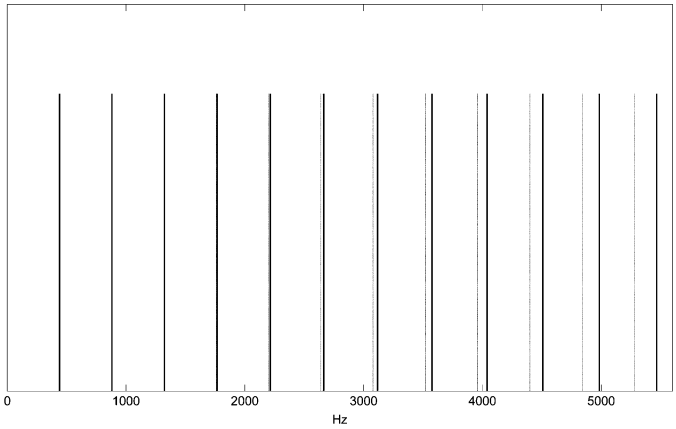

Fig. 1. Example of inharmonicity.

The positions of the partials corresponding to each source are then determined. The position of a partial, in the context of this work, refers to the central frequency of the band expected to be dominated by that partial. Simply taking multiples of F0 sometimes works, but the inherent inharmonicity [33], [34] of some instruments may cause this approach to fail, especially if one needs to take several partials into consideration. Fig. 1 shows an example of the effects of inharmonicity, where the light lines represent the positions of the partials if there was no inharmonicity, and the dark lines represent the actual positions. As can be seen, the values begin to differ considerably as higher harmonics are considered. To make the estimation of each partial frequency more accurate, the following procedure was adopted for each F0.

1) The discrete Fourier transform (DFT) is calculated for the frame under analysis, from which the magnitude spectrum M is extracted. The DFT has the same length as the frame, a rectangular window is applied, and there is no overlap between the frames.
2) The ideal position of the current partial (p_n) is made equal to $p_{n-1} + F0$, with $p_0 = 0$.
3) The adjusted position of the current partial $(\hat{p}_n)$ is given by the highest peak in the interval $[p_n - s_w, p_n + s_w]$ of M, where $s_w = 0.1 \cdot F0$ is the search width. This search width contains the correct position of the partial in nearly 100% of the cases; a broader search region was avoided in order to reduce the chance of interference from other sources.

A partial is used in the next steps of the algorithm if there is no other partial whose position is within the interval $[p_n - s_w, p_n + s_w]$, and if it has at least 1% of the energy of the most energetic partial. In some cases, no partial satisfies those conditions, in which case the corresponding instrument will not be classified.

C. Partial Filtering

The isolated partials identified in the previous step are separated by means of a filter. In real signals, the bandwidth of a partial depends on its frequency modulation and amplitude modulation rates, as well as on the amplitudes and reverberation, which in turn depend on instrument type, environment, and other factors. Therefore, a filter with a narrow pass-band may be appropriate for some kinds of sources, but may ignore relevant parts of the spectrum for others. On the other hand, a broad pass-band will certainly include the whole relevant portion of the spectrum, but may also include spurious components resulting from noise and even neighbor partials. Experiments have indicated that the most appropriate band to be considered around the peak of a partial is given by the interval $[0.5\cdot(p_{n-1}+p_n), 0.5\cdot(p_n+p_{n+1})]$, where p_n is the frequency of the partial under analysis, and p_{n-1} and p_{n+1} are the frequencies of the closest partials with lower and higher frequencies, respectively.

The partials are separated using the overlap-add method [35]. First, the frame under analysis is divided into sub-frames, whose length N is such that $L+N-1$ is a power of two, where L is the filter length as defined below. The (Hamming) window method [36] is used to dynamically design a new FIR filter for each partial. The filters must meet two requirements: the cutoff frequencies must match the edges of the interval presented above, and the cutoff slope must be sharp enough to block at least 99% of the energy belonging to any other partial. The first requirement aims to guarantee that the pass-band will be wide enough to include all relevant content even in cases in which the central frequencies of the partials vary significantly, and the second requirement aims to avoid that inconsistent data be fed to the rest of the algorithm. Several tests were performed to determine which is the lowest filter order that meets those requirements. The ideal filter order was found to be inversely proportional to the bandwidth of the pass-band (bw), and is given by 500 000/bw. This may result in a rather high filter order (the highest order found in the tests was 12 500), but a relatively low computational complexity is still achieved by employing the overlap-add method.

III. The Algorithm

Fig. 2 shows the basic structure of the algorithm; each numbered part is described in the following subsections.

A. Feature Extraction

After the partials are separated (step 1 in Fig. 2), 34 features are extracted from each one of them. Most of the features were implemented based on [37] and [38], slightly modified to be extracted in a partialwise basis. Some of the tested features were original. For details about the feature calculation, see the Appendix.

The method proposed here is based on a pairwise approach, in which the classification of each partial is performed for every possible pair of instruments. Since 25 instruments are considered here, there are 300 possible combinations of two instruments. Nine features from the whole set of 34 are then associated to each pair of instruments (step 2 in Fig. 2). The features that represent each pair were selected from the whole set using the procedure suggested by Deng *et al.* [39]. Such a procedure

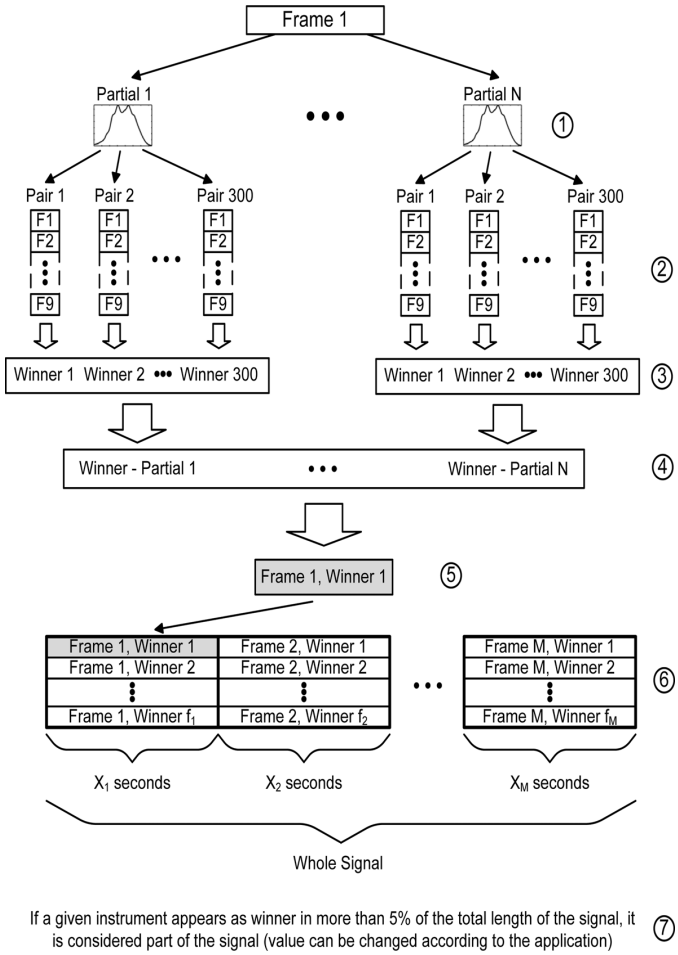

Fig. 2. Basic structure of the algorithm.

searches for the features that best correlates with the dataset it should represent and, at the same time, tries to eliminate redundant features, thus picking a subset of the original features. Normally, this would result in a different number of features for each pair. However, experiments revealed that the voting system and majority rules to be applied in the following work better if nine features are considered. A list with the features used for each pair of instruments can be found in the address http://opihi.cs.uvic.ca/music_instrument_featuresTASLP.pdf.

B. Classification Scheme

In the next steps, successive majority rules are applied. The first and most basic classification is given by individual features within each pair of instruments. This first classification consists of two steps.

1) The fundamental frequency of the source is compared to the normal frequency range of both instruments. If the F0 is outside both ranges, there is no winner instrument; if it is within the range of only one of the instruments, such an instrument is automatically taken as the winner; if it is within both ranges, the second step takes place.
2) A simple linear discrimination is applied. If the value of the feature is larger than a given value, it represents an instrument, otherwise it represents the other one. Those threshold values were determined through experiments over the training set. This simple approach yielded to

good results in comparison with other discrimination methods—it achieved an accuracy of 63.1%, against 55.5% and 64.1% using a single perceptron and support vector machines (SVM), respectively. Although the results using SVM were slightly better, tests using a different database revealed that the SVM actually resulted in a poorer generalization capacity—in this case, the SVM achieved an accuracy of 57.9%, against 59.7% achieved by the proposed approach. Although there is no definitive explanation for this rather counterintuivive result, it seems that the relative importance of each feature is reduced by considering each one of them separately and then summarizing the results through majority rules. As a result, bad features have limited impact in the overall accuracy. On the other hand, the SVM considers a multidimensional space determined by all the features considered at once. In this case, a single bad feature may have significant impact on the overall classification scheme.

The results of all nine features are then summarized, and the instrument that appears more times is taken as the winner for that particular pair (step 3 in Fig. 2). Since the number of features is odd, equality between the instruments is not possible at this point.

A new summarization is then performed along all 300 pairs of instruments. The instrument that appears more times among the pair winners is taken as partial winner (step 4 in Fig. 2). If an equality occurs, the total number of wins considering each feature separately is used as a tiebreaker. A new equality in this case is very unlikely but, if it happens, the winner is taken randomly among the tied instruments.

A further summarization is then performed along all isolated partials, whose number may vary depending on the number and frequencies of the other simultaneous instruments. If there is only one isolated partial, the partial winner will also be the frame winner, otherwise the instrument that wins for the greatest number of partials is taken as the frame winner (step 5 in Fig. 2). At this point, multiple equalities are common. The first tiebreaker is the total number of pair wins within the frame, and if a (unlikely) new tie occurs, the total number of wins considering individual features is taken into account.

The same procedure is repeated for all simultaneous sources present in the frame, and then for all frames of the signal (step 6 in Fig. 2). Finally, if an instrument is present in more than 5% of the total duration of the signal, it is definitely considered as being part of the signal (step 7 in Fig. 2). The threshold of 5% was chosen to avoid propagating isolated frame misclassifications to the entire signal. In this way, an instrument that is mistakenly considered as being part of a frame (which will be likely smaller than 5% of the whole signal) is prevented from being considered part of the whole signal. On the other hand, instruments that are actually present for only a short period may be discarded. Because of that, the threshold is a parameter left open in the final algorithm, so the user can set it according to the expected characteristics of the signals to be analyzed.

IV. Experimental Results

The training of the method was performed using individual instrument notes taken from the RWC database [40]. All samples used in the training process—about one-third of the notes available in the RWC database corresponding to two hours of material—were removed from the experimental tests to avoid biased results. The tests were performed using the remaining two-thirds of the RWC signals, individual notes taken from the University of Iowa musical instrument samples database [41], and a number of real recordings, as will be described throughout this section.

The experimental tests performed here can be divided into three main stages. The first stage aimed to measure the specific performance of the method to identify instruments, and is described in Section IV-A. The second stage aimed to analyze the effects of errors in the onset identification (Section IV-B) and in the fundamental frequency estimation (Section IV-C). Since the effects of errors in the number of instruments are straightforward—underestimates imply in instruments remaining unclassified, and overestimates imply in the inclusion of nonexistent instruments—no specific study on those was carried out. Finally, the third stage aimed to assess the performance of the entire system (all supporting tools included) using both artificial mixtures (Section IV-E) and real recordings (Section IV-F). The 95% confidence interval for all results presented in this section is given, in average, by $\mathrm{value}(\%) \pm 3\%$.

A. Performance of the Instrument Identification

This part of the tests aimed to measure the specific performance of the method in identifying instruments, in which case the onset positions, number of instruments and fundamental frequencies were supposed to be known. A large number of signals was generated for the tests as follows. Each signal is composed by a 100-ms segment of silence, followed by the active part created by summing a number of individual notes from different instruments, and ends with another 100-ms segment of silence. The active part starts with the alignment of the onsets of all simultaneous notes, and ends with the offset of the last note still active. The duration of each signal is thus directly related to the duration of the longest note in the mixture—all signals have a duration between 0.5 and 5 s (most in the range 1.5–3 s). As can be seen, the signals are actually composed by only one frame, because in this part of the tests the onset identification is not under investigation. The number of simultaneous sources in each signal can vary between two and five, and the instruments and respective notes were taken randomly, provided that each instrument has at least one isolated partial. This means that any harmonic relation between the notes is allowed, with exception of the 1:1 ratio. As a result, most signals have at least one pair of closely related notes (2:1, 3:2, and 4:3 frequency ratios). A given instrument can appear more than once in the same mixture, and the entire note range of each instrument was considered. In total, a little less than 9 hours of audio material was used in the tests. Table I shows the number of signals generated from each database (RWC and University of Iowa), in terms of the number of simultaneous notes.

Fig. 3 presents the confusion matrix for individual partials. Those results are equivalent to assuming that all instruments have always only one isolated partial available, so a partial winner will always be a frame winner (step five does not take place—see Section III-B). Fig. 4 presents the confusion matrix

	vi	va	ce	cb	gu	pn	eg	eb	fl	pc	cl	as	bs	ss	ts	ob	bn	tb	tp	hn	tu	ac	ha	vp	vo
vi	85	1	11	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1
va	1	86	1	1	0	0	0	0	1	2	0	0	0	0	0	0	0	0	2	2	3	1	0	0	0
ce	11	2	82	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4
cb	0	0	0	79	2	7	7	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0
gu	0	0	0	3	46	21	22	7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
pn	0	0	0	0	14	67	3	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	14	0
eg	0	0	0	2	16	15	61	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
eb	0	0	0	7	17	13	3	60	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
fl	0	0	0	5	0	0	0	0	55	7	7	0	0	1	0	7	0	4	6	2	0	3	3	0	0
pc	0	5	0	0	0	0	0	0	4	88	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0
cl	0	0	0	1	0	0	0	0	3	0	76	1	1	2	0	1	1	1	3	0	4	6	0	0	0
as	0	0	0	0	0	0	0	0	2	1	0	50	9	1	1	4	16	2	11	1	0	1	1	0	0
bs	0	0	0	0	0	0	0	0	0	0	0	13	73	1	1	0	4	1	4	3	0	0	0	0	0
ss	0	0	0	0	0	0	0	0	4	1	1	15	5	46	2	14	3	1	4	1	1	1	1	0	0
ts	0	0	0	1	0	0	0	0	3	0	0	6	17	6	55	3	2	0	4	2	0	1	0	0	0
ob	0	1	0	0	0	0	0	0	16	4	4	11	2	2	2	40	6	1	3	0	0	3	5	0	0
bn	0	0	0	0	0	0	0	0	5	0	3	8	7	1	0	4	45	5	13	3	3	2	1	0	0
tb	0	1	0	1	0	0	0	0	13	2	6	2	1	0	1	2	1	43	12	15	0	0	0	0	0
tp	0	1	0	0	0	0	0	0	6	3	2	8	6	4	0	3	2	2	46	13	0	4	0	0	0
hn	0	3	0	0	0	0	0	0	3	0	0	1	1	2	3	1	0	4	26	56	0	0	0	0	0
tu	0	0	0	3	0	0	0	0	0	0	4	0	1	0	0	0	8	3	4	0	77	0	0	0	0
ac	0	0	0	0	1	0	0	0	1	0	5	8	1	1	1	12	2	0	6	1	1	45	15	0	0
ha	0	0	0	0	1	3	0	0	1	0	1	12	4	4	1	1	3	3	0	1	0	0	65	0	0
vp	0	0	0	1	0	12	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	83	1
vo	6	1	3	1	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	87

Fig. 3. Results for isolated partials (in %).

TABLE I
NUMBER OF SIGNALS USED IN THE TESTS

Database	Number of simultaneous notes			
	2	3	4	5
RWC	2000	2000	2000	2000
Iowa	1000	1000	1000	1000

for the cases in which at least six isolated partials were available for each instrument. Table II shows the codes used for each instrument (in alphabetical order). In Figs. 3–5, the darkest is the shade of gray, the most related are the instruments. The main diagonal in the matrices represent the correct classifications.

The analysis of Fig. 3 reveals some important information. First, the results are better for instruments that have only a few related instruments. Indeed, it should be expected that voice signals, which have very particular characteristics, lead to much better results (87% of correct classifications) than, for instance, alto saxophone (50%), which has four related instruments in the database (other types of saxophone). It is easy to notice that most misclassifications occur amidst related instruments, which indicates that the method is capable to at least identify the correct class of instruments in most cases. Good results are also achieved for instruments whose typical frequency range is in one of the extremes of the spectrum, as is the case of piccolo (88%), whose notes have frequencies that are higher than the highest notes produced by most instruments. This is because the F0 of the source with respect to the ranges of the instruments is one of the criteria of classification, as described in Section III-B. If the F0 is outside the range of most instruments, the number of potential winners is small, increasing the chances of a correct classification. Overall, the results shown in Fig. 3 can be considered remarkably good, considering that they were obtained using only individual partials.

As expected, the results shown in Fig. 4 are better than those in Fig. 3, since there are more partials to summarize the results—the final classification is given by the instrument with more wins along all partials. In some cases, having several partials available nearly double the accuracy, like in the case of accordion (44% to 82%). The overall accuracy of the algorithm when several partials are available was 80.3%. Previous studies show that the instrument recognition rates by human listeners are usually significantly lower, especially if many instruments are involved [42]–[45]. For example, Srinivasan *et al.* [45] carried out a study in which conservatory students were asked to associate isolated tones to one of 27 possible instruments, with an average recognition rate of 55.7%.

The use of the same database to train and test the algorithm may, in some cases, lead to deceptively good results [46]. Because of that, samples from the University of Iowa musical instrument samples database [41] were used to provide a cross-database validation, whose results for isolated partials are shown in Fig. 5. Because this database includes only 15 of the 25 instruments used in the training, ten rows of the confusion matrix were removed, but the columns were kept as the signals can still be classified as one of the missing instruments. As a result, there is no main diagonal anymore, and the correct classifications are given by the cells with the darkest shade of gray. As can be seen, the results are slightest worse for most instruments, and a little better for a few of them. The overall recognition rate was 60.5%, against 62.2% for the RWC database (with the ten missing instruments excluded in Fig. 3). Considering the cases with at least six partials present, the accuracy rates were 76.7% for the University of Iowa database and 78.7% for the RWC database. Such small differences provide some evidence that the algorithm has good generalization capacity.

	vi	va	ce	cb	gu	pn	eg	eb	fl	pc	cl	as	bs	ss	ts	ob	bn	tb	tp	hn	tu	ac	ha	vp	vo
vi	91	0	7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0
va	0	93	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	0	2	0	0	0	0
ce	10	0	90	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
cb	0	0	0	93	0	0	7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
gu	0	0	0	0	59	17	15	11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
pn	0	0	0	0	7	80	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	13	0
eg	0	0	0	0	14	6	73	7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
eb	0	0	0	12	0	0	0	88	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
fl	0	0	0	3	0	0	0	0	77	3	2	0	0	0	0	2	0	0	5	0	0	0	8	0	0
pc	0	3	0	0	0	0	0	0	3	94	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
cl	0	0	0	0	0	0	0	0	0	0	95	0	0	0	0	0	0	0	0	0	0	5	0	0	0
as	0	0	0	0	0	0	0	0	0	0	0	67	6	3	0	0	9	0	15	0	0	0	0	0	0
bs	0	0	0	0	0	0	0	0	0	0	0	13	78	0	0	0	4	0	5	0	0	0	0	0	0
ss	0	0	0	0	0	0	0	0	0	0	0	23	6	60	0	5	0	0	6	0	0	0	0	0	0
ts	0	0	0	0	0	0	0	0	0	0	0	0	14	0	79	4	0	0	3	0	0	0	0	0	0
ob	0	0	0	0	0	0	0	0	6	6	0	3	0	0	0	69	11	0	3	0	0	0	2	0	0
bn	0	0	0	0	0	0	0	0	4	0	0	7	0	0	0	0	72	7	7	3	0	0	0	0	0
tb	0	0	0	0	0	0	0	0	14	3	3	3	0	0	0	0	0	66	6	5	0	0	0	0	0
tp	0	0	0	0	0	0	0	0	6	6	3	8	6	6	0	0	3	0	52	7	0	3	0	0	0
hn	0	3	0	0	0	0	0	0	0	0	0	0	0	4	4	0	0	0	14	75	0	0	0	0	0
tu	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0
ac	0	0	0	0	0	0	0	0	0	0	6	0	0	0	0	6	0	0	2	0	0	83	3	0	0
ha	0	0	0	0	0	11	0	0	0	0	0	5	0	0	0	0	5	0	0	0	0	0	79	0	0
vp	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0
vo	2	1	1	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	94

Fig. 4. Results when at least six partials are present (in %).

	vi	va	ce	cb	gu	pn	eg	eb	fl	pc	cl	as	bs	ss	ts	ob	bn	tb	tp	hn	tu	ac	ha	vp	vo
vi	79	8	7	2	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	2
va	9	78	4	1	0	0	0	0	3	3	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0
ce	9	5	80	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3
cb	0	0	0	79	4	5	7	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
pn	0	0	0	0	15	69	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	12	0
fl	0	0	0	1	0	0	0	0	59	9	4	0	0	2	0	4	0	6	7	1	0	4	3	0	0
cl	0	0	0	1	0	0	0	0	5	0	72	2	2	2	0	1	0	0	5	0	6	4	0	0	0
as	0	0	0	0	0	0	0	0	2	1	0	49	7	2	5	7	11	0	13	0	0	3	0	0	0
ss	0	0	0	0	0	0	0	0	5	0	3	17	5	46	0	14	1	1	5	0	1	2	0	0	0
ob	0	1	0	0	0	0	0	0	13	6	5	9	1	0	0	42	6	4	4	0	0	5	4	0	0
bn	0	0	0	0	0	0	0	0	6	2	2	9	6	1	1	3	40	6	12	4	5	0	3	0	0
tb	0	0	0	0	0	0	0	0	14	3	3	0	0	0	2	5	2	47	13	11	0	0	0	0	0
tp	0	0	0	0	0	0	0	0	10	2	1	9	3	5	1	0	4	3	42	14	0	4	2	0	0
hn	0	2	0	0	0	0	0	0	5	0	0	3	0	4	2	1	2	7	24	50	0	0	0	0	0
tu	0	0	0	4	0	0	0	0	0	0	5	0	0	0	0	0	11	2	2	0	76	0	0	0	0

Fig. 5. Results for isolated partials using the University of Iowa database [41] (in %).

TABLE II
INSTRUMENT CODES

Code	Instrument		Code	Instrument
ac	accordion		ob	oboe
as	alto saxophone		pc	piccolo
bn	bassoon		pn	piano
bs	bass saxophone		ss	soprano saxophone
cb	contrabass		tb	trombone
ce	cello		tp	trumpet
cl	clarinet		ts	tenor saxophone
eb	electric bass		tu	tuba
eg	electric guitar		va	viola
fl	flute		vi	violin
gu	guitar		vo	voice
ha	harmonica		vp	vibraphone
hn	french horn			

Table III shows a condensed performance comparison for some selected instruments, reinforcing the idea that the best results are indeed obtained considering several partials, and that there is only a small drop in the accuracy when signals from a database not used in the training are submitted to the algorithm. In Table III the overall results presented for the RWC database consider only the 15 instruments present in the University of Iowa database.

All results presented to this point were obtained under the assumption that onset positions, number of simultaneous sources and fundamental frequencies are known, so the instrument identification procedure could be assessed in isolation. The next subsections present some additional studies and results in order to provide a more comprehensive picture of how a complete system would perform. The tests presented in Sec-

TABLE III
CONDENSED COMPARISON BETWEEN DIFFERENT CASES (IN %)

	vi	ce	pn	fl	cl	as	tb	hn	all
RWC (1 partial)	85	82	67	55	76	50	43	56	62
Iowa (1 partial)	79	80	69	59	72	49	47	50	60
RWC (6 partials)	91	90	80	77	95	67	66	75	79
Iowa (6 partials)	86	87	80	78	92	67	67	71	77

tions IV-B to IV-E were performed using the signals generated from the University of Iowa database in order to provide results as unbiased as possible.

B. Effects of Onset Misidentifications

This section analyses the effects of onset misplacements. The following kinds of onset location errors may occur.

1) *Small errors*: Errors smaller than 10% of the frame length have little impact on the accuracy of the instrument identification, because the characteristics of the partials are only slightly altered.
2) *Large errors, estimated onset placed after the actual position*: This kind of error has little effect over sustained instruments, because the characteristics of the note are approximately the same for its whole extent. On the other hand, this kind of error may cause problems for instruments whose notes decay over time (e.g., piano and guitar), because the main content of the note, which is usually near the actual onset, may be lost. In those cases, the instrument identification accuracy drops almost linearly with the onset error—for example, a 30% forward error in the onset position will result in 30% less accurate estimates for instruments with decaying notes.
3) *Large errors, estimated onset placed before the actual position*: In this case, a part of the signal that does not contain the new note is considered. This may not affect the accuracy at all, if the notes in the spurious segments are not harmonically related to the notes in the actual segment. If there is some kind of harmonic relationship, then the severeness of the effects will be directly linked to the number of common partials between the spurious and actual notes, and to the length of the onset displacement. Table IV shows the effects of the onset misplacements (given in percentage of the actual frame) when the interfering note has 1/3, 1/2 and all partials in common with the note to be classified. The values given in Table IV represent the relative accuracy given by A_e/A_i, where A_e is the accuracy of the method when there is onset misplacement, and A_i is the accuracy of the method when the onset is in the correct position. As can be seen, when there are some partials that are not affected by the interfering note, the method is able to compensate in part the problems caused by the onset misplacement. Even when the interfering note has the same frequency as the target note, the algorithm is able to compensate the damaging effects to some degree because, when the amplitudes of interfering partials are low, the characteristics of the actual partial to be classified may still stand out. Fig. 6 summarizes the effects of both backward and forward onset misplacements.

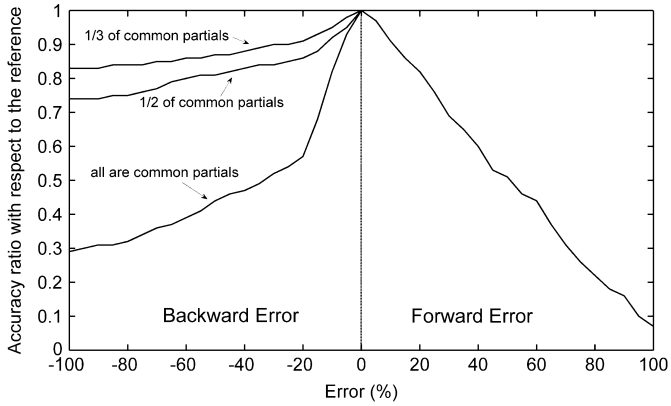

Fig. 6. Effect of the backward and forward onset misplacements.

TABLE IV
EFFECT OF ONSET BACKWARD ONSET MISPLACEMENTS

Common Partials	20%	40%	60%	80%	100%
1/3	.91	.88	.86	.84	.83
1/2	.86	.83	.80	.75	.74
1	.57	.47	.39	.32	.29

C. Effects of F0 Misestimates

The F0 estimator used in the system [32] has an overall accuracy around 85%. This does not mean that the results shown in Figs. 3–5 will be 15% worse when this tool is incorporated. Actually, the impact is less severe, and depends on the kind of error. The most common error in F0 estimator are the so-called octave errors, in which the estimate is actually a multiple or submultiple of the correct F0. If the misestimate is a multiple of the correct F0, the only effect is that fewer partials will be available for summarization, hence the impact is very limited: the accuracy drops by 2% if the estimated F0 is one octave above the correct one, by 5% if it is two octaves above, and by 7% if it is three octaves above.

If the misestimate is a submultiple of the actual F0, all correct partials will be considered, together with a number of spurious ones. The impact here is also limited: if the spurious partials collide with actual partials from other instruments, the only effect will be the removal of the partials from the process, as only isolated partials are considered; on the other hand, if the spurious partials are in a noise-only part of the spectrum, they will generate wild data that will result in a variety of classifications, and if the actual partials are classified correctly, they will dominate the scoring process and no error will occur. Experiments showed that the accuracy drops in average 4% if the estimated F0 is one octave below the correct one, 7% if it is two octaves below, and 13% if it is three octaves below.

Other kinds of F0 misestimates have a much more pronounced impact, and such impact will be as severe as less actual partials are taken into account. In general, no accuracy whatsoever can be expected if the misestimated F0 has no harmonic relation with the actual one. Table V summarizes the information presented in this section. In this table, the first row presents the estimated frequency with respect to the correct

TABLE V
EFFECT OF F0 MISESTIMATES

Freq.	f	f/2	f/4	f/8	3f/2	2f	3f	4f
Accuracy	0.76	0.72	0.69	0.63	0.72	0.74	0.71	0.69

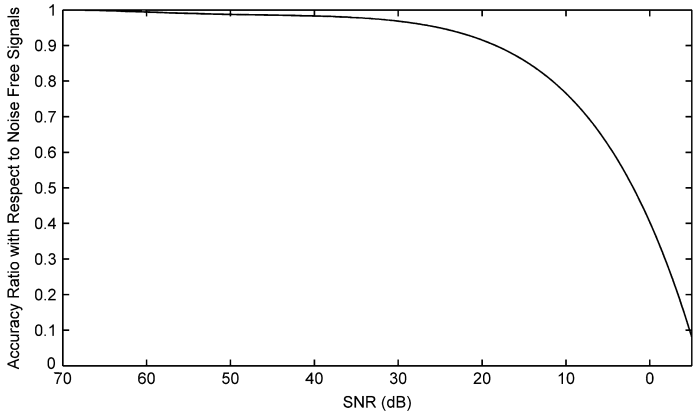

Fig. 7. Effect of noise on the algorithm performance.

TABLE VI
PERFORMANCE UNDER NOISY CONDITIONS

SNR (dB)	60	50	40	30	20	15	10	5	0
Accuracy	.99	.99	.98	.97	.95	.90	.81	.69	.40

frequency f (e.g., $f/2$ indicates that the estimated frequency is one octave below the correct one).

D. Performance Under Noisy Conditions

Because the strategy presented here depends on fine-tuned features and thresholds, it is sensitive to noise. In general, it was observed that any colored noise is less harmful than white noise, because in the former case some parts of the spectrum are only mildly affected by the noise, so some partials may still maintain their original characteristics. On the other hand, white noise is likely to affect all partials, potentially making the scoring process and majority rules unreliable. Fig. 7 and Table VI show the performance degradation for different signal-to-noise ratios (SNRs), using white noise. As in Table IV, the numbers represent the relative accuracy given by A_e/A_i, where A_e is the accuracy of the method under noisy conditions, and A_i is the accuracy of the method when the signal is noise-free. As can be seen, the algorithm begins to lose reliability for SNR below 15 dB. However, it is important to notice that the noise levels of musical signals are, in general, low, so the algorithm is expected to work properly for the vast majority of the signals.

E. Precision and Recall in Isolated Frames

The results presented in this subsection were obtained using the entire system, which includes the tools for location of onsets, estimation of the number of simultaneous instruments, estimation of the fundamental frequencies, and the instrument classifier itself. Since the number of instruments is unknown, measures like precision and recall [19] provide useful information on the performance of the algorithm. Table VII shows the precision and recall values as a function of the number of simultaneous instruments. As can be seen, the results deteriorate significantly as the number of simultaneous instruments increase.

TABLE VII
PRECISION AND RECALL FOR THE ARTIFICIAL MIXTURES

Number of	RWC database		Univ. Iowa database	
Instruments	Precision	Recall	Precision	Recall
2	0.84	0.77	0.83	0.75
3	0.77	0.70	0.75	0.69
4	0.69	0.59	0.69	0.57
5	0.59	0.51	0.56	0.50
6	0.50	0.43	0.50	0.41
all	0.70	0.58	0.68	0.57

TABLE VIII
PERFORMANCE OF THE SYSTEM FOR REAL RECORDINGS

Number of Instruments	Number of Excerpts	Precision	Recall
2	12	0.55	0.98
3	29	0.59	0.96
4	29	0.60	0.95
5	17	0.61	0.93
6	10	0.60	0.93
7	3	0.56	0.94
all	100	0.59	0.95

This is due to two factors: first, more instruments mean that there will be less clean partials available; second, the accuracy of the number of sources is better when there are only a few instruments present. Fortunately, the musical signals to be analyzed normally have several frames that can be analyzed by the algorithm, and the subsequent integration of the results over the entire signal is able to partially compensate those problems. This can be seen in Section IV-F, which presents the results for real recordings.

F. Classification of Real Recordings

The results presented in this subsection were also obtained using the entire system. One-hundred 1-min excerpts were taken from commercial recordings, comprising the musical genres pop/rock, classical, and jazz. The excerpts were chosen in such a way they do not include any nonpercussive instruments other than those used in the main tests. Each excerpt may have between two and seven instruments. Twenty-nine of the excerpts include percussion instruments, which cannot be classified by the algorithm, so they act as noise. In average, an instrument is present in 16 excerpts; the piano (17 occurrences) is the most frequent instrument, and vibraphone (2 occurrences) is the least represented one.

Since the entire system was used here, the only input to the algorithm is the signal to be classified. Table VIII presents the results in terms of the number of instruments in the excerpt. Table IX presents the results for each instrument.

As can be seen in Tables VIII and IX, the recall values are very high, meaning that almost all instruments present in the signal are identified. On the other hand, the precision value is close to 0.6. This means that, for each ten instruments correctly identified, between six and seven instruments that are not present are mistakenly identified. However, it is important to highlight that most false positives come from instruments that are related to those present in the signal. This can be seen in Table X, which shows the results for ten selected signals. In this table, the *instruments* column shows the instruments present in the signal, the *number of instruments* column shows the total number of

TABLE IX
PERFORMANCE OF THE SYSTEM IN TERMS OF INDIVIDUAL INSTRUMENTS

Instrument	Number of Appearances	Precision	Recall
violin	15	0.75	0.94
viola	7	0.70	1.00
cello	8	0.80	1.00
contrabass	8	0.62	1.00
guitar	6	0.50	0.86
piano	17	0.47	0.94
elec. guitar	10	0.63	0.91
elec. bass	12	0.67	0.92
flute	8	0.47	1.00
piccolo	3	0.75	1.00
clarinet	11	0.65	1.00
alto sax	8	0.40	0.89
bass sax	4	0.50	1.00
soprano sax	6	0.60	0.86
tenor sax	5	0.71	1.00
oboe	10	0.40	0.91
bassoon	7	0.54	0.88
trombone	7	0.54	0.88
trumpet	13	0.39	0.93
french horn	4	0.57	1.00
tuba	6	0.67	1.00
accordion	3	0.50	1.00
harmonica	3	0.60	1.00
vibraphone	2	0.67	1.00
voice	12	0.86	1.00

TABLE X
PERFORMANCE FOR SELECTED EXCERPTS

	instruments	n. instr.	identified	false hits (related)
Sig. 1	cb,ce,va,vi	4	4	1 (0)
Sig. 2	as,eb,pn,tp,ts	5	5	4 (2 - bs,ss)
Sig. 3	bn,cb,ob,tb,vi	5	4	2 (1 - va)
Sig. 4	eb,eg,vo	3	3	3 (2 - gu,pn)
Sig. 5	ce,cl,fl,ob,pn,vi	6	5	2 (2 - as,gu)
Sig. 6	gu,vo	2	2	1 (0)
Sig. 7	bn,tb,tu	3	3	2 (2 - hn,ob)
Sig. 8	as,eb,eg,ss	4	4	2 (2 - pn,ts)
Sig. 9	bs,eg,pn,tb,tp	5	5	6 (3 - gu,hn,ts)
Sig.10	as,cl,fl,ob	4	4	3 (2 - pc,ss)

instruments (percussion instruments not included), the *identified* column shows the number of instruments correctly identified by the algorithm, and the *false hits (related)* column shows the number of instruments mistakenly identified by the algorithm and, between parenthesis, how many among those false hits are related to the other instruments in the signal, according to the second darkest shade of gray in Fig. 3. As can be seen, all instruments were recognized for eight signals, and only one instrument was missed for the other two. This reinforces the claim that the algorithm is effective in identifying the correct instruments in real signals. On the other hand, the number of false hits is, as expected, relatively high, and every signal had at least one false hit. However, as commented before, many of those false hits are of instruments closely related to the correct ones (e.g., misidentifying oboe as bassoon), which indicates that the algorithm is able to recognize the instrument family in most cases. The number of false hits can be reduced by changing the threshold for a instrument to be considered part of the signal from 5% to larger values. However, this also reduces the number of correct hits. Thus, such a parameter must be set taking into consideration the tradeoff between the number of correct and false hits.

As stated before, there are 29 excerpts that contain nonharmonic instruments. In most of those excerpts, the nonharmonic instruments are not strong or frequent enough to cause any discernible drop in the performance of the algorithm. However, in some cases they act like a very strong noisy interference, which can cause the SNR of some segments to drop below 0 dB. This happens in six of the excerpts—considering only those, both the recall and precision have values around 0.6. This means that the algorithm may have problems if the signal under analysis has very strong percussive elements.

V. CONCLUSION

This paper presented a new method to identify musical instruments in polyphonic musical signals. The method uses a pairwise comparison approach to determine which instrument corresponds to each individual partial, and summarizes the results to provide an overall estimate of the instruments present in the signal. Tests performed with notes extracted from 25 instruments and two databases showed that the method has comparable or even possibly superior performance than human listeners, and works well for SNRs above 15 dB.

A possible shortcoming of the proposed algorithm would be its dependency on other tools to perform tasks like onset detection, estimation of the number of simultaneous instruments, and estimation of the fundamental frequencies. Tests have shown that, although errors caused by imperfections on those supporting tools indeed propagate throughout the system, their overall effect is actually relatively mild. Further evidence is provided by the tests performed with the whole system using real recordings, which revealed that the correct instruments are identified in the vast majority of the cases, although the number of false hits is still relatively high.

A direct comparison with other methods for instrument recognition was not presented here due to several practical constraints. A fair comparison is only possible if the same signals and the same number of instruments are considered. This eliminates the possibility of using the results published by the respective authors, since the signals and number of instruments vary wildly. The only option would be implementing the methods and testing them over the same database. This option poses three problems: first, many methods are described in short conference papers that do not provide all information necessary to reproduce exactly the algorithm implemented by the authors; second, many of the methods are quite complex, making the debugging process very difficult without having some specific knowledge; and third, many of the proposals use a quite different classification philosophy, like, for example, using an entire instrument phrase at once, making it difficult to make a direct comparison even if the original algorithm is available. This indicates that there is a need for a standardized metric and database for evaluating instrument classification in polyphonic music. We intend to explore possible solutions to this problems in the near future.

The results presented here can be extended in a number of directions. One important improvement would be including the ability to recognize sound sources that lack harmonic structure, as most percussion instruments. Future work can also concentrate in reducing the dependency of the algorithm on side

information provided by other tools (onset position, number of instruments, fundamental frequencies). Finally, it would be useful to create mechanisms to improve the tradeoff between the number of correct and false hits. We expect that the algorithm proposed here help to expand the applicability and effectiveness of a number of digital music applications.

APPENDIX

This Appendix presents the way each one of the 34 features used in this work were calculated. All features are calculated for individual partials.

A. Features Based on the Energy Spectrum

Features 1 to 3—Bandwidth: These features measure the bandwidth containing a given percentage of the total energy in the partial, which is centered around the frequency bin where the maximum magnitude occurs in the DFT. The DFT is calculated across the entire length of the partial. The features are given by

$$\text{band} = \min(B) : \frac{\sum_{k=i}^{i+B} X^2(k)}{\sum_{k=1}^{K} X^2(k)} \geq R \tag{1}$$

where $X(k)$ is the magnitude spectrum resulting from applying a discrete Fourier transform (DFT) to the filtered partial, B is the length (in frequency bins) of the subband to be tested, i is the index of the first frequency bin of the subband, k represents the index of the frequency bins within the partial, K is the total number of frequency bins in the partial, and R is the proportion of energy, which is 0.9 for feature 1, 0.95 for feature 2, and 0.99 for feature 3.

Feature 4—Relative Centroid: It measures the asymmetry of the energy spectrum with respect to the center of the partial, and is given by

$$\text{rcent} = \frac{\sum_{k=1}^{K}(k - (K+a)/2) \cdot X^2(k)}{\sum_{k=1}^{K} X^2(k)} \tag{2}$$

where $a = 1$ for K odd, and $a = 2$ for K even.

B. Features Based on the Amplitude Envelope

Features 5 to 10—Amplitude Modulation Features: These features are inspired on the homonymous ones suggested by Eronen [47]. They are based on the amplitude envelopes extracted from the temporal signals corresponding to each partial. The envelope y is generated by dividing the signal into 10-ms frames with 50% overlap, and calculating their RMS values. The Fourier transform Y of the envelope y is then extracted, which is the basis for the calculation of the features, as described next.

Feature 5: is given by the frequency of the highest peak of Y in the 4–8 Hz range.

Feature 6: same as feature 5, but in the 10–40 Hz range.

Feature 7: is given by the magnitude of the highest peak in the 4–8 Hz range, divided by the mean magnitude of all frequency bins.

Feature 8: same as feature 7, but in the 10–40 Hz range.

Feature 9: is given by the magnitude of the highest peak in the 4–8 Hz range, divided by the mean magnitude of all frequency bins within that range.

Feature 10: same as feature 9, but in the 10–40 Hz range.

Feature 11—Crest Factor: This feature is inspired by the homonymous one suggest by Eronen [47], and is given simply by the ratio between the maximum and the RMS value of the amplitude envelope.

Feature 12—Onset Duration: This feature has also a counterpart in [47]. The beginning and end of the onset are given by the points where the magnitude of the amplitude envelope rises above -10 dB and -3 dB with respect to the RMS value of the entire note, respectively.

Feature 13—Slope of Amplitude Decay: Also inspired in [47], this feature is given by the gradient of the line fitting the segment of the amplitude envelope between its maximum and the point where it falls below -10 dB with respect to the RMS value of the entire note. Before the fitting, the amplitude envelope is transformed to the logarithmic domain.

Feature 14—MSE Between Line And Real Data: This feature is given by the mean squared error between the line used in feature 20 and the data it tries to approximate.

Feature 15—Amplitude Roughness: This feature is given by the ratio between the standard deviation and the mean of the amplitude envelope.

Feature 16—Relative Variation of Amplitude Envelope: This feature is calculated in the same way as Feature 30, but here the amplitude envelope is used instead of the frequency trajectory.

An alternative way of calculating the amplitude envelope using the Hilbert transform was suggested by Every [48]. Although the objective is the same as the procedure suggested by Eronen [47], the resulting curves are sufficiently different to reveal different characteristics of the signals. Features 17 to 22 are calculated based on this Hilbert transform-based amplitude envelope, which is obtained by performing the Hilbert transform of the waveform and low-pass filtering the magnitude of the result using a IIR Butterworth filter with a cutoff frequency of 20 Hz.

Feature 17—Centroid: This feature calculates the center of gravity of the amplitude envelope, and is given by

$$\text{cent} = \frac{1}{T}\sum_{t=1}^{T} x(t) \cdot t \tag{3}$$

where t is the time index, T is the number of samples and $x(t)$ is the amplitude of signal x in the instant t.

Feature 18—Note Spread: This feature is given by

$$\text{spread} = \sum_{t=1}^{T} x(t) \cdot \left(\frac{t}{T} - \text{cent}\right)^2 \tag{4}$$

Feature 19—Note Skewness: This feature is given by

$$\text{skew} = \frac{\sum_{t=1}^{T} x(t) \cdot \left(\frac{t}{T} - \text{cent}\right)^3}{\text{spread}^{1.5}} \tag{5}$$

Feature 20—Note Kurtosis: This feature is given by

$$\text{kurt} = \frac{\sum_{t=1}^{T} x(t) \cdot \left(\frac{t}{T} - \text{cent}\right)^4}{\text{spread}^{2}} \tag{6}$$

Features 21 and 22—AM Frequency and AM Amplitude: These features are calculated using the guidelines presented

in [48, p. 191]. Since the description for their extraction is extensive, it will not be presented here.

C. Features Based on the Frequency Trajectory

Features 23 to 28—Amplitude Modulation of the Frequency Trajectory of the Partials: These features are calculated exactly in the same way as features 5 to 10. The only difference is that the amplitude envelopes are replaced by the frequency trajectories of the partials, which are calculated using the normalized autocorrelation [49].

Feature 29—Standard Deviation of the Frequency Trajectory: This feature consists simply in calculating the standard deviation of the frequency trajectories of the partials.

Feature 30—Jitter: Inspired on the homonymous feature suggested by [38], it measures the stability of the partial over time using the frequency trajectories cited in Features 23 to 28. This feature is given by

$$\text{jitter} = \frac{\sum_{n=2}^{N-1} \left| f(n) - \frac{f(n-1)+f(n)+f(n+1)}{3} \right|}{\sum_{n=1}^{N} f(n)} \tag{7}$$

where n is the time index, N is the total number of samples in the frequency trajectory, and $f(n)$ is the frequency of the partial in the instant n.

Features 31 to 34—Frequency Trajectory Curve Features: The four final features are original, and are extracted from the frequency trajectory of the partials. First, a smoothed version of the frequency trajectory is generated by applying a low-pass filter with cutoff frequency at 25 Hz. Then, all local maxima and minima are identified. Finally, the following two features are extracted:

Feature 31: is given by the standard deviation of the distance between the peaks. This feature aims to determine if the frequency envelope fluctuates wildly (high standard deviation) or has a marked pattern.

Feature 32: is given by the standard deviation of the difference between the amplitudes of each local maximum and the next local minimum. This feature also aims to characterize the fluctuations present in the smoothed frequency trajectory.

Features 33 and 34 are calculated in the same way, but in this case the smoothed frequency trajectory is obtained by applying a low-pass filter with cutoff frequency at 50 Hz. Features 31 and 32 work better for some instruments, while features 33 and 34 work better for others. Due to their similarity, they rarely appear together in the group of nine features used to characterize each pair of instruments.

REFERENCES

[1] G. Agostini, M. Longari, and E. Pollastri, "Musical instrument timbres classification with spectral features," *EURASIP J. Appl. Signal Process.*, vol. 2003, pp. 5–14, 2003.

[2] E. Benetos, M. Kotti, and C. Kotropoulos, "Musical instrument classification using non-negative matrix factorization algorithms and subset feature selection," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2006, pp. 221–224.

[3] N. Chetry and M. Sandler, "Linear predictive models for musical instrument identification," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2009, pp. 173–176.

[4] A. Eronen and A. Klapuri, "Musical instrument recognition using cepstral coefficients and temporal features," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2000, pp. 753–756.

[5] S. Essid, G. Richard, and B. David, "Musical instrument recognition by pairwise classification strategies," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 14, no. 4, pp. 1401–1412, Jul. 2006.

[6] A. M. Fanelli, L. Caponetti, G. Castellano, and C. A. Buscicchio, "Content-based recognition of musical instruments," in *Proc. IEEE Int. Symp. Signal Process. Inf. Tech.*, 2004, pp. 361–364.

[7] M. Ihara, S.-I. Maeda, and S. Ishii, "Instrument identification in monophonic music using spectral information," in *Proc. IEEE Int. Symp. Signal Process. Inf. Tech.*, 2007, pp. 595–599.

[8] C. Joder, S. Essid, and G. Richard, "Temporal integration for audio classification with application to musical instrument classification," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 17, no. 1, pp. 174–186, Jan. 2009.

[9] A. G. Krishna and T. V. Sreenivas, "Music instrument recognition: From isolated notes to solo phrases," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2004, pp. 265–268.

[10] K. D. Martin and Y. E. Kim, "Musical instrument identification: A pattern-recognition approach," in *Proc. Meeting Acoust. Soc. Amer.*, 1998.

[11] C. Pruysers, J. Schnapp, and I. Kaminskyj, "Wavelet analysis in musical instrument sound classification," in *Proc. Int. Symp. Signal Process. Applicat.*, 1998.

[12] J. C. Brown, "Computer identification of musical instruments using pattern recognition with cepstral coefficients as features," *J. Acoust. Soc. Amer.*, vol. 105, pp. 1933–1941, 1999.

[13] J. C. Brown, O. Houix, and S. McAdams, "Feature dependence in the automatic identification of musical woodwind instruments," *J. Acoust. Soc. Amer.*, vol. 109, pp. 1064–1072, 2001.

[14] D. Fragoulis, C. Papaodysseus, M. Exarhos, G. Roussopoulos, T. Panagopoulos, and D. Kamarotos, "Automated classification of piano-guitar notes," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 14, no. 3, pp. 1040–1050, May 2006.

[15] J. J. Burred, A. Robel, and T. Sikora, "Polyphonic musical instrument recognition based on a dynamic model of the spectral envelope," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2009, pp. 173–176.

[16] P. Jincahitra, "Polyphonic instrument identification using independent subspace analysis," in *Proc. IEEE Int. Conf. Multimedia Expo*, 2004, pp. 1211–1214.

[17] K. Kashino and H. Murase, "A sound source identification system for ensemble music based on template adaptation and music stream extraction," *Speech Commun.*, vol. 27, pp. 337–349, 1999.

[18] T. Kitahara, M. Goto, K. Komatani, T. Ogata, and H. G. Okuno, "Instrument identification in polyphonic music: Feature weighting to minimize influence of sound overlaps," *EURASIP J. Appl. Signal Process.*, vol. 2007, pp. 1–15, 2007.

[19] L. G. Martins, J. J. Burred, G. Tzanetakis, and M. Lagrange, "Polyphonic instrument recognition using spectral clustering," in *Proc. Int. Conf. Music Inf. Retrieval*, 2007.

[20] E. Vincent and X. Rodet, "Instrument identification in solo and ensemble music using independent subspace analysis,," in *Proc. Int. Conf. Music Inf. Retrieval*, 2004, pp. 576–581.

[21] J. Eggink and G. J. Brown, "Application of missing feature theory to the recognition of musical instruments in polyphonic audio," in *Proc. Int. Conf. Music Inf. Retrieval*, 2003, pp. 1–7.

[22] J. Eggink and G. J. Brown, "Instrument recognition in accompanied sonatas and concertos," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2004, pp. 217–220.

[23] P. Leveau, D. Sodoyer, and L. Daudet, "Automatic instrument recognition in a polyphonic mixture using sparse representations," in *Proc. Int. Conf. Music Inf. Retrieval*, 2007.

[24] S. Essid, G. Richard, and B. David, "Instrument recognition in polyphonic music based on automatic taxonomies," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 14, no. 1, pp. 68–80, Jan. 2006.

[25] P. Somerville and A. L. Uitdenbogerd, "Multitimbral musical instrument classification," in *Proc. Int. Symp. Comp. Science Applic.*, 2008, pp. 269–274.

[26] J. Eggink and G. J. Brown, "A missing feature approach to instrument identification in polyphonic music," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2003, pp. 553–556.

[27] B. Kostek, "Musical instrument classification and duet analysis employing music information retrieval techniques," *Proc. IEEE*, vol. 92, no. 4, pp. 712–729, Apr. 2004.

[28] H. Thornburg, R. J. Leistikow, and J. Berger, "Melody extraction and musical onset detection via probabilistic models of framewise STFT peak data," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 15, no. 4, pp. 1257–1272, May 2007.
[29] G. Hu and D. Wang, "Auditory segmentation based on onset and offset analysis," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 15, no. 2, pp. 396–405, Feb. 2007.
[30] R. Zhou, M. Mattavelli, and G. Zoia, "Music onset detection based on resonator time frequency image," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 16, pp. 1685–1695, 2008.
[31] J. G. A. Barbedo, A. Lopes, and P. J. Wolfe, "Empirical methods to determine the number of sources in single-channel musical signals," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 17, no. 8, pp. 1435–1444, Nov. 2009.
[32] J. G. A. Barbedo, A. Lopes, and P. J. Wolfe, "High time-resolution estimation of multiple fundamental frequencies," in *Proc. Int. Conf. Music Inf. Retrieval*, 2007, pp. 399–403.
[33] J. Rauhala, H.-M. Lehtonen, and V. Valimaki, "Fast automatic inharmonicity estimation algorithm," *J. Acoust. Soc. Amer.*, vol. 121, pp. EL184–EL189, 2007.
[34] J. C. Brown, "Frequency ratios of spectral components of musical sounds," *J. Acoust. Soc. Amer.*, vol. 99, pp. 1210–1218, 1996.
[35] A. V. Oppenheim and R. W. Schafer*, Digital Signal Processing.*. Englewood Cliffs, NJ: Prentice-Hall, 1975.
[36] J. O. Smith, III, Spectral Audio Signal Processing. [Online]. Available: https://ccrma.stanford.edu/~jos/sasp
[37] A. Eronen, "Comparison of features for music instrument recognition," in *Proc. IEEE Workshop Applicat. Signal Process Audio Acoust.*, 2001, pp. 19–22.
[38] M. R. Every, "Discriminating between pitched sources in music audio," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 16, no. 2, pp. 267–277, Feb. 2008.
[39] J. D. Deng, C. Simmermacher, and S. Cranefield, "A study on feature analysis for musical instrument classification," *IEEE Trans. Syst., Man Cybern. B: Cybern.*, vol. 38, no. 2, pp. 429–438, Apr. 2008.
[40] M. Goto, "Development of the RWC music database," in *Proc. Int. Cong. Acoustics*, 2004, pp. 553–556.
[41] University of Iowa Musical Instrument Samples Database. [Online]. Available: http://theremin.music.uiowa.edu
[42] C. Elliott, "Attacks and releases as factors in instrument identification," *J. Res. Music Educat.*, vol. 23, pp. 35–40, 1975.
[43] R. A. Kendall, "The role of acoustic signal partitions in listener categorization of musical phrases," *Music Percep.*, vol. 4, pp. 185–214, 1986.
[44] K. D. Martin, "Sound-source recognition: A theory and computational model," Ph.D. dissertation, Mass. Inst. of Technol., Cambridge, U.K., 1999.
[45] A. Srinivasan, D. Sullivan, and I. Fujinaga, "Recognition of isolated instrument tones by conservatory students," in *Proc. Int. Conf. Music Percept. Cogn.*, 2002, pp. 17–21.
[46] A. Livshin and X. Rodet, "The importance of cross database evaluation in sound classification," in *Proc. Int. Conf. Music Inf. Retrieval*, 2003.
[47] A. Eronen, "Automatic musical instrument recognition," Ph.D. dissertation, Tampere Univ. of Technol., Tampere, Finland, 2001.
[48] M. R. Every, "Separation of musical sources and structure from single-channel polyphonic recordings," Ph.D. dissertation, Univ. of York, York, U.K., 2006.
[49] M. Wu, D. L. Wang, and G. J. Brown, "A multi-pitch tracking algorithm for noisy speech," *IEEE Trans. Speech Audio Process.*, vol. 11, no. 3, pp. 229–241, May 2003.

Jayme G. A. Barbedo received the B.S. degree in electrical engineering from the Federal University of Mato Grosso do Sul, Campo Grande, Brazil, in 1998, and the M.Sc. and Ph.D. degrees from the State University of Campinas, Campinas, Brazil, in 2001 and 2004, working on objective assessment of speech and audio.

From 2004 to 2005, he worked with the Source Signals Encoding Group of the Digital Television Division, CPqD Telecom and IT Solutions, Campinas. From 2006 to 2007, he was with the School of Engineering and Applied Sciences, Harvard University, Cambridge, MA, as a Postdoctoral Fellow. In 2008, he was awarded a fellowship from the Foreign Affairs and International Trade Canada (DFAIT) to work as a Postdoctoral Fellow at the Department of Computer Science of the University of Victoria, Victoria, BC, Canada, where he conducted studies in the areas of instrument identification and audio source separation. Currently, he is with the Department of Communications, School of Electrical and Computer Engineering, State University of Campinas as a Researcher. He has published several papers in some of the most important journals and conferences of the area of digital audio. His research interests also include audio and video encoding applied to digital television broadcasting, automatic music transcription, and super-resolution spectral estimation, among others.

George Tzanetakis (S'98–M'02) received the B.Sc. degree in computer science from the University of Crete, Heraklion, Greece, and the M.A. and Ph.D. degrees in computer science from Princeton University, Princeton, NJ. His Ph.D. work involved the automatic content analysis of audio signals with specific emphasis on processing large music collections.

In 2003, he was a Postdoctoral Fellow at Carnegie-Mellon University, Pittsburgh, PA, working on query-by-humming systems, polyphonic audio-score alignment, and video retrieval. Since 2004, he has been an Assistant Professor of Computer Science (also cross-listed in Music and Electrical and Computer Engineering) at the University of Victoria, Victoria, BC, Canada. His research deals with all stages of audio content analysis such as analysis, feature extraction, segmentation, and classification, with specific focus on music information retrieval (MIR). He is the principal designer and developer of the open source Marsyas audio processing software framework (http://marsyas.info). His work on musical genre classification is frequently cited.

Prof. Tzanetakis received an IEEE Signal Processing Society Young Author Award in 2004.