QUANTIFYING THE RELEVANCE OF LOCALLY EXTRACTED INFORMATION FOR MUSICAL INSTRUMENT RECOGNITION FROM ENTIRE PIECES OF MUSIC

Ferdinand Fuhrmann and Perfecto Herrera
Music Technology Group
Universitat Pompeu Fabra
Barcelona, Spain
{ferdinand.fuhrmann,perfecto.herrera}@upf.edu

ABSTRACT

In this work we study the problem of automatic musical instrument recognition from entire pieces of music. In particular, we present and evaluate 4 different methods to select, from an unknown piece of music, relevant excerpts in terms of instrumentation, on top of which instrument recognition techniques are applied to infer the labels. Since the desired information is assumed to be redundant (we may extract just a few labels from a thousands of audio frames) we examine the recognition performance, the amount of data used for processing, and their possible correlation. Experimental results on a collection of Western music pieces reveal state-of-the-art performance in instrument recognition together with a great reduction of the required input data. However, we also observe a performance ceiling with the currently applied instrument recognition method.

1. INTRODUCTION

Content-based Music Information Retrieval (MIR) aims at automatically extracting higher-level concepts from music data in order to enhance methods for an intelligent and user-friendly management of music collections. Here, information about the instrumentation plays a fundamental role in the semantic description of a music piece. Given the sizes of nowadays music archives, typical MIR applications such as indexing or retrieval demand for algorithms with low or moderate computational load. However, related literature in the field of automatic musical instrument recognition from polyphonies mostly concentrated on developing discrimination strategies, while disregarding aspects related to the computational complexity of the algorithms. Therefore, many approaches towards musical instrument recognition are costly and were designed for simplified test scenarios (e.g. [7, 8]). Furthermore, global properties of the music related to the instrumentation, which can help to reduce the amount of data to analyse and improve recognition robustness, were either only partially used or completely neglected (e.g. [1, 9, 10]). Moreover, most of the works incorporate restrictions such as reduced number of instruments, aseptic or limited data, and/or other a priori assumptions (e.g. [3, 6]).

In general, the auditory scene produced by a musical composition can be regarded as a multiple source environment, where the different sound sources – the musical instruments – are temporarily active, while often recurring along the piece. We therefore expect that the instrumentation's temporal evolution of a given music piece shows a repetitive character, so that the information related to the individual sources becomes redundant (we may extract a few labels from a thousands of audio frames). This suggests that, for automatic recognition systems, analysing only a fraction of the data is enough to extract the available information. Thereby the overall computational load of such algorithms is reduced which enables the implementation of fast recognition systems, indispensable for analysing big music collections. Moreover, this so-obtained data reduction can further be exploited by any other MIR related algorithm, e.g. music visualisation or summarisation.

In the present work we study the effect of data reduction on instrument recognition performance from entire music pieces for real world applications, e.g. music collection indexing. We thereby address two of the above-identified aspects lacking in the related literature, namely the development of both robust and efficient methods for automatic instrument recognition. In particular, we introduce and compare several track-level approaches, i.e. aimed to roughly assign labels to a whole track, which pre-process a given music piece to output a set of segments. Labels are then inferred from these segments using our previously presented

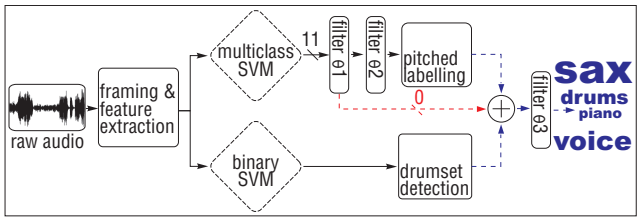

Figure 1. Graphical illustration of the label inference.

recognition method [5]. We further show that by applying this methodology we can significantly reduce the amount of data needed for analysis, while maintaining high recognition performance. In doing so we explore the redundancy of the information along a music track, and study the influence of locally obtained data on recognition, i.e. how much data needs to be extracted from which part of the track to obtain a sufficient description of its instrumentation.

Since our focus lies on developing approaches for real world applications, e.g. music collection indexing, we do not impose any restrictions on the input data, hence evaluating our approaches only on music pieces taken from real recordings. Furthermore, all information used in the labelling process is directly taken from the mixture signal without applying a priori information.

Below, we first present the basic methodology to extract instrumental labels from an unknown musical excerpt of arbitrary length (Sec. 2). We then give details about the different approaches to process entire pieces of music (Sec. 3), which is followed by a description of the data used in the experiments (Sec. 4). In Sec. 5 we define the evaluation metrics and present the obtained results. After a discussion, Sec. 6 concludes this article.

2. LABEL INFERENCE

Here we describe the basic process of extracting instrumental labels given an unknown audio excerpt of arbitrary length. First, the method sequentially applies previously trained predominant instrument classifiers to the audio. The resulting frame series is then analysed to extract the labels (Fig. 1).

2.1 Classification

To extract information about musical instruments from a short section of the audio signal we applied parts of the work previously presented in [4]. That is, our method uses statistical models of predominant musical instruments to estimate the presence of both pitched and percussive instruments for a 3-seconds excerpt of a polyphonic mixture signal. In particular, we applied the support vector machine (SVM) model [1] for 11 pitched instruments (*Cello*, *Clarinet*, *Flute*, *acoustic* and *electric Guitar*, *Hammond Organ*, *Piano*, *Saxophone*, *Trumpet*, *Violin*, and *singing Voice*) as developed in [4] ("multiclass SVM" in Fig. 1), and a separate model for estimating the presence of the drumkit ("binary SVM" in Fig. 1). Both SVMs output probabilistic estimates, i.e. a real value between 0 and 1, for each of the target classes. The models were trained with automatically pre-selected low-level audio features, describing the spectral and pitch related properties of the signal [2], extracted from proper training data. In particular, the features were computed frame-wise in the applied 3 second window, using a frame size of 46 ms with 50% overlap, and integrated over time via mean and variance statistics of the instantaneous and first difference values.

The training data itself consisted of 3 second excerpts containing predominant pitched target instruments, taken from more than 2,000 – presumably polytimbral – music recordings [4]. Besides for training the pitched instruments model, this collection was also annotated according to the presence of the drumkit, i.e. labels *drums* and *no-drums*, and used for constructing the percussive classifier.

2.2 Labelling

To extract labels of an audio signal of arbitrary length, the method first sequentially applies the above-described classifiers, using a hop size of 0.5 sec. The temporal behaviour of the obtained probabilistic time series is then exploited for label inference. Since the output of the pitched and the percussive model is merged (Fig. 1), we developed separate approaches corresponding to each of the two models for extracting the desired labels.

2.2.1 Percussive Instruments

First, a decision boundary of 0.5 is applied to binarize each prediction of the classifier. Then, a majority vote among all so-obtained binary decisions of the analysed signal is performed to indicate the target label. The corresponding confidence value is set to the relative amount of positive binary decisions.

2.2.2 Pitched Instruments

The method first uses the mean values of each instrument's probabilistic curve along the analysed audio to determine those instruments for label analysis. Thereby a threshold θ_1 is applied to these mean values; if all of them fall below the threshold, the whole audio under analysis is skipped and not labelled at all, indicating a potential confusion due to unknown or heavily overlapped instruments. If approved, a second threshold θ_2 is applied to the mean values; if an

[1] We used the libSVM implementation, available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm/`.

[2] A complete list of all applied audio features can be accessed under `http://mtg.upf.edu/system/files/publications/ismir11_ffuhrmann_sup.pdf`.

instrument falls below this threshold, it is regarded as inactive and not used in the further analysis. The probabilistic curves of the remaining instruments are then searched for sections, where a single instrument predominates the mixture, i.e. it holds the highest probability value among all instruments for a certain minimal amount of time. If such a section is found, the corresponding instrument is added to the list of labels for the analysed audio, along with a confidence value as defined by the section's length relative to the overall length of the audio [3] . This process is repeated for all determined active instrument. Finally, a label threshold θ_3 is applied to discard unreliable tags. Fig. 2 exemplifies the labelling process for a 30 second excerpt.

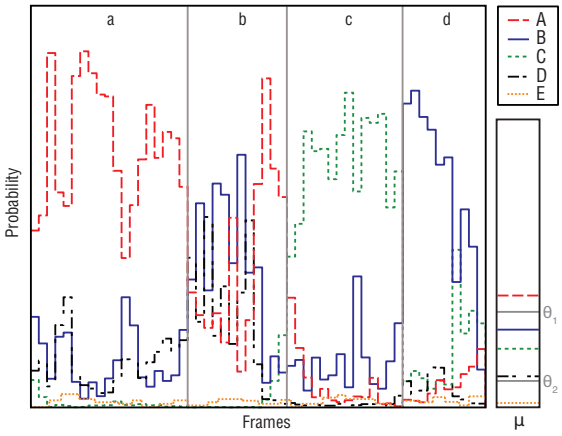

Figure 2. An example of the labelling method for pitched instruments. The main figure shows the probabilistic estimates for sources A-E, the right panel the mean values together with the thresholds used for instrument filtering. Since E is discarded as its mean value falls below θ_2, the curves A-D are scanned for sections, where a single instruments predominates. Depending on the parameter for the minimal length of these sections, up to three different instruments can be detected here (a,c,d → A,C,B), whereas sections containing instrument confusions are not used for labelling (b).

3. TRACK-LEVEL APPROACHES

In this section we present 4 different approaches to process and label an entire piece of music. Since the instrumentation and its temporal evolution of a piece of music usually follows a clear structural scheme, we expect, inside a given music track, a certain degree of repetitiveness of its different instrumentations. This property of music and the resulting redundancy is exploited by the described approaches to reduce the amount of data to process. We then apply the label inference method described in Sec. 2 on their respective output and evaluate the algorithms in terms of labelling performance and the amount of used data. In short, the presented approaches are accounting – some of them more than others – for the time-varying character of instrumentation inside a music piece. Their output consist of a set of segments which are then used to infer the instrumental labels for the given music track. Fig. 3 depicts the underlying ideas.

[3] For multiple occurrences of the same instrument the respective confidence values are summed.

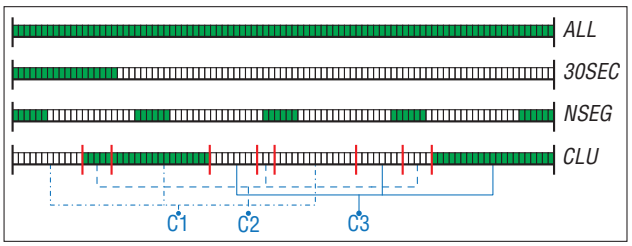

Figure 3. Illustration of the presented track-level approaches; the green filled frames denote the respective data used for labelling. Segmentation (red) and clustering (blue) are indicated for the *CLU* method, while *NSEG* applies a value of $n = 5$. See text for details.

3.1 All-frame processing (*ALL*)

Probably the most straightforward approach given the above-described labelling methodology. By processing all frames we automatically account for the time-varying character of musical instruments via a global analysis of the track. However, no data reduction is performed. Since this approach uses all data available, it acts as a kind of upper baseline both in terms of recognition performance and amount of data processed, which all other methods using less data compete with.

3.2 30 seconds (*30SEC*)

This widely used approach in MIR assumes that by reducing the data to 30 sec of audio most of the semantic information is maintained. Many genre, mood, or artist classification systems use an excerpt of this length to represent an entire music track (e.g. [11]). The process can be regarded as an extrapolation of the locally obtained information to the global scope, i.e. the entire piece of music. Since the aforementioned concepts are rather stable across one single piece, the data reduction does not affect the significance of the obtained results. However, instrumentations usually change with time, so that the targeted information is inadequately represented by this data amount. In our experiments we extracted the data from 0 to 30 sec of the track.

3.3 Segment sampling (*NSEG*)

Here, we obtain excerpts by uniformly sampling the track to incorporate the time-varying characteristics of instrumentation. This enables a local extraction of the information which is combined to a global estimate of the instrumental labels. In particular we extract n equal-distant excerpts of 10 seconds length from the track (for n equals 1 or 2 a single segment from the beginning, or one segment from the beginning and the end of the music track is taken, respectively). The labels inferred from each of the segments are then merged, where small values of n lead to a great data reduction while still considering the instrumentation's time-varying character. The parameter n is kept variable for the experiments conducted in Sec. 5.

3.4 Cluster representation (*CLU*)

Certainly the most elaborated approach from the perceptual point-of-view; a given piece of music is represented with a cluster structure where each cluster corresponds to a different instrumentation. This approach explicitly uses an estimate of the global distribution of the musical instruments to locally infer the labels from a reduced set of the data by exploiting redundancies of the instrumentations inside the piece of music. In particular, it applies unsupervised segmentation and clustering algorithms to locate the different instrumentations and their repetitions. At the end, only one segment per cluster is taken for further analysis. Hence this approach is directly exploiting repetitions in the instrumentation to reduce the amount of data to process, while the local continuity of the individual instruments is preserved to guarantee a maximum in recognition performance.

3.4.1 Segmentation

Since instrumentation is closely related to timbre, a timbral representation of the track is processed to find local changes therein, applying an unsupervised segmentation algorithm based on the Bayesian Information Criterion (BIC) [2]. To represent timbre the approach uses 13 frame-wise extracted Mel Frequency Cepstral Coefficients (MFCCs).

3.4.2 Clustering

Here, an agglomerative clustering step builds a hierarchical tree (i.e. a so-called dendrogram) on the pair-wise similarities of all generated segments. The segments are merged iteratively to form the tree, where a linkage method further measures proximities between groups of segments at higher levels [12]. The final clusters are then found by cutting the tree according to an inconsistency coefficient, which measures the compactness of each link in the tree. Furthermore, to estimate the pair-wise segment similarities, we model each segment as a single Gaussian distribution of the raw MFCC frames with diagonal covariance matrix and calculate the symmetric Kullback-Leibler divergence (KL) between pairs of segments.

Finally, the longest segment of each resulting cluster is passed to the label inference algorithm. The predictions from all segments are then merged to form the set of labels for the track under analysis.

4. DATA

For our experiments we used a data corpus consisting of 220 music pieces taken from various genres of Western music. In these tracks, all perceptually audible instruments were annotated manually along with their start and end times. Since no limitations in the vocabulary size were imposed to the human annotators, this evaluation data includes, additional to the 12 modelled classes, instruments which are not modelled by the classifier. Moreover, if the annotator could not recognize a certain instrument's sound, the label *unknown* was used [4] .

An analysis of the set of labels used in the annotations revealed 28 different instrumental categories, at which the label *unknown* was the third-most frequently used, directly after the labels *bass* and *drums*. It should be noted that none of the tracks used for training the instrumental models was used in this evaluation collection.

5. GENERAL RESULTS

5.1 Metrics

To estimate the labelling performance we regarded the problem as multi-class, multi-label classification. That is, each instance to evaluate can hold an arbitrary number of unique labels of a given dictionary. Given a collection of music tracks $X = \{x_i\}, i = 1 \dots N$, with N items, we define, respectively, $Y = \{\hat{y}_i\}, i = 1 \dots N$, and $\tilde{Y} = \{\tilde{y}_i\}, i = 1 \dots N$, the set of ground truth and predicted labels for each x_i. Together with the label dictionary $L = \{l_i\}, i = 1 \dots M$, we define the weighted precision and recall metrics,

$$P = \frac{1}{\sum_{l,i} \tilde{y}_{l,i}} \sum_{l,i} \tilde{y}_{l,i} \cdot \hat{y}_{l,i}, R = \frac{1}{\sum_{l,i} \hat{y}_{l,i}} \sum_{l,i} \tilde{y}_{l,i} \cdot \hat{y}_{l,i}, \quad (1)$$

where $\hat{y}_{l,i}$ ($\tilde{y}_{l,i}$) represents a boolean variable indicating the presence or absence of the label l in the annotation (generated instrumental tags) of track i. Additionally, we define an F-measure to estimate the overall labelling performance,

[4] A complete list of all tracks contained in the evaluation dataset, along with the annotated instruments and genre labels, can be accessed via `http://mtg.upf.edu/system/files/publications/ismir11_ffuhrmann_sup.pdf`.

$$F = \frac{2\sum_{l,i} \tilde{y}_{l,i} \cdot \hat{y}_{l,i}}{\sum_{l,i} \tilde{y}_{l,i} + \sum_{l,i} \hat{y}_{l,i}}. \quad (2)$$

5.2 Results

In order to provide a robust estimate of the methods' performance with respect to the parameters to evaluate, we performed a 3-fold Cross Validation (CV). For each turn we used the data of 2 folds for estimating the optimal parameter settings and subsequently tested on the remaining fold. We then obtained mean values and corresponding standard deviations by averaging the evaluation results of the respective predictions of all three runs [5] .

The upper panel of Table 1 contains the results (mean values) of the CV obtained for the studied algorithms. The parameter n of the *NSEG* method was set to 3 and 6, generating systems processing 30 sec (*3SEG* – an equivalent in terms of data size to the *30SEC* method) and 1 min of audio data (*6SEG*). Additionally, figures regarding the relative amount of data used for label inference are shown in the lower panel (relative with respect to the all-frame processing algorithm *ALL*). A lower bound was generated by drawing each label from its respective prior binomial distribution, inferred from all tracks of the collection, averaging the resulting performance over 100 independent runs (*PRIOR*).

Table 1. *Precision, recall, and F measures of the studied approaches together with the relative amount of data used for label inference (data). The asterisk indicates average values over 100 independent runs.*

	*PRIOR**	*30SEC*	*3SEG*	*6SEG*	*CLU*	*ALL*
P	0.4	0.62	0.64	0.60	0.64	0.66
R	0.4	0.5	0.6	0.71	0.74	0.73
F	0.4	0.55	0.62	0.65	0.69	0.69
data	–	0.11	0.11	0.25	0.66	1

The figures presented in Table 1 show that all considered approaches are outperforming the prior baseline *PRIOR*, operating well above a knowledge-informed chance level. Moreover, two clear dependencies of the resulting performance can be observed; first, a correlation with the absolute amount of data processed (e.g. *3SEG* → *6SEG* → *ALL*), and second, a dependency on the location where the information is extracted (*30SEC* → *3SEG*).

Comparing the sampling methods with the timbre analysis of *CLU* we can see that the knowledge introduced by the latter positively affects the recognition performance. Besides the greater values of *R* and *F*, the precision *P* is remarkable here, which holds the same value as for the *3SEG* method, although *CLU* processes 55 percent points more data. The segmentation and clustering preserves the temporal continuity of the instrumentation, therefore exhibiting less data variability, ensuring the high value of the *P* metric. The same local continuity of musical instruments otherwise enforces the lower recall value in the *30SEC* approach, in comparison to the *3SEG* method. However, with more analysed segments from different parts of the track, the variation in the data increases. This affects the recall value *R*, resulting in a trade-off between the two aforementioned metrics.

Furthermore, the similar performance figures of the *CLU* and *ALL* approaches suggest that there exists a minimal amount of data from which all the extractable information can be derived [6] . Hence more data will then not result in an improvement of the labelling performance. The next section will examine this phenomenon in more detail, in particular by determining the minimum of audio data required to maximize labelling performance.

5.3 Scaling and computational aspects

The observations in the previous section suggest that there seems to be a strong amount of repetitiveness present inside a music piece. Additionally, many excerpts – even though differing in instrumentation – produce the same label output when processed with the used label inference method. To quantify those effects we used the *CLU* and *NSEG* methods to process the entire piece under analysis, as both offer a straightforward way to vary the amount of data used by the label inference algorithm. In particular, we studied the effect of an increasing amount of segments to process on the labelling performance. In case of the *NSEG* method we constantly increased the amount of segments used by the label inference, thus augmenting the method's parameter n. For the *CLU* method we sorted the clusters downwards by the accumulated length of their respective segments, started processing just the first one, and iteratively added the next longest cluster. For both methods we then tracked the performance figures as well as the amount of data used for inference. Fig. 4 depicts both performance and amount of data for the first 20 steps on the evaluation data (mean value of CV outputs).

As can be seen from Fig. 4 the performance of both *CLU* and *NSEG* systems stagnates at a certain amount of segments processed. Due to the different amount of data processed, those values represent, respectively, 3 and 5 segments. Hence, incorporating global timbral structure, as implemented by *CLU*, benefits labelling performance at the ex-

[5] Parameter estimation itself was performed via a grid search procedure over the relevant parameter space. For each of the studied approaches described in Sec. 3 the parameters were evaluated separately to guarantee maximal comparativeness of the respective results.

[6] The small differences in *P* and *R* result from individual parameter settings, estimated by the CV by determining the best performing configuration in respect to the *F* metric, and can be compensated by manually choosing proper values. However, the F metric would not be affected, since there will always be a trade-off between *P* and *R*.

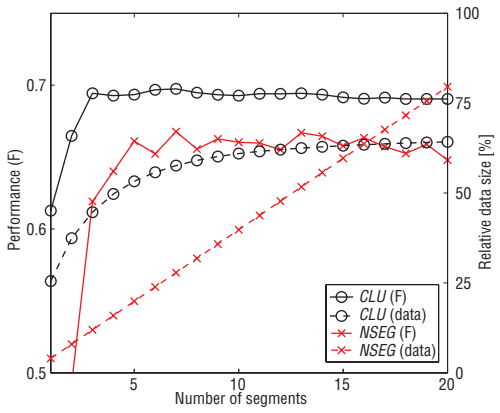

Figure 4. Scaling properties of the studied algorithms. Solid lines refer to the respective labelling performance in terms of *F*, dashed ones show the respective data amount used for label inference, relative to the maximum as produced by *ALL*. Mean values across CV-Folds are shown.

pense of algorithmic pre-processing. By preserving the continuity of musical instruments the method shows a slightly superior performance compared to *NSEG*, which segment extraction is unaware of any contextual properties. In terms of the used data amount, *NSEG* is superior whilst processing less than around 40% of the data (i.e. $n \leq 10$), whereas when processing more, *CLU* returns the better overall labelling performance. However, the results suggest that, on average, a timbre-informed clustering does not result in a significant increase in performance, thus it might be of advantage in specialized applications (e.g. working on a single genre which exhibits clear recurrent structural sections).

Finally, the stagnation of labelling performance indicates a kind-of "glass ceiling" that has been reached. It seems that with the presented classification and labelling methodology we are not able to extract more information about the instrumentation. Nevertheless, we can observe that predominant instrumental information is highly redundant inside a given Western piece of music from which 70% of the labels can be obtained. Furthermore, this fact allows for a reduction of the effective amount of data used for label inference of around 55%. Remarkably, the same factor of about $1/2$ can also be observed when comparing the number of different instrumentations to the overall number of segments in the ground truth annotations of all files in the used music collection.

6. CONCLUSIONS

In this article we studied the problem of extracting labels corresponding to the instrumentation from entire pieces of music. We designed our approach to be applied in a real world context, hence the presented methods work on any piece of music, without imposing restrictions to the input data. In particular we analysed different methods to preprocess the entire tracks, studying the effect of data reduction on recognition performance. Evaluation on a dataset of 220 musical pieces showed that by using the best performing approach we are able to score a global F-measure of 0.69 while examining 12 musical instruments. On the other hand, a proper preprocessing of the data allows for a reduction of the amount of data used for label inference of more than 50% while the recognition performance is preserved.

7. ACKNOWLEDGEMENTS

This work has been supported by the following projects: Classical Planet: TSI-070100- 2009-407 (MITYC) and DRIMS: TIN2009-14247-C02-01 (MICINN).

8. REFERENCES

[1] J. Burred, A. Robel, and T. Sikora. Dynamic spectral envelope modeling for timbre analysis of musical instrument sounds. *IEEE Transactions on Audio, Speech, and Language Processing*, 18(3):663–674, 2010.

[2] S. Chen and P. Gopalakrishnan. Speaker, environment and channel change detection and clustering via the bayesian information criterion. In *Proc. of the DARPA*, 1998.

[3] S. Essid, G. Richard, and B. David. Instrument recognition in polyphonic music based on automatic taxonomies. *IEEE Transactions on Audio, Speech, and Language Processing*, 14(1):68–80, 2006.

[4] F. Fuhrmann, M. Haro, and P. Herrera. Scalability, generality and temporal aspects in the automatic recognition of predominant musical instruments in polyphonic music. In *Proc. of ISMIR*, 2009.

[5] F. Fuhrmann and P. Herrera. Polyphonic instrument recognition for exploring semantic similarities in music. In *Proc. of DAFx*, 2010.

[6] T. Heittola, A. Klapuri, and T. Virtanen. Musical instrument recognition in polyphonic audio using source-filter model for sound separation. In *Proc. of ISMIR*, 2009.

[7] T. Kitahara, M. Goto, K. Komatani, T. Ogata, and H. Okuno. Instrogram: A new musical instrument recognition technique without using onset detection nor f0 estimation. In *Proc. of ICASSP*, 2006.

[8] P. Leveau, D. Sodoyer, and L. Daudet. Automatic instrument recognition in a polyphonic mixture using sparse representations. In *Proc. of ISMIR*, 2007.

[9] A. Livshin and X. Rodet. Musical instrument identification in continuous recordings. In *Proc. of DAFx*, 2004.

[10] S. Pei and N. Hsu. Instrumentation analysis and identification of polyphonic music using beat-synchronous feature integration and fuzzy clustering. In *Proc. of ICASSP*, 2009.

[11] N. Scaringella, G. Zoia, and D. Mlynek. Automatic genre classification of music content: A survey. *IEEE Signal Processing Magazine*, 23(2):133–141, 2006.

[12] R. Xu and D. Wunsch. *Clustering*. IEEE Press Series on Computational Intelligence, 2008.