Representation of Musical Signals using Instrument-Specific Dictionaries

Mohammadali Azamian
Department of Electrical and Computer Engineering
Tarbiat Modares University, Iran

ABSTRACT

A new simple method is proposed to synthesize the instrument-specific dictionaries and its use is examined in the time domain musical signal representation. By investigating the spectrum of musical note signals, it is seen that only a small number of frequency elements are significant in the inherent structure of a musical note, and other elements could be omitted. This sparsity is utilized to synthesize note-specific atoms. Firstly, some basic functions are defined from the long-term spectrum of the note signal, called primary atoms. Then the primary atoms that satisfy some conditions are selected as basic atoms and are incorporated to synthesize note-specific atoms. Some usual signal processing windows also are examined such as Gaussian and Hamming windows to synthesize note-specific atoms. The note-specific atoms of an instrument are integrated in an instrument-specific dictionary. A musical signal is represented by mapping to this dictionary by means of the Matching Pursuit algorithm. The proposed method was evaluated on the RWC musical sound database. The results showed that it improves the quality of signal representation compared to some previous methods.

Keywords

Signal Representation, Signal Mapping, Audio Signal Processing, Spectral Analysis, Signal Reconstruction

1. INTRODUCTION

Considering the large size of multimedia data such as sound and image, storing their basic information in low volume seems necessary. An important challenge is to preserve the useful information while non-important data is removed [1].
The basic information of different sounds is less in common with one another. Therefore, the difference between the main information of sounds causes discrepancy among them. The basic information of sounds can be considered in the form of a limited number of the basic functions. These basic functions are called atoms and a group of these atoms is called dictionary [2].

The time structure of the sound sources could be obtained through the learning of a group of functions in time domain, so that these basic functions encode the signal perfectly using statistical methods [3]. First, some particular basic functions are chosen. Then, the weights of these functions are determined using the Maximum Likelihood (ML) method. Next, each sound source is modeled as a weighted linear summation of the basic functions (see Figure 1).

This method leads to a sparse representation of sources only if each basis function has a high correlation with one of the sources and low correlation with others. Otherwise, the sparse representation will not occur.

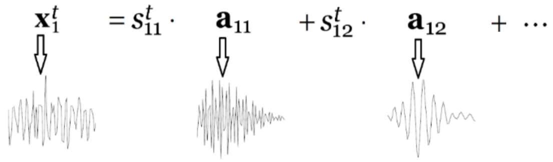
$\mathbf{x}_1^t = s_{11}^t \cdot \mathbf{a}_{11} + s_{12}^t \cdot \mathbf{a}_{12} + \cdots$

Fig 1: Learning the basic functions. a_{ij} is the basic function j of the source i and s_{ij} is its corresponding weight learned from data X [3]

Signal representation using sparse dictionaries are considered in several recent researches as an efficient method for different audio processing issues such as audio structure analysis, automatic music transcription, and audio source separation [4-9].

The sparse representation of music signals using source-specific dictionary, has been proposed in [10]. The main purpose in their research is to separate music signals from background signals or speech. To achieve this, a source-specific dictionary is formed for each musical source using some synthesized atoms. Then, the mixed signal is decomposed to the atoms of this dictionary, and thus, only those parts of the signal having high correlation with the related atoms are extracted and background signals or speech are omitted. The selection and extraction of suitable atoms for representing the signal is done by means of the Matching Pursuit (MP) algorithm [11].

The main goal of this study is to propose a new simple method of synthesizing some time-domain function, called note-specific atoms, which improve the efficiency of musical signal representation. In the proposed method, the structural elements of a musical note are extracted by signal analysis in the frequency domain, which are then used for synthesizing the time-domain note-specific atoms. Next, an instrument-specific dictionary is made by collecting the note-specific atoms. This dictionary is used for musical signal representation. The proposed method is experimented for the test signals produced from the Piano, Clarinet, Classical Guitar and Violin.

2. SYNTHESIZING NOTE-SPECIFIC ATOMS

The proposed method in synthesizing note-specific atoms consists of three main steps, as depicted in Figure 2, which are described in the following subsections. We used this method also in audio source separation [12].

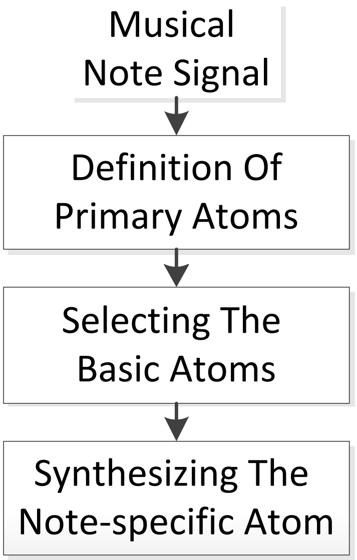
Musical Note Signal
Definition Of Primary Atoms
Selecting The Basic Atoms
Synthesizing The Note-specific Atom

Fig 2: The steps of synthesizing note-specific atoms

2.1 Definition of the Primary Atoms

The primary atoms are defined from the FFT components of the note signal as:

$$h_{A,f,\emptyset}(n) = A\cos(2\pi f n + \emptyset)\,;\, 0 \le n \le (N-1) \qquad (1)$$

where A, f and φ are the amplitude, frequency and the phase of an FFT component of the note signal, respectively, n is the sample number, and N is the arbitrary length of a primary atom. In order to avoid the FFT components in reverse phase due to the symmetry, the number of primary atoms is set to half the length of the original signal.

2.2 Selecting the Basic Atoms

The note-specific are synthesized atoms using only the primary atoms corresponding to the main components of the frequency spectrum, called basic atoms. These atoms contain the main inherent structure of the note signal. A primary atom which is a candidate for being a basic atom should satisfy two conditions: First, it should be greater than the lateral atoms within a predetermined neighborhood, and second, the average of its lateral atoms in the specified neighborhood must be greater than a lateral threshold. These conditions can be summarized as follows:

$$A_n > A_m; \qquad n - L \le m \le n + L \qquad (2)$$

$$\sqrt{\sum_{n-L}^{n+L} A_i^2} \; > \; TH_L \qquad (3)$$

in which A_n is the amplitude of the primary atom, TH_L is the lateral threshold, and L is the neighborhood distance from each side.

By checking the first condition, only one atom having the highest peak in a frequency neighborhood is selected, resulting in sparsity. For example, consider the spectrum details of the piano note C4 around 263 Hz as depicted in Figure 3. By checking this condition, the primary atoms around 263 Hz are omitted, which although their amplitude is large enough, they are smaller than an atom in their neighborhood. So, only one atom is selected around 263 Hz. The second condition should be satisfied to ensure that the signal energy is sufficient around the basic atom. So, we check if the integration of amplitudes around the test atom is greater than a lateral threshold (TH_L). Note that before applying the algorithm, the spectrum must be normalized so that the total energy of spectra is 1, i.e.:

$$\sum_i A_i^2 \; = \; 1 \qquad (4)$$

After extracting the atoms in peaks, they are sorted according to their amplitude and use the first M atoms of this list as the basic atoms. The parameters TH_L, L and M are assigned adaptively to yield best representation performance.

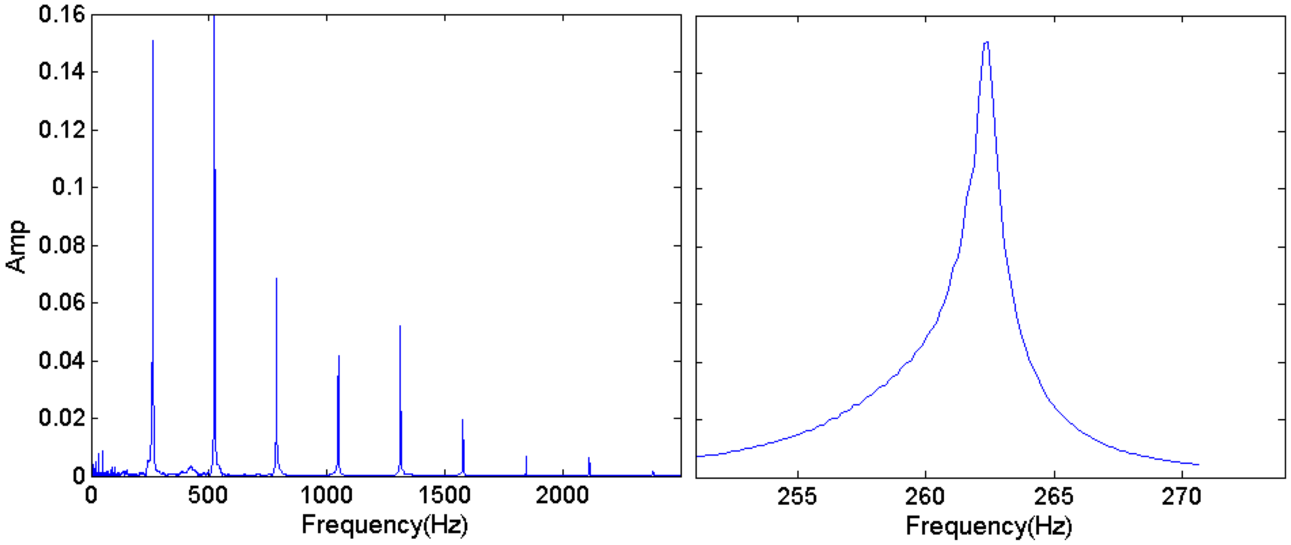

Fig 3: The spectrum of piano note C4 and its detail around 263 Hz

2.3 Synthesizing the note-specific atoms

The note-specific atoms is calculated as:

$$H(n) = k\,w(n)\sum_{m=1}^{M} h_m\ (n) \qquad (5)$$

where H(n) is a note-specific atom, w(n) is a signal processing window, M is the number of basic atoms, $h_m(n)$ is the mth basic atom chosen from primary atoms as described in the previous section, and k is the normalizing factor such that the total energy of note-specific atom is 1.

Common signal processing windows can be used to synthesize note-specific atoms. Hamming, Gaussian and Rectangular windows are examined, and according to the experiments, using either Hamming or Gaussian windows improve the efficiency of signal representation.

3. MUSIC SIGNAL REPRESENTATION

Musical signal representation is done by mapping the signal to the instrument-specific dictionaries. The instrument-specific dictionary is composed of a set of note-specific atoms of each instrument. Decomposition of instrument music signal into note-specific atoms is done by the MP algorithm [11], as described in Algorithm 1, where an input signal X(t) is decomposed into the atoms of dictionary D in Q iterations. The existing atoms in the dictionary D are expressed as G_i. In the beginning, the initial residual signal R_0 is set equal to X. Then decomposition is performed for Q iterations. At iteration n, the selected atom G_n is the atom with the highest correlation with the last residual signal R_{n-1}. To update the new residual R_n, the selected atom weighted by the correlation factor is subtracted from the last existing residual signal.

Algorithm 1: MP algorithm for signal decomposition [11]

```
$R_0(t) = X(t);$
**For** $n=1:Q$
    $G_n(t) = \underset{G_i \in D}{argmax} |\langle R_{n-1}, G_i\rangle|;$
    $R_n(t) = R_{n-1}(t) - \langle R_{n-1}, G_n\rangle * G_n(t);$
**End For**
```

After decomposition, the original signal could be estimated as:

$$X'(t) = \sum_{n=1}^{N} \langle R_{n-1}, G_n\rangle\, G_n(t) \qquad (6)$$

It is clear that the original signal will be:

$$X(t) = X'(t) + R_N(t) \qquad (7)$$

After decomposition, the Signal to Distortion Ratio (SDR) can be calculated as:

$$SDR = 10\log\frac{\|X\|}{\|X' - X\|} \qquad (8)$$

It is noted that the dictionary used by the MP algorithm implicitly includes all possible time-delayed functions of an atom. Generally, it is assumed that once an atom is included in a dictionary, all atoms resulted from its time-delayed functions also exist in that dictionary.

4. EXPERIMENTS

The RWC musical instrument sound dataset [13] is used to evaluate our proposed method. In that dataset, there are three variations of different musical instruments notes. Two variations are used for building the instrument-specific dictionary and one to make the test signals.

4.1 Synthesizing note-specific atoms

The note-specific atoms are synthesized for two variations of piano, clarinet, classical guitar and violin notes in the RWC dataset using our method. The sampling rate in the dataset is 44100 samples per second. Since there are no frequency elements above 5 kHz in the used note signals, according to the Nyquist theorem, a sample rate of 10000 would be sufficient, and thus the sample rate is reduced to 25% of the original, i.e. 11025 samples per second for the experiments.

The lengths of the used note signals were lower than 65536 samples after down-sampling. Therefore, to calculate the long term spectrum, the lengths of all note signals are equalized to 65536 samples by zero-padding, resulting in a spectrum resolution of 0.168 Hz. After computing the spectrum, it was normalized so that the total energy is 1.

Subsequently, the basic atoms of each note signal spectrum was extracted. A sliding window sweeps the spectrum and the atoms which satisfy the conditions are extracted and sorted according to their amplitudes. By investigating the note signal spectrums, it is observed that the minimum distance between the peaks was around 50 Hz. On the other hand, there was no considerable frequency element in more than 10 Hz far from the peaks. If the neighborhood around 30 Hz is select from each side, there would be no overlaps and no frequency element would be lost. So, considering the FFT resolution of 0.168 Hz, the parameter L is selected as 200. Also, there were not more than 5 effective peak elements in any spectrum. So the parameter M is set as 5. A wide range of values is tried for the parameter TH_L. The best results in the representation stage were achieved when we used 0.038 for TH_L. So the proposed algorithm is applied to all note signals using the parameter set 0.038, 200 and 5 for TH_L, L, and M, respectively

Figure 4 shows the note-specific atom, computed for the piano note C4 using different windows for length of 1024 samples. It is seen that the Gaussian and Hamming windows represent better characteristics in the frequency domain. However, based on our experimental results, no preference is observed in choosing these two windows.

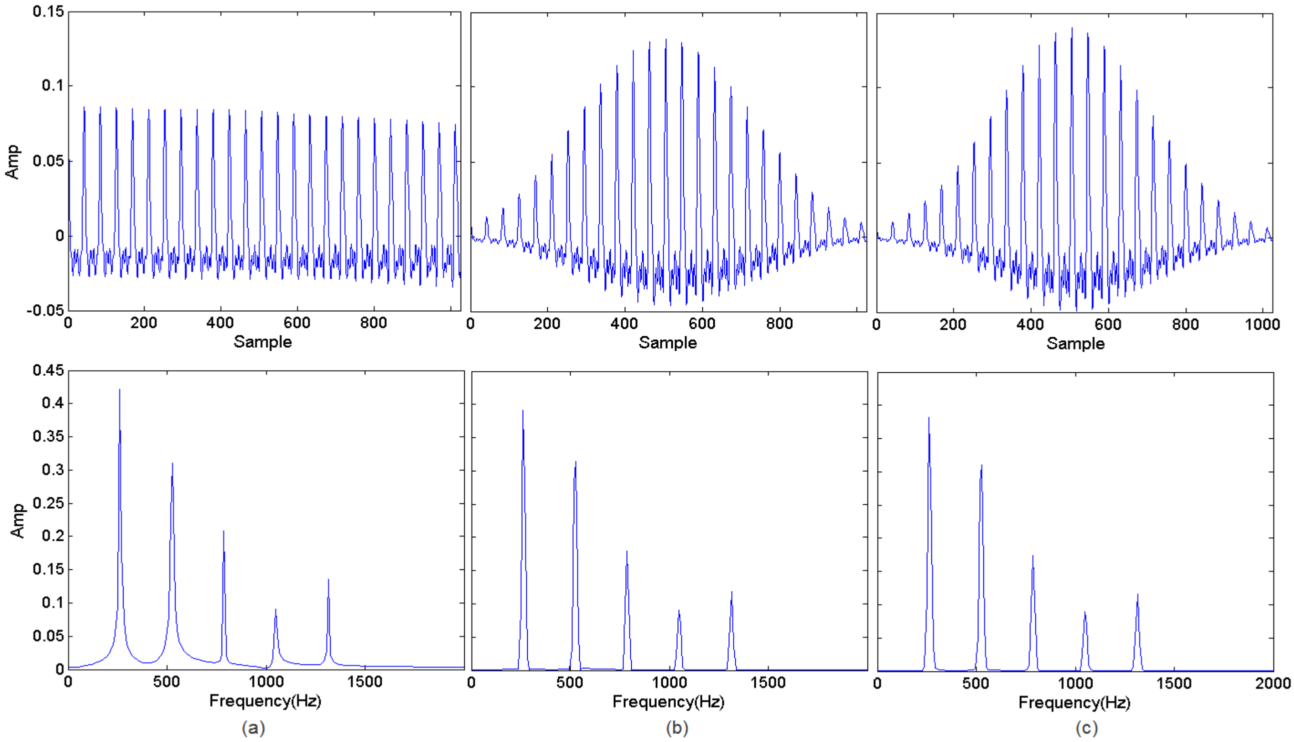

Fig 4: Atoms obtained for the Piano note C4: a) Rectangular window, b) Hamming window, c) Gaussian

4.2 Constructing the instrument-specific dictionaries

The instrument-specific dictionaries are constructed for the piano, clarinet, classical guitar and violin. The lengths of the note-specific atoms were selected as N = 512, 1024, and 2048. Our observations in the experiments demonstrated that the best representation efficiency can be typically achieved in lengths of 2048 for lower frequencies, and in lengths of 512 for higher frequency notes. So, the lengths of 512, 1024 and 2048 are used to synthesize note-specific atoms.

For each instrument, distinct dictionaries are constructed using different windows. Two variations of dataset are used as training data for each note signal and synthesized note-specific atoms in three lengths. So, for each note there are six note-specific atoms in each instrument-specific dictionary. For comparisons, source-specific dictionaries also are constructed (SSD) [10]. A Gabor atom dictionary is used to extract new atoms. To perform a fair comparison, all these new atoms were synthesized in three different lengths for two training data. The instruments-specific and the source-specific dictionaries were used for evaluation of the proposed method.

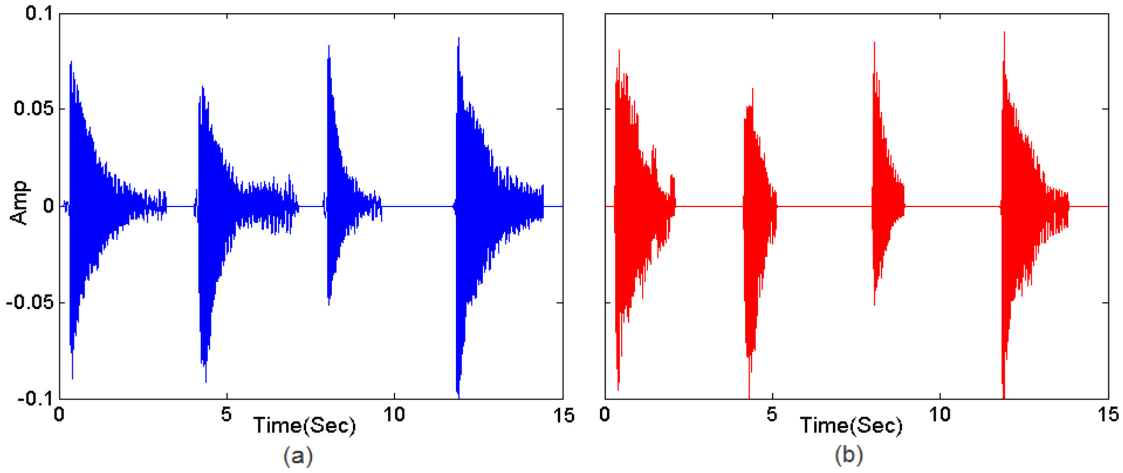

Fig 5: a) A piano test signal, b) Its representation using hamming window

4.3 Representation of musical signals

The constructed dictionaries are used to represent synthesized music signals of four different instruments. Test signals are produced by weighted summation of note signals for each instrument. Figure 4 shows the note-specific atom, computed for the piano note C4 using different windows for length of 1024 samples. It is seen that the Gaussian and Hamming windows represent better characteristics in the frequency domain. However, based on our experimental results, we did not observe any preference in choosing these two windows.

The efficiency of representation is calculated using (8). The proposed algorithm is evaluated, comparing to SSD method [10]. The average SDR of test signal representation is shown in Table 1. According to the results, the proposed method demonstrates a better efficiency. Also, in [10], the MP algorithm was used to synthesize note-specific atoms, which is a complicated process. However, in our method there is no need to use that algorithm for making an instrument-specific dictionary, and note-specific atoms are computed with a fast and simple process in the frequency domain.

As we aimed to propose a new source-dependent method for music representation, the results are compared with source-specific dictionaries as a relevant baseline, and showed that our algorithm outperforms the SSD method. Thus, comparison with source-independent methods is out of the scope of this paper, since this comparison has been done previously and the effectiveness of source-dependent algorithms has been proved in [10].

Table 1. The Average SDR of Reconstructed Signals

Instrument	**Average SDR**			
	SSD	**Proposed Method**		
		Rectangular Window	**Gaussian Window**	**Hamming Window**
Piano	8.7	10.1	11.3	11.5
Clarinet	12.5	12.3	13.3	13.3
Violin	12.8	11.4	15.1	15.1
Guitar	13.3	12.8	16.2	16.3

5. CONCLUSION

In this paper, a novel method is proposed for constructing the instrument-specific dictionaries, which was used for music signal representation. The inherent time structures of the musical note signals are extracted by spectrum analysis and used them to synthesize note-specific atoms. These atoms were integrated to the instrument-specific dictionary and musical signals were represented by mapping to this dictionary.

Compared to the method proposed in [10], a better quality of musical signal representation was achieved in the SDR criterion. The proposed method is also simpler in making an instrument-specific dictionary.

As a future work, conceptual structure of note signals can be used in synthesizing note-specific atoms. Quality of signal representation could be enhanced using this feature.

6. REFERENCES

[1] D. P. W. Ellis, 2006. "Extracting information from music audio," Communications of the ACM, vol. 49, no. 8, p. 32.

[2] N. Cho, C. C. J. Kuo, 2010. "Sparse representation of musical signals using source-specific dictionaries," IEEE Signal Processing Letters, vol. 17, no. 11, pp. 913-916. doi:10.1109/LSP.2010.2071864

[3] G. J. Jang, T. W. Lee, Y. H. Oh, 2003. "Single-channel signal separation using time-domain basis functions," IEEE Signal Processing Letters, vol. 10, no.6, pp. 168-171. doi:10.1109/LSP.2003.811630

[4] H. Huang, J. Yu, W. Sun, 2014. "Super-resolution mapping via multi-dictionary based sparse representation," Int. Conf. on Acoustics Speech and Signal Processing, IEEE, Florence, pp. 3523-3527.

[5] Y. Xu, G. Bao, X. Xu, Z.Ye, 2015. "Single-channel speech separation using sequential discriminative dictionary learning," Signal Processing, vol. 106, pp. 134–140. doi:10.1016/j.sigpro.2014.07.012

[6] M. Yaghoobi, T. Blumensath, M. E. Davies, 2009. "Dictionary learning for sparse approximations with the majorization method," IEEE Transactions on Signal Processing, vol. 57, no. 6, pp. 2178–2191. doi:10.1109/TSP.2009.2016257

[7] T. Blumensath and M. Davies, 2006. "Sparse and shift-invariant representations of music," IEEE Transactions on Audio, Speech and Language Processing, vol. 14, no. 1, pp. 50–57. doi:10.1109/TSA.2005.860346

[8] S. A. Abdallah and M. D. Plumbley, 2006. "Unsupervised Analysis of Polyphonic Music by Sparse Coding," IEEE Transactions on Neural Networks, vol. 17, no. 1, pp. 179–196. doi:10.1109/TNN.2005.861031

[9] Y. Vaizman, B. McFee, G. Lanckriet, 2014. "Codebook-based audio feature representation for music information retrieval," IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 22, no. 10, pp. 1483-1493. doi:10.1109/TASLP.2014.2337842

[10] N. Cho, C. C. J. Kuo, 2011. "Sparse music representation with source-specific dictionaries and its application to signal separation," IEEE Transactions on Audio Speech Lang. Process., vol. 19, no. 2, pp. 337-348. doi:10.1109/TASL.2010.2047810

[11] S. G. Mallat, Z. Zhang, 1993. "Matching pursuit with time-frequency dictionaries," IEEE Transactions on Signal Processing, vol. 41, no. 12, pp. 3397–3415. doi:10.1109/78.258082

[12] M. Azamian, E. Kabir, S. Seyedin, E. Masehian, 2017. "An adaptive sparse algorithm for synthesizing note-specific atoms by spectrum analysis, applied to musical signal separation," Advances in electrical and computer engineering, vol. 17, no. 2, pp. 103-112. doi:10.4316/AECE.2017.02014

[13] M. Goto, H. Hashiguchi, T. Nishimura, R. Oka, 2003. "RWC music database: musical instrument sound database," ISMIR, pp. 229-230.