# PAPER TEMPLATE FOR ISMIR 2016

**First author**
Affiliation1
author1@ismir.edu

**Second author**
Affiliation2
author2@ismir.edu

**Third author**
Affiliation3
author3@ismir.edu

**Fourth author**
Affiliation4
author4@ismir.edu

## ABSTRACT

The abstract should be placed at the top left column and should contain about 150-200 words.

## 1. INTRODUCTION

*Source separation* is field of research that seeks to separate the components of an audio signal present in a record. Such separation has many applications in music : noise suppression [**?**] (if a source is noise) , up-mixing [**?**](spatialization of the sources) or automatic transcription [**?**] (it is easier to operate on single source). The task is difficult due to the complexity and the variability of the music mixtures. Most datasets used for Blind Source Separation (BSS) research are small in size and they do not allow for a thorough comparison of the algorithms. Using a larger database is crucial to benchmark the separation algorithms in order to obtain a true evaluation rather than particular case results.

The large variety of audio signal can be classified into different musical genres. Genres are labels created and used by humans for categorizing and describing music. They have no strict definitions and boundaries but particular genre share certain characteristics typically related to the instrumentation, rhythmic structure, and pitch content of the music and the resemblance between two pieces of music have been used to perform chord transcription [**?**, **?**] or for downbeat detection [**?**]. Finally when the genre information is not available, it is possible to perform automatic genre classification [**?**].

In the context of BSS, Non-negative Matrix Factorization (NMF) is a widely used method for source separation. The goal of NMF is to approximate a data matrix $V \in \mathbb{R}_+^{n \times m}$ as $V \approx \tilde{V} = WH$ with $W \in \mathbb{R}_+^{n \times k}$, $H \in \mathbb{R}_+^{k \times m}$ and where $k$ is the rank of factorization [**?**]. In audio signal processing, the input data is usually a Time-Frequency (TF) representation such as a short time Fourier transform (STFT) or a constant-Q transform spectrogram. Blind source separation is a difficult problem and the plain NMF decomposition does not provide satisfying results. To perform a satisfying results, it is necessary to exploit various features that make each sources distinguishable from one another. Supervised algorithms in the NMF framework exploit training data or prior information in order to guide the decomposition process. For example information from the scores or from midi signals [**?**] can be used to initialize the learning process. The downside of this approach is that it requires well organized prior information that is not always available. Another supervised method consists in performing prior training on specific databases. For example a dictionary matrix $W_{train}$ is learned from a big database in order to separate an instrument [**?**, **?**]. A common method to build a dictionary for NMF is to perform a decomposition on a large training set. After the convergence, the $W$ matrix from the decomposition is used as the dictionary matrix $W_{train}$ in the separation [**?**]. Another method is detailed in [**?**], a dictionary matrix is created by extracting template spectra from isolated drum samples. The dictionary is then used in a NMF decomposition to perform drum transcription. This method requires minimum tuning from the user. However, the dictionary should match the target instrument for satisfying performances.

The problem of recent method using dictionary matrices is that, within a database, an instrument can sound differently depending on the recording condition and post processing treatment. In order to represent correctly one instrument, ones can decide to learn a dictionary on a large database, however, the problem of overfitting the data exist. In order to overcome this problem and to build effective dictionaries we decided to use genre specific training data. Genre specific information can provide an insight on the structure of the audio signal as they share similar features.

In this paper, we focus on the task of harmonic/percussive source separation (HPSS) using the method developed in [**?**]. We adapt the method to be used with a drum dictionary to extract the percussive instruments. This method is explained in detail in the preprint. The main contribution of this article is that we developed a genre specific method to build a drum NMF dictionary that obtains consistent results on a HPSS task. Overall using a fixed dictionary for drum extraction is an underused method in the literature as it is difficult to create a drum dictionary that provide robust results on a large variety of signal. By using genre specific dictionary we were able to improve the separation score and decrease the computation time as the dictionary are smaller in size.

## 2. STRUCTURED PROJECTIVE NMF (SPNMF)

In this section we present our semi-supervised algorithm for harmonic/percussive source separation.

## 2.1 Presentation of the orthogonal and projective NMF

Using a squared Euclidean distance between the data matrix $V$ and its approximation $WH$, the NMF problems reads:

$$\min_{W,H \geq 0} \|V - WH\|^2 ,$$

where $\|.\|^2$ is the squared Euclidean distance.

The aim of PNMF is to find a non negative projection matrix $P \in \mathbb{R}_+^{n \times n}$ such that $V \approx \tilde{V} = PV$. In [?] Yuan & al. propose to seek $P$ as an approximative projection matrix under the form $P = WW^T$ with $W \in \mathbb{R}_+^{n \times k}$ with $k \leqslant n$. The PNMF problem reads :

$$\min_{W \geqslant 0} \|V - WW^T V\|^2 \qquad (1)$$

PNMF is similar to the NMF problem and can be simply obtained by replacing the activation matrix $H$ by $W^T V$. It is shown in [?] that the PNMF gives a much sparser decomposition than NMF.

Another very similar approach is the ONMF [?]. It consists in solving the following problem:

$$\min_{W \geqslant 0, H \geqslant 0} \|V - WH\|^2 \quad \text{s.t.} \quad W^T W = I_k \qquad (2)$$

In this method, orthogonality between nonnegative basis functions is enforced during the optimization process. In theory, it seems that PNMF and ONMF lead to similar decompositions, as the $W$ matrix estimated by PNMF is almost orthogonal (i.e., $\|W^T W - I_k\|^2$ is small). However in practice, enforcing the orthogonality between the base at every iteration is a constraint too strong to decompose audio signal [?].

The sparsity of the dictionary matrix is an interesting property for the decomposition of audio signals and especially for the decomposition of harmonic instruments with very localized harmonic spectra. Contrary to the NMF, the sparsity of PNMF and is an inherent features of the decomposition. These key properties of PNMF motivated us to decompose the harmonic instruments with the orthogonal basis functions.

## 2.2 Principle of the SPNMF

The orthogonal basis functions of PNMF are not flexible enough to decompose a complex audio signal. As stated in [?], harmonic instruments have sparse basis functions whereas percussive instruments have much flatter spectra. As the columns of $W$ are orthogonal, when two sources overlap in the Time-Frequency (TF) plane only one basis function will represent the mixture which is not adequate for efficient separation. To overcome this problem, we propose to add a standard NMF decomposition term to the PNMF. We can expect that most of the harmonic components will be represented by the orthogonal part while the percussive ones will be the regular NMF components. Using a similar model as in our preliminary work [?], let $V$ be the magnitude spectrogram of the input data. The model is then given by

$$V \approx \tilde{V} = V_H + V_P, \qquad (3)$$

with $V_P$ the spectrogram of the percussive part and $V_H$ the spectrogram of the harmonic part. $V_H$ is approximated by the PNMF decomposition while $W_P$ is decomposed by NMF components as :

$$V \approx \tilde{V} = W_H W_H^T V + W_P H_P. \qquad (4)$$

The data matrix is approximated by an almost orthogonal sparse part that codes the harmonic instruments $V_H = W_H W_H^T V$ and a non constrained NMF part that codes the percussive instruments $V_P = W_P H_P$. However, a fully unsupervised SPNMF model does not allow for a satisfying harmonic/percussive source separation [?]. To alleviate this problem, we use here a fixed drum dictionary $W_p$ in the percussive part of the SPNMF.

## 2.3 Algorithm Optimization

In order to obtain such a decomposition, we can use a measure of fit $D(x|y)$ between the data matrix $V$ and the estimated matrix $\tilde{V}$. $D(x|y)$ is a scalar cost function and in this article, we use the Itakura Saito (IS) divergence.

The SPNMF model gives the cost function :

$$\min_{W_H, W_P, H_P \geq 0} D(V | W_H W_H^T V + W_P H_P) \qquad (5)$$

A solution of this problem can be obtained by iterative multiplicative update rules following the same strategy as in [?,?] which consists in splitting the gradient with respect to (wrt) one variable (here $W_H$ for exemple) $\nabla_{W_H} D(V|\tilde{V})$ in its positive $[\nabla_{W_H} D(V|\tilde{V})]^+$ and negative parts $[\nabla_{W_H} D(V|\tilde{V})]^-$. The multiplicative updates for SPNMF are then given by:

$$W_H \leftarrow W_H \otimes \frac{[\nabla_{W_H} D(V|\tilde{V})]^-}{[\nabla_{W_H} D(V|\tilde{V})]^+},$$

where $\otimes$ is the Hadamard product or element-wise product. The SPNMF algorithm with a fixed dictionary matrix is:

---

Input: $V \in \mathbb{R}_+^{m \times n}$ Output: $W \in \mathbb{R}_+^{m \times k}$,
$W_{train} \in \mathbb{R}_+^{m \times e}$ and $H \in \mathbb{R}_+^{e \times n}$ Initialization;
**while** $i \leq$ *number of iterations* **do**

$\quad H_P \leftarrow H_P \otimes \frac{[\nabla_{H_P} D(V|\tilde{V})]^-}{[\nabla_{H_P} D(V|\tilde{V})]^+}$

$\quad W_H \leftarrow W_H \otimes \frac{[\nabla_{W_H} D(V|\tilde{V})]^-}{[\nabla_{W_H} D(V|\tilde{V})]^+}$

$\quad i = i + 1$

**end**
$X_P = W_{train} H_P$ and $X_H = W_H W_H^T V$

**Algorithm 1:** SPNMF with the drum dictionary matrix.

---

## 2.4 Signal reconstruction

The percussive signal $x_p(t)$ is synthesized using the magnitude percussive spectrogram $X_P = W_P H_P$. To reconstruct the phase of the percussive part, we use a generalized Wiener filter [?] to create a percussive mask as:

$$\mathcal{M}_P = \frac{X_P^\alpha}{X_M^\alpha + X_P^\alpha}. \qquad (6)$$

| Genre | Length(min) |
|---|---|
| Classical | 22.06 |
| Electronic/Fusion | 18.66 |
| Jazz | 10.96 |
| Pop | 12.53 |
| Rock | 11.43 |
| Sing/Songwriter | 9.36 |
| World/Folk | 9.53 |
| Non Specific | 11.03 |

**Table 1**: Source separation performance for the synthetic signal.

Where $\alpha \in (1, 2)$. To retrieve the percussive signal as,

$$x_p(t) = InverseSTFT(\mathcal{M}_P \otimes X). \qquad (7)$$

Where $X$ is the complex spectrogram of the mixture. Similarly for the harmonic part, we obtain:

$$\mathcal{M}_H = \frac{X_H^\alpha}{X_M^\alpha + X_P^\alpha}, \qquad (8)$$

and:

$$x_h(t) = InverseSTFT(\mathcal{M}_H \otimes X). \qquad (9)$$

### 3. CONSTRUCTION OF THE DICTIONARY

Length of the dictionary:
### 3.1 Database

The dataset is taken from medley-dB [**?**], it is composed of polyphonic real-world music excerpts. It has 122 music signals and 77 of them contain percussive instruments, harmonic instruments and vocals. The signals that do not contain a percussive part are not part of the evaluation. We will be using the song of the genre, *Singer/Songwriter* (17 songs), *Pop* (10 songs), *Rock* (20 songs), *Jazz* (11 songs), *Electronic/Fusion* (13 songs) and *World/Folk* (6 songs). Because the notion of genre is quite subjective, the medley-dB database uses general genre labels. These labels should not be considered to be "precise" genre labels. There are many instances where a song could have fallen in multiple genres, and the choices were made so that each genre would be as acoustically homogeneous as possible. As we are only working with the instrumental part of the song "Pop" label (for example) are similar to the "Singer/Songwriter". We use 3 song of each genre to use as a training database. The songs used for the training part are not part of the evaluation. The files selected are: Length of the dictionary:
### 3.2 Supervised NMF for source separation

The NMF model is:

$$V \approx \tilde{V} = WH. \qquad (10)$$

If $V$ is the power spectrum of a drum signal, The matrix $W$ is a *dictionary* or a set of *patterns* that codes the frequency information of the drum. Building a dictionary specific to an instrument that performs well on a large database is a complicated problem. Here we build genre specific drum

| Genre | Length(min) |
|---|---|
| Classical | JoelHelander Definition |
| | MatthewEntwistle AnEveningWithOliver |
| | MusicDelta Beethoven |
| Electronic/Fusion | EthanHein 1930sSynthAndUprightBass |
| | TablaBreakbeatScience Animoog |
| | TablaBreakbeatScience Scorpio |
| Jazz | CroqueMadame Oil |
| | MusicDelta BebopJazz |
| | MusicDelta ModalJazz |
| Pop | DreamersOfTheGhetto HeavyLove |
| | NightPanther Fire |
| | StrandOfOaks Spacestation |
| Rock | BigTroubles Phantom |
| | Meaxic TakeAStep |
| | PurlingHiss Lolita |
| Sing/Songwriter | AimeeNorwich Child |
| | ClaraBerryAndWooldog Boys |
| | InvisibleFamiliars DisturbingWildlife |
| World/Folk | AimeeNorwich Flying |
| | KarimDouaidy Hopscotch |
| | MusicDelta ChineseYaoZu |

**Table 2**: Source separation performance for the synthetic signal.

dictionary using the medley-dB database. Using dictionary specific to the genre of music allows us to have smaller dictionaries that a more specific to the signal to decompose. It grants us lower computation time and better separation score The dictionary are build has follow. For every song of the medley-dB database, we perform and NMF with $k = 100$ on the drum signals. The $W$ matrices are then concatenated depending on the genre of the song to form a dictionary matrix specific to a genre of music.