# GENRE SPECIFIC DICTIONARIES FOR HARMONIC/PERCUSSIVE SOURCE SEPARATION


| Clément Laroche | Second author | Third author | Fourth author |
|---|---|---|---|
| Affiliation1 | Affiliation2 | Affiliation3 | Affiliation4 |
| author1@ismir.edu | author2@ismir.edu | author3@ismir.edu | author4@ismir.edu |



## ABSTRACT

Supervised algorithms for audio source separation use apriori information like training data and dictionary to acheive a decomposition. Audio signals are diverse and it is generaly impossible to build prior information that is relevant on any audio signal. Most of the method are tested on small databases that do not allow exhaustive comparison of the algorithms. The problem of using a large database is that the algorithms are not robust to the wide variety of audio signals and they do not provide satisfying results without user intervention. In this article, we focus on the particular case of harmonic/percussive source separation and we propose to use musical genre information to guide the decomposition on a large database. Our method proves that the musical genre is a relevant feature as the dictionaries built using genre specific information perform better that an universal dictionary built on all genre.


## 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 [5] (if a source is a noise) , up-mixing [9](spatialization of the sources) or automatic transcription [3] (it is easier to work on single source). The task is difficult due to the complexity and the variability of the music mixtures. Most datasets used for Blind Audio Source Separation (BASS) 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 different 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 [23]. 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. This resemblance between two pieces of music have been used to perform chord transcription [16, 20] or for downbeat detection [10]. Finally when the genre information is not available, it is possible to perform automatic genre classification [17].

In the context of BASS, 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 [14]. 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 decomposition, 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 [8] can be used to initialize the learning process. The downside of these approachs are that they require well organized prior information that is not always available. Another supervised method consists in performing prior training on specific databases. A dictionary matrix $W_{train}$ is learned from a big database in order to separate an instrument [11,25]. A common method to build a dictionary is to perform a NMF 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 [11]. Another method is detailed in [25], 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. These methods require minimum tuning from the user. However, the dictionaries must match the target instruments for satisfying performances.

In this paper, we focus on the task of harmonic/percussive source separation (HPSS) using the method developed in [13]. We adapt the algorithm to use a drum dictionary to extract the percussive instruments. This method is explained in detail in the preprint. The problem of using a fixed dictionary matrices is that within a database, the same instrument can sound differently depending on the recording conditions and post processing treatments. In order to represent correctly one instrument, ones can decide to learn a dictionary on a large database.

However, the problem of over-fitting the data exist. In order to overcome this problem and to build effective dictionaries, we decided to use genre specific training data. As they share similar features, genre specific information can provide an insight on the structure of the audio signal. The main contribution of this article is that we developed a genre specific method to build NMF drum dictionaries that obtain consistent and robust results on a HPSS task. The genre specific dictionaries are able to improve the separation score compared to a universal dictionary.

## 2. GENRE SPECIFIC INFORMATION

Musical genre is probably the most popular high level music descriptor. Electronic Music Distribution have become more popular in recent years and music catalogues never stop to increase (the biggest online services propose around 1 million tracks); in that context, associating a genre to a musical piece is crucial to help users finding what they are looking for. If an explicit definitions of musical genres is still out of reach [2], musical genre classification can be performed automatically using different set of features [19, 23].

Source separation have been used extensively in order to help the classification process [12, 22]. However, using genre information have not been exploited to guide the decomposition process.

## 3. STRUCTURED PROJECTIVE NMF (SPNMF)

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

### 3.1 Presentation of the orthogonal and projective NMF

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 [27] 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 :

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

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 [26] that the PNMF gives a much sparser decomposition than NMF.

Another very similar approach is the ONMF [7]. It consists in solving the following problem:

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

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 [13].

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.

### 3.2 Principle of the SPNMF

The orthogonal basis functions of PNMF are not flexible enough to decompose a complex audio signal. As stated in [6], 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 [13], let $V$ be the magnitude spectrogram of the input data. The model is then given by

$$V \approx \tilde{V} = V_H + V_P, \tag{3}$$

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 :

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

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 $V_P = W_P H_P$. We use here a fixed drum dictionary $W_p$ in the percussive part of the SPNMF as a fully unsupervised SPNMF model does not allow for a satisfying harmonic/percussive source separation [13].

### 3.3 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 :

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

A solution of this problem can be obtained by iterative multiplicative update rules following the same strategy as in [15, 27] 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 algorithm 1 gives us the SPNMF optimization process.

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.

### 3.4 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 [18] to create a percussive mask as:

$$\mathcal{M}_P = \frac{X_P^2}{X_M^2 + X_P^2}. \tag{6}$$

To retrieve the percussive signal as,

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

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

$$\mathcal{M}_H = \frac{X_H^2}{X_M^2 + X_P^2}, \tag{8}$$

and:

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

## 4. CONSTRUCTION OF THE DICTIONARY

In this section, we present the test conducted on the SiSec database in order to find the optimal parameter to build the genre specific dictionaries.

### 4.1 Optimal size for the dictionary

The first step to build a NMF drum dictionary is to select the rank of factorization. We run the optimization tests on the public SiSec database from [1] to avoid overtraining our algorithm. The dataset is composed of polyphonic real-world music excerpts and each music signal contains percussive, harmonic instruments and vocals. The duration of the four recording is ranging from 14 to 24 s. The goal is to perform an harmonic/percussive decomposition. Following the same protocol as [6], we will not consider the vocal part and we will build the mixture signals from the percussive and harmonic instruments only. All the signals are sampled at $44.1kHz$. We compute the STFT with a and 2048 sample-long Hann window with a 50% overlap.

The drum signal used for the training comes from the database [21] and the signal is around 3 min long. We used 14 files from the database were the drummer is playing a *drum phrase*. We compute an NMF decomposition with different rank of factorization ($k = 12$, $k = 50$, $k = 100$, $k = 500$, $k = 1000$ and $k = 2000$) on the drum signal alone to obtain 6 drum dictionaries.

The dictionaries are then used to perform a HPSS on the four songs of the SiSEC database using the SPNMF algorithm. The results are compared by means of the Signal-to-Distortion Ratio (SDR), the Signal-to-Interference Ratio (SIR) and the Signal-to-Artifact ratio (SAR) of each of the separated sources using the BSS Eval toolbox provided in [24].

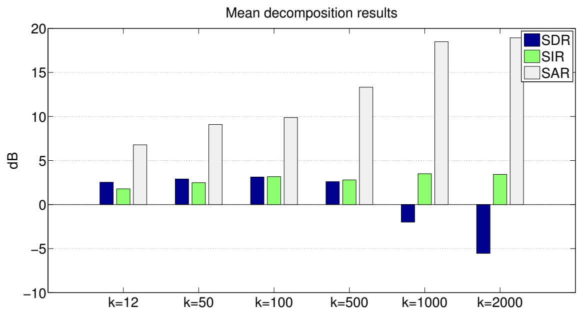


**Figure 1**: Average results on the SiSec database.

The results on the figure 1 show that the optimal value for the SDR and SIR is reached for $k = 100$, then the SDR decrease rapidly for $k \geqslant 500$. The high value of SAR (for $k \geqslant 500$) are explained because the separation process is not satisfying. The harmonic signal given at the end of the algorithm is composed of most of the original signal therefore the SAR is very high but the decomposition quality is poor.

For a 3 min drum signal, the optimal dictionary size for the SPNMF algorithm is $k = 100$.

### 4.2 Training and evaluation database

The dataset medley-dB [4] is used for our tests. It is composed of polyphonic real-world music excerpts. It has 122 music signals and 87 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 are using the song of the genre, *Classical* (8 songs), *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 (see Section 2), the medley-dB database uses general genre labels. These labels should not be considered to be precise genre labels. There are many

| Genre | Artist Song |
| --- | --- |
| Classical | JoelHelander Definition<br>MatthewEntwistle AnEveningWithOliver<br>MusicDelta Beethoven |
| Electronic/Fusion | EthanHein 1930sSynthAndUprightBass<br>TablaBreakbeatScience Animoog<br>TablaBreakbeatScience Scorpio |
| Jazz | CroqueMadame Oil<br>MusicDelta BebopJazz<br>MusicDelta ModalJazz |
| Pop | DreamersOfTheGhetto HeavyLove<br>NightPanther Fire<br>StrandOfOaks Spacestation |
| Rock | BigTroubles Phantom<br>Meaxic TakeAStep<br>PurlingHiss Lolita |
| Singer/Songwriter | AimeeNorwich Child<br>ClaraBerryAndWooldog Boys<br>InvisibleFamiliars DisturbingWildlife |
| World/Folk | AimeeNorwich Flying<br>KarimDouaidy Hopscotch<br>MusicDelta ChineseYaoZu |
| Non specific | JoelHelander Definition<br>TablaBreakbeatScience Animoog<br>MusicDelta BebopJazz<br>DreamersOfTheGhetto HeavyLove<br>BigTroubles Phantom<br>AimeeNorwich Flying<br>MusicDelta ChineseYaoZu |

**Table 1**: Song selected for the training database.

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, the *Pop* label (for example) are similar to the *Singer/Songwriter*.

The training dataset is built using 3 song of each genre. The songs used for the training part are not part of the evaluation. To compare the genre specific dictionary, we build a non specific/universal dictionary built using one half of one song of each genre. The files selected for the training are given in Table 1.

### 4.3 Genre specific dictionaries

The NMF model is:

$$V \approx \tilde{V} = WH. \tag{10}$$

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 dictionary using the medley-dB database. Using dictionary specific to the genre of music allows us to have dictionaries that a more specific to the signal to decompose.

Table 2 gives us the length and the energy of the training signal for each genre. *Classical* and *Electronic/Fusion* songs are composed of songs where the drum is only playing for a few moments and even though the signals are longer than the other, the energy is around the same for all the signals. With the results from Section 4.1 the dictionary are built as follow. For every genre specific database

| Genre | Length(min) | Energy |
| --- | --- | --- |
| Classical | 22.06 | 12200 |
| Electronic/Fusion | 18.66 | 49800 |
| Jazz | 10.96 | 49900 |
| Pop | 12.53 | 39300 |
| Rock | 11.43 | 79000 |
| Singer/Songwriter | 9.36 | 48300 |
| World/Folk | 9.53 | 46700 |
| Non Specific (Mix) | 11.03 | 30000 |

**Table 2**: Length and energy of the genre specific database.

of the training database, we perform and NMF on the drum signals with $k = 300$ as the signal are tree times the size of the one in Section 4.1 (we choose $k = 100$ per song for the NMF). Then the dictionaries are used in the SPNMF algorithm as the matrix $W_P$ (see algorithm 1).

## 5. RESULTS

### 5.1 Comparison of the dictionaries

In this section we present the results of the algorithm with the genre specific dictionaries on the 66 song from test database Medley-dB. We perform an HPSS on the audio files using the SPNMF with the 8 dictionaries created on Section 4.3. The results on each of the songs are then sorted by genres and the average results are displayed using box-plot. Each box-plot is made up of a central line indicating the median of the data, upper and lower box edges indicating the $1^{st}$ and $3^{rd}$ quartiles while the whiskers indicate the minimum and maximum values.

The Figures 2, 3 and 4 show the SDR, SAR and SIR results for all the dictionaries on the *Pop* subsection. It gives us a overall idea on how all the dictionaries perform on the same database. The results using the *Pop* dictionary has the highest SDR and SIR results. The non specific dictionary is not performing as well as the *Pop* dictionary. On this database, the genre specific method is giving relevant information to the algorithm. As stated in Section 4.2, some genre are similar to other. This explains why the *Rock* and the *Singer* dictionaries are giving good results too. An interesting result is that compared to the non specific dictionary, the *Pop* dictionary has a lower variance. Genre information allows for a higher robustness to the variety of the songs within the same genre.

On Table 3, we display the mean separation score for all the genre specific dictionaries compared to the non specific dictionary. The genre specific dictionaries outperform the universal dictionary by a considerable margin on 5 of the 7 genres. The results are discussed in the next Section.

### 5.2 Discussion

On the database *Singer/Songwriter*, *Pop*, *Rock*, *Jazz* and *World/Folk*, the genre specific dictionaries outperform the universal dictionary. The similar pitch content of the music of the same genre is not altered by the NMF compression

| Genre | Classical | Electronic/Fusion | Jazz | Pop | Rock | Singer/Songwriter | World/Folk |
|---|---|---|---|---|---|---|---|
| Percussive separation | | | | | | | |
| Genre specific (dB) | | | | | | | |
| SDR | -1.64 | -0.63 | **0.36** | **2.45** | **-0.17** | **0.64** | **-3.02** |
| SIR | 8.21 | 15.17 | **9.60** | 12.34 | **19.79** | 11.45 | **2.54** |
| SAR | 5.88 | 0.31 | **2.08** | **3.36** | 0.31 | **4.46** | **11.90** |
| Non specific (dB) | | | | | | | |
| SDR | **-0.04** | **-0.25** | -0.68 | 2.01 | -2.15 | -0.01 | -3.84 |
| SIR | **11.3** | **17.01** | 9.57 | **12.60** | 18.30 | **13.04** | 2.46 |
| SAR | **8.07** | **0.39** | 0.87 | 2.74 | **2.34** | 1.83 | 11.89 |
| Harmonic Separation | | | | | | | |
| Genre specific (dB) | | | | | | | |
| SDR | **7.49** | **1.63** | **13.05** | **5.06** | **2.14** | 7.20 | **4.86** |
| SIR | **10.60** | **1.84** | **13.27** | **5.02** | 2.19 | **11.45** | 12.19 |
| SAR | 18.19 | 23.48 | 28.50 | 24.48 | **35.97** | 28.48 | **21.23** |
| Non specific (dB) | | | | | | | |
| SDR | 6.04 | 1.33 | 12.71 | 4.78 | 1.92 | 7.46 | 4.80 |
| SIR | 7.14 | 1.36 | 12.82 | 4.85 | **2.86** | 7.50 | 12.60 |
| SAR | **27.21** | **27.72** | **29.87** | **26.17** | 34.25 | **31.87** | **20.92** |

**Table 3**: Mean harmonic SDR, SIR and SAR results on the Medley-dB database.

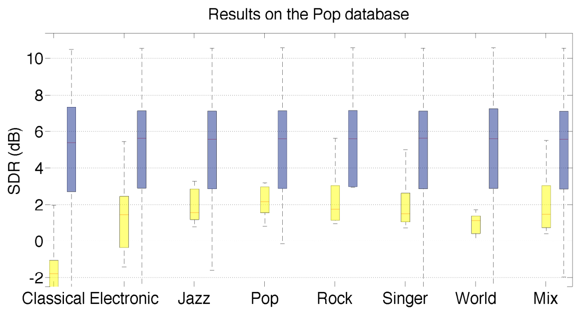


**Figure 2**: Harmonic (left bar)/percussive (right bar) SDR results on the Pop sub-database using the SPNMF with the 8 dictionaries.

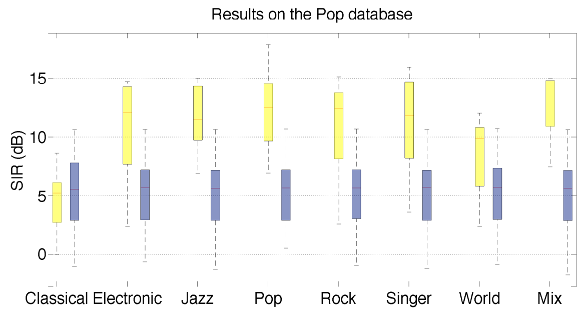


**Figure 3**: Harmonic (left bar)/percussive (right bar) SIR results on the Pop sub-database using the SPNMF with the 8 dictionaries.

and this information improve the separation. The database *Classical* and *Electronic/Fusion* are composed of songs where the drum is only playing for a few moments. Similarly on some songs of the *Electronic/Fusion* database, the electronic drum reproduces the same pattern during the whole song making the drum part very redundant thus the drum dictionary does not contain a sufficient amount of information to outperform the universal dictionary. Because of these two factors, the genre specific dictionaries are not performing correctly.

Overall the harmonic separation is giving much better results than the percussive extraction. The fixed dictionaries are creating artefact as the percussive templates do not correspond exactly to the drum signal to decompose. A possible way to alleviate this problem is to adapt the dictionaries but this require the use of an hyper parameters that is not the philosophy of this work.

The main drawback of using a NMF dictionary is that the decomposition is not unique and can any permutation is also a solution. Because of that, we lose the temporal structure of the original drum signal.

## 6. CONCLUSION

Using genre specific information in order to build more relevant drum dictionaries is a powerful method to improve the HPSS. The dictionaries still have an imprint of the genre even after the NMF decomposition and the additional information is properly used by the SPNMF to improve the source separation.

This is a first step in order to produce dictionaries capable of extracting a wide variety of audio signal.

Future work will be dedicated into building a blind method to select the genre specific dictionary in order to perform the same technique on database were the genre information is not present.

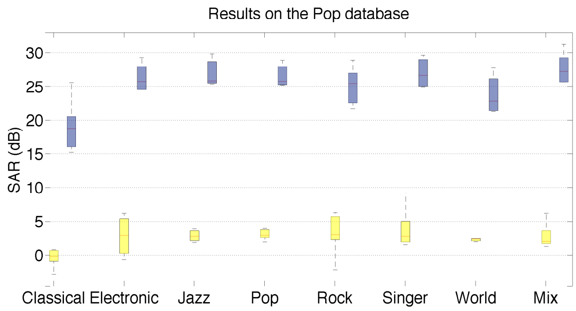


**Figure 4**: Harmonic (left bar)/percussive (right bar) SAR results on the Pop sub-database using the SPNMF with the 8 dictionaries.

## 7. ANNEXE

### 7.1 Itakura Saito divergence

The Itakura Saito divergence gives us the problem,

$$\min_{W_1,W_2,H_2\geq 0} \frac{V}{\tilde{V}} - log(\frac{V}{\tilde{V}}) + 1.$$

The gradient wrt $W_1$ gives

$$[\nabla_{W_1} D(V|\tilde{V})]^-_{i,j} = (ZV^TW_1)_{i,j} + (VZ^TW_1)_{i,j},$$

with $Z_{i,j} = (\frac{V}{W_1W_1^TV+W_2H_2})_{i,j}$. The positive part of the gradient is

$$[\nabla_{W_1} D(V|\tilde{V})]^-_{i,j} = (\phi V^TW_1)_{i,j} + (V\phi^TW_1)_{i,j},$$

with

$$\phi_{i,j} = (\frac{I}{W_1W_1^TV + W_2H_2})_{i,j}.$$

and $I = ones(size(V))$.

Similarly, the gradient wrt $W_2$ gives

$$[\nabla_{W_2} D(V|\tilde{V})]^- = VH_2^T$$

and

$$[\nabla_{W_2} D(V|\tilde{V})]^+ = W_1W_1^TVH_2^T + W_2H_2H_2^T.$$

Finally, the gradient wrt $H_2$ gives

$$[\nabla_{H_2} D(V|\tilde{V})]^- = W_2^TV$$

and

$$[\nabla_{H_2} D(V|\tilde{V})]^+ = 2W_2^TW_1W_1^TV + W_2^TW_2H_2.$$

## 8. REFERENCES


[1] S. Araki, A. Ozerov, V. Gowreesunker, H. Sawada, F. Theis, G. Nolte, D. Lutter, and N. Duong. The 2010 signal separation evaluation campaign : audio source separation. In *Proc. of LVA/ICA*, pages 114–122, 2010.

[2] J.J. Aucouturier and F. Pachet. Representing musical genre: A state of the art. *Journal of New Music Research*, 32(1):83–93, 2003.

[3] N. Bertin, R.Badeau, and G.Richard. Blind signal decompositions for automatic transcription of polyphonic music: Nmf and k-svd on the benchmark. In *Proc. of IEEE ICASSP*, pages 65–68, 2007.

[4] R. Bittner, J. Salamon, M. Tierney, M. Mauch, C. Cannam, and J. Bello. Medleydb: A multitrack dataset for annotation-intensive mir research. In *Proc. of ISMIR*, 2014.

[5] S. Boll. Suppression of acoustic noise in speech using spectral subtraction. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 27(2):113–120, 1979.

[6] F. Canadas-Quesada, P. Vera-Candeas, N. Ruiz-Reyes, J. Carabias-Orti, and P. Cabanas-Molero. Percussive/harmonic sound separation by non-negative matrix factorization with smoothness/sparseness constraints. *EURASIP Journal on Audio, Speech, and Music Processing*, 2014(1):1–17, 2014.

[7] S. Choi. Algorithms for orthogonal nonnegative matrix factorization. In *Proc. of IEEE IJCNN*, pages 1828–1832, 2008.

[8] S. Ewert and M. Müller. Score-informed source separation for music signals. *Multimodal music processing*, 3:73–94, 2012.

[9] D. Fitzgerald. Upmixing from mono-a source separation approach. In *Proc. of IEEE DSP*, pages 1–7, 2011.

[10] J. Hockman, M. Davies, and I. Fujinaga. One in the jungle: Downbeat detection in hardcore, jungle, and drum and bass. In *Proc. of ISMIR*, pages 169–174, 2012.

[11] X. Jaureguiberry, P. Leveau, S. Maller, and J. Burred. Adaptation of source-specific dictionaries in non-negative matrix factorization for source separation. In *Proc. of IEEE ICASSP*, pages 5–8, 2011.

[12] A. Lampropoulos, P. Lampropoulou, and G. Tsihrintzis. Musical genre classification enhanced by improved source separation technique. In *Proc. of ISMIR*, pages 576–581, 2005.

[13] C. Laroche, M. Kowalski, H. Papadopoulos, and G. Richard. A structured nonnegative matrix factorization for source separation. In *Proc. of EUSIPCO*, 2015.

[14] D. Lee and S. Seung. Learning the parts of objects by nonnegative matrix factorization. *Nature*, 401:788–791, 1999.

[15] D. Lee and S. Seung. Algorithms for non-negative matrix factorization. *Proc. of NIPS*, pages 556–562, 2001.

[16] K. Lee and M. Slaney. Acoustic chord transcription and key extraction from audio using key-dependent hmms trained on synthesized audio. *IEEE Transactions on Audio, Speech, and Language Processing*, 16(2):291–301, 2008.

[17] T. Li, M.Ogihara, and Q. Li. A comparative study on content-based music genre classification. In *Proc. of ACM*, pages 282–289, 2003.

[18] A. Liutkus and R. Badeau. Generalized wiener filtering with fractional power spectrograms. In *Proc. of IEEE ICASSP*, pages 266–270, 2015.

[19] C. McKay and I. Fujinaga. Musical genre classification: Is it worth pursuing and how can it be improved? In *Proc. of ISMIR*, pages 101–106, 2006.

[20] Y. Ni, M. McVicar, R. Santos-Rodriguez, and T. De Bie. Using hyper-genre training to explore genre information for automatic chord estimation. In *Proc. of ISMIR*, pages 109–114, 2012.

[21] O.Gillet and G.Richard. Enst-drums: an extensive audio-visual database for drum signals processing. In *Proc. of ISMIR*, pages 156–159, 2006.

[22] H. Rump, S. Miyabe, E. Tsunoo, N. Ono, and S. Sagayama. Autoregressive mfcc models for genre classification improved by harmonic-percussion separation. In *Proc. of ISMIR*, pages 87–92, 2010.

[23] G. Tzanetakis and P. Cook. Musical genre classification of audio signals. *IEEE transactions on Speech and Audio Processing*, 10(5):293–302, 2002.

[24] E. Vincent, R. Gribonval, and C. Févotte. Performance measurement in blind audio source separation. *IEEE Transactions on Audio, Speech, Language Process.*, 14:1462–1469, 2006.

[25] C. Wu and A. Lerch. Drum transcription using partially fixed non-negative matrix factorization. In *Proc. of EUSIPCO*, 2008.

[26] Z. Yang and E. Oja. Linear and nonlinear projective nonnegative matrix factorization. *IEEE Transactions on Neural Network.*, 21(5):734–749, 2010.

[27] Z. Yuan and E. Oja. Projective nonnegative matrix factorization for image compression and feature extraction. *Image Analysis*, pages 333–342, 2005.