NON-NEGATIVE MATRIX FACTORIZATION OF SIGNALS WITH OVERLAPPING EVENTS FOR EVENT DETECTION APPLICATIONS

Shiqiang Wang and Jorge Ortiz

IBM T.J. Watson Research Center, Yorktown Heights, NY 10598
{wangshiq, jjortiz}@us.ibm.com

ABSTRACT

In many event detection applications, training data may contain tags with multiple, simultaneous events. This is particularly likely when the definition of "event" is broad and includes events that can persist for an extended period of time. Decomposing a mixed signal into signals corresponding to individual events is non-trivial. In this paper, we propose a non-negative matrix factorization (NMF) method that generates independent dictionaries for different events from training data with overlapping events. The proposed method adds a mask matrix into the regularization term in conventional NMF approaches. This mask matrix captures known event labels in the training data, so that only related dictionary terms are updated during iteration. The effectiveness of the proposed approach is evaluated using both synthetic and real data.

Index Terms— Event detection, Internet of Things (IoT), non-negative matrix factorization, signal decomposition

1. INTRODUCTION

Internet of Things (IoT) applications have become increasingly popular over the recent years [1]. Many of them aim at detecting events from sensor signals. For example, acoustic signal processing for sound-event detection has been examined by researchers for the last several years [2–11], where the goal is to infer events in a physical environment from acoustic signals. Other examples include the detection of the type and number of electrical appliances in buildings from consumption patterns in the power line [12], the monitoring of driver behavior from various sensors in a car [13], etc. These applications enable us to better understand the environment and provide personalized services to users.

One challenge in event detection applications is that signals captured from the physical environment often contain components that belong to multiple events. Even for a single event, the observed signal may be a combination of multiple base signals. For example, an acoustic signal captured in an outdoor environment can simultaneously include the sounds of cars, people speaking, and the wind blowing, each of which can be defined as an event. The sound of a single event can also consist of multiple acoustic atoms that are mixed together. To detect different events, one needs to decompose the original signal into base signals that capture the fundamental components.

1.1. Non-Negative Matrix Factorization (NMF)

A common approach of performing such decomposition is to use non-negative matrix factorization (NMF) [14]. Here, the N observed non-negative signals (or features, such as spectrograms, extracted from these signals) are expressed as an M-by-N matrix $\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, ..., \mathbf{v}_N]$. The goal is to decompose $\mathbf{V}$ into an M-by-K matrix $\mathbf{W} = [\mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_K]$ multiplied by a non-negative K-by-N matrix $\mathbf{H} = [\mathbf{h}_1, \mathbf{h}_2, ..., \mathbf{h}_N]$, such that

$$\mathbf{V} \approx \mathbf{WH}, \quad \text{s.t. } \mathbf{W}, \mathbf{H} \geq 0 \tag{1}$$

where the "$\geq$" comparison is element-wise. Each column vector $\mathbf{v}_n$ represents an observed signal, which can be captured in different time frames or at different receivers. The column vectors $\mathbf{w}_k$ are the bases (also known as *dictionary*) of which the signals $\mathbf{v}_n$ are composed. The column vector $\mathbf{h}_n$ (known as *representation*) specifies how the dictionary atoms $\mathbf{w}_k$ are combined in the signal $\mathbf{v}_n$.

Many NMF techniques have been proposed to minimize the approximation error in (1), where the error can be expressed in the form of Euclidean distance (i.e., $\|\mathbf{V} - \mathbf{WX}\|^2$ where $\|\cdot\|$ stands for the Frobenius norm) or K-L divergence between $\mathbf{V}$ and $\mathbf{WH}$. The initial work in [15] proposed iterative algorithms to approximately solve the NMF problem, where it was also argued that NMF is a non-convex problem so that one should not expect to find efficient algorithms for finding the exact solution. Subsequent extensions to the original NMF problem include adding a regularization term to the error term to encourage sparsity of $\mathbf{H}$ [16–19], as well as more efficient solution approaches [20].

This research was sponsored by the U.S. Army Research Laboratory and the U.K. Ministry of Defence under Agreement Number W911NF-16-3-0001. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the U.S. Army Research Laboratory, the U.S. Government, the U.K. Ministry of Defence or the U.K. Government. The U.S. and U.K. Governments are authorized to reproduce and distribute reprints for Government purposes notwithstanding any copy-right notation hereon.

1.2. NMF Applied to Event Detection

The application of NMF to sound event detection has received attention in recent years. In [8], a method that learns different dictionaries for different events was proposed, where each event $d \in [1, D]$ has its own dictionary $\mathbf{W}^{(d)}$ that is part of the overall dictionary $\mathbf{W} = [\mathbf{W}^{(1)}, \mathbf{W}^{(2)}, ..., \mathbf{W}^{(D)}]$. It is assumed that clean single-event signals are available at the dictionary generation (i.e., training) phase, so that for event d, the dictionary $\mathbf{W}^{(d)}$ can be learned by decomposing the signal matrix $\mathbf{V}^{(d)}$ (which only contains event d) into the learned dictionary $\mathbf{W}^{(d)}$ and its representation $\mathbf{H}^{(d)}$. In the event detection phase, signals in matrix $\mathbf{V}$ may contain multiple events, and $\mathbf{V}$ is decomposed as follows:

$$\mathbf{V} \approx [\mathbf{W}^{(1)}, \mathbf{W}^{(2)}, ..., \mathbf{W}^{(D)}] \begin{bmatrix} \mathbf{H}^{(1)} \\ \mathbf{H}^{(2)} \\ \vdots \\ \mathbf{H}^{(D)} \end{bmatrix}, \quad (2)$$

$$\text{s.t. } \mathbf{W}^{(d)}, \mathbf{H}^{(d)} \geq 0, \forall d$$

The likelihood of the presence of event d in signal $\mathbf{v}_n$ is then proportional to the sum of elements in $\mathbf{h}_n^{(d)}$, where $\mathbf{H}^{(d)} = [\mathbf{h}_1^{(d)}, \mathbf{h}_2^{(d)}, ..., \mathbf{h}_N^{(d)}]$.

It was later found in [5, 10] that the convex hull of the different events' dictionaries generated from the above approach may overlap, which can lead to inaccuracies in event detection when directly using the above method. A preprocessing step to the training data using unsupervised NMF and K-means clustering was proposed in [5, 10], so that subsequent NMF processes are constrained within the clusters to avoid the region overlapping problem. This approach received very good results in the DCASE 2016 Challenge [21]. Although sound event detection has been predominantly studied in the literature, NMF-based approaches can also be applied to many other domains such as electricity monitoring [12].

1.3. Multiple Overlapping Events in Training Data

A limitation in the above approaches is that they require clean signals with single events for model training. Such signals may not be always available in practice, because signals captured in a real-world environment may contain multiple events at the same time. In complex environments, multiple events can be present for the majority of time. Therefore, how to train a model from training data with multiple overlapping events is an important problem.

One possible approach of performing training with multi-event signals is to separate the signal into components corresponding to single events using source separation techniques [22]. However, these techniques usually require some prior knowledge of signal statistics and it is often difficult to match the separated sources with event labels.

An NMF-based approach was proposed in [7], where the training signal vector $\mathbf{v}_n$ is extended to $N + D$ dimensions with the last D dimensions containing binary values representing the activation of events. A separate dictionary $\mathbf{W}^{(l)}$ is learned for the event labels. At the detection phase, the representation matrix $\mathbf{H}$ is first found from the observed signal $\mathbf{V}$ and the signal dictionary $\mathbf{W}^{(s)}$ such that $\mathbf{V} \approx \mathbf{W}^{(s)}\mathbf{H}$. Then, the label dictionary $\mathbf{W}^{(l)}$ is applied to $\mathbf{H}$, and $\mathbf{W}^{(l)}\mathbf{H}$ contains the likelihoods of different events.

In this paper, different from existing approaches, we propose a more direct method of incorporating overlapping event information in NMF. We consider a grouped dictionary model as in (2), while noting that this model can be integrated into larger systems such as those in [5, 10] to improve performance. We focus on the standalone NMF problem in this paper, and leave system integration aspects for future work. The approach we take is inspired by the approaches for NMF with sparsity constraints as in [16–19], where a regularization term is included in the objective function to encourage sparseness. We propose a novel regularization approach using mask matrix in this paper, so that event-based dictionaries can be generated from training data with overlapping events.

2. PROPOSED METHOD

We consider the grouped dictionary model as in (2). Let K_d denote the number of dictionary atoms for event d. Then, the sizes of matrices $\mathbf{W}^{(d)}$ and $\mathbf{H}^{(d)}$ are M-by-K_d and K_d-by-N, respectively. Note that $\sum_{d=1}^{D} K_d = K$.

2.1. Dictionary Generation

In the dictionary generation phase, we are given a training dataset $\mathbf{V}$ with N different signals. Each of these signals are labeled with one or multiple events. Our goal is to find the $\mathbf{W}$ and $\mathbf{H}$ matrices in (2). The resulting $\mathbf{W}$ matrix is used for detecting events from new signals later.

To maintain the grouping structure, we define a K-by-N matrix $\mathbf{A} = [\mathbf{a}_1, \mathbf{a}_2, ..., \mathbf{a}_N]$, where each column vector $\mathbf{a}_n$ has D groups, i.e., $\mathbf{a}_n = [\mathbf{a}_n^{(1)}, \mathbf{a}_n^{(2)}, ..., \mathbf{a}_n^{(D)}]^{\mathrm{T}}$. Here, $\mathbf{a}_n^{(d)}$ is a K_d dimensional vector. For the n-th training sample $\mathbf{v}_n$, we set $\mathbf{a}_n^{(d)} = \mathbf{0}$ if $\mathbf{v}_n$ *has* event d, and we set $\mathbf{a}_n^{(d)} = \mathbf{1}$ otherwise, where $\mathbf{0}$ and $\mathbf{1}$ are vectors (of suitable size) containing all zeros and ones, respectively.

Because all elements in $\mathbf{h}_n^{(d)}$ should be equal to zero if the training sample $\mathbf{v}_n$ does not include event d, with the above definition of $\mathbf{A}$, we can equivalently say that $H_{ij} = 0$ if $A_{ij} = 1$, where H_{ij} and A_{ij} are the (i, j)-th element of $\mathbf{H}$ and $\mathbf{A}$, respectively. We therefore say that $\mathbf{A}$ is a *mask* of $\mathbf{H}$.

We then solve the following:

$$\min_{\mathbf{W},\mathbf{H}} \|\mathbf{V} - \mathbf{W}\mathbf{H}\|^2 + \lambda\|\mathbf{A} \odot \mathbf{H}\|_1, \quad (3)$$

$$\text{s.t. } \mathbf{W}, \mathbf{H} \geq 0$$

where $\|\cdot\|$ denotes the Frobenius norm, i.e., $\|\mathbf{X}\| = \left(\sum_i \sum_j |X_{ij}|^2\right)^{\frac{1}{2}}$ for an arbitrarily defined matrix $\mathbf{X}$, $\|\cdot\|_1$ is defined as $\|\mathbf{X}\|_1 = \sum_i \sum_j |X_{ij}|$, "$\odot$" denotes the element-wise multiplication of two matrices, and $\lambda > 0$ is a constant parameter of the regularization term. It is easy to see that if λ is large enough, H_{ij} is forced to zero when $A_{ij} = 1$. In this way, only those dictionaries for events that are present in the training signal are associated to the error minimization procedure.

The above definition uses Euclidean distance as the error metric. A similar objective can be defined for K-L divergence. We focus on the Euclidean distance metric in this paper for simplicity, but the proposed approach can be easily extended to K-L divergence or any other error metrics by modifying the objective function and solution algorithm.

A modified version of existing NMF algorithms can be used to solve the optimization problem in (3), to take into account the new regularization term. For simplicity, we propose an iterative solution approach based on the work in [16, 17] in this paper, while noting that other solution approaches can be applied as well after proper modifications. The iterative algorithm for solving (3) includes the following steps:

1. Generate mask matrix $\mathbf{A}$ based on the event labels in training dataset $\mathbf{V}$.
2. Initialize $\mathbf{W}$ and $\mathbf{H}$ with random positive values between 0 and 1.
3. Normalize each column of $\mathbf{W}$, i.e., $\mathbf{w}_k \leftarrow \mathbf{w}_k / \|\mathbf{w}_k\|_1$.
4. Update $\mathbf{H}$ using

$$\mathbf{H} \leftarrow \mathbf{H} \odot \frac{\mathbf{W}^{\mathrm{T}}\mathbf{V}}{\mathbf{W}^{\mathrm{T}}\mathbf{W}\mathbf{H} + \lambda\mathbf{A}} \tag{4}$$

   where the division $\frac{\mathbf{X}}{\mathbf{Y}}$ is defined element-wise.
5. Update $\mathbf{W}$ using

$$\mathbf{W} \leftarrow \mathbf{W} \odot \frac{\mathbf{V}\mathbf{H}^{\mathrm{T}} + \mathbf{E}(\mathbf{W}\mathbf{H}\mathbf{H}^{\mathrm{T}} \odot \mathbf{W})}{\mathbf{W}\mathbf{H}\mathbf{H}^{\mathrm{T}} + \mathbf{E}\mathbf{V}\mathbf{H}^{\mathrm{T}} \odot \mathbf{W}} \tag{5}$$

   where $\mathbf{E}$ is an M-by-M matrix with all elements equal to one.
6. Repeat from step 3 until convergence.

Using the same methodology as in [15, 16], one can show that the above algorithm is guaranteed to converge to a local minima. The global optimum cannot be guaranteed with this or any existing NMF algorithm, because NMF is essentially a non-convex optimization problem [15]. However, these algorithms work reasonably well in practice, and thus NMF is widely used in many applications.

2.2. Event Detection

In the event detection phase, for newly observed signals $\mathbf{V}$ (which may contain either one or multiple signals), we fix $\mathbf{W}$ found from Section 2.1, and update $\mathbf{H}$ according to (4) until convergence. Since $\mathbf{W}^{(d)}$ is the dictionary for event d only, the values of the elements in $\mathbf{H}^{(d)}$ represent the likelihood of event d. In particular, for the n-th observed signal $\mathbf{v}_n$, the likelihood that $\mathbf{v}_n$ has event d can be expressed as $\|\mathbf{h}_n^{(d)}\|_1$. We can define a threshold $\gamma > 0$, so that $\mathbf{v}_n$ is classified as containing event d if

$$\|\mathbf{h}_n^{(d)}\|_1 \geq \gamma \tag{6}$$

By checking the condition (6) for all n and d, we can obtain result on which events are present in each $\mathbf{v}_n$.

3. EXPERIMENTATION

We evaluate the performance of the proposed method via experimentations with three different datasets, which are summarized as follows.

The *synthetic dataset* contains training and testing signals synthetically generated according to (2), where we first randomly choose $\mathbf{W}$, then randomly generate separate matrices $\mathbf{H}_{\text{train}}$ and $\mathbf{H}_{\text{test}}$. The signals $\mathbf{V}_{\text{train}} = \mathbf{W}\mathbf{H}_{\text{train}}$ and $\mathbf{V}_{\text{test}} = \mathbf{W}\mathbf{H}_{\text{test}}$ are then respectively used for training and testing. The random matrices have values uniformly distributed between 0 and 1. Parameters related to the total number of events, number of dictionary atoms per event etc. are chosen as $D = 10$, $K_d = 5$ for all d, $M = 1000$, $N_{\text{train}} = N_{\text{test}} = 2000$. There is also a limit on the maximum number of events present in each single signal $\mathbf{v}_n$, which we will specify later. The dictionary generation and event detection algorithms *do not* have knowledge of the signal generation procedure.

The experiment also uses two real sound datasets. One is an *industrial sounds dataset* containing sounds of drilling, idle engine, jackhammer, water pump (new), and water pump (old) [11]. This is an example of industrial IoT applications where the goal is to use acoustic signals to monitor the condition of machine rooms and detect any abnormalities. The other is an *office sounds dataset* from the training data[1] of Task 2 in DCASE 2016 Challenge [21], containing sounds of clearing throat, coughing, door slam, drawer, keyboard, keys dropping, knocking, human laughter, page turning, phone, and speech. This is an example of home/office IoT applications for detecting human living/working condition.

Each label in the real sound dataset is considered as an event. For each event, approximately 25% of the data is used as testing data, and the rest is used as training data. Because most sound clips in the available dataset only contain a single event, we generate multi-event sounds by randomly mixing the sound clips. Each sound clip is split into frames of 0.5 seconds, frames with different labels are mixed with different amplification factors to generate an acoustic signal with multiple events. The spectrograms (with five windows) of mixed acoustic signals are used as the input signals $\mathbf{V}$ for

[1] At the time of submission, labeled evaluation dataset is not available to the public. Therefore, we only use the training dataset of Task 2 in DCASE 2016 Challenge and split this dataset further into training and testing data.

Table 1: Performance results of different approaches under different settings

Parameter Q		$Q=1$			$Q=3$			$Q=5$		
Approach		Proposed	[7]	[8]	Proposed	[7]	[8]	Proposed	[7]	[8]
Synthetic	F1 score	0.9977	0.3753	0.9977	0.9934	0.3533	0.9840	0.9857	0.3445	0.9521
	EER	0.0033	0.7054	0.0033	0.0087	0.7234	0.0205	0.0182	0.7680	0.0611
Industrial sounds	F1 score	0.7752	0.7342	0.7752	0.7116	0.7038	0.6999	0.6963	0.6957	0.6756
	EER	0.3462	0.3621	0.3462	0.3227	0.3689	0.3237	0.3236	0.3739	0.3435
Office sounds	F1 score	0.4714	0.4405	0.4714	0.4465	0.3855	0.4106	0.4362	0.3845	0.3908
	EER	0.5424	0.5689	0.5424	0.5776	0.6364	0.6210	0.5845	0.6354	0.6533

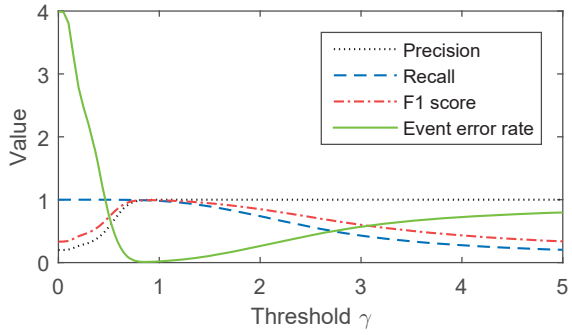

Fig. 1: Performance metrics under different values of γ for the proposed approach on synthetic dataset.

NMF. For the real sound datasets, we set $K_d = 50$ for all d, $N_{\text{train}} = 1500$, $N_{\text{test}} = 500$, while the values of D and M depend on characteristics of the real data.

We consider the overall precision, recall, F1 score, and event error rate (EER) as performance metrics. See [3, 7] for details of their definitions. At the high level, the F1 score can be understood as the accuracy jointly considering the precision and recall. The EER is an error rate that can have values above one because it is normalized by the ground truth number of events instead of the maximum number of possible events. A higher F1 score, precision, recall, and a lower EER indicates a better performance.

We fix $\lambda = 10$ in the experiments. The number of active events in each testing signal is uniformly distributed among $\{1, 2, 3\}$. In each training signal, the number of active events is uniformly distributed among $\{1, 2, ..., Q\}$, where the value of Q is defined later. All experiments were run with 10 different random seeds and the average performance is shown.

We first study the impact of the threshold value γ in (6). Fig. 1 shows the performance results on the synthetic dataset with $Q = 3$ and different values of γ. We see that, as one should expect, the precision increases and the recall decreases when γ increases, because the false positive rate decreases and the false negative rate increases with increasing γ. There exists an optimal γ that brings the highest F1 score and lowest EER. This optimal threshold can be found through cross validation during the training process in practice.

We then compare the performance of the proposed approach against the label dictionary approach in [7] and the single event grouping approach in [8], for all three datasets. For the method in [8], if the training signal has multiple events, a single event is randomly chosen. The results are shown in Table 1. We see that the proposed approach performs best in terms of F1 score and EER in all cases. The performance difference between the proposed approach and other approaches differ on a case-by-case basis. For example, the approach in [7] performs significantly worse than other approaches for the synthetic dataset, because the synthetic signals for different events are generated from largely different bases due to the random generation procedure. Therefore, a dictionary with event grouping such as the proposed method and the method in [8] gives much better performance than a method that does not perform this grouping. The performance generally becomes worse for all approaches when Q becomes large, i.e., when the training signal contains more overlapping events. This is intuitive because it should be harder to learn the dictionaries from signals with mixed events than from clean signals with a single event. When $Q = 1$, the proposed approach becomes the same as the approach in [8], thus their performances are the same in this case. The performance gain increases with larger Q, because the proposed method takes into account multiple labels for dictionary generation while [8] does not.

4. CONCLUSION

In this paper, we have proposed a simple but efficient method of performing NMF on training data with multiple overlapping event labels. The proposed method uses a mask matrix to restrict the values of the representation matrix during dictionary generation, so that separate dictionaries can be learned for different events. A simple thresholding approach is used to detect events in a new signal. Experiments using multiple datasets have shown superior performance of the proposed approach compared to other comparable approaches. Because NMF is an important building block in modern event detection systems, we have focused on the NMF process alone in this paper. We note that the proposed method can be integrated into a larger event detection system that has multiple processing stages to further improve performance. Such system integration aspects can be studied in the future. Future work can also consider the use of more realistic datasets, such as those with partly noisy labels and real-world signal mixtures, for evaluation.

5. REFERENCES

[1] L. Atzori, A. Iera, and G. Morabito, "The internet of things: A survey," *Computer networks*, vol. 54, no. 15, pp. 2787–2805, 2010.

[2] G. Parascandolo, H. Huttunen, and T. Virtanen, "Recurrent neural networks for polyphonic sound event detection in real life recordings," in *2016 IEEE ICASSP*. IEEE, 2016, pp. 6440–6444.

[3] E. Cakir, T. Heittola, H. Huttunen, and T. Virtanen, "Polyphonic sound event detection using multi label deep neural networks," in *2015 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2015, pp. 1–7.

[4] T. Heittola, A. Mesaros, T. Virtanen, and A. Eronen, "Sound event detection in multisource environments using source separation," in *Workshop on Machine Listening in Multisource Environments*, 2011, pp. 36–40.

[5] T. Komatsu, Y. Senda, and R. Kondo, "Acoustic event detection based on non-negative matrix factorization with mixtures of local dictionaries and activation aggregation," in *2016 IEEE ICASSP*. IEEE, 2016, pp. 2259–2263.

[6] A. Dessein, A. Cont, and G. Lemaitre, "Real-time detection of overlapping sound events with non-negative matrix factorization," in *Matrix Information Geometry*. Springer, 2013, pp. 341–371.

[7] O. Dikmen and A. Mesaros, "Sound event detection using non-negative dictionaries learned from annotated overlapping events," in *2013 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*. IEEE, 2013, pp. 1–4.

[8] J. F. Gemmeke, L. Vuegen, P. Karsmakers, B. Vanrumste *et al.*, "An exemplar-based NMF approach to audio event detection," in *2013 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*. IEEE, 2013, pp. 1–4.

[9] S. Adavanne, G. Parascandolo, P. Pertila, T. Heittola, and T. Virtanen, "Sound event detection in multichannel audio using spatial and harmonic features," in *Detection and Classification of Acoustic Scenes and Events 2016*, Sept. 2016.

[10] T. Komatsu, T. Toizumi, R. Kondo, , and Y. Senda, "Acoustic event detection method using semi-supervised non-negative matrix factorization with a mixture of local dictionaries," in *Detection and Classification of Acoustic Scenes and Events 2016*, Sept. 2016.

[11] B. J. Ko, J. Ortiz, T. Salonidis, D. Verma, S. Wang, X. Wang, and D. Wood, "Demo: Acoustic signal processing for anomaly detection in machine room environments," in *ACM BuildSys 2016*, Nov. 2016.

[12] M. Zeifman, C. Akers, and K. Roth, "Nonintrusive appliance load monitoring (nialm) for energy control in residential buildings: Review and outlook," in *IEEE Transactions on Consumer Electronics*, 2011.

[13] T. Toledo, O. Musicant, and T. Lotan, "In-vehicle data recorders for monitoring and feedback on drivers behavior," *Transportation Research Part C: Emerging Technologies*, vol. 16, no. 3, pp. 320–331, 2008.

[14] A. N. Langville, C. D. Meyer, R. Albright, J. Cox, and D. Duling, "Initializations for the nonnegative matrix factorization," in *Proceedings of the twelfth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2006, pp. 23–26.

[15] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," in *Advances in neural information processing systems*, 2001, pp. 556–562.

[16] J. Eggert and E. Korner, "Sparse coding and NMF," in *2004 IEEE International Joint Conference on Neural Networks*, vol. 4. IEEE, 2004, pp. 2529–2533.

[17] M. N. Schmidt, J. Larsen, and F.-T. Hsiao, "Wind noise reduction using non-negative sparse coding," in *2007 IEEE Workshop on Machine Learning for Signal Processing*. IEEE, 2007, pp. 431–436.

[18] J. Kim, R. Monteiro, and H. Park, "Group sparsity in nonnegative matrix factorization." in *SDM*. SIAM, 2012, pp. 851–862.

[19] A. Lefevre, F. Bach, and C. Févotte, "Itakura-saito non-negative matrix factorization with group sparsity," in *2011 IEEE ICASSP*. IEEE, 2011, pp. 21–24.

[20] H. Kim and H. Park, "Sparse non-negative matrix factorizations via alternating non-negativity-constrained least squares for microarray data analysis," *Bioinformatics*, vol. 23, no. 12, pp. 1495–1502, 2007.

[21] "DCASE 2016 challenge," 2016. [Online]. Available: http://www.cs.tut.fi/sgn/arg/dcase2016/challenge

[22] A. Ozerov, E. Vincent, and F. Bimbot, "A general flexible framework for the handling of prior information in audio source separation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 4, pp. 1118–1133, 2012.