Probabilistic Modelling of Signal Mixtures with Differentiable Dictionaries

Lukáš Samuel Marták[1,2], Rainer Kelz[1], and Gerhard Widmer[1,2]
[1]Institute of Computational Perception & [2]LIT Artificial Intelligence Lab
Johannes Kepler University Linz, Austria
{first.last}@jku.at

***Abstract*—We introduce a novel way to incorporate prior information into (semi-) supervised non-negative matrix factorization, which we call *differentiable dictionary search*. It enables general, highly flexible and principled modelling of mixtures where non-linear sources are linearly mixed. We study its behavior on an audio decomposition task, and conduct an extensive, highly controlled study of its modelling capabilities.**

***Index Terms*—non-negative matrix factorization, normalizing flows**

I. Introduction

Non-negative matrix factorization (NMF) [1] can be used to decompose a spectrogram $\mathbf{S}$ of an audio mixture into a spectral representation of its individual sources [2]. In its unsupervised form, NMF simultaneously tries to learn a representation of the individual sources, a *dictionary* $\mathbf{W}$, as well as the points in time at which those sources can be heard — their *activity* over time $\mathbf{H}$, such that $\mathbf{S} \approx \mathbf{WH}$. This decomposition succeeds if one crucial assumption about the audio mixture is met: all the individual sound sources that are mixed together can also be heard in isolation for at least some amount of time. If this assumption is not true, NMF will not be able to separate sound sources that only appear together [3]. To still achieve a useful decomposition in such cases, some prior information about the individual sound sources needs to be incorporated, for example as structural constraints on the dictionary of sources, the vectors in $\mathbf{W}$ that describe the spectral representation of the sound sources. This approach is called *supervised* or *semi-supervised* NMF [4] or alternatively *task-driven dictionary learning* [5]. For this to work we need to have access to recordings of individual sound sources, and assume that *similar* sound sources will be present in the audio recording that we try to decompose.

As there are multiple ways to introduce prior knowledge and constraints, if given a choice, we would strongly prefer one that has the following desirable properties: it fully captures the distribution over the spectral representation of sound sources, meaning it has high modelling capacity; it is capable of modelling the underlying data generating process to some extent, and hence flexible enough to extrapolate to unseen data; it fits well into the existing NMF framework, meaning that sound sources can conveniently be added or removed from the dictionary $\mathbf{W}$.

The approach we are proposing has all of these desirable properties. Before we go on and discuss them in detail, we briefly review two much more basic ways of inferring the necessary prior information from an appropriate, additional dataset. One simple way to learn about the dictionary elements a priori, is to compute the mean spectral representation of individual sound sources. This is done by averaging over the individual frames of the spectrogram obtained from sources. Sometimes, one can already obtain reasonable decompositions with this straightforward technique. Another approach is to incorporate *all* the individual spectrogram frames for a sound source directly into an *overcomplete* dictionary. One can then take the sum of activations of these bases as an indicator for the presence of the sound source at a particular point in time, at the cost of additional computation.

Both of these methods have shortcomings. Simple averaging over example frames is too simplistic in most cases, and cannot adequately model realistic, high dimensional distributions. Working with overcomplete dictionaries becomes cumbersome quickly, due to both runtime and memory complexities of the decomposition, which directly depend on the number of dictionary entries — the dictionary can not grow indefinitely in general. Both methods are still *linear* and have difficulties generalizing to *unseen* data.

II. Proposed Method

We propose a novel, flexible and principled way to incorporate prior information about the spectral characteristics of individual sound sources into the non-negative matrix factorization framework. As in supervised NMF, we assume we have access to recordings of individual sound sources that are sufficiently similar to the ones that will appear in the actual signals we want to decompose. For each of these sources, we train a *normalizing flow* [6] that is capable of modelling the density of the spectrogram frames of this source. Instead of a fixed vector or a set of fixed vectors, to describe a sound source, we now have a parametrized density estimator, a kind of *differentiable dictionary* at our disposal. At decomposition time, we use a collection of these differentiable dictionaries to search for a mixture of spectral representations of the sound sources that simultaneously *minimizes reconstruction error* on the mixture, while *staying likely* with respect to the density of the spectrogram frames of the individual sound sources.

This approach enables us to decompose an input audio mixture into linear combinations of sound sources that are best described by nonlinear processes. One scenario with

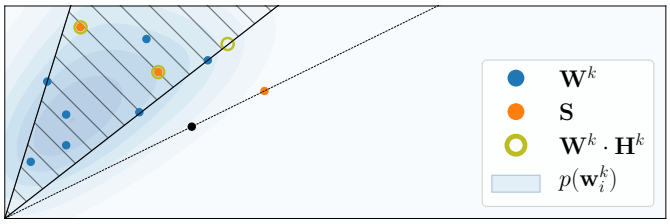

Fig. 1. A simplified two dimensional sketch to illustrate nonlinear extrapolation enabled by normalizing flows.

these characteristics is the decomposition of piano recordings into individual notes. The mixing process in a piano is predominantly linear [7], whereas the sound generating physical process is not [8]. We will use this scenario as our testbed to characterise the modelling capacity of the method.

A. Nonlinear Extrapolation

We will illustrate the difference between *linear interpolation* with an overcomplete dictionary and *nonlinear extrapolation* utilizing normalizing flows with the help of the sketch shown in Figure 1. Please note that this is a severely simplified example in two dimensions only, to visually support the description. The blue dots represent single feature vectors $\mathbf{w}_i^k$ from the training set that describe a particular sound source with index k, with $\mathbf{W}^k$ denoting the sub-matrix of the full dictionary $\mathbf{W}$ that contains only these feature vectors. Due to the non-negativity constraints of NMF, these vectors form a cone $\mathcal{C} = \{\mathbf{c} | \mathbf{c} = \mathbf{W}^k\mathbf{h}^k, \mathbf{W}^k \succcurlyeq 0, \mathbf{h}^k \succcurlyeq 0\}$. To reconstruct the three spectrogram frames $\mathbf{S}$ represented as orange dots in terms of this cone, and to *associate* them with a sound source, they need to lie *inside* this cone. This is the case for the orange dots inside the olive circles. Datapoints that lie *outside* this cone cannot be reconstructed as a non-negative linear combination of the sound source feature vectors without reconstruction error. As an unfortunate secondary effect, the strength of association with this source is diminished as well.

In contrast, a normalizing flow estimates the density of the feature vectors of the sound source. It models how the feature vectors were generated, it can easily *generate new samples* from this density, and is capable of *nonlinearly extrapolating* to unseen new feature vectors. An example of such an unseen feature vector is shown as a black dot. It is *still likely* under the density $p(\mathbf{w}_i^k)$ that is modeled by the flow, which is indicated by the blue contours. The extrapolation capabilities of the flow enable us to fully associate new, unseen, yet similar feature vectors to a given sound source, with high likelihood and almost zero reconstruction error.

B. Related Work

The concept of normalizing flows has been introduced in [9]. The particular kind of normalizing flow that we use is called *RealNVP*, and has been introduced as a parametric density model in [6]. Although Generative Adverserial Networks have been used in the context of audio decomposition [10]–[13] they do not allow explicit access to the likelihood of a data sample. The likelihood of a data sample under a Variational Autoencoder is cumbersome to approximate as well, and necessitates Monte Carlo approximation of an expectation of a lower bound on the likelihood [14].

III. Differentiable Dictionary Search

We introduce a novel, constrained dictionary adaptation method that we call *Differentiable Dictionary Search* (DDS). This method was devised to address the problems outlined in the introduction, when decomposing mostly linear mixtures of non-linearly behaving sound sources.

A. The DDS Model

We denote the magnitude spectrogram that we would like to decompose as $\mathbf{S} \in \mathbb{R}_+^{D \times T}$, having D spectral bins and T time frames. We denote the fixed dictionary with N entries as $\mathbf{W} \in \mathbb{R}^{D \times N}$, and the activation matrix as $\mathbf{H} \in \mathbb{R}^{N \times T}$. Supervised NMF seeks to approximate the spectral frame $\mathbf{s}_t \approx \sum_n h_t^n \mathbf{w}^n$, where h_t^n is the activation of the n-th dictionary entry $\mathbf{w}^n$ at time t. We will now describe how to integrate DDS with the NMF framework.

To estimate the density of the spectrogram frames for each sound source, we use a simplified version of RealNVP [6] without multi-scale architecture, batch normalization, or spatial masking. Instead, we use plain multi-layer perceptrons to parametrize the affine coupling layers, and randomly chosen, fixed permutation layers in between the coupling layers.

For each of K sound sources, we train a separate normalizing flow to obtain a parametrized density estimator f_k. A flow allows explicit evaluation of the likelihood $p(\mathbf{x})$ of a sample $\mathbf{x}$ by computing $p_Z(f_k(\mathbf{x}))$. A flow can generate samples by drawing $\mathbf{z} \sim p_Z$ and computing $f_k^{-1}(\mathbf{z})$, where $p_Z(\mathbf{z})$ is a prior distribution, which we choose to be a standard isotropic Gaussian $\mathcal{N}(\mu = \mathbf{0}, \mathbf{\Sigma} = I)$, following [6].

DDS approximates an arbitrary spectrogram frame $\mathbf{s}_t$ as a weighted sum of dictionary components $\mathbf{s}_t \approx \hat{\mathbf{s}}_t = \sum_k h_t^k \mathbf{w}_t^k$ of K sound sources. The k-th component is generated by the k-th flow as $\mathbf{w}_t^k = f_k^{-1}(\mathbf{z}_t^k)$. For decomposition, DDS updates the component activations $\mathbf{h}_t \in \mathbb{R}_+^K$ and the dictionary entries *in latent space* $\mathbf{Z}_t \in \mathbb{R}^{D \times K}$ that *generate components in data space* via the flows $\{f_k\}_{k=1}^K$, using (projected) gradient descent on the loss $\mathcal{L}$. Minimizing the loss jointly minimizes reconstruction error of the mixture, and maximizes likelihoods of individual components:

$$\mathcal{L}(\mathbf{s}_t, \hat{\mathbf{s}}_t) = \|\mathbf{s}_t - \hat{\mathbf{s}}_t\|_2 - \frac{c}{D \sum_k h_t^k} \sum_k h_t^k \log p_Z(\mathbf{z}_t^k) \quad (1)$$

The likelihood penalties $-\log p_Z(\mathbf{z}_t^k)$ on the latent vectors of individual sources are normalized to *nats per dimension* by $\frac{1}{D}$ and weighted by the activation components h_t^k normalized by their sum $\sum_k h_t^k$. The global likelihood penalty weight c is a hyperparameter of the decomposition and allows to fine-tune the behavior of the method, balancing reconstruction quality and deviations from *likely* dictionary entries. We provide an illustration of the process in Figure 2.

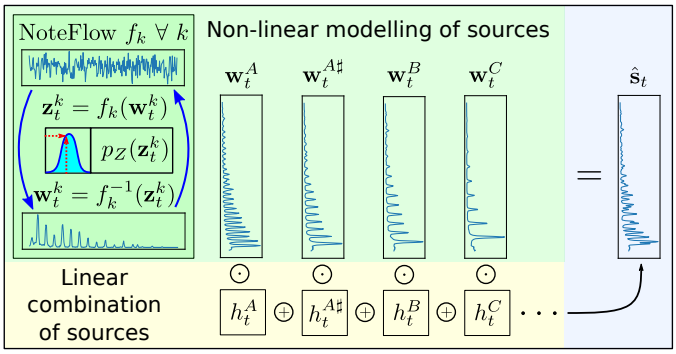

Fig. 2. An illustration of DDS decomposing a spectrogram frame.

B. Expected Benefits and Open Questions

Our design addresses the performance limitations of the strictly linear, overcomplete baseline when we are dealing with non-linearity of sound sources, while preserving interpretability. Due to DDS modelling the data generating process, the ability to represent arbitrary mixtures of similar sound sources should be considerably improved. Another benefit is the ability to better cope with the inevitable *distribution shift* between the data used for training the individual sound source models, and the data which is encountered during decomposition, while preserving association strength with a sound source.

To investigate the usefulness of the added non-linear extrapolation capability, we designed a set of highly controlled experiments aimed at revealing the differences between overcomplete NMF and DDS. Both methods need to approximate both the mixing and the reconstruction aspects of the signal-generating process. Both assume a linear mixing process; the approximate reconstruction of sound sources is entirely different, however. Thus, we will first study and compare the *reconstruction capabilities*. We quantitatively evaluate the differences in ability to generalize to new, unseen data for isolated sound sources in Section IV-A. This gives us a first appraisal of the beneficial aspects of DDS, before we compare the decomposition capabilities. We ensured equal train and test conditions for both methods, for a fair comparison.

Similarly, we would like to study the *discrimination ability* of individual NoteFlows in isolation. NoteFlows need to assign high likelihoods to unseen samples that are similar to the training samples, and low likelihoods to unseen and dissimilar samples. This is a necessary prerequisite for a clear decomposition into pre-defined components. We quantify the one-vs-all discrimination ability of the individual NoteFlows, by determining *likelihood thresholds*. These permit both soft and hard constraints on what samples can be generated by the sound source models, and hence facilitate calibration of the method. They act as a kind of *natural similarity measure* to the data generating process: we can minimize misattributions by constraining generated samples to have likelihoods below these thresholds. We will quantify one-vs-all discrimination on a set of models in Section IV-B. The complete DDS decomposition scheme will be put to a practical test in IV-C.

IV. Experimental Evaluation

In order to compare how the methods deal with *distribution shift* from sound sources seen during training to (similar) sound sources encountered during testing, we extract a set of notes played on various acoustic piano instruments, with different volume levels (also called "velocity", in MIDI jargon) and varying acoustics conditions. We divide the isolated notes into multiple splits consisting of two disjoint subsets: a training dataset and a test dataset. We draw upon a commercial VSTi sample-based synthesizer plugin, called "Spectrasonics KeyScape"[1], which contains a multitude of high quality samples of isolated notes played on different acoustic pianos in various microphone conditions and room reverberation settings, that provide a rich variety of timbre.

We sample the 4 notes A1, A2, A3, A4 and an additional full octave A2-A3 (12 notes). For each isolated note, we extract audio recordings played with 4 representative velocity values (32, 64, 96, 127) on *all* of the 43 available acoustic piano presets. We create 9 different splits by using different subsets of presets for training and testing, to create multiple distribution shift scenarios that challenge both methods, and serve as a realistic testbed displaying real-world characteristics.

We downsample the isolated note recordings to 16 [kHz] mono waveforms and compute logarithmic magnitude spectrograms with a Hann window of size 2048 and hop size 512. We only keep the lowest 512 resulting spectral bins that represent the frequency range from $[0; 4]$ [kHz], and normalize the magnitudes to the interval $[0; 1]$. Depending on the particular split, we have N spectrogram frames as 512-dimensional training examples, where N lies in the range $[3504, 25228]$.

To evaluate DDS, we train a RealNVP model for each isolated note in the training set. The models have 16 affine coupling layers, and each coupling function is realized as a multi-layer perceptron with 4 dense layers of 256 units with SELU activations [15]. The Adam optimizer [16] is used with a learning rate of $1 \cdot 10^{-3}$ to train each model for up to 1000 epochs, with a minibatch size of 512. The early stopping criterion was configured to have a patience of 50 epochs. Density estimation performance is monitored by computing the mean log-likelihood over a held-out validation set. The validation set consists of 20% randomly selected training samples. We refer to these models, trained on isolated notes, as "NoteFlows" throughout this manuscript. The parameters of each NoteFlow are fixed during decomposition.

We compare DDS to an overcomplete variant of NMF, with its dictionary $\mathbf{W}$ initialized to the training set and held fixed. The activation matrix $\mathbf{H}$ is initialized to constant values of $\frac{1}{N}$ where N is number of components given by size of the training set.

NoteFlows generate spectrogram frames determined by the noise vectors $\mathbf{z}_t^k$. We initialize these noise vectors to $\mathbf{0}$ and set the global likelihood penalty weight (see Eq.1) to $c = 1 \cdot 10^{-3}$ for all experiments that follow.

[1]https://www.spectrasonics.net/products/keyscape/

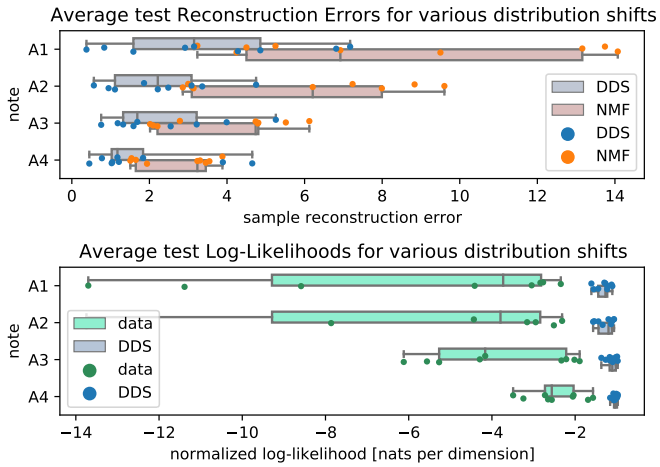

Fig. 3. Quantitative assessment of robustness to distribution shift of the compared methods DDS and overcomplete NMF.

The decomposition is run for a maximum of 10000 update steps for both methods, overcomplete NMF and DDS. Early stopping is possible, as soon as the cost term changes by less than $\epsilon = 1 \cdot 10^{-15}$, between update steps. If the cost term fluctuates, but there is no progress for 10 consecutive steps, the learning rate is reduced by a factor of 2. For all decompositions, the Adam [16] optimizer is used. After each update step, the current solution is projected back into the non-negative orthant to satisfy all constraints.

A. Non-linear Extrapolation

After DDS is trained using the aforementioned procedure, and the dictionary matrix of overcomplete NMF is initialized with the spectrogram frames of the same train set, we study the reconstruction error of each method on new, unseen sound sources. This is shown in the upper half of Figure 3, where each data point represents the average reconstruction error over the test samples of a particular split. In the lower half of Figure 3 we look at the tradeoff between reconstruction error and sample likelihood,. The likelihoods of test samples with zero reconstruction error (labeled as `data`) are contrasted with the likelihoods of test samples where a small amount of reconstruction error is allowed, while the sample is kept likely (labeled as `DDS`).

We can see clear evidence for the advantage of DDS over the overcomplete, linear NMF baseline when it comes to reconstruction quality. This is especially apparent for the low note A1, but can be observed across all four octaves. The lower half of Figure 3 shows a noticable decrease in likelihood if one insists on a perfect reconstruction. What we can observe is that DDS is perfectly capable of producing reconstructions with *low error* while still staying *close* to the training data distribution.

B. Discrimination Ability of NoteFlows

To assess the potential for DDS to wrongly attribute spectral activity, we measure *one-sided discriminativeness* on one

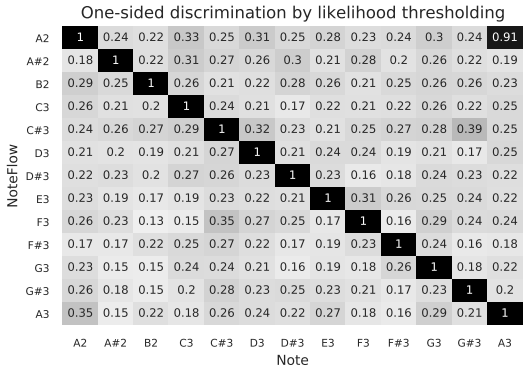

Fig. 4. Confusion of NoteFlows in terms of likelihood-based one-sided discrimination ability between their "correct" notes, and all the other notes.

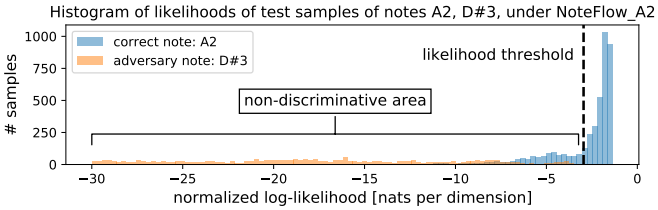

Fig. 5. Illustration of likelihood thresholding in definition of one-sided discriminativeness quantifier.

representative split. Given a model of source k and a set of N test samples $\mathbf{x}^k$ similar to source k as well as test samples $\mathbf{x}^j$ of a different source, the one-sided discriminativeness measure is defined as the ratio of the number of samples of source k (blue histogram in Figure 5) below the "likelihood threshold", to the number of *all* samples of source k (entirety of blue area in Figure 5). The likelihood threshold θ is defined as the *maximum* over the likelihoods of the samples of the *different* source j under the NoteFlow model for source k. This leads to the expression $d_{os} = \left(\sum_{n=1}^{N}[p(\mathbf{x}_n^k) < \theta]\right)/N$ and, intuitively speaking, measures how many of the samples that the NoteFlow should model as "likely" – using likelihood as a kind of natural similarity measure – are confused with samples of some other source, that should be deemed "unlikely" under this particular NoteFlow.

We trained 13 NoteFlows to model one full octave of piano notes, and computed the one-sided discriminative measure for all of them, using unseen data from the test set. The entries in the confusion matrix in Figure 4 are the ratios as previously defined. Lower numbers are better. The main diagonal is always one by definition. Out of curiosity, we computed the one-sided discriminativeness with samples solely from the training data, which yielded a confusion matrix with *all* entries very close to zero. This shows that the method is capable of explaining the majority of spectral evidence by assigning correct sources. We find this encouraging, and note that there is still room for improvement regarding the generalization capabilities of the parametric density estimation models we currently employ.

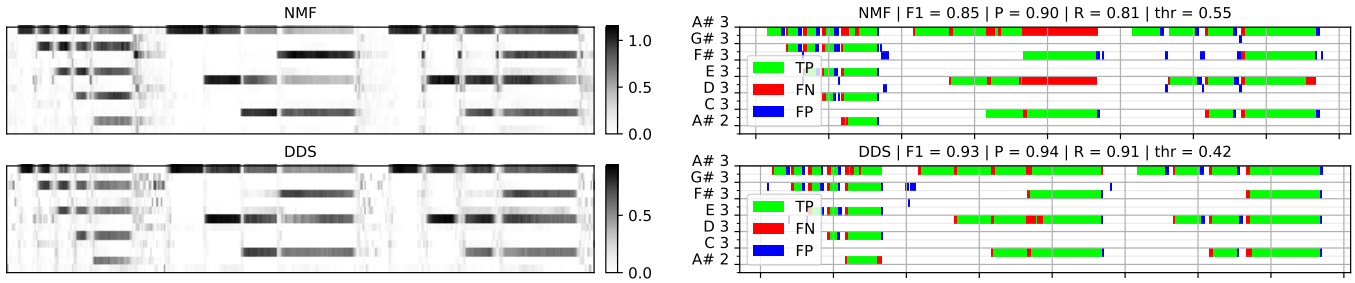

Fig. 6. Demonstration of semantic decomposition capability. Activation matrices; on the left. Thresholded activation matrices visualizing the underlying computation of the (M)IR metrics by highlighting where True Positives (TP), False Negatives (FN) and False Positives (FP) are; on the right.

C. Semantic Decomposition

Finally, we compare the abilities of DDS and overcomplete NMF to decompose a short piece of polyphonic piano music into its individual notes. We rendered the piece twice, once with an instrument seen during training, and once with a new, unseen instrument, using the same splits as in Section IV-B. Each method needs a global threshold to binarize the activity matrix $\mathbf{H}$, so the F1-score for frame-level source attribution can be computed. The optimal threshold for each method is a trainable parameter, and is determined on the training piece. During test time the two respective thresholds are held fixed and used to binarize the activity matrices of the decompositions. Finally, we compute the F1-score for the binary activity matrices on the test piece.

The results of these decompositions are shown in Figure 6. The left column shows the raw activations of source components for each method. For the overcomplete NMF approach, each row of the activation matrix is computed as the sum of the activities of all dictionary entries belonging to a given source. The right column shows the binarized activations, contrasted with the ground truth. The resulting F1-score can be found in the titles.

V. Conclusion

We observe that despite the possibility for confusion of sources, as measured in Figure 4, DDS demonstrates its ability to strongly associate spectral activity to the appropriate source. This demonstrates a clear improvement of decomposition performance over the linear, overcomplete NMF baseline, as measured by the frame-level F1 metric. The improvement can be directly attributed to the great improvement in recall, and we interpret this improvement as direct evidence for the usefulness of introducing non-linear extrapolation capability into the general NMF framework to jointly achieve clearer decompositions and better reconstructions of the sources.

References

[1] D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, pp. 788–791, 1999.

[2] P. Smaragdis, "Non-negative matrix factor deconvolution; extraction of multiple sound sources from monophonic inputs," in *Independent Component Analysis and Blind Signal Separation, Fifth International Conference, ICA 2004, Granada, Spain*, vol. 3195. Springer, 2004, pp. 494–499.

[3] D. L. Donoho and V. Stodden, "When does non-negative matrix factorization give a correct decomposition into parts?" in *Advances in Neural Information Processing Systems 16 [Neural Information Processing Systems, NIPS 2003, Vancouver and Whistler, British Columbia, Canada]*. MIT Press, pp. 1141–1148.

[4] P. Smaragdis, B. Raj, and M. V. S. Shashanka, "Supervised and semi-supervised separation of sounds from single-channel mixtures," in *Independent Component Analysis and Signal Separation, 7th International Conference, ICA 2007, London, UK*, vol. 4666. Springer, 2007, pp. 414–421.

[5] J. Mairal, F. R. Bach, and J. Ponce, "Task-driven dictionary learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 4, pp. 791–804, 2012.

[6] L. Dinh, J. Sohl-Dickstein, and S. Bengio, "Density estimation using Real NVP," in *5th International Conference on Learning Representations, ICLR 2017, Toulon, France*.

[7] K. Ege, X. Boutillon, and M. Rébillat, "Vibroacoustics of the piano soundboard: (Non)linearity and modal properties in the low- and mid-frequency ranges," *Journal of Sound and Vibration*, vol. 332, no. 5, pp. 1288 – 1305, 2013.

[8] B. Bank and H.-M. Lehtonen, "Perception of Longitudinal Components in Piano String Vibrations," *The Journal of the Acoustical Society of America*, vol. 128, no. 3, pp. EL117–EL123, 2010.

[9] E. G. Tabak and E. Vanden-Eijnden, "Density estimation by dual ascent of the log-likelihood," *Communications in Mathematical Sciences - COMMUN MATH SCI*, vol. 8, 03 2010.

[10] Y. C. Sübakan and P. Smaragdis, "Generative adversarial source separation," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2018, Calgary, AB, Canada*. IEEE, 2018, pp. 26–30.

[11] L. Li, H. Kameoka, and S. Makino, "Determined audio source separation with multichannel star generative adversarial network," in *30th IEEE International Workshop on Machine Learning for Signal Processing, MLSP 2020, Espoo, Finland*. IEEE, 2020, pp. 1–6.

[12] D. Stoller, S. Ewert, and S. Dixon, "Adversarial semi-supervised audio source separation applied to singing voice extraction," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2018, Calgary, AB, Canada*. IEEE, 2018, pp. 2391–2395.

[13] R. Tanabe, Y. Ichikawa, T. Fujisawa, and M. Ikehara, "Music source separation with generative adversarial network and waveform averaging," in *53rd Asilomar Conference on Signals, Systems, and Computers, ACSCC 2019, Pacific Grove, CA, USA*. IEEE, pp. 1796–1800.

[14] Y. Bando, M. Mimura, K. Itoyama, K. Yoshii, and T. Kawahara, "Statistical speech enhancement based on probabilistic integration of variational autoencoder and non-negative matrix factorization," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2018, Calgary, AB, Canada*, pp. 716–720.

[15] G. Klambauer, T. Unterthiner, A. Mayr, and S. Hochreiter, "Self-normalizing neural networks," in *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, Long Beach, CA, USA*, pp. 971–980.

[16] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA*.