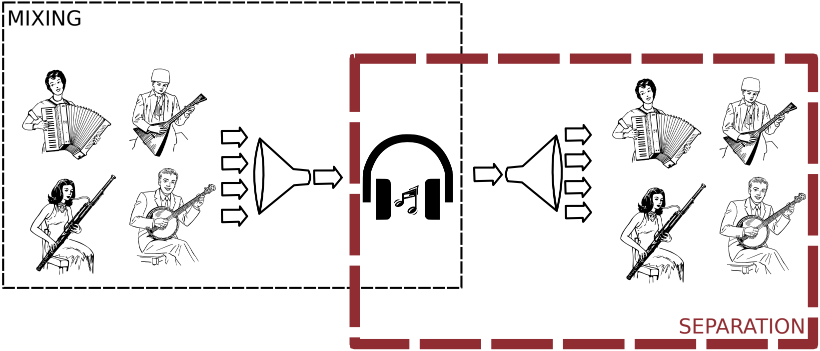
Scalable audio separation with light Kernel Additive Modelling

Antoine Liutkus[1], Derry Fitzgerald[2], Zafar Rafii[3]

[1]Inria, Université de Lorraine, LORIA, UMR 7503, France
[2]NIMBUS Centre, Cork Institute of Technology, Ireland
[3]Gracenote, Media Technology Lab, Emeryville, CA, USA

ICASSP, Brisbane, April 21^{st} 2015

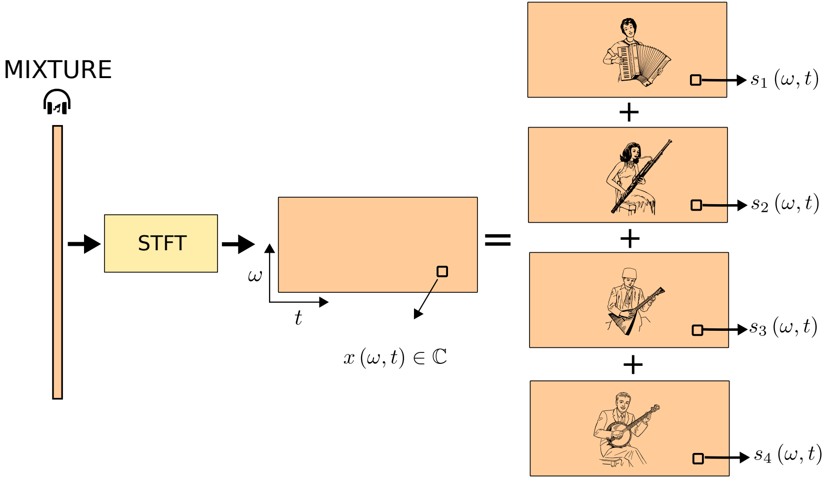
Separating audio sources



In this presentation: mono mixtures

⇒ General multichannel case in the paper

Notations


MIXTURE
STFT
ω
t
$x(\omega, t) \in \mathbb{C}$
=
$s_1(\omega, t)$
+
$s_2(\omega, t)$
+
$s_3(\omega, t)$
+
$s_4(\omega, t)$

Time frequency masking

$\hat{s}_1(\omega,t)$

$=$

$x(\omega,t)$

$\circ$

$w_1(\omega,t)$

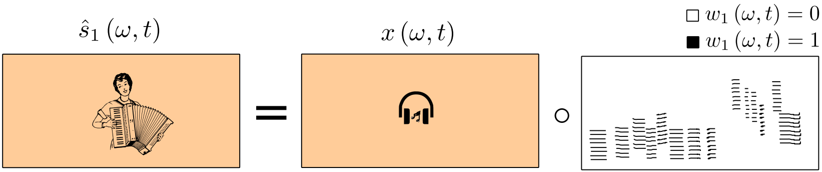
- Each source STFT $s_j(\omega,t)$ is obtained by *filtering* the mixture

$$\hat{s}_j(\omega,t) = x(\omega,t)\, w_j(\omega,t)$$

- Underdetermined separation $\Rightarrow$ w_j varies with both ω and t
- Waveforms obtained by inverse STFT

Many different ways to get a Time-Frequency (TF) mask $w_j(\omega,t)$

Time frequency masking



- $s_j(\omega, t)$ is assumed equal either to $x(\omega, t)$or to 0
- A **classification task** over the mixture STFT x

  $\Rightarrow$ based on **features**

  - pitch detection+harmonics selection (CASA)
  - panning position (DUET)

Y. Han and C. Raphael. Informed source separation of orchestra and soloist. In *Proceedings of the 11th International Society for Music Information Retrieval Conference (ISMIR)*, pages 315–320, 2010

O. Yilmaz and S. Rickard. Blind separation of speech mixtures via time-frequency masking. *IEEE Trans. on Signal Processing*, 52(7):1830–1847, 2004

Getting the mask

Binary masking yields **musical noise**
$\Rightarrow$ Soft masking $w_j(\omega, t) \in [0\,1]$ is better!

Example: Wiener filtering for Gaussian processes

- Sources energies $p_j(\omega, t) \geq 0$ add up to get mix energy

$$\sum_j p_j(\omega, t)$$

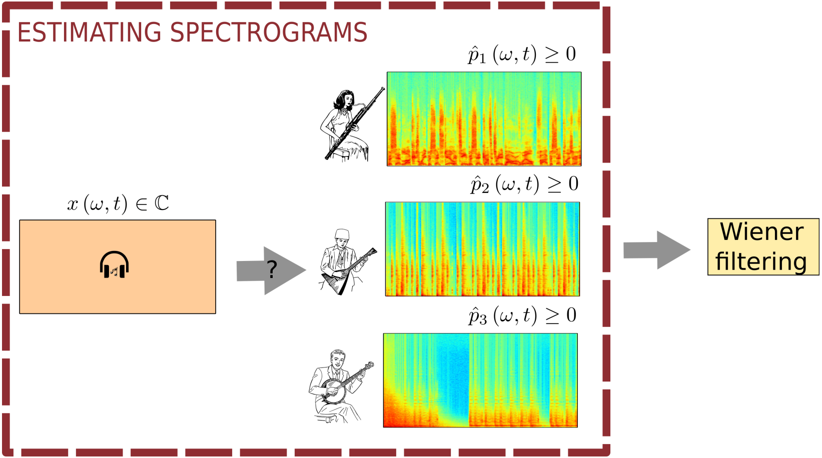
- $w_j(\omega, t)$ taken as proportion of source j in mix

$$w_j(\omega, t) = \frac{p_j(\omega, t)}{\sum_{j'} p_{j'}(\omega, t)} \in [0\,1]$$

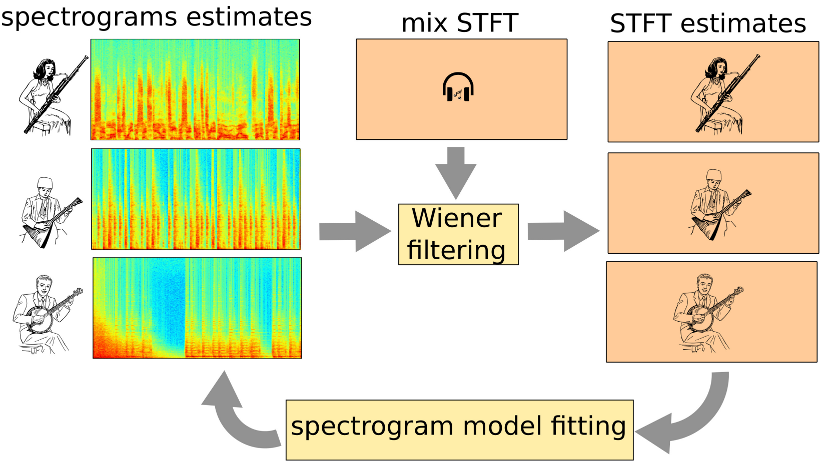
L. Benaroya, F. Bimbot, and R. Gribonval. Audio source separation with a single sensor. *IEEE Trans. on Audio, Speech and Language Processing*, 14(1):191–199, January 2006

Time-Frequency masking

challenges



Iterative approaches
main ideas



The need for spectrograms models

- For each time frequency bin (ω, t)
  - $\rightarrow$ we have J unknowns $p_j(\omega, t) \geq 0$
  - $\rightarrow$ we have 1 observation $x(\omega, t) \in \mathbb{C}$
  - $\Rightarrow$ **The problem is ill-posed**

$\Rightarrow$ We need to:

- $\rightarrow$ exploit redundancies (e.g. multichannel data)
- $\rightarrow$ reduce the number of parameters

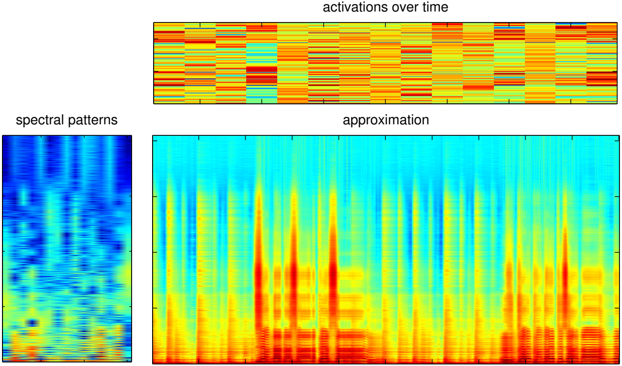
We should use prior knowledge on p_j

$\Rightarrow$ exploit **expected structure of spectrograms**

N.Q.K. Duong, E. Vincent, and R. Gribonval. Under-determined reverberant audio source separation using a full-rank spatial covariance model. *Audio, Speech, and Language Processing, IEEE Transactions on*, 18(7):1830–1840, sept. 2010

Global spectrogram models

nonnegative matrix factorization



A. Ozerov, E. Vincent, and F. Bimbot. A general flexible framework for the handling of prior information in audio source separation. *Audio, Speech, and Language Processing, IEEE Transactions on*, PP(99):1, 2011

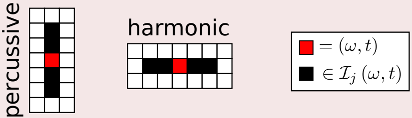
Y. Salaün, E. Vincent, N. Bertin, N. Souviraà-Labastie, X. Jaureguiberry, D. Tran, and F. Bimbot. The Flexible Audio Source Separation Toolbox (FASST) version 2.0. In *ICASSP*, 2014

Kernel spectrogram models
principles

- NMF is a **global** single model for all of p_j
- Sometimes, our knowledge is only **local**
  $\Rightarrow$ We assume $p_j(\omega, t)$ is equal to some **neighbours** $\mathcal{I}_j(\omega, t)$

Example: harmonic/percussive local models

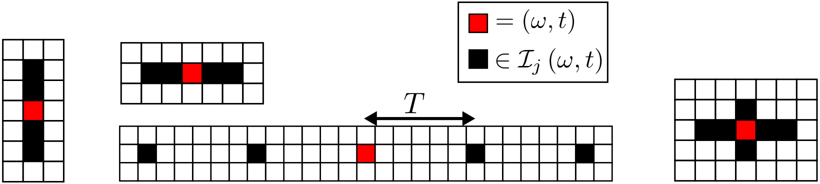
- Percussive sounds are locally constant through frequency
- Harmonic sounds are locally constant through time



D. Fitzgerald. Harmonic/percussive separation using median filtering. In *Proc. of the 13th Int. Conference on Digital Audio Effects (DAFx-10)*, Graz, Austria, September 2010

Kernel spectrogram models
examples

$$\forall\left(\omega', t'\right) \in \mathcal{I}_j\left(\omega, t\right),\ p_j\left(\omega', t'\right) \approx p_j\left(\omega, t\right)$$



D. Fitzgerald. Harmonic/percussive separation using median filtering. In *Proc. of the 13th Int. Conference on Digital Audio Effects (DAFx-10)*, Graz, Austria, September 2010

Z. Rafii and B. Pardo. A simple music/voice separation method based on the extraction of the repeating musical structure. In *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*, pages 221 –224, may 2011

D. FitzGerald. Vocal separation using nearest neighbours and median filtering. In *Proceedings of the 23nd IET Irish Signals and Systems Conference*, pages 583–588, Maynooth, 2012

Z. Rafii and B. Pardo. Music/voice separation using the similarity matrix. In *Proceedings of the 13th International Conference on Music Information Retrieval (ISMIR)*, pages 583–588, 2012

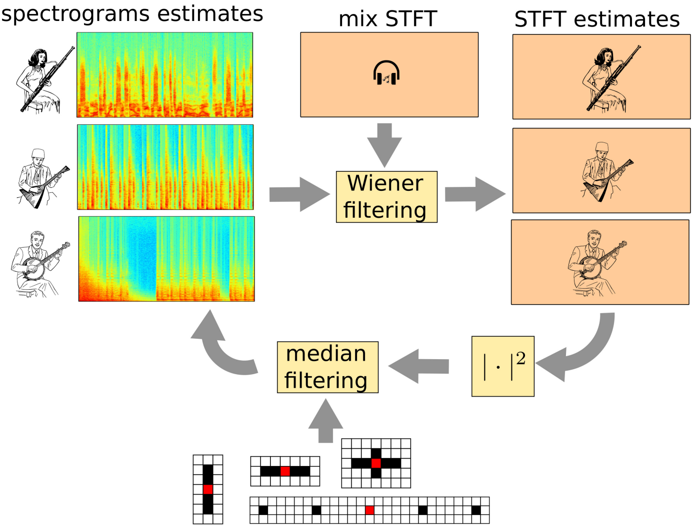
Kernel spectrogram models
objective

Combining all those local models together!

Example: voice/music separation

- Musical background
  5 sources repeating at different scales (beat, downbeat, ...)
  +1 source which is stable along time (strings, synths)
- Voice
  with a locally constant spectrogram (cross-like kernel)

Kernel backfitting algorithm



Kernel backfitting algorithm

monochannel *standard* version

Input

Mixture STFT $x(\omega, t)$

Neighbourhoods $\mathcal{I}_j(\omega, t)$, also called "proximity kernels"

Initialization:

$\forall j, \hat{p}_j(\omega, t) \leftarrow |x(\omega, t)|^2$: simply take mix spectrogram

Iterate

Separation with Wiener filtering

compute

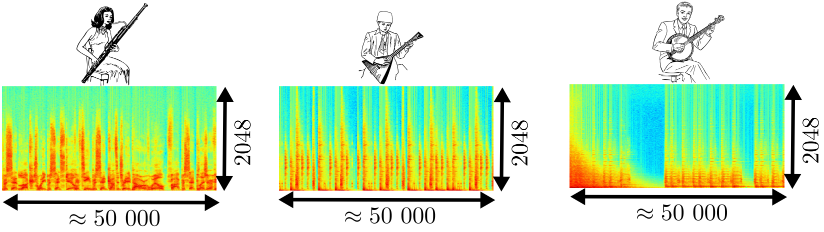
estimates $\hat{s}_j(\omega, t) = \left[\hat{p}_j(\omega, t) / \sum_{j'} \hat{p}_{j'}(\omega, t)\right] x(\omega, t)$

Spectrograms fitting

$\hat{p}_j(\omega, t) \leftarrow$ median filter $|\hat{s}_j|^2$ with kernel $\mathcal{I}_j(\omega, t)$

Output: source estimates $\hat{s}_j$

Scalability issues



Kernel models: no compact parameterization

⇒ all spectrograms must be stored in full resolution

for 10 sources and a full length track:
approximately 32Gb of RAM needed!

Low-rank models for compression

$$p_j(\omega, t) = \sum_{k=1}^{K} W(\omega, k) H(k, t)$$

Different possible approaches

Nonnegative matrix factorization

- W and H have nonnegative entries
- meaningful decompositions, but **slow**

Truncated singular values decompositons (SVD)

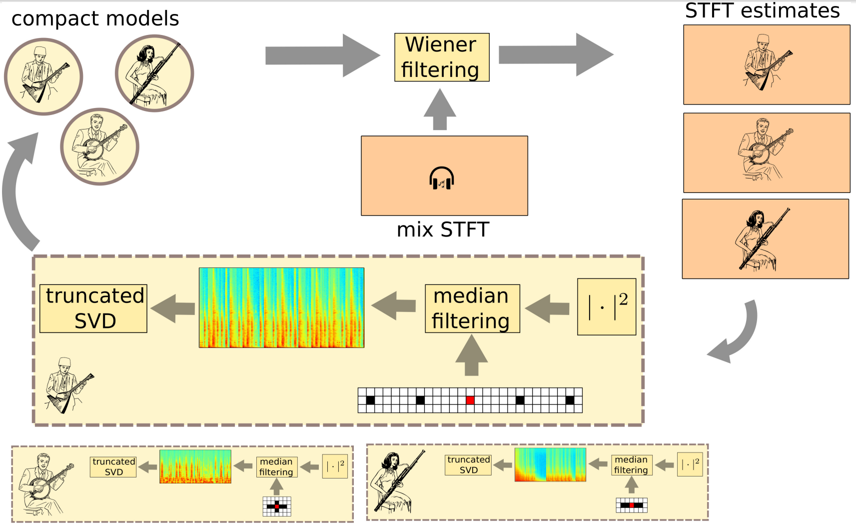
- W and H are real
- not physically meaningful, but **fast**

After filtering, **compress** spectrograms with a low-rank model

A. Ozerov, E. Vincent, and F. Bimbot. A general flexible framework for the handling of prior information in audio source separation. *Audio, Speech, and Language Processing, IEEE Transactions on*, PP(99):1, 2011

N. Halko, P. Martinsson, and J. Tropp. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM review*, 53(2):217–288, 2011

Kernel Additive Modelling Light (KAML)



Demo

external demo

Kernel Additive Modelling: conclusion

- A general framework for combining different kernel models
- Handles multichannel, full-length mixtures
- Easy to implement and fast algorithms
  $\Rightarrow$ full demo at `www.loria.fr/~aliutkus/kaml/`

To go further

Formalization

$\Rightarrow$ optimization framework with robust cost-functions

$\Rightarrow$ equivalence with EM algorithm in some cases

Combination with other techniques

Learning source kernels automatically?

$\Rightarrow$ maximizing size of kernel (robustness)

$\Rightarrow$ maximizing invariance to median filtering