A Diagonal Plus Low-Rank Covariance Model for Computationally Efficient Source Separation

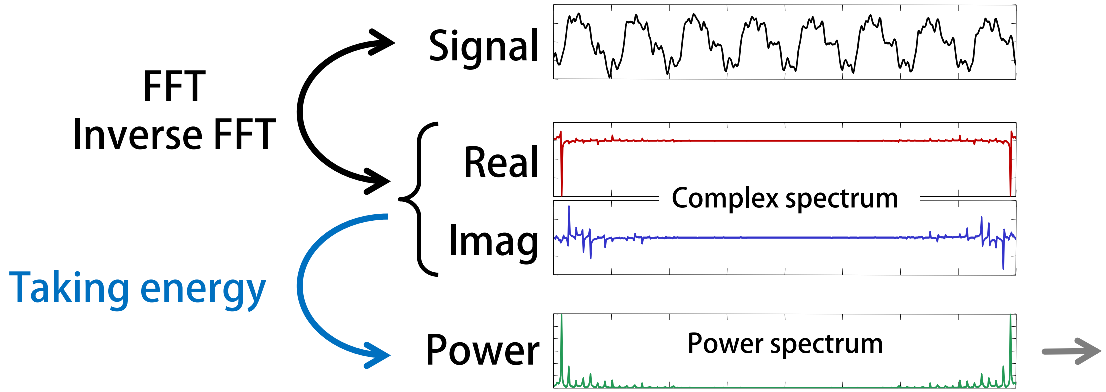
Antoine Liutkus (INRIA, France)
Kazuyoshi Yoshii (Kyoto University/RIKEN AIP, Japan)

Outline

- We introduce positive semidefinite tensor factorization (PSDTF) based on the Log-Det divergence
  - A natural extension of nonnegative matrix factorization (NMF) based on the Itakura-Saito divergence
  - Estimation of locally-stationary Gaussian processes
- We propose a constrained version of LD-PSDTF for reducing computational complexity
  - Kernel matrices are restricted to diagonal + low-rank matrices
  - Woodbury formula is used for inversing kernel matrices

Background

- Source separation is essential for various applications
  - Speech recognition and understanding
  - Automatic music transcription
- Phase information has not been used in most studies
  - The characteristics of sounds can be represented well in the power domain by discarding the phase information
  - The low-rankness and sparseness are useful clues



Nonnegative matrix factorization (NMF)
- Interpretable
- Easy to implement
- Efficient

Motivation

- Phase-aware source separation is promising
  - NMF can be extended based on additivity of complex spectra

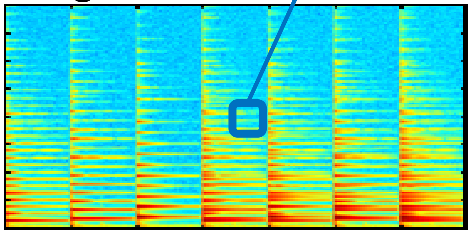
	Frequency bins	Time frames
Complex NMF [Kameoka 2009]	Independent	Independent
High Resolution NMF [Badeau 2011]	Independent	Autoregressive
PSDTF [Yoshii 2013]	Correlated	Independent

Additivity of time-domain signals

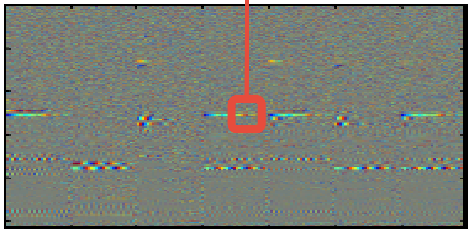
Complex value

$$x_{ft} = r_{ft}(\cos\theta_{ft} + i\sin\theta_{ft})$$

Magnitude r



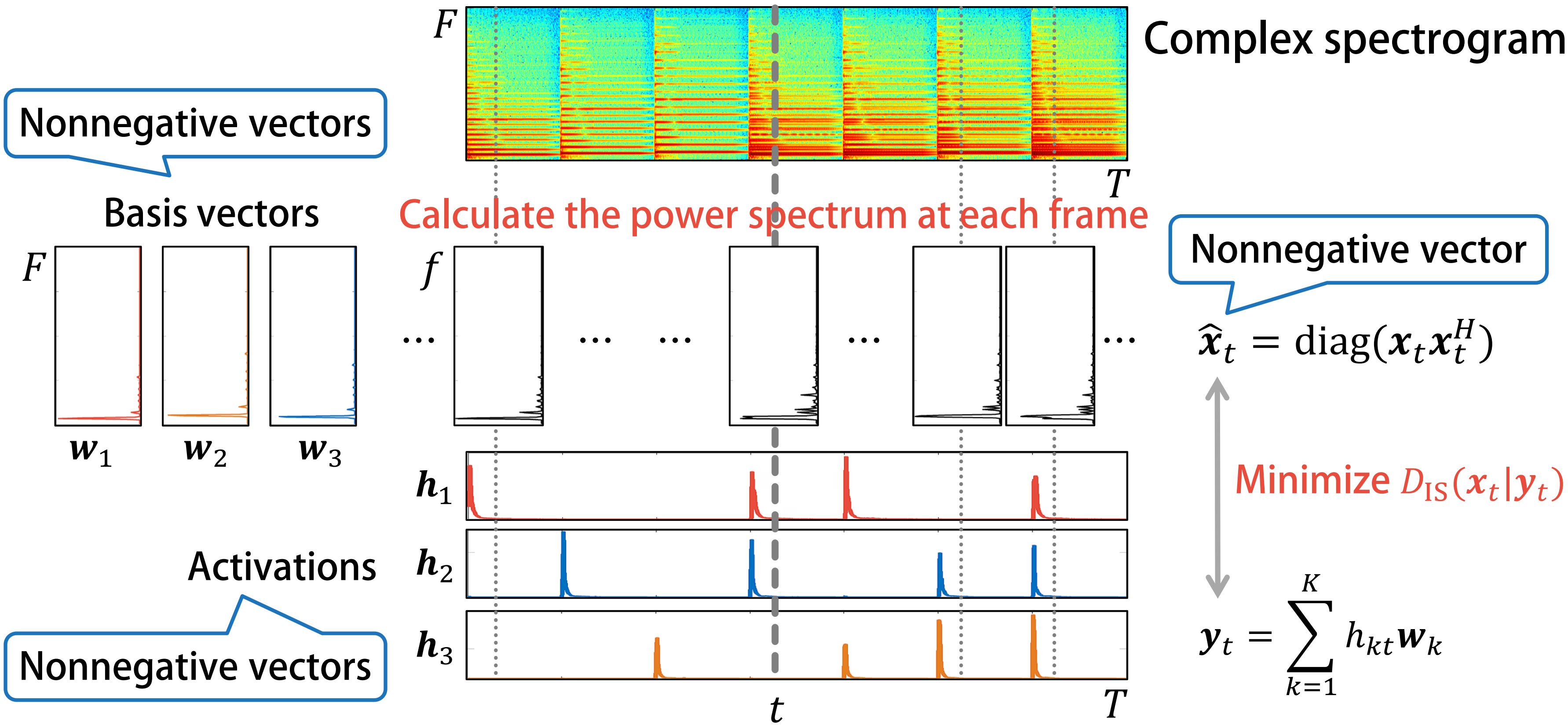
Phase θ



The values of magnitude and phase are not determined independently at frequency bins

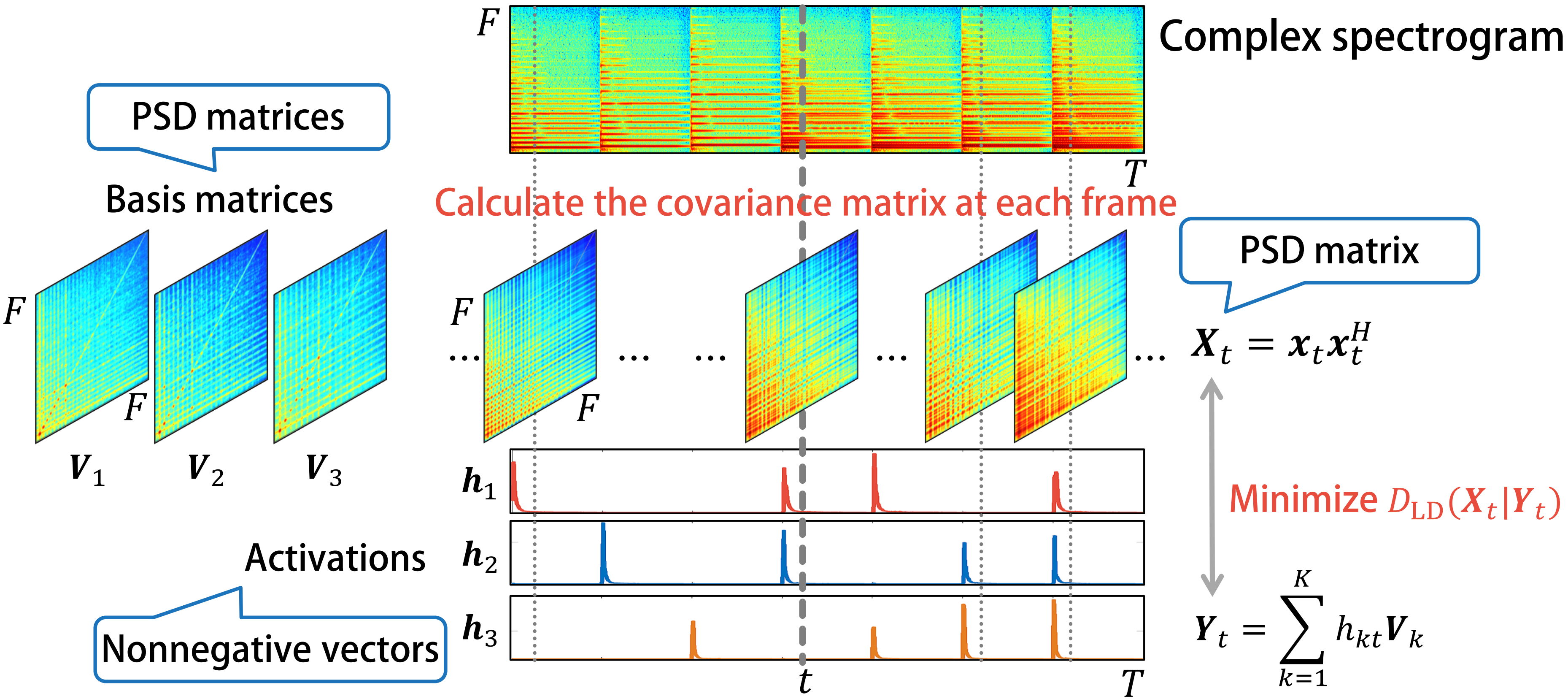
Itakura-Saito NMF (IS-NMF) [Févotte 2009]

- Each observed nonnegative vector is approximated as the weighted sum of basis nonnegative vectors



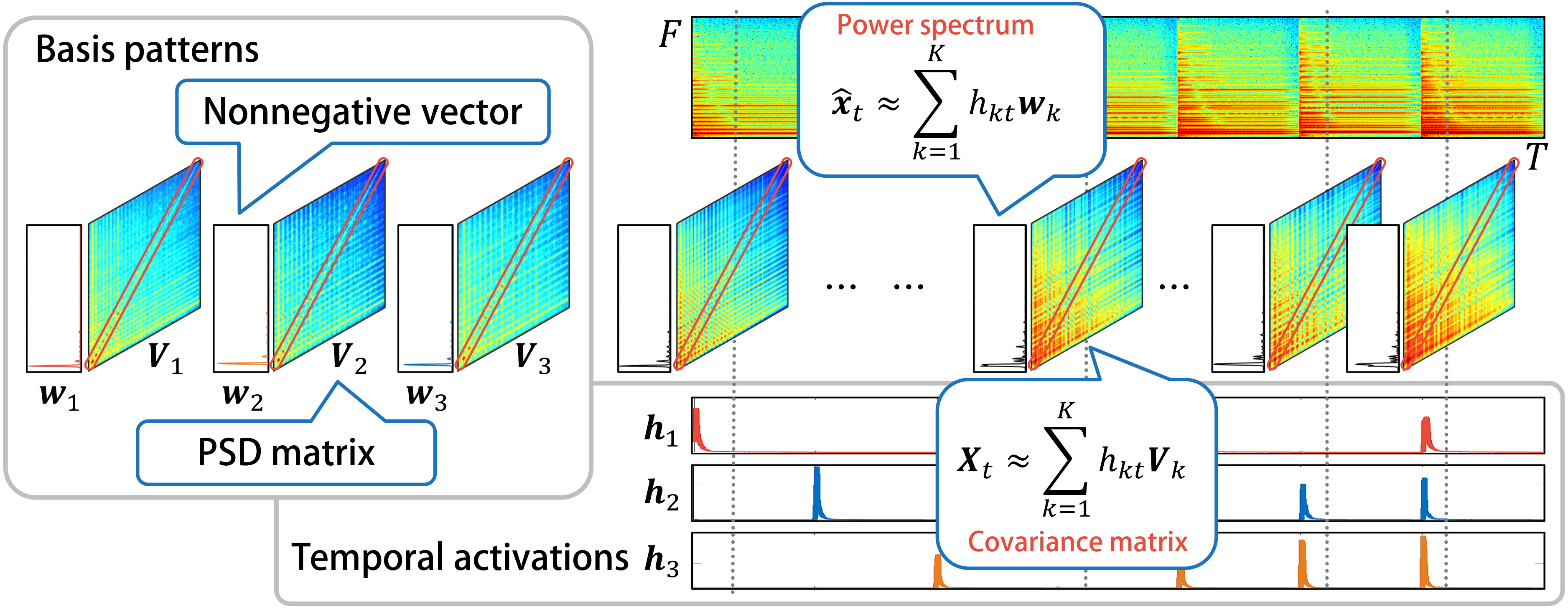
Log-Det PSDTF (LD-PSDTF) [Yoshii 2013]

- Each observed pos. semidef. matrix is approximated as the weighted sum of basis pos. semidef. matrices



IS-NMF vs LD-PSDTF

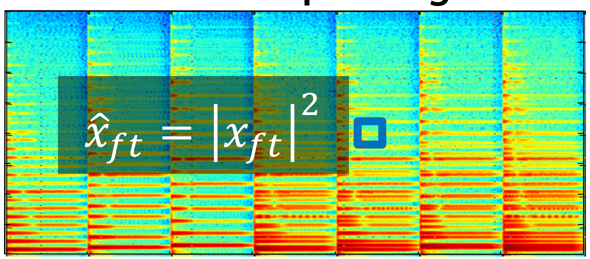
- LD-PSDTF is a natural extension of IS-NMF
  - Nonnegativity of scalars → Positive semidefinitenss of matrices
  - PSDTF reduces to NMF when all PSD matrices are diagonal



Itakura-Saito NMF (IS-NMF) [Févotte 2009]

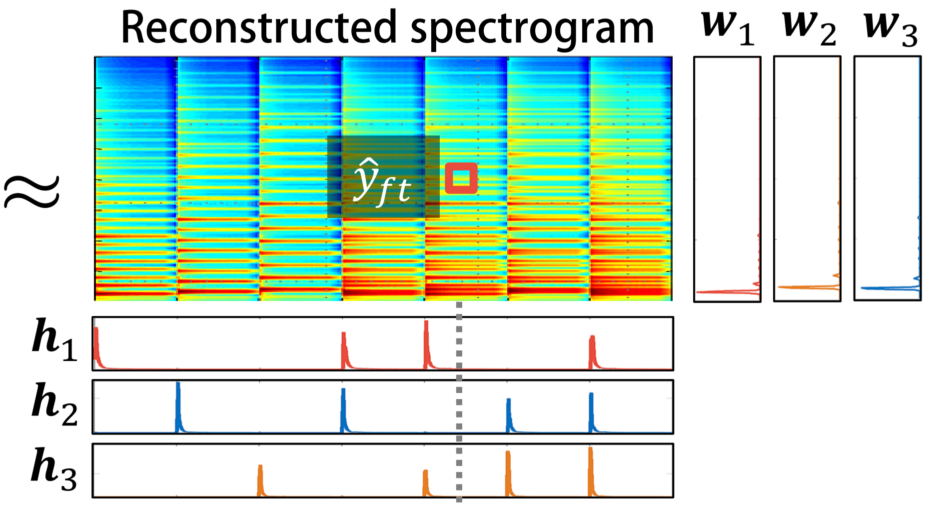
- NMF based on the Itakura-Saito divergence
  - The mixture spectrogram is approximated as a low-rank matrix
  - The number of sources K should be specified in advance

Observed spectrogram



$\approx$

Reconstructed spectrogram



$$\hat{x}_{ft} \approx \hat{y}_{ft} = \sum_{k=1}^{K} w_{kf} h_{kt}$$

$$D_{\mathrm{IS}}(\hat{x}_{ft} | \hat{y}_{ft}) = -\log \frac{\hat{x}_{ft}}{\hat{y}_{ft}} + \frac{\hat{x}_{ft}}{\hat{y}_{ft}} - 1$$

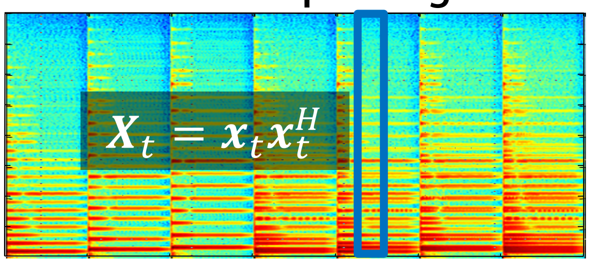
Scale-invariant measure

$$D_{\mathrm{IS}}(\hat{x}_{ft} | \hat{y}_{ft}) = D_{\mathrm{IS}}(\alpha\hat{x}_{ft} | \alpha\hat{y}_{ft})$$

Log-Det PSDTF (LD-PSDTF) [Yoshii 2013]

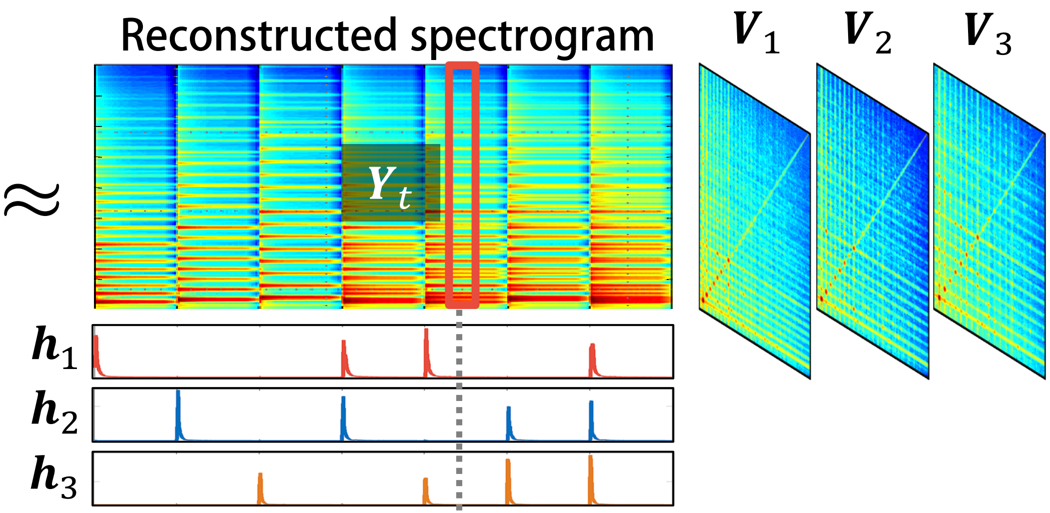
- PSDTF based on the log-determinant divergence
  - The covariance matrix at each frame is approximated as the weighted sum of covariance matrices (basis matrices)

Observed spectrogram



$\approx$

Reconstructed spectrogram



$$X_t \approx Y_t = \sum_{k=1}^{K} V_k h_{kt}$$

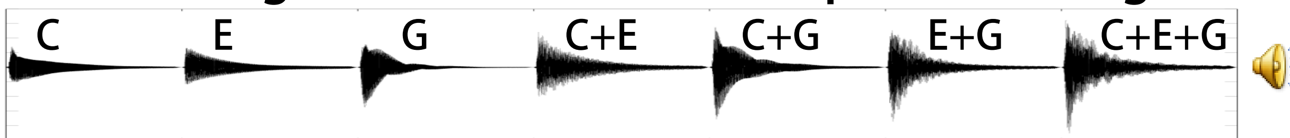
$$D_{\mathrm{LD}}(X_t | Y_t) = -\log|X_t Y_t^{-1}| + \mathrm{tr}(X_t Y_t^{-1}) - F$$

Scale-invariant measure

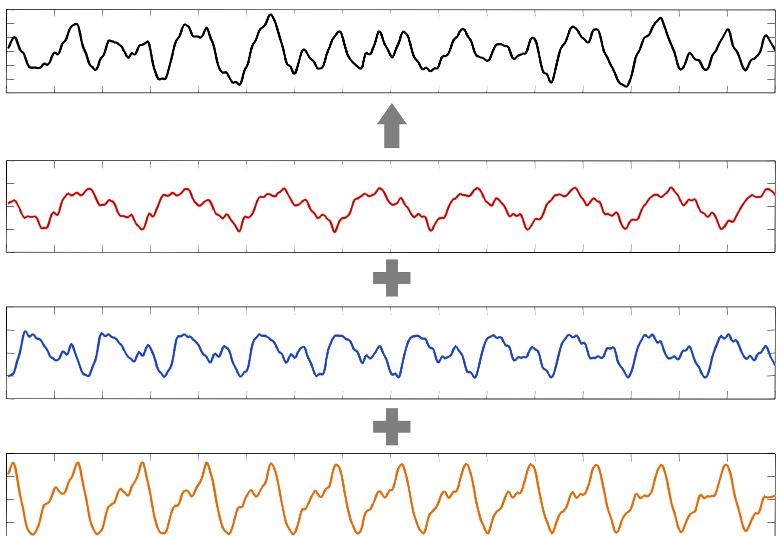
$$D_{\mathrm{LD}}(X_t | Y_t) = D_{\mathrm{LD}}(\alpha X_t | \alpha Y_t)$$

Probabilistic Formulation

- The source signals are assumed to follow independent locally-stationary Gaussian processes in the time domain
  - A mixture signal is the sum of multiple source signals



Assume the signals to be stationary in a short window t



Mixture signal: $\boldsymbol{x}_t = \boldsymbol{x}_{1t} + \boldsymbol{x}_{2t} + \boldsymbol{x}_{3t}$

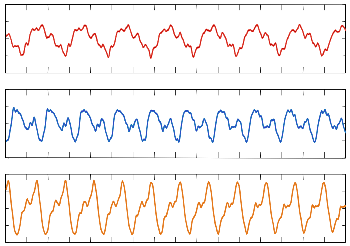
Source signal 1 (C): $\boldsymbol{x}_{1t} \sim N_c(0, h_{1t}\boldsymbol{V}_1)$

Source signal 2 (E): $\boldsymbol{x}_{2t} \sim N_c(0, h_{2t}\boldsymbol{V}_2)$

source signal 3 (G): $\boldsymbol{x}_{3t} \sim N_c(0, h_{3t}\boldsymbol{V}_3)$

Mixing Process & Demixing Process

- Sum of Gaussian variables → Gaussian variable



$\boldsymbol{x}_{1t} \sim N_c(0, \boldsymbol{Y}_{1t})$

$\boldsymbol{x}_{2t} \sim N_c(0, \boldsymbol{Y}_{2t})$

$\boldsymbol{x}_{3t} \sim N_c(0, \boldsymbol{Y}_{3t})$

?

$\boldsymbol{x}_t = \boldsymbol{x}_{1t} + \boldsymbol{x}_{2t} + \boldsymbol{x}_{3t}$

$\sim N_c(0, \boldsymbol{Y}_{1t} + \boldsymbol{Y}_{3t} + \boldsymbol{Y}_{3t} = \boldsymbol{Y}_t)$

- Gaussian variable → Sum of Gaussian variables

?

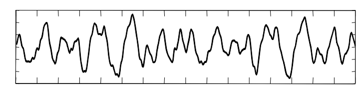
?

?

$\boldsymbol{x}_{1t} \sim N_c(0, \boldsymbol{Y}_{1t})$

$\boldsymbol{x}_{2t} \sim N_c(0, \boldsymbol{Y}_{2t})$

$\boldsymbol{x}_{3t} \sim N_c(0, \boldsymbol{Y}_{3t})$



$\boldsymbol{x}_t = \boldsymbol{x}_{1t} + \boldsymbol{x}_{2t} + \boldsymbol{x}_{3t}$

$\sim N_c(0, \boldsymbol{Y}_{1t} + \boldsymbol{Y}_{3t} + \boldsymbol{Y}_{3t} = \boldsymbol{Y}_t)$

$$\boldsymbol{x}_{1t}|\boldsymbol{x}_t \sim N_c(\boldsymbol{Y}_{1t}\boldsymbol{Y}_t^{-1}\boldsymbol{x}_t, \boldsymbol{Y}_{1t} - \boldsymbol{Y}_{1t}\boldsymbol{Y}_t^{-1}\boldsymbol{Y}_{1t})$$

$$\boldsymbol{x}_{2t}|\boldsymbol{x}_t \sim N_c(\boldsymbol{Y}_{2t}\boldsymbol{Y}_t^{-1}\boldsymbol{x}_t, \boldsymbol{Y}_{2t} - \boldsymbol{Y}_{2t}\boldsymbol{Y}_t^{-1}\boldsymbol{Y}_{2t})$$

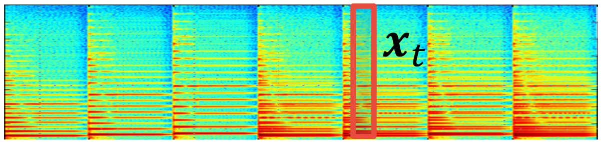
$$\boldsymbol{x}_{3t}|\boldsymbol{x}_t \sim N_c(\boldsymbol{Y}_{3t}\boldsymbol{Y}_t^{-1}\boldsymbol{x}_t, \boldsymbol{Y}_{3t} - \boldsymbol{Y}_{3t}\boldsymbol{Y}_t^{-1}\boldsymbol{Y}_{3t})$$

All the frequency components of each source spectrum can be estimated jointly via Wiener filtering

Maximum Likelihood Estimation

- We aim to estimate $\boldsymbol{H}, \boldsymbol{V}$ that maximizes the likelihood

Observed complex spectrogram



$$\boldsymbol{x}_t \sim N_c\left(\mathbf{0}, \sum_{k=1}^{K} h_{kt} \boldsymbol{V}_k\right) \longrightarrow \text{Maximize}$$

Observed covariance matrix

$$\boldsymbol{X}_t = \boldsymbol{x}_t \boldsymbol{x}_t^H$$

↔

Approx. covariance matrix

$$\boldsymbol{Y}_t = \sum_{k=1}^{K} h_{kt} \boldsymbol{V}_k$$

Gaussian log-likelihood

$$\log p(\boldsymbol{X}_t | \boldsymbol{Y}_t) = -\frac{1}{2} \log |\boldsymbol{Y}_t| - \frac{1}{2} \mathrm{tr}(\boldsymbol{X}_t \boldsymbol{Y}_t^{-1}) \longrightarrow \text{Maximize}$$

<u>Log-Det divergence</u>

$$D(\boldsymbol{X}_t | \boldsymbol{Y}_t) = -\log |\boldsymbol{X}_t \boldsymbol{Y}_t^{-1}| + \mathrm{tr}(\boldsymbol{X}_t \boldsymbol{Y}_t^{-1}) - F \longrightarrow \text{Minimize}$$

} Equivalent!

Generalized EM Algorithm (Proposed)

- Iteratively update latent sources and parameters
  - Expectation step
    - Calculate covariance matrices $\boldsymbol{Y}_{kt} = h_{kt}\boldsymbol{V}_k \quad \boldsymbol{Y}_t = \sum_{k=1}^{K} \boldsymbol{Y}_{kt}$
    - Calculate posteriors of source spectra

$$\boldsymbol{x}_{kt}|\boldsymbol{x}_t \sim N_c(\underbrace{\boldsymbol{Y}_{kt}\boldsymbol{Y}_t^{-1}\boldsymbol{x}_t}_{\mathrm{E}[\boldsymbol{x}_{kt}]}, \underbrace{\boldsymbol{Y}_{kt} - \boldsymbol{Y}_{kt}\boldsymbol{Y}_t^{-1}\boldsymbol{Y}_{kt}}_{\mathrm{V}[\boldsymbol{x}_{kt}]})$$

    - Calculate second-order statistics

$$\mathrm{E}\left[\boldsymbol{x}_{kt}\boldsymbol{x}_{kt}^H\right] = \mathrm{E}[\boldsymbol{x}_{kt}]\mathrm{E}\left[\boldsymbol{x}_{kt}^H\right] + \mathrm{V}[\boldsymbol{x}_{kt}]$$

IS-NMF: $O(KTF)$
LD-PSDTF: $O(KTF^3)$

  - Maximization step
    - Update parameters (depend on each other)

$$h_{kt} \leftarrow \frac{\mathrm{tr}\left(\boldsymbol{V}_k^{-1}\mathrm{E}\left[\boldsymbol{x}_{kt}\boldsymbol{x}_{kt}^H\right]\right)}{F} \qquad \boldsymbol{V}_k \leftarrow \frac{\sum_{t=1}^{T} h_{kt}^{-1}\mathrm{E}\left[\boldsymbol{x}_{kt}\boldsymbol{x}_{kt}^H\right]}{T}$$

Computational Bottleneck

- Inversion of big matrices is computationally prohibitive
  - E step: updating source spectra

$$\mathrm{E}\left[\boldsymbol{x}_{kt}\boldsymbol{x}_{kt}^{H}\right] = \boldsymbol{Y}_{kt}\boldsymbol{Y}_{t}^{-1}\boldsymbol{x}_{t} + \boldsymbol{Y}_{kt} - \boldsymbol{Y}_{kt}\boldsymbol{Y}_{t}^{-1}\boldsymbol{Y}_{kt}$$

  - M step: updating parameters

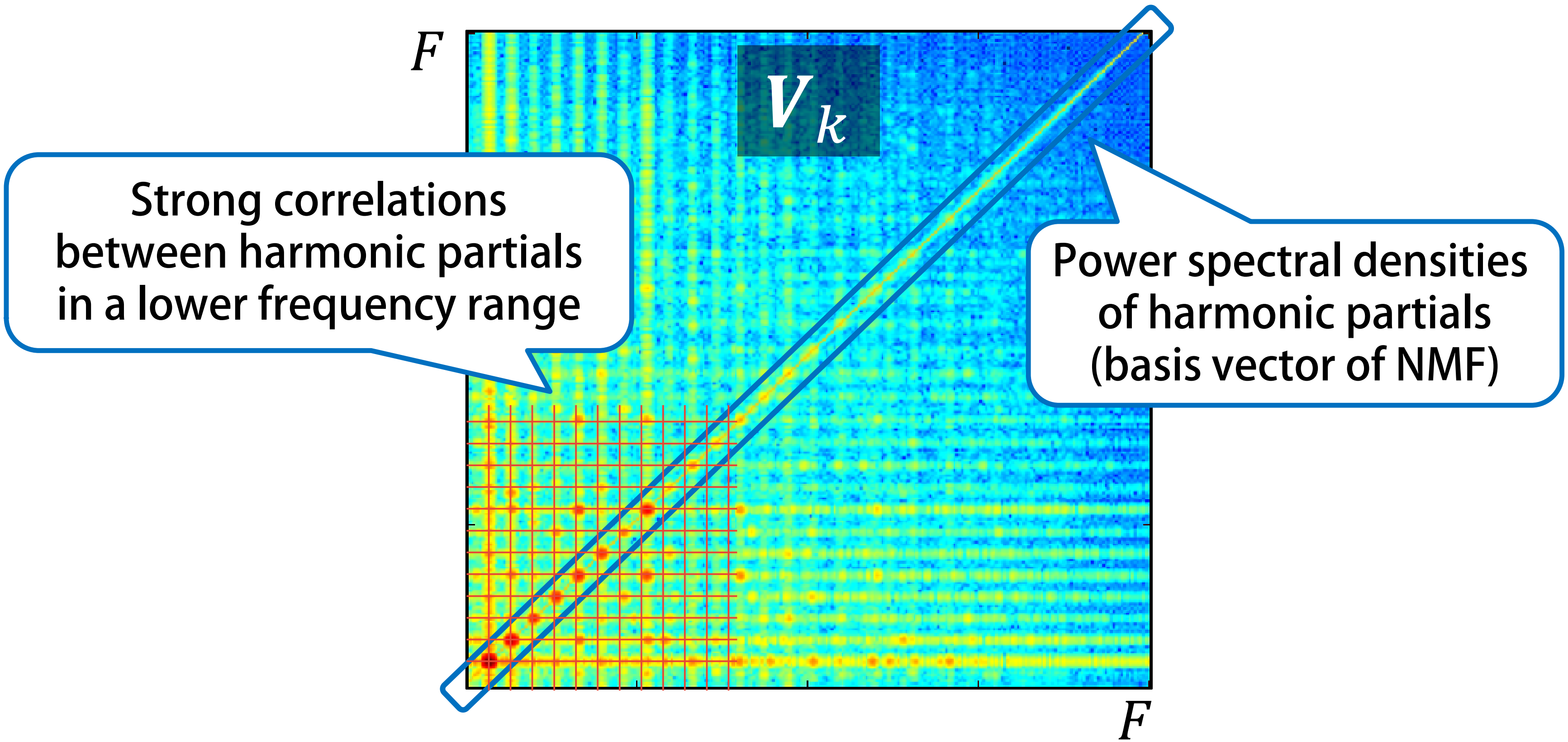
$$h_{kt} \leftarrow \frac{\mathrm{tr}\left(\boldsymbol{V}_{k}^{-1}\mathrm{E}\left[\boldsymbol{x}_{kt}\boldsymbol{x}_{kt}^{H}\right]\right)}{F} \qquad \boldsymbol{V}_{k} \leftarrow \frac{\sum_{t=1}^{T} h_{kt}^{-1}\mathrm{E}\left[\boldsymbol{x}_{kt}\boldsymbol{x}_{kt}^{H}\right]}{T}$$

The inverse matrices $\boldsymbol{Y}_{t}^{-1}$ and $\boldsymbol{V}_{k}^{-1} \in \mathbb{C}^{F\times F}$ are required: $O(F^{3})$

How to calculate these inversions in a more efficient manner?

Covariance Matrix Revisited

- Basis covariance matrices have diagonal + grid patterns
  - Especially for complex spectra with harmonic structures



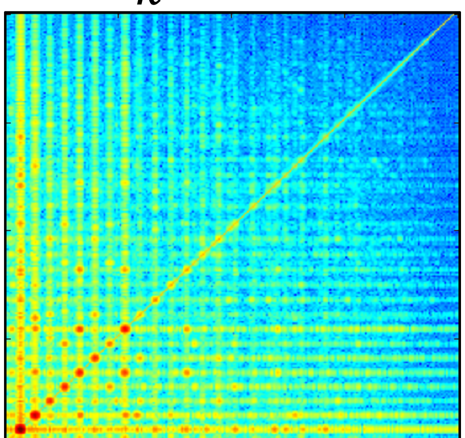
Covariance Approximation (Proposed)

- Each V_k is approximated as a diagonal + low-rank matrix
  - The rank N can be around the number of harmonic partials

$$\boldsymbol{V}_k = [\boldsymbol{w}_k] + \boldsymbol{L}_k[\boldsymbol{s}_k]\boldsymbol{L}_k^H$$

Basis covariance matrix

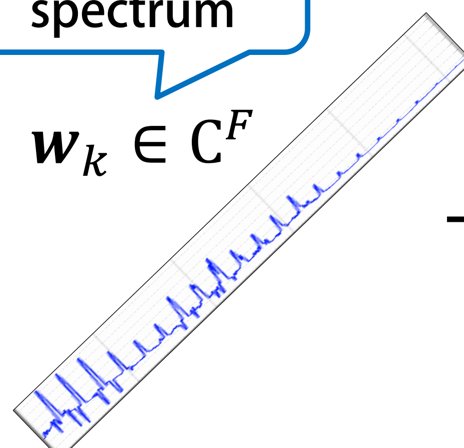
$\boldsymbol{V}_k \in \mathbb{C}^{F\times F}$



$=$

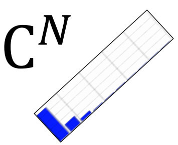
Basis power spectrum

$\boldsymbol{w}_k \in \mathbb{C}^{F}$

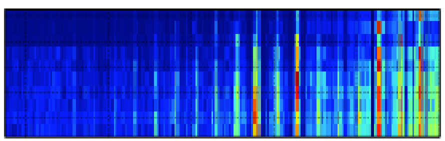


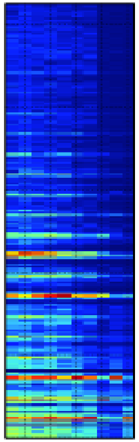
$+$

$\boldsymbol{s}_k \in \mathbb{C}^{N}$



$\boldsymbol{L}_k^H \in \mathbb{C}^{N\times F}$





$\boldsymbol{L}_k \in \mathbb{C}^{F\times N}$

EM Algorithm Revisited

- The inversion of big matrices are required
  - E step: updating source spectra

$$\mathrm{E}\left[\boldsymbol{x}_{kt}\boldsymbol{x}_{kt}^{H}\right] = \boldsymbol{Y}_{kt}\boldsymbol{Y}_{t}^{-1}\boldsymbol{x}_{t} + \boldsymbol{Y}_{kt} - \boldsymbol{Y}_{kt}\boldsymbol{Y}_{t}^{-1}\boldsymbol{Y}_{kt}$$

  - M step: updating parameters

$$h_{kt} \leftarrow \frac{\mathrm{tr}\left(\boldsymbol{V}_{k}^{-1}\mathrm{E}\left[\boldsymbol{x}_{kt}\boldsymbol{x}_{kt}^{H}\right]\right)}{F} \qquad \boldsymbol{V}_{k} \leftarrow \frac{\sum_{t=1}^{T} h_{kt}^{-1}\mathrm{E}\left[\boldsymbol{x}_{kt}\boldsymbol{x}_{kt}^{H}\right]}{T}$$

$$\begin{cases} \boldsymbol{V}_{k} = [\boldsymbol{w}_{k}] + \boldsymbol{L}_{k}[\boldsymbol{s}_{k}]\boldsymbol{L}_{k}^{H} \\ \boldsymbol{Y}_{t} = \sum_{k=1}^{K} h_{kt}\boldsymbol{V}_{k} = \sum_{k=1}^{K} h_{kt}[\boldsymbol{w}_{k}] + \sum_{k=1}^{K} h_{kt}\boldsymbol{L}_{k}[\boldsymbol{s}_{k}]\boldsymbol{L}_{k}^{H} \end{cases}$$

Each term can be inverted efficiently

Efficient Matrix Inversion

- Use Woodbury formula for covariance matrices

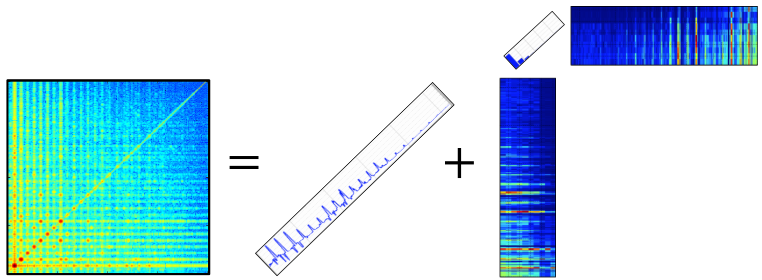
$$(\boldsymbol{A} + \boldsymbol{U}\boldsymbol{C}\boldsymbol{V})^{-1} = \boldsymbol{A}^{-1} - \boldsymbol{A}^{-1}\boldsymbol{U}\left(\boldsymbol{C}^{-1} + \boldsymbol{V}\boldsymbol{A}^{-1}\boldsymbol{U}\right)^{-1}\boldsymbol{V}\boldsymbol{A}^{-1}$$

This formula is useful when $\boldsymbol{A}$ and $\boldsymbol{C}$ can be inverted efficiently

Diagonal matrices!

$$\underset{F \times F}{\boldsymbol{V}_k} = \underset{F \times F}{[\boldsymbol{w}_k]} + \underset{F \times N}{\boldsymbol{L}_k}\,\underset{N \times N}{[\boldsymbol{s}_k]}\,\underset{N \times F}{\boldsymbol{L}_k^H}$$

$(N \ll F)$



$N \times N$

$$\underset{F \times F}{\boldsymbol{V}_k^{-1}} = \underset{F \times F}{[\boldsymbol{w}_k]^{-1}} - \underset{F \times F}{[\boldsymbol{w}_k]^{-1}}\,\underset{F \times N}{\boldsymbol{L}_k}\left(\underset{N \times N}{[\boldsymbol{s}_k]^{-1}} + \underset{N \times F}{\boldsymbol{L}_k^H}\,\underset{F \times F}{[\boldsymbol{w}_k]^{-1}}\,\underset{F \times N}{\boldsymbol{L}_k}\right)^{-1}\underset{N \times F}{\boldsymbol{L}_k^H}\,\underset{F \times F}{[\boldsymbol{w}_k]^{-1}}$$

Inversion of a compact matrix!

Recursive Matrix Inversion

- Use Woodbury formula in a recursive manner

$$\boldsymbol{Y}_t = \sum_{k=1}^{K} h_{kt}[\boldsymbol{w}_k] + \sum_{k=1}^{K} h_{kt}\boldsymbol{L}_k[\boldsymbol{s}_k]\boldsymbol{L}_k^H \quad \Rightarrow \quad \boldsymbol{Y}_t^{-1}$$

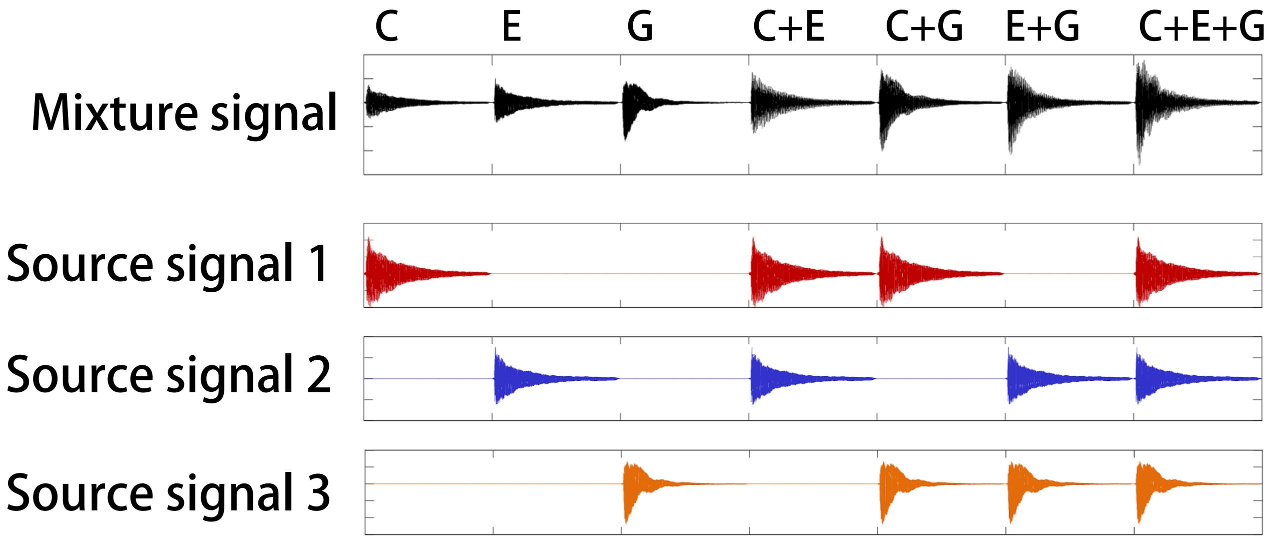
$$\boldsymbol{Y}_t^{(p)} \stackrel{\text{def}}{=} \sum_{k=1}^{K} h_{kt}[\boldsymbol{w}_k] + \sum_{k=1}^{p} h_{kt}\boldsymbol{L}_k[\boldsymbol{s}_k]\boldsymbol{L}_k^H = \boldsymbol{Y}_t^{(p-1)} + h_{pt}\boldsymbol{L}_p[\boldsymbol{s}_p]\boldsymbol{L}_p^H$$

$$\left(\boldsymbol{Y}_t^{(p)}\right)^{-1} = \left(\boldsymbol{Y}_t^{(p-1)}\right)^{-1} - \left(\boldsymbol{Y}_t^{(p-1)}\right)^{-1}\boldsymbol{L}_p \underbrace{\left(h_{pt}^{-1}[\boldsymbol{s}_p]^{-1} + \boldsymbol{L}_p^H\left(\boldsymbol{Y}_t^{(p-1)}\right)^{-1}\boldsymbol{L}_p\right)^{-1}}_{N\times N}\boldsymbol{L}_p^H\left(\boldsymbol{Y}_t^{(p-1)}\right)^{-1}$$

Recurrence formula starting at $\boldsymbol{Y}_t^{(0)} = \sum_{k=1}^{K} h_{kt}[\boldsymbol{w}_k]$ (NMF)

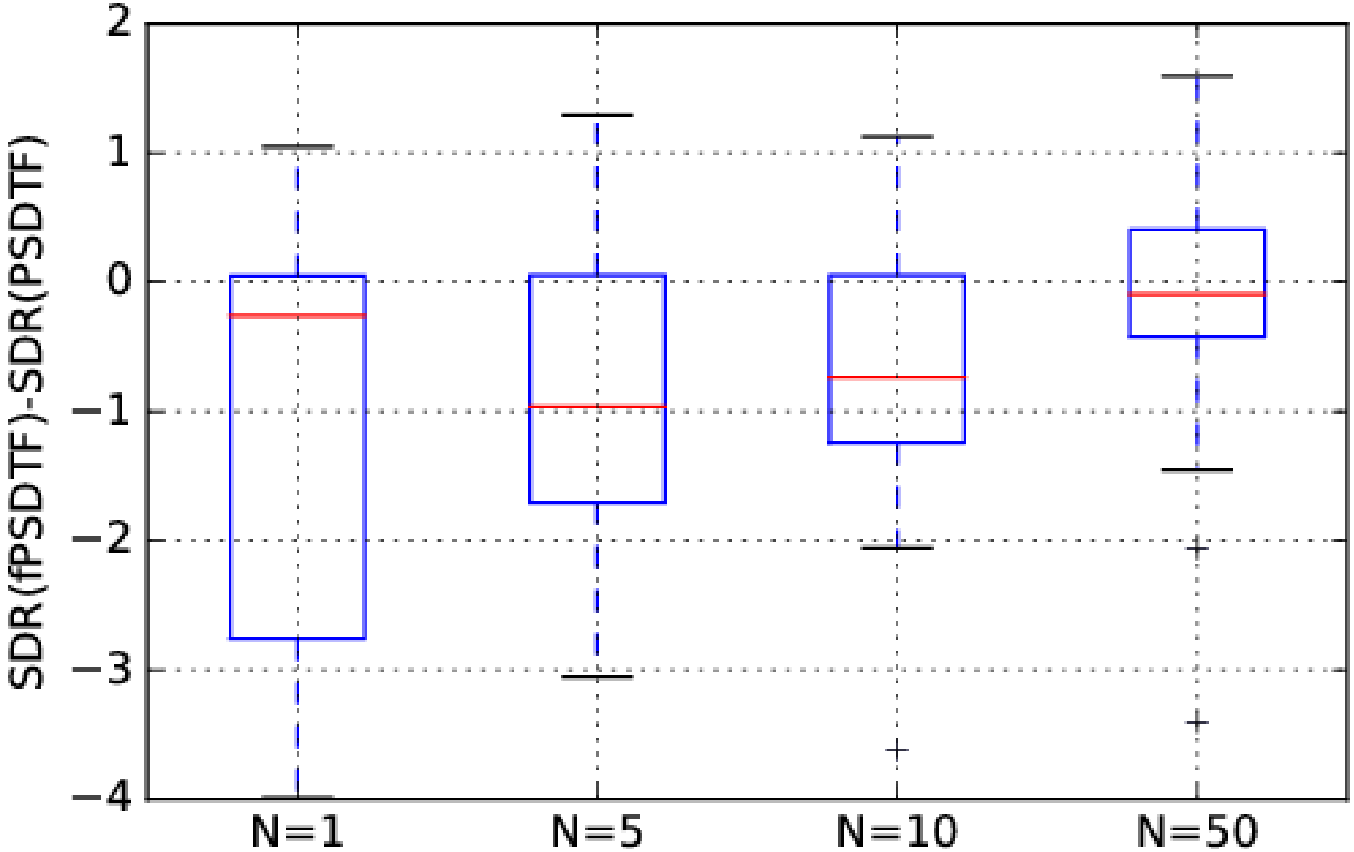
Evaluation

- Separation performance vs covariance approximation
  - Synthesize a mixture signal sampled at 16 [kHz]
    - $K = 3$ (C4, E4, G4, piano) • $F = 256, T = 840$
  - Test "fast" PSDTF wih $N = 0$ (NMF), $1, 5, 10, 50, 256$ (PSDTF)
  - Use BSS Eval Toolbox [Vincent2006]



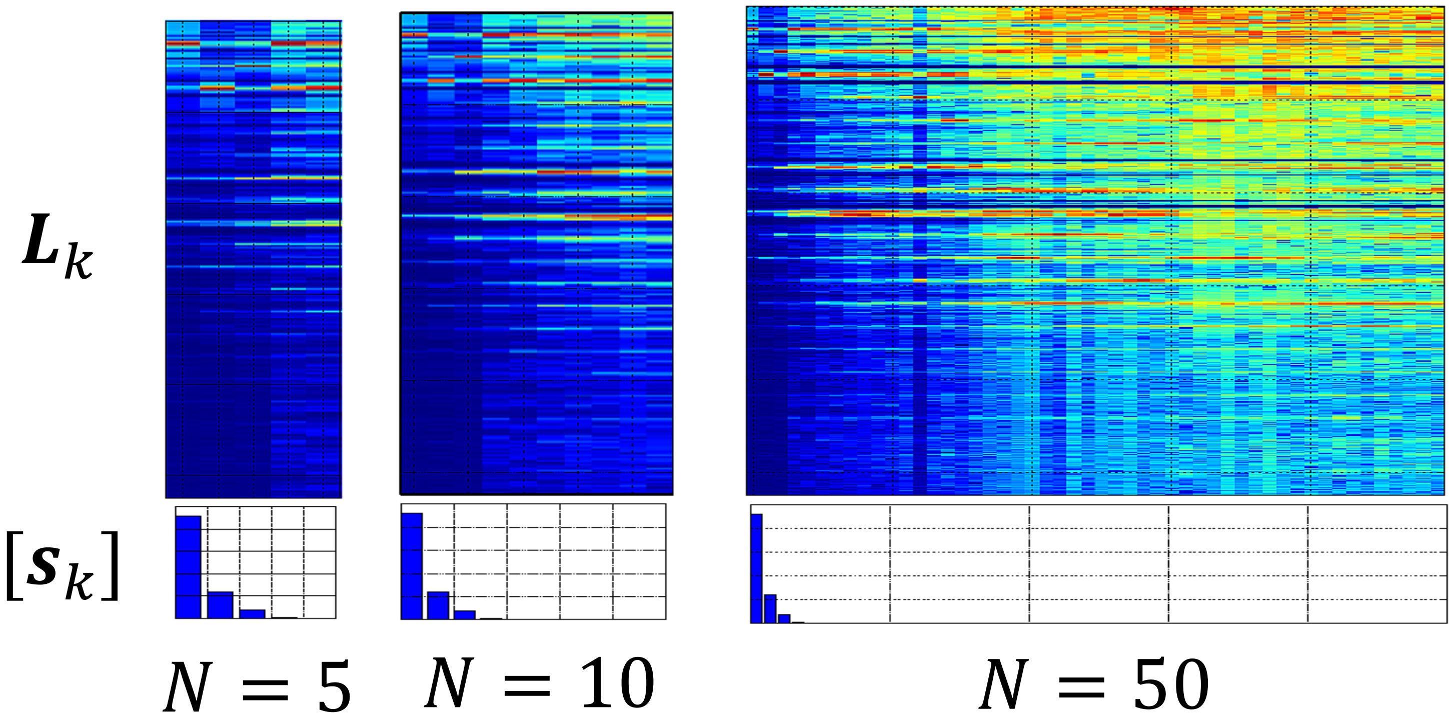
Separation Performance

- PSDTF $(N = 50)$ was comparable with PSDTF $(N = 256)$



Estimated Results

- The off-diagonal elements of each V_k (inter-frequency correlations) can be approximated by a low-rank matrix
  - A limited number of eigenvalues are significantly larger than 0



Conclusion

- We introduced positive semidefinite tensor factorization (PSDTF) based on the Log-Det divergence
  - A natural extension of nonnegative matrix factorization (NMF) based on the Itakura-Saito divergence
  - Estimation of locally-stationary Gaussian processes
- We proposed a constrained version of LD-PSDTF for reducing computational complexity
  - Kernel matrices are restricted to diagonal + low-rank matrices
  - Woodbury formula is used for inversing kernel matrices (in a recursive manner)

$$\boldsymbol{V}_k = [\boldsymbol{w}_k] + \boldsymbol{L}_k[\boldsymbol{s}_k]\boldsymbol{L}_k^H$$