卒業研究論文

論文題目

ユーザーからの補助情報を用いる
インタラクティブ音源分離システムの開発

提　出　年　月　日	令和 2 年 2 月 28 日
学　　科	電気情報工学科
氏　　名	中野 将生　　印
指 導 教 員 (主 査)	北村 大地 助教　　印
副　　査	柿元 健 准教授　　印
学　科　長	辻 正敏 教授　　印

香川高等専門学校

Development of Interactive Audio Source Separation Informed by User Annotation

Masaki Nakano

Department of Electrical and Computer Engineering
National Institute of Techonology, Kagawa College.

ABSTRACT

As a blind audio source separation technique for time-convolutive mixtures, independent low-rank matrix analysis (ILRMA) was proposed. Although ILRMA achieves better separation performance compared with the other methods, ILRMA sometimes fails to align permutation of wideband source components in a frequency domain, which is the so-called “block permutation problem”.

In this thesis, I develop an interactive audio source separation system that utilizes user annotation in a parameter optimization of ILRMA for avoiding the block permutation problem. I proposed two types of annotation: (a) frequency annotation that directly assigns a permutation-incorrect frequency band and (b) time annotation that indicates silent time frames of a specific source. The efficacy of the proposed annotation system is evaluated by two-speech separation tasks.

keywords: ILRMA, blind source separation, user interaction

論文概要

残響による時間畳み込み混合に適用可能なブラインド音源分離手法として独立低ランク行列分析（ILRMA）が提案されている．ILRMA は他のブラインド音源分離手法と比較して高精度な分離が可能であるが，広い周波数帯域でまとまって分離音源の順番を間違える「ブロックパーミュテーション問題」を起こす場合がある．

本論文ではブロックパーミュテーションを解決するため，ILRMA に対しアノテーションを与えるインタラクティブな音源分離システムを開発する．アノテーションには，（a）分離音源において順番を間違えた周波数帯域を直接指定するものと，（b）特定の音源が沈黙している時間区間を示すものの 2 通りを提案する．提案システムの効果は，2 話者の音声混合信号の分離実験により評価する．

目次

シンボル表

シンボル	意味
I	時間周波数領域での周波数ビン数
J	時間周波数領域での時間フレーム数
N	音源数
M	観測チャネル数
K	NMF における基底数
L	短時間フーリエ変換の窓長（短時間区間に分割した際の信号長）
i	時間周波数領域での周波数ビンインデックス
j	時間周波数領域での時間フレームインデックス
n	音源インデックス
m	観測チャネルインデックス
k	NMF における基底インデックス
$\tilde{x}(\tau)$	時間領域の観測信号
$\boldsymbol{S}_n$	時間周波数領域での音源信号 （複素スペクトログラム）
$\boldsymbol{X}_m$	時間周波数領域での観測信号 （複素スペクトログラム）
$\boldsymbol{Y}_n$	時間周波数領域での分離信号 （複素スペクトログラム）
$\boldsymbol{s}_{ij}$	時間周波数領域での多音源信号
$\boldsymbol{x}_{ij}$	時間周波数領域での多チャネル観測信号
$\boldsymbol{y}_{ij}$	時間周波数領域での多音源分離信号
$\boldsymbol{A}_i$	複素混合行列
$\boldsymbol{a}_{i,n}$	ステアリングベクトル
$\boldsymbol{B}$	複素スペクトログラム

シンボル	意味
$\boldsymbol{W}_i$	分離行列
$\boldsymbol{w}_{i,n}$	分離フィルタ
$\boldsymbol{T}_n$	基底行列 （スペクトルパターン行列）
$\boldsymbol{V}_n$	アクティベーション行列
$\boldsymbol{R}_n$	モデルパワースペクトログラム行列
$t_{ik,n}$	基底行列の要素
$v_{kj,n}$	アクティベーション行列の要素
$r_{ij,n}$	モデルパワースペクトログラム行列の要素 （時間周波数毎のパワー値）
$w(\tau)$	短時間フーリエ変換の窓関数
c	短時間フーリエ変換のシフト長
i_s	周波数帯域交換アノテーションの開始位置
i_e	周波数帯域交換アノテーションの終了位置
j_s	沈黙時間区間指定アノテーションの開始位置
j_e	沈黙時間区間指定アノテーションの終了位置
n_t	アノテーションの指定先
n_s	周波数帯域交換アノテーションの交換元
$.^{\cdot p}$	行列の要素毎の指数乗を取った （p乗） を取った行列
$.^{\mathrm{T}}$	転置
$.^{\mathrm{H}}$	エルミート転置
$\mathcal{D}\big(\cdot \mid \cdot\big)$	行列の要素間のダイバージェンスの総和
$\mathcal{L}$	負対数尤度
$\mathbb{C}$	複素数集合
$\mathbb{R}$	実数集合
j	虚数単位
e	自然対数の底
ε	マシンイプシロン
α	ある程度乱雑さを含んだ十分に大きな値
ρ	0 から 1 までの乱数

1 章
緒言

1.1. 音源分離の背景

音源分離とは複数の音源からの音声信号が混合した観測信号から元の音源毎の音声信号を推定する技術である．この技術は，カーナビゲーションシステム，スマートフォン，スマートスピーカーなどの音声認識を行う多くのデバイスで音声認識の精度を上げるために不可欠な要素として組み込まれている．また，音楽分野においても楽曲の再編集，特定の楽器音の抽出などの目的にも利用可能である．

音源分離手法には複数の条件による分類があり，まず観測に用いるマイクの数が音源の数よりも多いか否かで大別できる．前者の条件を優決定条件と呼び，後者を劣決定条件と呼ぶ．

劣決定条件では観測時の制約が弱いため幅広い分野での適用が期待できるものの，観測信号から得られる情報が少ないため，音源とマイクの位置関係や抽出したい音声の特徴などの事前情報無しでの音源分離は難しい．このような背景に対応するため，音声信号，特に調波楽器による信号が持つ低ランク性（管楽器や弦楽器などの調波楽器で見られる，同じスペクトルパターンの繰り返しが支配的になる性質）を利用し，非負値行列因子分解（nonnegative matrix factorization: NMF）[1-2] に基づく教師あり音源分離手法 [3] などが検討されている．近年では，深層学習技術の発展により，深層ニューラルネットワークを用いた教師あり音源分離手法も多数検討されている [4-6]．

一方，優決定条件では最低でも分離したい音源数以上のマイクを用意する制約があるものの，利用可能な情報が多いことから後者に比べ，事前に学習したスペクトルパターンなどの音源に関する事前情報がより少ない場合においても分離可能である．また，近年ではマイクロホンセンサの製造価格も低下しており，品質も向上していることから物理的な制約は緩和されている．マイクの位置関係と音源の到来方向が分かる条件では，ビームフォーミング [7] と呼ばれる古典的な手法が適用可能であり，特定方向からの音声信号が抽出可能である．また，音源の到来方向やマイク位置の事前情報が得られない条件下でも，信号の統計的

な性質を仮定することで音源分離を実現するブラインド音源分離（blind source separation: BSS）が研究されている．特に信号間の独立性を仮定する独立成分分析（independent component analysis: ICA）[8] に基づく手法が多い．また，ICA の応用として，残響を含む音源信号の混合に対処するため，時間周波数領域での線形時不変（混合系が線形であり，時間によって変化しないこと）な瞬時混合（その瞬間の畳み込みとして表現されること）を仮定して周波数ビン（特定の周波数で取った時間方向のベクトル）毎に ICA を適用する周波数領域 ICA（frequency-domain ICA: FDICA）[9] が提案されている．

FDICA に基づく音源分離では，音源間の独立性しか仮定されず分離信号の周波数毎の順番とスケールが不定となる．この内，スケールに関しては projection back 法 [10] によって高速に解決可能であるが，分離された信号がどちらの音源に所属するかを確実に解決することは困難である．時間領域における ICA では順番は大きな問題とはならないが，前述の FDICA においては周波数ビン毎に ICA を適用するため，Figure 1.1 に示すように，分離された信号を全周波数ビンに対して同じ音源順に整列しない限り，時間領域に戻しても正しく音源分離することができない．これはパーミュテーション問題 [11] と呼ばれ，これまでに様々な解決法が提案されてきた [10-12]．近年では，パーミュテーション問題を極力起こさずに音源分離を行う手法として，同一音源の分散（パワー）が全周波数ビンで共起性を持つことを仮定する独立ベクトル分析（independent vector analysis: IVA）[13-14] や，同一音源の時間周波数の共起性が低ランク構造を持つことを仮定する独立低ランク行列分析（independent low-rank matrix analysis: ILRMA）[15-16] が提案されている．特に ILRMA は，音楽信号のように音源の時間周波数構造の低ランク性仮定が良く合致する信号において，頑健かつ高精度な BSS を実現しているが，音声信号の分離問題においては IVA と同程度の性能となる場合も多く [17]，実用においては課題が残る．

IVA や ILRMA が音声信号の BSS において分離に失敗する主な原因は，ブロックパーミュテーション問題と呼ばれる現象が起こることである [13, 18-19]．これは，Figure 1.2 のように，隣接する周波数ビン間でのパーミュテーション問題が解決されているにもかかわらず，まとまった周波数帯域がブロックとしてパーミュテーション不整合を起こす問題である．図中左側の列が正しいスペクトログラムであり，右側の列がブロックパーミュテーションを起こしているスペクトログラムである．

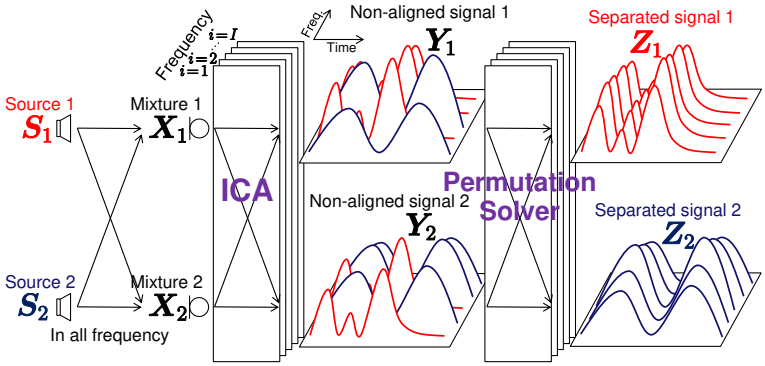

Figure 1.1 Permutation problem.

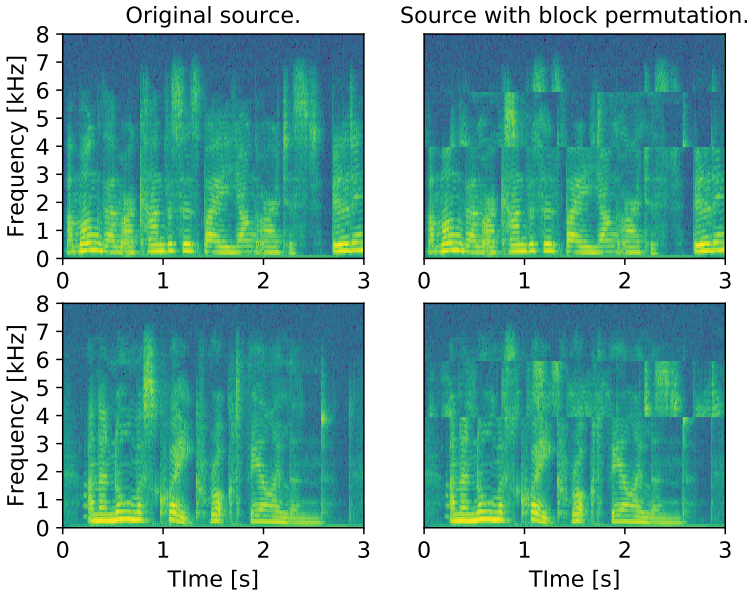

Figure 1.2 Example of block permutation problem.

1.2. 本論における主題

本論文では，Figure 1.3 に示すような，ILRMA に基づく BSS に対してユーザとのインタラクションを導入することで，安定的に高精度な音源分離を達成する手法を提案する．また，この手法を評価するため，ブロックパーミュテーションを判別可能な程度の知識を持ったユーザーを対象にしたシステムを構築する．具体的には，ILRMA の最適化の途中で現状の分離結果を Web フロントエンドによりユーザに提示し，ユーザーがブロックパーミュテーション問題が生じていると判断した場合はユーザによるアノテーション情報を利用することで，パーミュテーション不整合を修正しつつ最適化を続けるサーバー・クライアント型のシステムを構築する．

1.3. 本論の構成

2 章では，ベースとなる ILRMA について述べる．3 章では，ILRMA に対して行う補正方法を定式化し述べる．4 章では，実際に構築したシステムの実装方法と仕様を述べる．5 章では，構築したシステムを用い実施した実験の結果を述べる．6 章では，本論文の結論を述べる．

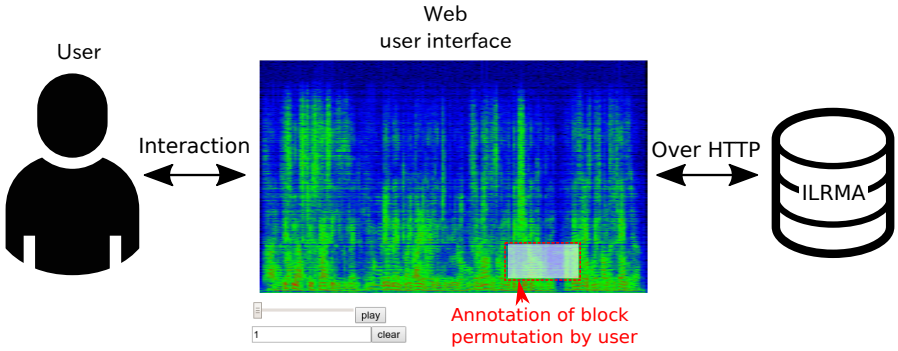

Figure 1.3 Interaction between user and ILRMA.

2 章
従来手法

2.1. はじめに

本章では，提案手法において必要となる理論を説明するため，本論において扱う問題の定式化と共に従来手法について述べる．まず 2.2 節において本論で扱う問題を定式化する．2.3 節では音響信号処理において頻繁に用いられる短時間フーリエ変換を導入する．2.4 節では，ILRMA のベースとなった周波数領域 ICA と独立ベクトル分析の 2 手法を導入する．2.5 節では，ILRMA で用いる非負値行列因子分解を導入する．2.6 節では本論で構成したシステムにおいて基礎となる手法である ILRMA を導入する．

2.2. 定式化

音源数と観測チャネル数をそれぞれ N 及び M とし，各時間周波数における音源信号，観測信号及び分離信号をそれぞれ

$$\boldsymbol{s}_{ij} = \left(s_{ij,1},\cdots s_{ij,N}\right)^{\mathrm{T}} \qquad \in \mathbb{C}^{N\times 1} \tag{2.1}$$

$$\boldsymbol{x}_{ij} = \left(x_{ij,1},\cdots x_{ij,M}\right)^{\mathrm{T}} \qquad \in \mathbb{C}^{M\times 1} \tag{2.2}$$

$$\boldsymbol{y}_{ij} = \left(y_{ij,1},\cdots y_{ij,N}\right)^{\mathrm{T}} \qquad \in \mathbb{C}^{N\times 1} \tag{2.3}$$

と表す．ここで，$i = 1,\cdots,I$ は周波数インデックス，$j = 1,\cdots,J$ は時間インデックス，$n = 1,\cdots,N$ は音源インデックス，$m = 1,\cdots,M$ はチャネルインデックスを示し，$\cdot^{\mathrm{T}}$ はベクトルまたは行列の転置を表す．また，各信号の複素スペクトログラム行列を $\boldsymbol{S}_n \in \mathbb{C}^{I\times J}$，$\boldsymbol{X}_m \in \mathbb{C}^{I\times J}$，$\boldsymbol{Y}_n \in \mathbb{C}^{I\times J}$ で表す．これらの行列の要素はそれぞれ $s_{ij,n}$，$x_{ij,m}$，$y_{ij,n}$ である．混合系が線形時不変であり，時間周波数領域での複素瞬時混合で表現できると仮定すると，周波数毎の時不変な複素混合行列 $\boldsymbol{A}_i = (\boldsymbol{a}_{i,1} \ \cdots \ \boldsymbol{a}_{i,N}) \in \mathbb{C}^{M\times N}$（$\boldsymbol{a}_{i,n} = (a_{i,n1},\cdots,a_{i,nM})^{\mathrm{T}}$ は各音源のステアリングベクトル）が定義でき，観測信号を式 (2.4) で表現できる．

$$\boldsymbol{x}_{ij} = \boldsymbol{A}_i \boldsymbol{s}_{ij} \tag{2.4}$$

この混合モデルは，時不変混合系の残響時間が短時間フーリエ変換（short-time Fourier transform: STFT）の窓長よりも十分に短い場合に成立する．この時，$M = N$ かつ $\boldsymbol{A}_i$ が正則であれば，分離ベクトル $\boldsymbol{w}_{i,n} = (w_{i,n1}, \cdots, w_{i,nM})^{\mathrm{T}}$ で構成される分離行列 $\boldsymbol{A}_i^{-1} \approx \boldsymbol{W}_i = (\boldsymbol{w}_{i,1}, \cdots, \boldsymbol{w}_{i,N})^{\mathrm{H}} \in \mathbb{C}^{N \times M}$ が存在し，分離信号は式 (2.5) で与えられる．

$$\boldsymbol{y}_{ij} = \boldsymbol{W}_i \boldsymbol{x}_{ij} \tag{2.5}$$

ここで，$.^{\mathrm{H}}$ はベクトルまたは行列のエルミート転置を表す．線形時不変を仮定する優決定条件下における BSS では，この分離行列 $\boldsymbol{W}_i$ を全ての周波数ビン（$i = 1, \cdots, I$）で求めることが目標である．本論文では，観測チャネル数と音源数が等しいと仮定 $(M = N)$ する．

2.3. 短時間フーリエ変換

時間領域の実数で定義された原信号を時間周波数領域での複素数の信号に変換するため，Figure 2.1 に示すような，式 (2.6) で表される離散時間での STFT を使用する．STFT では，図のように原信号をシフト長毎に窓長で切り出し，そこに窓関数をかけたものを離散フーリエ変換して時間フレームとして並べていく．

$$\boldsymbol{X}(i,j) = \sum_{l=0}^{L-1} \tilde{x}\big(l + (j-1)c\big) w(l) e^{-\mathrm{j}\frac{2\pi i l}{L}} \tag{2.6}$$

ここで，$\tilde{x}(\tau)$ は原信号，c はシフト長，L は窓長（短時間区間信号の信号長）で，立体の j は虚数単位，$w(\tau)$ は窓関数である．実際の離散時間の STFT の計算においては，有限長のサンプルを取得し，それを離散フーリエ変換（discrete Fourier transform: DFT）しながら重ね合わせる事でスペクトログラムを算出する．この際取得した部分の開始点と終了点が連続するように窓関数を掛ける必要がある．

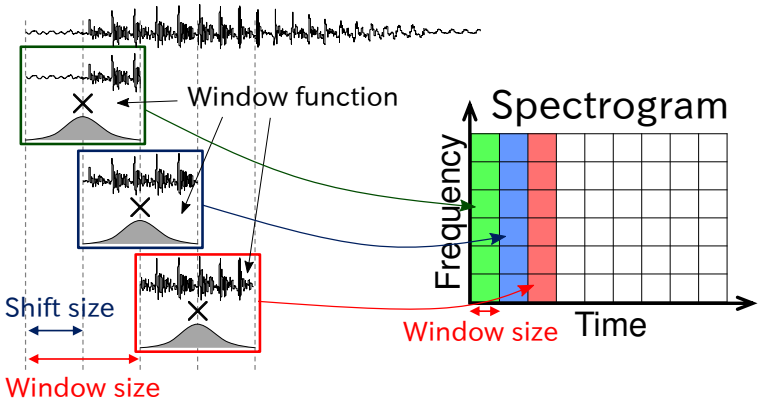

Figure 2.1 Short-time Fourier transform.

2.4. FDICA と IVA

FDICA では残響を含む事によって生じた音源の畳込み混合を分離するため，周波数ビン毎の複素時系列観測信号 $x_{i1,m} \cdots x_{iJ,m}$ に対し個別に ICA を適用し分離する． 各周波数ビンにおける分離行列 $\boldsymbol{W}_i$ は他の周波数とは無関係に推定されるため，ICA により分離された信号の順序が不定となることから，推定された信号を並べ直すパーミュテーション問題を解く必要がある．IVA では，全周波数成分を纏めた周波数ベクトル $\bar{\boldsymbol{x}}_{j,m} = (x_{1j,m}, \cdots, x_{Ij,m})^{\mathrm{T}}$ を考え，このベクトル確率変数に対して ICA を適用する． このとき，周波数ベクトルは I 次元の非ガウスかつ球対称な多変量分布を仮定する． これにより 1 つの周波数ベクトル内の各成分は高次相関を持ち，基本周波数とその倍音のような同時に生起する周波数成分を一つに纏められるようになり，パーミュテーション問題を回避しつつ時間周波数領域での音源分離が可能となる．

2.5. NMF

行列を低ランクな 2 つの行列の積で近似する手法に Figure 2.2 に示す非負値行列因子分解（nonnegative matrix factorization: NMF）[1] がある．単一チャネルの音響信号を対象とした NMF では，複素スペクトログラム行列 $\boldsymbol{B} \in \mathbb{C}^{I \times J}$ を非負化した行列 $|\boldsymbol{B}|^{\cdot p} \in \mathbb{R}_{\geq 0}^{I \times J}$ を別の 2 つの非負行列 $\boldsymbol{T} \in \mathbb{R}_{\geq 0}^{I \times K}$（基底行列）及び $\boldsymbol{V} \in \mathbb{R}_{\geq 0}^{K \times J}$（アクティベーション行列）の行列積により近似する．ここで，行列に対する絶対値記号は要素毎に絶対値を取った行列を表し， ドット付きの指数は要素毎の累乗を表す．従って，$|\boldsymbol{B}|^{\cdot p}$ は，$p = 1$ が振幅スペクトログラム，$p = 2$ がパワースペクトログラムに対応する．また，K は基底数であり，低ランク近似するために $K \ll \min(I, J)$ と設定される． 分解行列 $\boldsymbol{T}$ 及び $\boldsymbol{V}$ は， 式 (2.7) の最小化問題の解として推定される．

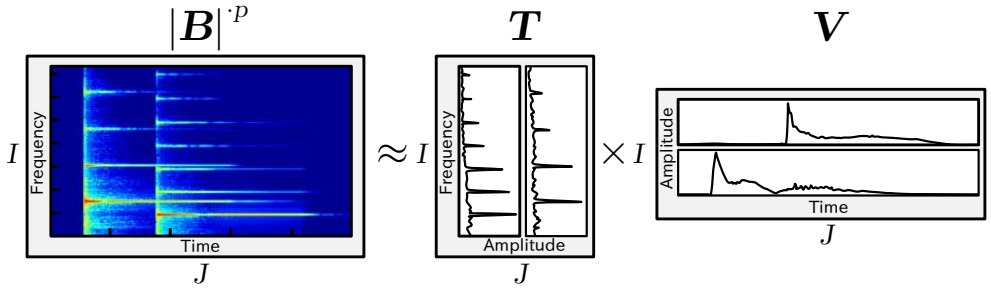

Figure 2.2 NMF decomposition of power spectrogram.

$$\min_{\boldsymbol{T},\boldsymbol{V}} \mathcal{D}\Big(|\boldsymbol{B}|^{\cdot p} \;\Big|\; \boldsymbol{T}\boldsymbol{V}\Big) \text{ s.t. } t_{ik}, v_{kj} \geq 0 \;\forall i,j,k \tag{2.7}$$

ここで，$\mathcal{D}(\cdot \mid \cdot)$ は 2 つの行列引数の要素間のダイバージェンスの総和，t_{ik} 及び v_{kj} は行列 $\boldsymbol{T}$ 及び $\boldsymbol{V}$ の要素，$k = 1, \cdots, K$ は基底インデックスを示す．行列積 $\boldsymbol{T}\boldsymbol{V}$ が低ランクとなる制約から，$\boldsymbol{T}$ の列ベクトルは $|\boldsymbol{B}|^{\cdot p}$ の頻出スペクトルパターン，$\boldsymbol{V}$ の行ベクトルは $|\boldsymbol{B}|^{\cdot p}$ における各スペクトルパターンの時間的な強度変化をそれぞれ示す．

2.6. ILRMA の生成モデルと更新式

FDICA におけるパーミュテーション問題を避けるため，IVA では同一音源において全周波数ビンがパワーの共起性を持つ（低周波域と高周波域のパワーが正比例する）と仮定した上で，複数音源間の独立性を最大化している．このように，同一音源の時間周波数構造に関する仮定をする音源モデルをより一般化するため，IVA の音源モデルに NMF を導入した手法が ILRMA [15-16] である．ILRMA では式 (2.8) の複素ガウス分布を音源信号の生成モデルとして仮定する．

$$\begin{aligned} p(\boldsymbol{Y}_1, \cdots, \boldsymbol{Y}_n) &= \prod_n p(\boldsymbol{Y}_n) \\ &= \prod_{n,i,j} p\Big(y_{ij,n}\Big) \\ &= \prod_{n,i,j} \frac{1}{\pi r_{ij,n}} \exp\left(-\frac{\big|y_{ij,n}\big|^2}{r_{ij,n}}\right) \end{aligned} \tag{2.8}$$

ここで，$r_{ij,n}$ は音源 n の時間周波数毎のパワーであり，$r_{ij,n} = \mathrm{E}\Big[\big|y_{ij,n}\big|^2\Big]$ で

ある．さらに，$r_{ij,n}$ を要素に持つ時間周波数行列を $\boldsymbol{R}_n \in \mathbb{R}_{\geq 0}^{I\times J}$ とおくと，この分散行列が式 (2.9)–(2.10) のように NMF でモデル化される低ランク構造を持っていることを仮定する．

$$\boldsymbol{R}_n = \boldsymbol{T}_n \boldsymbol{V}_n \tag{2.9}$$

$$r_{ij,n} = \sum_k t_{ik,n} v_{kj,n} \tag{2.10}$$

ここで，$\boldsymbol{T}_n \in \mathbb{R}_{\geq 0}^{I\times K}$ 及び $\boldsymbol{V}_n \in \mathbb{R}_{\geq 0}^{K\times J}$ はそれぞれ音源 n の分散行列をモデル化する非負行列（基底行列及びアクティベーション行列）であり，$t_{ik,n}$ 及び $v_{kj,n}$ はこれらの要素である．また，$k = 1, \cdots, K$ は基底行列 $\boldsymbol{T}_n$ 中の基底ベクトルのインデクスである．従って ILRMA では，同一音源の分散の時間周波数構造が高々 K 個の基底で表される低ランク構造を持つことが仮定されている．

ILRMA では，式 (2.8) の生成モデルに基づいて，変数 $\boldsymbol{W}_i$, $\boldsymbol{T}_n$，及び $\boldsymbol{V}_n$ の最尤推定を行う．式 (2.8) の負対数尤度は式 (2.11) となる．

$$\mathcal{L} = -2J\sum_i \log|\det \boldsymbol{W}_i| + \sum_{i,j,n}\left(\frac{\left|y_{ij,n}\right|^2}{r_{ij,n}} + \log r_{ij,n}\right) \tag{2.11}$$

ILRMA では，分離信号のパワースペクトログラム $|\boldsymbol{Y}_n|^{.2}$ を低ランク分散行列 $\boldsymbol{R}_n$（ランク K 行列）でモデル化し，その時間周波数の共起性を加味しながら分離行列 $\boldsymbol{W}_i$ を推定する．混合前の音源のパワースペクトグラム $|\boldsymbol{S}_n|^{.2}$ が低ランクであれば，混合信号のパワースペクトグラム $|\boldsymbol{X}_n|^{.2}$ のランクは基本的に増加することから，ILRMA は分離信号を低ランクに誘導することでパーミュテーション問題を避けつつ，互いに独立となる分離信号を推定している．LRMA の最尤推定問題は式 (2.11) の最小化問題と等価である．分離行列 $\boldsymbol{W}_i$ は，その行ベクトルである $\boldsymbol{w}_{i,n}$ を次に示す反復射影法（iterative projection: IP）[14] で全ての n について更新することで最適化される．

$$\boldsymbol{U}_{i,n} = \frac{1}{J}\sum_j \frac{1}{r_{ij,n}} \boldsymbol{x}_{ij}\boldsymbol{x}_{ij}^{\mathrm{H}} \tag{2.12}$$

$$\boldsymbol{w}_{i,n} \leftarrow \left(\boldsymbol{W}_i \boldsymbol{U}_{i,n}\right)^{-1} \boldsymbol{e}_n$$

$$\boldsymbol{w}_{i,n} \leftarrow \boldsymbol{w}_{i,n}\left(\boldsymbol{w}_{i,n}^{\mathrm{H}} \boldsymbol{U}_{i,n} \boldsymbol{w}_{i,n}\right)^{-\frac{1}{2}} \tag{2.13}$$

ここで，$\boldsymbol{e}_n \in \mathbb{R}_{\{0,1\}}^{N\times 1}$ は n 番目の要素が 1，それ以外が 0 となるようなベクトルである．分離ベクトル更新後の分離信号は式 (2.14) で更新する．

$$y_{ij,n} \leftarrow \boldsymbol{w}_{i,n}^{\mathrm{H}} \boldsymbol{x}_{ij} \tag{2.14}$$

分散行列の更新式は板倉斎藤ダイバージェンス [20] に基づく NMF [2] と同様に，乗算型反復更新式で最適化できる．

$$t_{ik,n} \leftarrow t_{ik,n} \sqrt{\frac{\sum_j |y_{ij,n}|^2 v_{kj,n} \left(\sum_{k'} t_{ik',n} v_{k'j,n} \right)^{-2}}{\sum_j v_{kj,n} \left(\sum_{k'} t_{ik',n} v_{k'j,n} \right)^{-1}}} \tag{2.15}$$

$$v_{kj,n} \leftarrow v_{kj,n} \sqrt{\frac{\sum_i |y_{ij,n}|^2 t_{ik,n} \left(\sum_{k'} t_{ik',n} v_{k'j,n} \right)^{-2}}{\sum_i t_{ik,n} \left(\sum_{k'} t_{ik',n} v_{k'j,n} \right)^{-1}}} \tag{2.16}$$

$\boldsymbol{T}_n$ 及び $\boldsymbol{V}_n$ の更新後は，推定分散を式 (2.17) で更新する．

$$\boldsymbol{R}_n \leftarrow \boldsymbol{T}_n \boldsymbol{V}_n \tag{2.17}$$

2.7. 本章のまとめ

本章では，提案手法の基礎となる ILRMA の理論及び ILRMA を構成する従来手法を導入した．次章からは本論文にて提案する手法の理論を述べる．

3 章
提案手法における最適化アルゴリズム

3.1. はじめに

本章では，提案手法の詳細と動機を述べる．まず 3.2 節において動機を述べる．3.3 節では実際に導入した，ILRMA に対し与えるアノテーション情報とその適用方法を述べる．3.4 節では，ILRMA が分離音源において順番を間違えた周波数帯を直接指定する場合の適用方法の詳細を述べる．特定の音源の沈黙している区間を指定する場合は 2 通りの手法を評価したため，3.5 節と 3.6 節でそれぞれ述べる．

3.2. 動機

IVA や ILRMA で複数音声を分離する場合，Figure 1.2 に示すようなブロックパーミュテーション問題が発生する場合がある．これは，音声信号の周波数帯域間の高次相関を推定する際に，共起性や低ランク性を仮定しても誤った局所最適解に陥り，別の音源の成分を同一音源の成分とみなす事が主な原因と考えられる．このため，最適化の途中でブロックパーミュテーション問題が生じた場合，そのまま反復計算による最適化を継続しても，ブロック帯域内での分離は進むものの，ブロックパーミュテーションが解決されることは殆どない．しかし，Figure 1.2 からも確認できるように，音声信号や音楽信号等のブロックパーミュテーションは（各周波数ビンで音源が十分分離できていれば）分離信号のパワースペクトログラムに対する目視で確認・判別できる可能性が高い．そこで本論文では，Web フロントエンドを用いたインタラクティブなアノテーション付与機能を作成し，ユーザからのアノテーション情報を ILRMA にフィードバックするシステムを構築する．

3.3. アノテーションの種類と指定方法

Figure 3.1 は，音源数 $N = 2$ の場合の Web フロントエンドのインターフェース画面を示している．ユーザはスペクトログラム上をマウスでドラッグすることで直接部分的な領域を指定することができる．Figure 3.1 では，ブロックパー

ミュテーション問題が生じている周波数ビンのみを含むような領域指定をしている．また，指定した領域が何番目の音源に属すかをテキストボックスに記述できる．最後に，画面下部の Frequency のラジオボタンをクリックし Submit を押すことで，ILRMA にブロックパーミュテーション問題が生じている周波数ビンの情報が送信される．また，提案するインターフェースでは，周波数の範囲だけでなく時間の範囲指定も可能である．例えば，2 人の話者の内一方しか発話していない時間範囲を指定し，画面下部の Time のラジオボタンをクリックし Submit を押すことで，ILRMA に 1 音源のみ沈黙している時間範囲の情報が送信される（この場合テキストボックスの内容は破棄される）．アノテーション情報が ILRMA に送信されると，再分離の後再び分離結果がユーザーに提示される．ここでも分離が十分でない場合は再びアノテーションを作成し，任意の回数再分離を実施することができる．なお，いずれのスペクトログラムも Play ボタンを押すことで分離音を聞くことができ，ユーザが領域を指定する際の参考にできる．

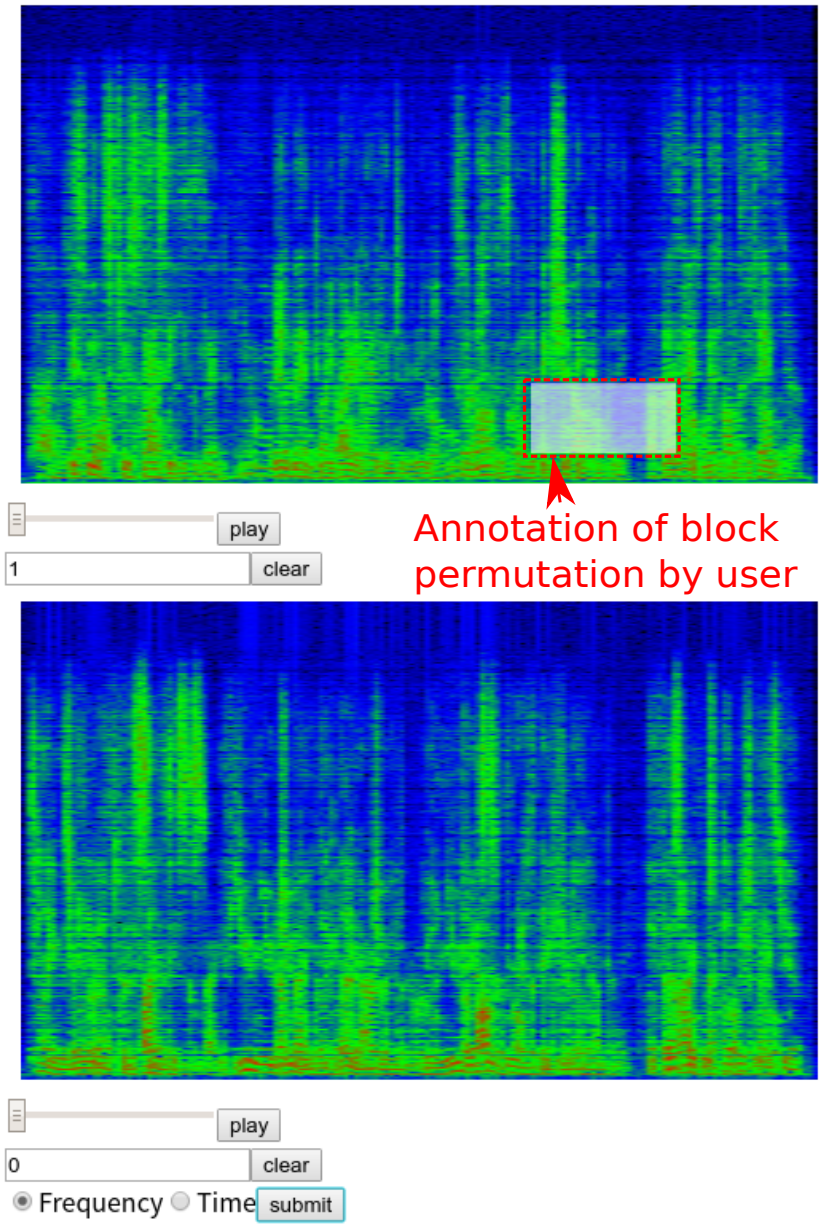

Figure 3.1 User interface (before annotated).

3.4. ブロックパーミュテーションを起こしている周波数帯の直接修正

概要を Figure 3.2 に示す．今，パーミュテーションが誤っている周波数範囲の開始点 $i = i_s$ と終了点 $i = i_e$（ここで $0 \leq i_s < i_e$）及び領域指定中の音源インデクス $n = n_s$ と正しいパーミュテーションとなる交換先の音源インデクス $n = n_t$（ブロックの移動先）がユーザから与えられた状況を考える．この場合，以後の ILRMA の更新では該当周波数ビンの分離フィルタ $\boldsymbol{w}_{i,n}$ と基底成分 $t_{ik,n}$ について入れ替え処理を行えばよいので，式 (3.18)–(3.20) のような処理を行う．

$$\boldsymbol{w}_{i_s n_s} \cdots \boldsymbol{w}_{i_e,n_s} \Leftrightarrow \boldsymbol{w}_{i_s,n_t} \cdots \boldsymbol{w}_{i_e,n_t} \tag{3.18}$$

$$t_{i_s k,n_s} \cdots t_{i_e k,n_s} \Leftrightarrow t_{i_s k,n_t} \cdots t_{i_e k,n_t} \qquad \forall k \tag{3.19}$$

$$v_{kj,n} \leftarrow \rho \qquad \forall k,j,n \tag{3.20}$$

ここで，$\Leftrightarrow$ は右辺と左辺の入れ替えを意味する．式 (3.18) 及び式 (3.19) で成分を入れ替え，式 (3.20) でアクティベーション行列を $0 \sim 1$ までの乱数 ρ で初期化する．その後一度全ての n について分散行列を式 (2.17) で更新し，式 (2.13)–(2.16) に示す ILRMA の反復最適化を再開する．

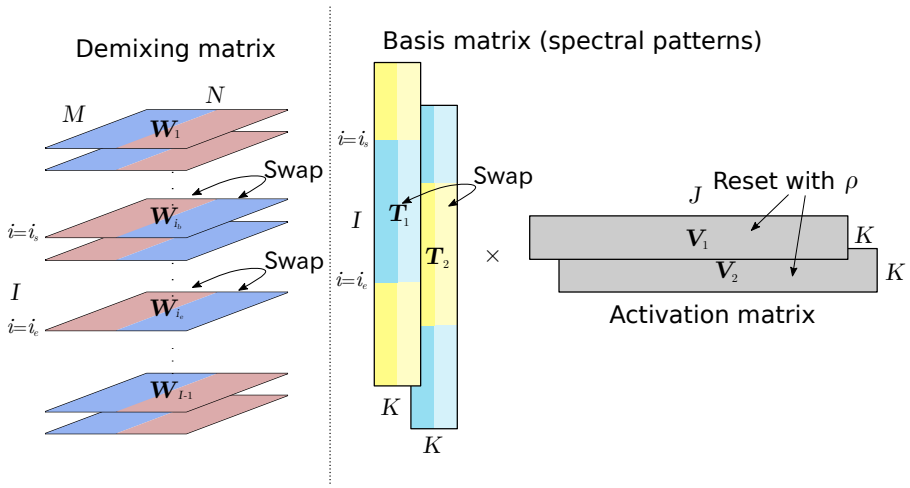

Figure 3.2 Overview of frequency annotation.

3.5. 沈黙している 1 音源の時間区間の修正 (a)

概要を Figure 3.3 に示す．今，n_t 番目の音源のみ沈黙している時間範囲の開始点 $j = j_s$ と終了点 $j = j_e$（ここで $0 \leq j_s < j_e$）がユーザから与えられた状況を考える．この場合，以後の ILRMA の更新では，式 (3.21)–(3.22) のように，該当時間フレームのアクティベーション成分 v_{kj,n_t} に非常に小さな値を代入する．

$$v_{kj_s,n_t} \cdots v_{kj_e,n} \leftarrow \varepsilon \cdots \varepsilon \qquad \forall k \tag{3.21}$$

$$\boldsymbol{w}_{j,n} \leftarrow \rho \qquad \forall j,n \tag{3.22}$$

ここで，ε は適当に定めた微小な値である．式 (3.21) でアクティベーション行列を更新し，式 (3.22) で分離行列をリセットする．更新後は，一度全ての n について分散行列を式 (2.17) で更新し，式 (2.13)–(2.16) に示す ILRMA の反復最適化を再開する．

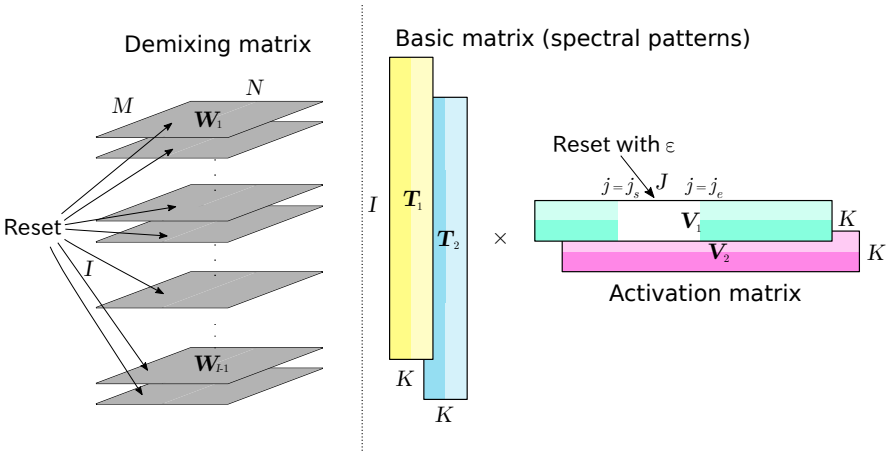

Figure 3.3 Overview of time annotation (a).

3.6. 沈黙している 1 音源の時間区間の修正 (b)

概要を Figure 3.4 に示す．3.5 節と同様に，式 (3.23)–(3.27) のように，該当時間フレームのアクティベーション成分 v_{kj,n_t} に非常に小さな値を代入する． この手法では NMF の再推定の際の自由度を上げるため沈黙区間以外はノイズを含んだある程度均一な値で初期化する．

$$v_{kj_s,n_t} \cdots v_{kj_e,n_t} \leftarrow \varepsilon \cdots \varepsilon \qquad \forall k \tag{3.23}$$

$$v_{k1,n_t} \cdots v_{kj_s,n_t} \leftarrow \alpha \cdots \alpha \qquad \forall k \tag{3.24}$$

$$v_{kj_e+1,n_t} \cdots v_{kJ,n_t} \leftarrow \alpha \cdots \alpha \qquad \forall k \tag{3.25}$$

$$v_{kj,n} \leftarrow \alpha \cdots \alpha \qquad \forall k,j,n \neq n_t \tag{3.26}$$

$$\boldsymbol{w}_{j,n} \leftarrow \rho \qquad \forall j,n \tag{3.27}$$

ここで，α は十分に大きく，ある程度乱雑さを含んだ値である．式 (3.23)–(3.26) の更新後は，一度全ての n について分散行列を式 (2.17) で更新し，式 (3.27) で分離行列をリセットする．その後，式 (2.13)–(2.16) に示す ILRMA の反復最適化を再開する．

3.7. 本章のまとめ

本章では本論文にて提案する手法の詳細を述べた． 次章では本手法を実装したシステムの技術的な詳細を述べる．

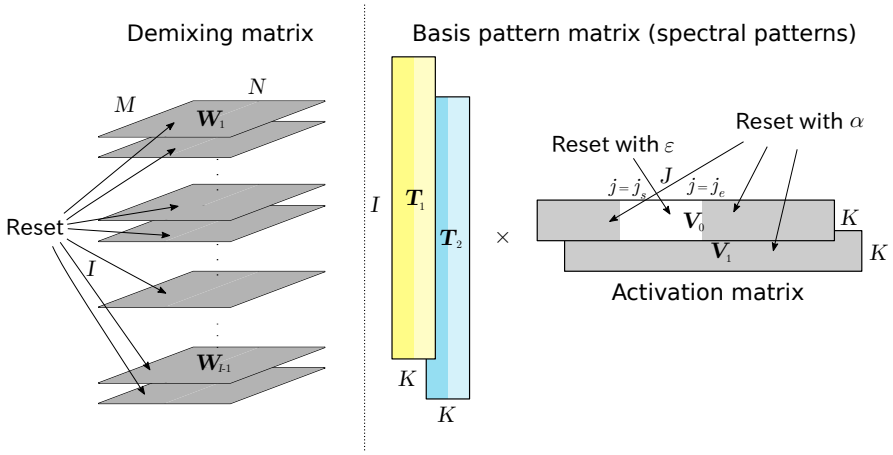

Figure 3.4 Overview of time annotation (b).

4 章
インタラクションシステムの実装

4.1. はじめに

本章では Figure 4.1 に示す本システムの実装に用いた技術とその詳細について述べる．4.2 節ではシステム全体の処理の流れについて述べる．4.3 節では通信に用いたプロトコル，及びその上で流れるデータの形式について述べる．4.4 節では実際に音源分離処理と補助情報の適用を行う実装について述べる．4.5 節では本システムにおいてユーザとのインタラクションを担うフロントエンドの実装について述べる．

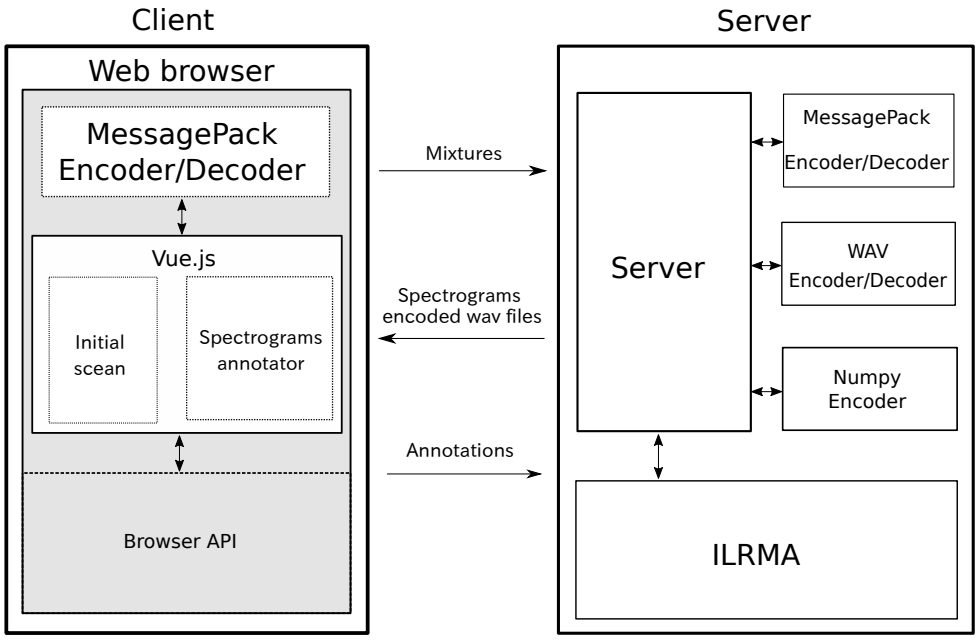

Figure 4.1 System overview.

4.2. 処理の流れ

ユーザと ILRMA のインタラクションを実現するシステムを実装するにあたり，サーバ・クライアントモデルのようにフロントエンドとバックエンドを分割する方法と一つのアプリケーションとしてモノリシックに実装する方法の二通りの方法が考えられる．前者には通信に伴うオーバーヘッドが存在し，後者にはシステムの柔軟性が失われる短所が存在する． 本研究では柔軟性とシステム全体の見通しを優先し前者の方法で実装している．

初期化時のフロントエンド側の画面を Figure 4.2 に示す．Figure 4.1 中の Initial scean に該当する．フロントエンド側は `connect` ボタンを押して WebSocket コネクションを確立する．NMF の基底数はテキストボックス `nb` で選択する．バックエンド側は一度起動されるとフロントエンドからのコネクションの確立を待機する． フロントエンド側はコネクションが確立されると，`Choose file` ボタンが有効化され， ボタンを押すとファイル選択ダイアログが開き分離したい音声ファイルが選択可能になる． 選択後は自動でファイルが WebSocket を通じてバックエンドに転送され， 受け取ったバックエンド側で処理が始まる．WebSocket 上では読み取った WAV ファイルがそのままバイナリデータとして転送されるため，バックエンド側で WAV ファイルをメモリ上でデコードする．デコード後，ILRMA に混合信号として渡し， 分離させ， その結果をスペクトログラムと WAV ファイルにそれぞれデコードして再び WebSocket にてフロントエンドに転送する．

フロントエンドはバックエンドからの返答により自動でアノテーション作成画面へ遷移する．Figure 4.1 中の Spectrogram annotator に該当する． その後は 3.3 節に示す方法によりアノテーションを作成し，Figure 3.1 中の `Submit` を押すとバックエンドにデータが転送される． バックエンドは WAV データデコード後にアノテーションを解釈し，3 章にて述べた手法で適用する他は初期化時と同様に処理する． 以上の流れを任意の回数繰り返すことでユーザーとのインタラクションを実現している．

Figure 4.2 Screenshot of ui in initizalizing scean.

4.3. 通信

4.3.1. 通信プロトコル

サーバとクライアントを接続する通信プロトコルには WebSocket [21] を採用した．これは Web 向けの双方向通信可能なプロトコルであり，Web 技術との親和性が高いことからフロントエンドの実装において既存の多くの資産が使える．また，双方向通信であるため本システムのように処理に時間がかかるため応答性の下がるシステムにも適している．WebSocket は HTTP（HTTPS）の Upgrade ヘッダを用い HTTP（HTTPS）上に双方向通信可能な通信を確立する．このため HTTP 通信しか許可されない場合でも通信可能であり，かつ HTTPS 通信を利用すればセキュリティも確保可能である．サーバからのレスポンスを遅延させることで仮想的な双方向通信を可能とする Comet，次世代通信規格である HTTP/2，Quic などの他の手段も存在する．しかし，Comet はクライアントからのリクエストを棚上げして遅延することにより擬似的にサーバーからのプッシュを実現するため，あまり効率的とは言えない．擬似的な再現でしか無いため WebSocket ほど実装も簡潔にならず，古いブラウザをサポートする必要が無いため，WebSocket に対する優位点が存在せず採用しなかった．また，HTTP/2 と Quic はまだ普及の途上にあるため採用しなかった．

WebSocket は HTTP を用いたコネクション確立後，相互にメッセージと呼ばれるデータを転送し合う．このメッセージは 1 つ以上のフレームからなる．送信時にフレームを分割する処理と再結合する処理は WebSocket 上で実施されるため使用者はこれについて意識する必要はない．WebSocket には現状 6 種類のフレーム種別が存在する．これを Table 4.1 に示す．本システムではこの内データ通信用にバイナリデータを使用する．これは本システムの扱うデータ量が膨大でありテキストにエンコード / デコードするオーバーヘッドが非常に大きくなるためである．

Table 4.1 Frame types on WebSocket

opcode	種別	説明
0x0	継続フレーム	先に送信されたフレームに繋がる
0x1	テキストフレーム	UTF-8 でエンコードされたテキストデータ
0x2	バイナリフレーム	バイナリデータ
0x3 - 0x7	予約済み	データ通信用に予約済み
0x8	close	接続切断
0x9	ping	ping 用
0xA	pong	pong 用
0xB - 0xF	予約済み	制御用に予約済み

4.3.2. シリアライズ形式

データ量の問題からデータ通信にはバイナリデータを用いるが，データ形式としては JSON のように予めエンコーダ / デコーダが存在し各プログラミング言語のデータ型に簡単に変換できるものが好ましいため，MessagePack [22] を用いた．これは高効率なバイナリ形式のフォーマットであり，JSON のようにデータ構造をデータ自身が保持するため，JSON の代替として扱うことが可能である．本システムでは通信の他ログの保存にも MessagePack を用いている．

4.3.3. 通信内容

実際に本システムで用いた通信内容を擬似コードの形式で示す．Figure 4.3 は接続時にクライアントからバックエンドに送信されるものであり， 予めバックエンドに音源を指定して起動した場合は省略される．Figure 4.4 はサーバからクライアントへのレスポンスである．Figure 4.5 はクライアントからサーバへの補助情報である．

```
{
  "type": "init",
  // 音源数
  "N": 2,
  "raw": [
    // WAV ファイル
    0x.....,
    // WAV ファイル
    0x.....
  ]
}
```

Figure 4.3 Packet for initialization.

```
{
  "type": "response",
  // 分離音数
  "N": 2,
  // 時間長
  "I": 1000,
  // 周波数ビン数
  "J": 1027,
  // 分離済み音源
  "separated": [
    {
      // パワースペクトログラムの log を取ったもの (numpy のデータ形式のダンプ)
      "freq": 0x.....,
      // wav ファイル
      "wav": 0x.....
    },
    {
    ....
    }
  ]
}
```

Figure 4.4 Packet for response.

```
{
  "type": "annot",
  // 沈黙区間指定では"time"
  // 周波数帯域指定では"freq"
  "method": "time",
  "annotations": [
    {
      // 矩形領域の始点と終点
      "start": {
        "x": 3,
        "y": 20
      },
      "end": {
        "x": 400,
        "y": 80
      },
      // freq の場合のみ存在．周波数帯域の置換先
      "target": 1,
      // 変更を適用する元となる音源の id
      "source": 0
    }
  ]
}
```

Figure 4.5 Packet for annotation.

4.4. バックエンド

バックエンドは Python を用いて実装した．利用したサードパーティライブラリを Table 4.2 に示す．

Table 4.2 Thirdparty libraries

パッケージ名	バージョン
numpy	1.17.4
soundfile	0.10.3.post1
Websockets	8.1
scipy	1.4.1
msgpack	0.6.2
mir-eval	0.5
matplotlib	3.1.2

バックエンドには予め分離音と混合音を与えて実行するモードとフロントエンドから混合音を与える 2 種類の実行モードを実装した．前者のモードは既存手法との比較のためのものである．msgpack と Websocket の扱いは外部のライブラリに任せているが，バイナリの wav ファイルをデコードできるライブラリは，ファイルからの読み込みのみサポートするものが多く見当たらなかったため自力で実装した．フォーマットは PCM と IEEE 浮動小数点数のみサポートしている．ILRMA は class を用いて実装した．スペクトログラムのデータ量はかなり大きく，WebSocket の規格上は送信するデータサイズの制限は無いものの，ライブラリ側で上限を定めている場合があるため注意が必要である．

4.5. フロントエンド

実際に構築したフロントエンドを Figure 3.1 に，フロントエンドを構成するコンポーネントの関係図を Figure 4.6 に示す．`Submit` ボタン等の各スペクトログラムで共通のコンポーネントは App に実装したが，各スペクトログラムに固有のコンポーネントは Pane 以下に実装した．モジュールバンドラなどの煩雑な設定を回避するため，vue-cli を用いた．UI 全体は，インタラクティブアプリケーションという性格を考慮し単一ページアプリケーション（single page application: SPA）として実装した．Vue.js は一つのファイル内にロジック，DOM 構造，CSS を記述する形式が利用可能であり見通しが良い．加えてデータ更新に伴い必要な部分のみのレンダリングをフレームワークが実行するため記述が短くなる利点もあるが，これは同時に正しく Vue.js が変更を検知できるよう書かなければならないため学習コストがやや高く，またデータとイベントの流れが単一方向ではなくなるため適切にモジュールを設計しなければ管理が極めて困難になるといった部分もある．

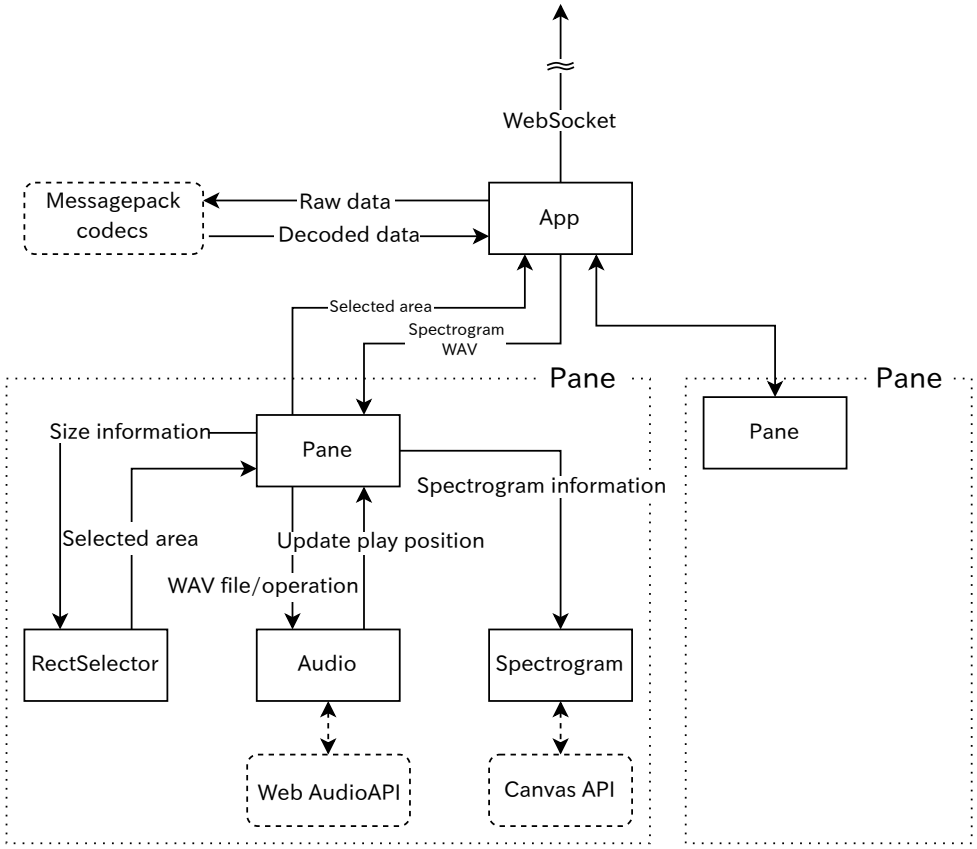

Figure 4.6 Diagram of relations between frontend components.

4.5.1. 音声再生処理

WAV データをファイルにせず MessagePack のバイナリに含め送受信するため，Binary large object を直接渡せる Audio API を用いて実装した．再生ボタンを押すと AudioContext を作成し，decodeAudioData 関数を呼び，デコード完了時に再生を始めるように実装した．再生位置を自由に変更できるような API は存在しないため，再生位置を変更した場合はデコードからやり直し再生開始位置を指定して再生することで擬似的に実現している．再生位置の表示は setTimeout を用いて 1 % 進む毎に再生位置を進めてそれをスライドバーに反映させている．スライドバーの値がユーザによって変更された場合は slider_change イベントが発行され，再生の停止と再生位置の変更を行う．

4.5.2. スペクトログラムの描画

送信されてきた MessagePack バイナリをデコードし，スペクトログラムの縦と横のサイズを取得してそれと同じサイズの画像を Canvas API を用いて作成す

る．スペクトログラムは numpy のダンプの形式で格納されており，これは単純に値の羅列であるので TypedArray を用いて Float64Array として解釈したあと，numpy のダンプと順番を合わせ読み込み，逐次値に応じて色を割り当て，前述の画像へ書き込む．書き込みを高速に実行するため，描画関数は用いず直接ピクセルデータを変更するよう実装した．

4.5.3. マウスドラッグ

スペクトログラムの上に透明の Canvas を重ね， そこに選択範囲を示す白色半透明の矩形を書き込むよう実装した． 座標情報はonmouseup,onmousedown, onmousemoveのコールバックを用いて取得した．更新がある度に Vue.js の emit 機能を用い上流に変更を通知する．

4.6. 本章のまとめ

本章では本論文で提案する手法を実装したシステムの詳細を述べた． 次章では本システムに対して実施した評価の条件とその結果を述べる．

5 章
実験

5.1. はじめに

本章では，提案手法に対して実施した実験の詳細を述べる．まず 5.2 節では，本実験に使用したデータセットとパラメータを述べる．5.3 節では，本実験により得られた結果を述べる．

5.2. 実験条件

比較のための信号には SiSEC2011 [23] の UND タスクに含まれる 6 信号を用いた．Table 5.3 に信号名を示す．STFT は 128 ms のハミング窓を 64 ms のシフトで行った．比較対象及び提案手法での NMF 部分の基底数は全て $K = 4$ とした． 提案手法にて使用した微小な値 ε は 10^{-15} とし，α は 10^5 に $0 \sim 10^4$ の一様乱数を加算した物を用いた．評価値には source-to-distortion ratio（SDR）[24] の改善量を用いた． 実験では ILRMA を 80 回反復した後， 従来手法ではそのまま残りの 80 回反復を継続し，提案手法ではユーザからのアノテーション情報を与えた上で同様に 80 回反復を継続した． 沈黙時間範囲を指定する手法では，アノテーションの検討のため 3.5 節と 3.6 節の 2 種を比較した．

Table 5.3 Sources used in experiment

Mixture	Source signals
No. 1	dev1_female3_synthconv_130ms_5cm_sim_1
	dev1_female3_synthconv_130ms_5cm_sim_2
No. 2	dev1_male3_synthconv_130ms_5cm_sim_1
	dev1_male3_synthconv_130ms_5cm_sim_2
No. 3	dev1_male3_synthconv_130ms_5cm_sim_1
	dev1_female3_synthconv_130ms_5cm_sim_2

5.3. 実験結果

Figure 5.1 の状態において， 沈黙している範囲を指定して ILRMA を 80 回反復した後に，残ったブロックパーミュテーションの周波数範囲を指定し同じく 80 回反復したときの結果が Figure 5.2 である．かなり分かりづらいが，沈黙区間に注目すると低周波域で発生しているブロックパーミュテーションが修復されている事が分かる．

Figure 5.3 にブロックパーミュテーションを直接指定した場合の SDR 改善量の推移を示す．破線が従来手法の，実践がが提案手法の SDR 改善料の推移を示す．SDR とは原信号に対する歪みを表す指標であり，値が高い程歪が少なく正確に分離されていることを示す．本実験では 2 話者の音源分離であるため，2 つの分離信号の SDR 改善率の算術平均をプロットしている．SDR は dB と同じくログスケールの指標である．Nos. 1 及び 2 の混合音源において，アノテーションを用いた補正後に SDR が改善する現象が確認できる．とくに No. 2 の例の改善量は大きく，提案システムが上手く動いた例といえる．一方で，従来の ILRMA ですでに十分な SDR 改善量を達成している No. 1 の混合音源については，アノテーション情報を用いても大きな変化は得られなかった．また，実際に本システムを利用した印象として，STFT の窓長が長くなるとブロックパーミュテーションの境界線が目視で確認しづらくなった． 従って， 残響の長い環境での混合信号の場合は， 周波数範囲指定よりも沈黙時間範囲の指定の方が幾分容易と思われる．

Figure 5.4 は，3.5 節の手法による SDR 改善量の推移である． アクティベーション行列及び分離行列を一度初期化するため一旦は SDR が減少するものの，その後は元の SDR より改善されている． 前述の通り沈黙区間はブロックパーミュテーションより目視での推定が容易であり，ブロックパーミュテーションを直接指定する場合に比べかなり高い確率で SDR を改善できた．一方で各混合音源ごとの最終的な改善量の差が大きく，実用面においては課題が残る．

Figure 5.5 は，3.6 節の手法による SDR 改善量の推移である．3.6 節の手法と同様に SDR が一旦は減少するものの， その後は安定して高い SDR に収束している点が異なる．理由としては，Figure 5.5 により部分的に値を変更するだけでは局所最適解を脱出することができない場合があるからではないかと考えられる． これは， 沈黙区間を誤って推定している以上， 他の時間区間も同様に誤って推定しているものと考えられるが， 誤った時間周波数構造を推定している行列 $\boldsymbol{T}$ 及び $\boldsymbol{V}$ の変更されていない部分が，再び誤った時間周波数構造へと導くためだと思われる．

Figure 5.6 は沈黙時間範囲指定の 2 手法の比較である．各音源において，分離

行列等の行列の初期化時及びアノテーションの適用時に用いられる乱数のシード値を 0 ~ 10 に変化させ，最終的な SDR 改善量を箱ひげ図で比較した．アノテーションの指定範囲は 2 手法とし，全てのシード値で共通のアノテーションを用いた．やはり 3.6 節の手法は 3.5 節の手法に比較して安定して高い改善量を達成している．

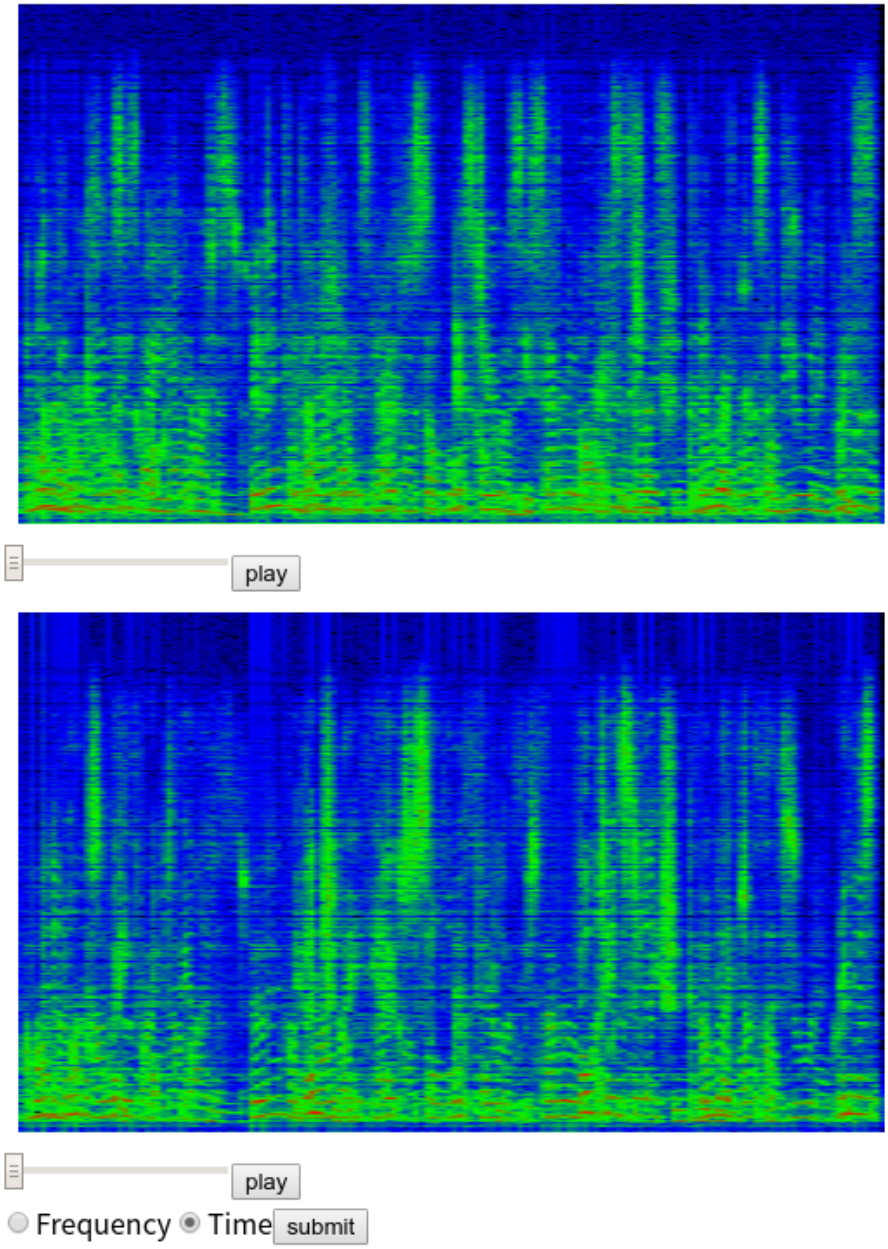

Figure 5.1 Spectrograms of before annotated signals.

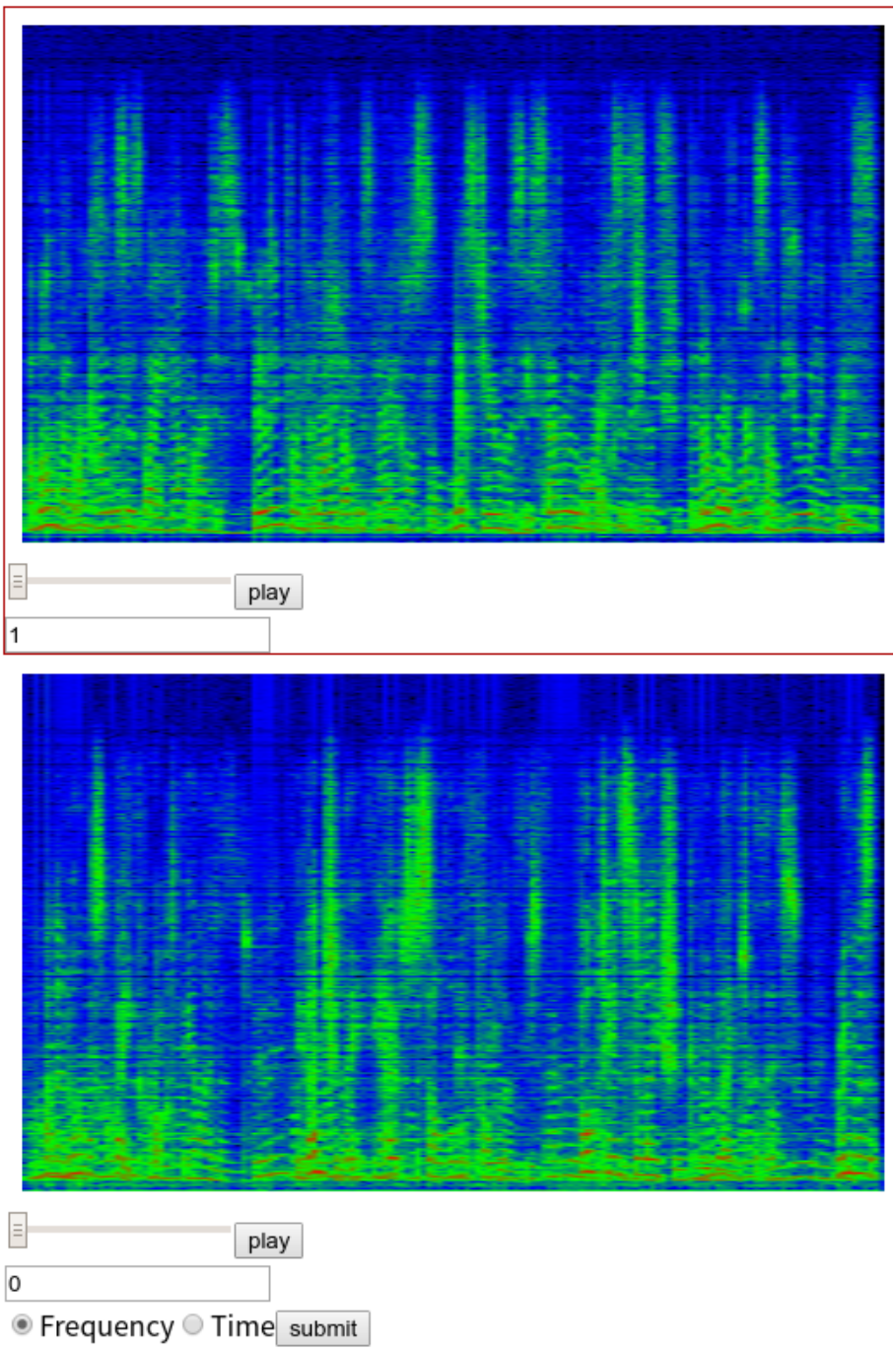

Figure 5.2 Spectrograms of block-permutation-fixed estimated signals.

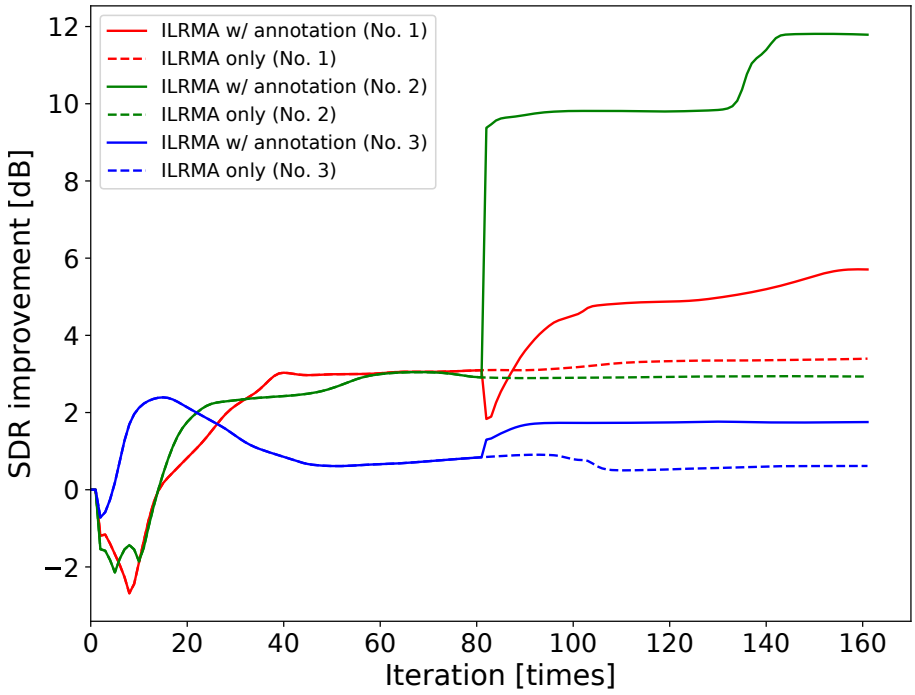

Figure 5.3 SDR improvement by frequency annotation.

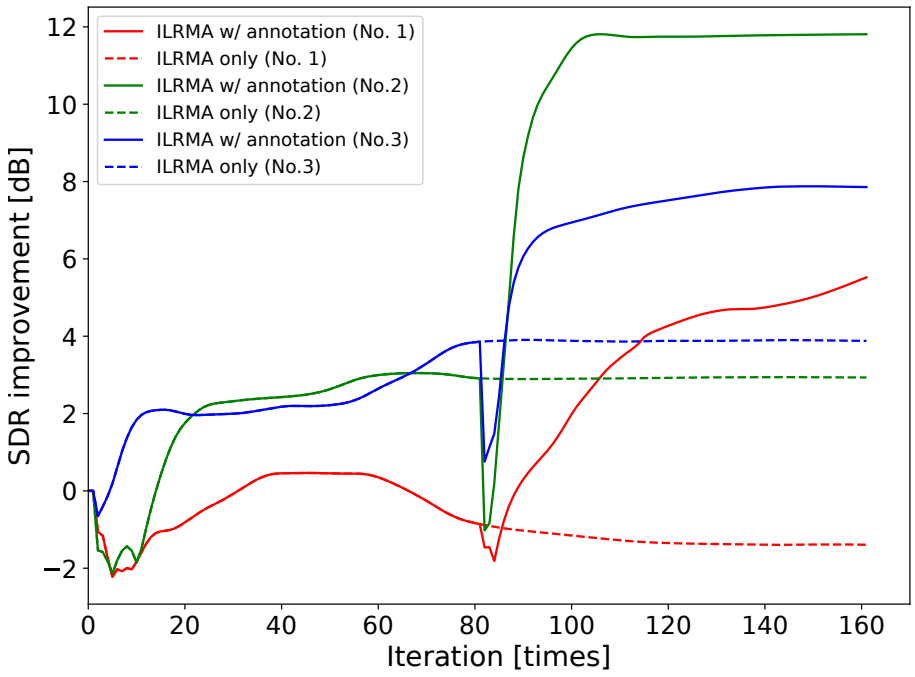

Figure 5.4 SDR improvement by time annotation (a).

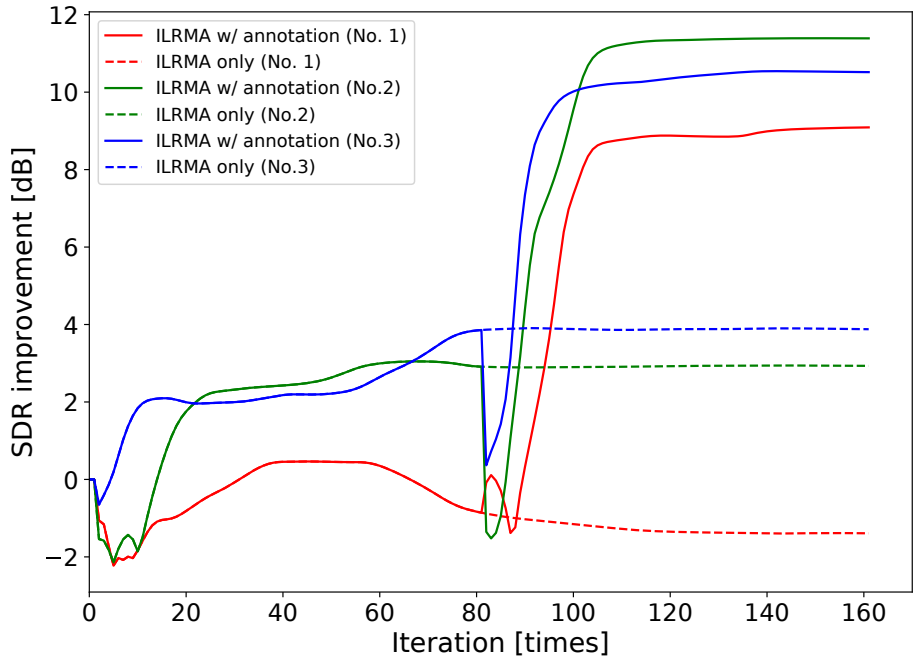

Figure 5.5 SDR improvement by time annotation (b).

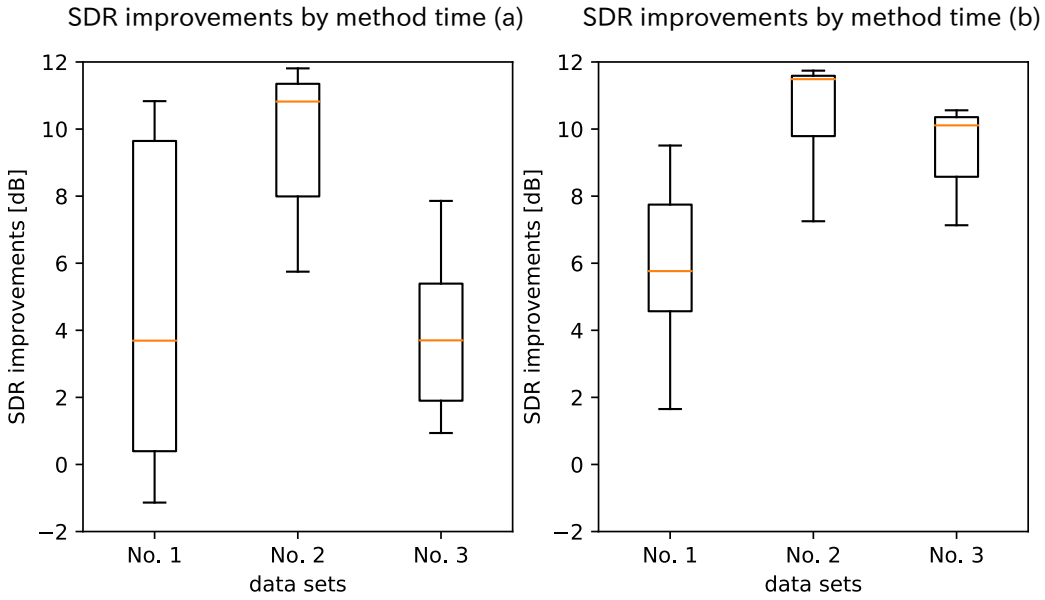

Figure 5.6 Comparison of two types of time annotation.

5.4. 本章のまとめ

本章では提案する手法を本論文で実装したシステムを用い，周波数帯域の交換と音源が沈黙している区間を指定する 2 手法を評価した． 音源の沈黙区間を指

定する手法では 2 種類のアノテーションの反映法を比較した．実施した実験により，両手法共に ILRMA が分離に失敗した場合でも，アノテーションを用い再分離を成功させられる事が示された．

6 章
結言

本論文では，ユーザーとのインタラクションを利用した音源分離手法として，ILRMA に対してアノテーション情報を与えるシステムを提案した．本システムでは，ブロックパーミュテーション問題が生じる帯域や特定の音源が沈黙している時間範囲をアノテーションとして与えることができる．実験の結果，音源分離精度の向上に一定の効果があることを確認した．一方で，5.3 節にて述べたように，アノテーションの作成難度が残響長に依存する問題が残るため，より適切なユーザインタフェースを作成することが課題となった．

謝辞

まず，本研究を進めるにあたり，指導教員の北村大地助教からは日頃の業務と研究の合間を縫い細部に至るまで熱心な指導と多大な助力を頂きました．厚く感謝いたします．ここでの研究の経験は私にとって大きな血肉となりました．

お忙しい中本論文の副査を引き受けてくださった柿元健准教授に感謝の意を示します．

北村研究室の山地修平氏には一年間に渡る研究室生活，特に受験においては過去問やノウハウなど多くの面で助けていただいた事心より感謝します．同研究室同期の岩瀬祐太氏，大藪宗一郎氏，渡辺瑠伊氏には日頃のゼミやディスカッションは勿論，精神的な支えともなりました．ここに感謝します．

時々作業環境を変えにラップトップを持ってお邪魔した柿元研究室の皆さんはいつも暖かく迎えてくださりました．厚く感謝します．

SATYSFI で本論文を書くにあたり様々な助言をくださった諏訪敬之さん，monaqa さん (https://github.com/monaqa) には心より感謝します．本研究で使用した Python，Numpy，Vue.js といった様々なオープンソースソフトウェアの開発に貴重な時間と技術を注いでいらっしゃるオープンソースソフトウェア開発者の方々には尊敬と感謝が絶えません．

最後になりますが，金銭，生活等様々な面で学生生活を支えていただいた両親，祖父母には大きく感謝します．本当にありがとうございました．

参考文献

[1] Daniel D. Lee and H. Sebastian Seung, “Learning the parts of objects by non-negative matrix factorization,” *Nature*, pp. 788–788, vol. 401, no. 6755, 1999.

[2] Nancy Bertin, Cédric Févotte, and Jean-Louis Durrieu, “Nonnegative matrix factorization with the Itakrua-Saito divergence: With application to musci analysis,” *Neural Computing*, pp. 793–793, vol. 21, no. 3, 2009.

[3] Bhiksha Raj, Paris Smaragdis, and Madhusudana Shashanka, “Supervised and semi-supervised separation of sounds from single-channel mixtures,” in *ICA'07 Proceedings of the 7th international conference on Independent component analysis and signal separation*, 2007, pp. 261–261.

[4] Naoya Takahashi, Thomas Kemp, Michael Enenkl, Franck Giron, Marcello Porcu, Stefan Uhlich, and Yuki Mitsufuji, “Improving music source separation based on deep neural networks through data augmentation and network blending,” in *International Conference on Acoustics, Speech and Signal Processing*, 2017, pp. 261–261.

[5] Emad M. Grais and Mark D. Plumbley, “Single channel audio source separation using convolutional denoising autoencoders,” in *Global SIP*, 2017, pp. 1265–1265.

[6] Rita Singh, Yuki Mitsufuji, Keiichi Osako, and Bhiksha Raj, “Supervised monoural source separation based on autoencoders,” in *International Conference on Acoustics, Speech and Signal Processing*, 2017, pp. 11–11.

[7] 浅野 太 , “音のアレイ信号処理 ,” コロナ社 , 2011.

[8] Pierre Comon, “Independent component analysis, a new concept?” *Signal Processing*, pp. 287–287, vol. 36, no. 3, 1994.

[9] Paris Smaragdis, “Blind separation of convoluved mixtures in the frequency domain,” *Neurocoputing*, pp. 21–21, vol. 22, no. 1, 1998.

[10] Shiro Ikeda, Noboru Murata, and Andreas Ziehe, "An approach to blind source separation based on temporal structure of speech signals," *Neurocoputing*, pp. 1–1, vol. 41, no. 1, 2001.

[11] Shoko Araki, Ryo Mukai, Hiroshi Sawada, and Shoji Makino, "A robust and precise method for solving the permutation problem of frequency-domain blind source separation," *IEEE Transactions on Speech and Audio Processing*, pp. 530–530, vol. 12, no. 5, 2004.

[12] Akinobu Lee, Tsuyoki Nishikawa, Toshiya Kawamura, Hiroshi. Saruwatari, and Kiyohiro Shikano, "Blind source separation based on a fast-convergence algorithm combining ICA and beamforming," *IEEE Transactions on Audio, Speech, and Language Processing*, pp. 666–666, vol. 14, no. 2, 2018.

[13] Soo-Young Lee, Hagai Thomas Attias, Taesu Kim, and Te-Won Lee, "Blind source separation exploiting higher-order frequency dependencies," *IEEE Transactions on Audio, Speech, and Language Processing*, pp. 70–70, vol. 15, no. 1, 2007.

[14] Nobutaka Ono, "Stable and fast update rules for independent vector analysis based on auxiliary function technique," in *Workshop on Applications of Signal Processing to Audio and Acoustics*, 2011, pp. 189–189.

[15] Hirokazu Kameoka, Hiroshi Sawada, Nobutaka Ono, Daichi Kitamura, and Hiroshi Saruwatari, "Determined blind source separation unifying independent vector analysis and nonnegative matrix factorization," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, pp. 1622–1622, vol. 24, no. 9, 2016.

[16] Hirokazu Kameoka, Hiroshi Sawda, Nobutaka Ono, Daichi Kitamura, and Hiroshi Saruwatari, "Determined blind source separation with independent low-rank matrix analysis," in *Audio Source Separation*, 2018, pp. 125–125.

[17] Nobutaka Ono, Daichi Kitamura, and Hiroshi Saruwatari, "Experimental analysis of optimal window length for independent low-rank matrix analysis," in *European Signal Processing Conference*, 2017, pp. 1170–1170.

[18] Syed Mohsen Naqvi, Yanfeng Liang, and Jonathon A. Chambers, "Overcoming block permutation problem in frequency domain blind source separation when using AuxIVA algorithm," *Electronics Letters*, pp. 460–460, vol. 48, no. 8, 2012.

[19] Yu Takamune, Hiroshi Saruwatari, Norihiro Takamune, Daichi Kitamura, Yoshiki Mitsui, and Kazunobu Kondo, "Independent low-rank matrix analysis based on parametric majorizaion-equalization algorithm," in *IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing*, 2007, pp. 1–1.

[20] Fumitada Itakura and Shuzo Saito, "Analysis synthesis telephony based on the maximum likelihood method," in *International Congress on Acoustics*, 1968, pp. C–17–C–17.

[21] Ian Fette, "The Websocket Protocol," *draft-ietf-hybi-thewebsocketprotocol-10*, pp. 1–1, vol. 6455, 2011.

[22] Sadayuki Furuhashi, Satoshi Tagomori, Yuichi Tanikawa, Stephen Colebourne, Stefan Friesel, René Kijewski, Michael Cooper, Kota Uenishi, wssbck, Gabe Appleton, Eric Cochran and Bernhard Mäser, *messagepack.org*, 2018.

[23] Andreas Ziehe, Guido Nolte, Zbyněk Koldovsky, Emmanuel Vincent, Francesco Nesta, Shoko Araki, and Alexis Benichoux, "The 2011 signal separation evaluation campaign (SiSEC2011):-audio source separation," in *International Conference on Latent Variable Analysis and Signal Separation*, 2012, pp. 414–414.

[24] Rémi Gribonval, Emmanuel Vincent, and Cédric Févotte, "Performance measurement in blind audio source separation," *IEEE Transactions on Audio, Speech, and Language Processing*, pp. 1462–1462, vol. 14, no. 4, 2006.

発表文献一覧

国内学会

[1] 中野将生 and 北村大地 , “ユーザーからの補助情報を用いるインタラクティブ音源分離システム ,” in 日本音響学会 *2020* 年春季研究発表会 , 2020, 2-P-38 (to appear).