能の謡分析のためのブラインド音源分離を用いたメロディ抽出

田本 篤喜
Atsuki Tamoto
法政大学大学院情報科学研究科情報科学専攻
atsuki.tamoto.6u@stu.hosei.ac.jp

Abstract

The purpose of this study is to extract singing melody from mixed sounds related to Noh performances. Noh sounds include singing, accompaniments, and other elements. For analyzing Noh singing, we need singing solos, but they are hard to collect since there are only a few sources of solo passages. Therefore, we focus on the extraction of singing melody from mixtures of accompaniments and singing. In this paper, we demonstrate that source separation can be introduced as an efficient preprocessing step for Noh singing melody extraction. In addition, we compare melody extraction based on a convolutional neural network (CNN) and Long short-term memory(LSTM) approach with Melodia, a plug-in for melody extraction which is particularly accurate in the presence of music with wide fluctuations in pitch. Raw Pitch Accuracy and Overall Accuracy are introduced as evaluation metrics. Our experimental results show that it is efficient for melody extraction to introduce source separation. We also demonstrated that Deep learning-based melody estimation can be efficiently trained using singing after source separation. Finally, LSTM with U-Net yields 91.7% as RPA, and Melodia with U-Net yields 91.3% as OA.

1 まえがき

音楽においてヴォーカルは主要な役割を担う．本研究では西洋の音楽ではなく，能を対象にする．日本の伝統芸能である能楽は，重要無形文化財に指定され，ユネスコ無形文化遺産にも登録されている．能楽は，狂言と能の総称である．能は，役に扮して舞台に立つ立方と，もっぱら音楽を受け持つ地謡方，囃子方とで成り立つ．立方のうち，主人公であるシテは，舞台の進行役を務める．囃子方は，笛方，小鼓方，大鼓方，太鼓方の四種の楽器で構成される．囃子は，声楽部である謡や動作部である所作とならぶ重要な表現要素である [1]．謡のみによって構成される場面，謡と囃子がともに演奏される場面，囃子のみが演奏される場面，が能を構成する音として挙げられる [2]．なお，謡のみの音源のことを素謡と言い，一人での素謡のことを独吟という．

能楽では，囃子の各楽器や立方の各役籍ごとに，流儀が複数存在する．実際の能の舞台は，異なる流儀の役籍が混じって構成されている．また，会場に観客が入るなど，練習環境と舞台本番の環境は異なる．そのため，練習時の素謡と，異なる流儀の囃子が混じった本番の舞台での謡では，謡が変化していることが考えられる．同じ人の同じフレーズでも音高や音高の遷移が異なることも考えられる．また，西洋音楽と異なり，能の謡には絶対的な音高は存在しない．謡の練習時には謡本というセリフが書かれた台本を用いるが，音高に関する指示は明確には記述されていない．謡の練習は，師匠の謡をまねるなどして練習する．そのため，謡の音高は，同じ流儀の師匠や個人の声質によって形作られていくことになり，謡手の身体的特徴や性別の違いによる謡の変化は許容される．以下の図 1 に謡本の一部分 (上部) と，対応する実際の謡のメロディ (下部) を示す．

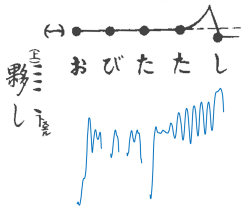

図 1. 謡本と実際の謡のメロディ遷移

図 1 に見られるように，実際のメロディは謡本に示されている音高の遷移と異なっていることがわかる．特に，ピッチが上下している揺らぎ部分やフレーズの謡い出しに見られる低いピッチから高くなっていくメロディ遷移は，謡本には記されていることがわかる．

このように，能では音階がはっきりしないために，音階の客観的な研究が必要である．そのために，メロディを正確・詳細に求める必要がある．西洋の音楽とは異なり絶対的な音高が決まっていない能においては，音階を分析するのにデータが多く必要である．多くの独吟の音源に対して謡を分析したいと思うが，入手できる音源は多くない．伴奏付きの音源は，比較的多く入手可能だが，囃子が混じっていることで，正確なメロディの抽出が困難である．伴奏付きの謡の音源からも分析できることで，より多くの謡の分析が可能になる．本研究では，伴奏付きの謡の音源に対して謡のみのメロディをアカペラ並みに分析することを目的とする．能の謡においては，演目が 200 程度と少ない．そのため，機械学習・深層学習等の手法を用いる場合は学習データの少なさも大きな問題である．

2 関連研究

これまで，複数の音が混じった音源からメロディを推定する従来手法が提案されている．Melodia は，ポリフォニックミュージックを対象としたメロディ推定器である [3]．ポリフォニックミュージックとは，メロディが一意に決められない音楽のことを言う．なお，Melodia[3] の開発では，ポリフォニックミュージックにおけるメロディの定義は [4] の定義を用いており，“口笛を吹いてください，と聴取者に頼んだときに吹く 1 本のピッチシーケンスがメロディである”としている．スペクトログラム上で顕著な倍音構造に着目した手法であり，Salience function によってスペクトログラムにおける顕著な値を抽出する．

ニューラルネットワークを用いたメロディ推定手法も近年提案されている．[5] では，混ざった音からメロディを推定するために CNN を導入している．CNN によるメロディ推定器では，混ざった音のスペクトログラムから直接所望の音のメロ

Supervisor: Prof. Katunobu Itou

ディを推定するネットワークを学習し, 使用する. CNN によるメロディ推定器は, 所望の音以外の音が混じる場合にロバストであるという結果がある [5]. また, CNN によるメロディ推定器を応用した研究が近年発表されている. [6] では, CNN によるメロディ推定と並行して, 自己相関を入力としてメロディを推定する DNN を用いて出力確率を算出しておく. 二つのネットワークの出力確率を入力とする全結合層で最終的なメロディを求める手法を提案している. また, [7] では, メロディを推定する CNN への入力特徴を別の 1 次元 CNN で構成しておき, メロディ推定 CNN へ入力・推定するという手法を提案している. この手法は音声時間波形が特徴抽出 CNN への入力となり, End-to-End での推定となる. CNN ではなく LSTM を導入して精度を比較した研究 [8] も存在し, 良い精度が得られている. 能は, シテの謡と楽器音だけでなく, 囃子方の掛け声や, 地謡方による謡なども含まれる. したがって, CNN を用いる上記の手法は有用であると考えている.

U-Net を用いた, 音源分離とメロディ推定の同時学習手法を提案している研究も存在する [9]. U-Net は画像を再構成するネットワークであり, [5] のようなフレーム単位でピッチクラスタリングを行うメロディ抽出ネットワークでは同時学習ができないため, メロディを表す画像情報をラベルデータとしている.

3 提案手法

最新の従来研究 [7] では, ピッチの変動の多いデータセットに対しては CNN ベースのメロディ推定器よりも Melodia が良い性能であるとの報告がある. 能の謡には西洋の音楽におけるビブラートよりもピッチ変動の大きい揺らぎが多く存在する. そこで本研究では, Melodia によるメロディ推定の前処理として音源分離を施すことで能の謡の正確なメロディ推定を狙う. Melodia はメロディを推定する上で, スペクトログラム上における倍音構造が重要である. そこで, 倍音構造まで高解像度で分離できることを狙い, U-Net による音源分離を導入する. 加えて, 音源分離後の音でメロディ推定 CNN を学習することで, メロディ推定 CNN によるメロディ推定性能も向上することも確かめる. 従来研究 [8] にて CNN よりも良い結果が得られている LSTM も導入し精度を比較する.

音源分離後の音に対してメロディを推定することで, 無音区間の検出もより正確にできることが考えられる. 図 2 が処理のダイアグラムである.

音源分離を施すことで, 囃子を抑圧し, 謡を強調した音源からメロディラインを推定することが可能になる.

深層学習モデルにおいては, トレーニングデータを増大させることで, よりロバストにできる. そこで, データ量を増やす処理を施す. 以下, 3.1 章では音源分離について, 3.2 章ではメロディ推定について, 4 章では Data augmentation について述べる.

3.1 ニューラルネットワークによる時間周波数マスク推定に基づく音源分離

二つ以上の音源の音から構成される混合音を分離する手法を音源分離という. 音源分離手法はいくつか存在する. 一つは音が伝わる方向を認識して音を強調するビームフォーミングである. ビームフォーミングでは, 複数のマイクから得られる情報から音の到達方向を推定する. 本研究では 1 チャンネル信号を対象とするため, ビームフォーミングを導入することはできない. また, 決まった音色のサンプルがある場合は, 混合音のスペクトログラムを行列とみなし, 音色のパターンに対応する行列 (事前に学習済み) と, その各スペクトルのパターンに対応する強度を表す行列に分解する, 非負値行列因子分解 (NMF) が適用できる. 関連研究 [10] でも, ギターの音と歌の混合音から歌のメロディを抽出するために音源分離をしているが, NMF を適用している. しかし, 本研究では決まったスペクトルパターンがわからない状況での分離であるため, 時間周波数マスクを用いた音源分離を実現する. 1 チャンネル音源を用いており, 音源の位置などは参考にできないため, ブラインド音源分離と呼ばれる.

3.1.1 時間周波数マスキング

能における主人公の謡と楽器の音が混じった混合音から音声部分を分離する手法として, 時間周波数マスキングを導入する [11]. 時間周波数マスキングとは, 混合音を時間周波数領域で分離する手法である. 時間周波数マスクには, 各点のデータ表現の違いで区別できる, バイナリ時間周波数マスクとソフト時間周波数マスクの 2 種類がある. 混合音がソース 1 とソース 2 の音源からの, 2 つの音が混じった音であると想定する場合, ソース 1 に関するバイナリ時間周波数マスク $\mathbf{M_b}$ は, 以下の式 (1) のように定義できる.

$$\mathbf{M_b} = \begin{cases} 1 & |\hat{\mathbf{y}}_{1_t}(f)| > |\hat{\mathbf{y}}_{2_t}(f)| \\ 0 & otherwise \end{cases} \tag{1}$$

ここで, $f = 1...F$ は異なる周波数を, $\mathbf{y}_1$ は 1 番目の音源のスペクトログラムを表している. この式 (1) では, 混合音のスペクトログラムのある点が, 所望の音か他方の音に完全に帰属すると仮定している. スペクトログラムの各点を 0 または 1 に分類するため, 以下で説明するソフト時間周波数マスクに比べて単純な問題になる.

ソース 1 に関する理想的なソフト時間周波数マスク $\mathbf{M_s}$ は以下の式 (2) のように定義できる.

$$\mathbf{M_s} = \frac{|\hat{\mathbf{y}}_{1_t}(f)|}{|\hat{\mathbf{y}}_{1_t}(f)| + |\hat{\mathbf{y}}_{2_t}(f)|} \tag{2}$$

この式 (2) は, 混合音のスペクトログラムの各点において, 所望の音と別の音がある割合で混ざり合っていることを表しており, 振幅スペクトルに基づいてその割合を計算している.

計算した時間周波数マスク $\mathbf{M}(\mathbf{M_b}$ or $\mathbf{M_s})$ を, 混合音に適用することで, 所望の音を抽出できる. ソース 1 に関する理想的なマスクを使用してソース 1 に対応する音を取り出したいときは,

$$\hat{\mathbf{s}}_{1_t}(f) = \mathbf{M}(f)\mathbf{X}_t(f) \tag{3}$$

の処理で取り出すことができ, ソース 2 に対応した音を取り出したいときは,

$$\hat{\mathbf{s}}_{2_t}(f) = (1 - \mathbf{M}(f))\mathbf{X}_t(f) \tag{4}$$

の処理で取り出せる. なお, $\mathbf{X}_t$ は混じった音のスペクトルを表している. 図 3 は理想的なソフトマスクとバイナリマスクの例を示している.

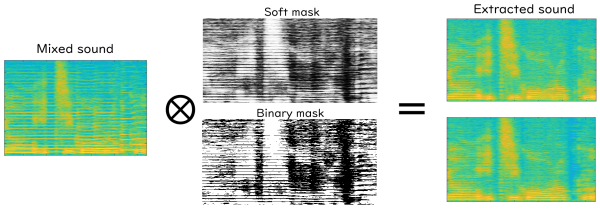

図 3. ソフト時間周波数マスクとバイナリ時間周波数マスクの例 アダマール積はスペクトログラムとマスクの要素ごとの積を意味する.

バイナリマスクの各点は 0 か 1 の値であることが確認できる. それぞれの理想的なマスクで分離した音源を聴取した限りでは, ほとんど差は感じないが, 本研究では, 倍音構造を崩しにくいことを期待し, ソフト時間周波数マスクを導入する.

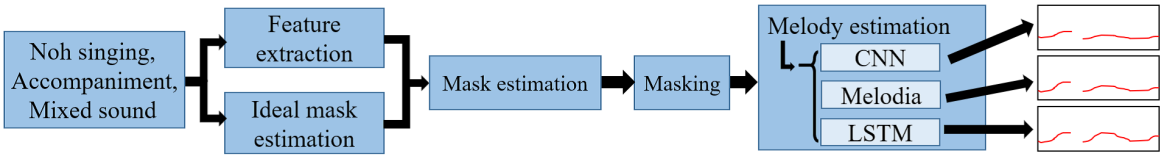

図 2. 処理の流れ

3.1.2 ニューラルネットワークによる時間周波数マスク推定

所望の音源からの音声だけを強調する理想的な時間周波数マスクを求めるためにニューラルネットワークを用いる．本研究では，DNN によるマスク推定と U-Net[12] による時間周波数マスク推定を実装し，評価する．

時間周波数マスク推定のための DNN の構成を以下の図 4 に示す．

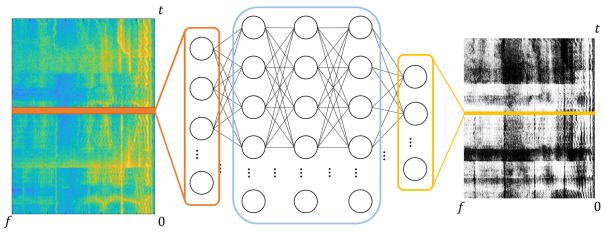

図 4. 時間周波数マスク推定 DNN の構成

音声のつながり具合を特徴としてニューラルネットワークに学習させる．つまり，学習時には混合音のスペクトル情報を複数フレーム連結させたデータ (スペクトログラム) を入力，事前に計算した理想的な時間周波数マスクを正解として与えて学習させる．実際の推論時には，学習時の入力と同じフレーム数分のスペクトル情報を学習済みのニューラルネットワークに入力することで，時間周波数マスクが出力される．

U-Net は CNN ベースのネットワークであり，医療画像の高解像度化に関する研究に用いられた手法である．[12] で，音源分離に用いるマスク推定器として導入され，有効性が示されている．時間周波数マスク推定のための U-Net の構成を以下の図 5 に示す．

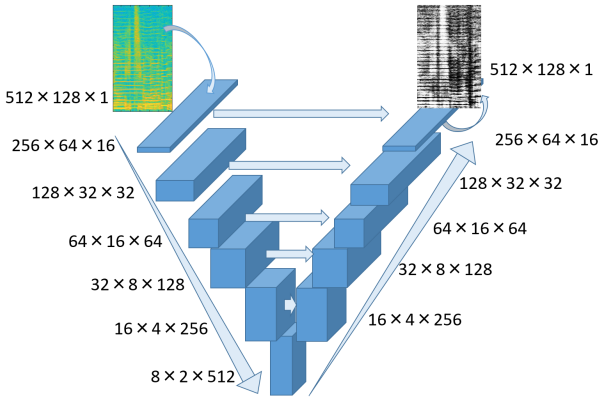

図 5. 時間周波数マスク推定 U-Net の構成

すべてのエンコード層は 2 ストライド，カーネルサイズ 4×4 の 2 次元畳み込み，バッチノーマライゼーション，Leaky relu で構成される．すべてのデコード層は，2 ストライド，カーネルサイズ 4×4 の逆畳み込み，バッチノーマライゼーション，Relu で構成され，始めの 3 層において 50% のドロップアウトを設定している．最終層の活性化関数には Sigmoid 関数を使用している．Optimizer は Adam を使用している．

各畳み込み層での出力を同じレベルの逆畳み込み層の入力データに連結して演算する．この処理は Low level skip connection と呼ばれる．従って，逆畳み込み演算は，一層前の 2 倍のサイズのデータに対して適用される．Low level slip connection によって各デコーダー層が同じレベルのエンコーダー層の解像度を考慮し，高解像度のマスク推定器を効率的に学習する．実際に U-Net を用いて分離したメロディの例を図 6 に示す．

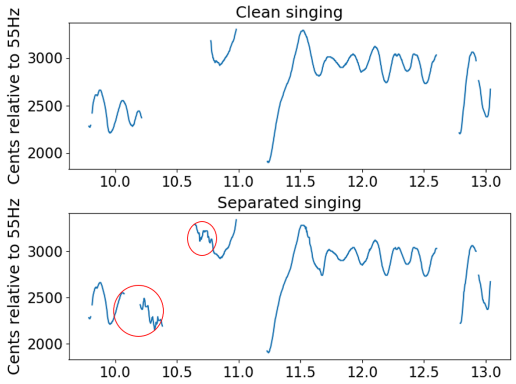

図 6. 独吟 (上部) と分離後の謡 (下部) のメロディ

大部分では正確に分析出来ているが，赤い丸印の部分のように，微かに残ってしまう囃子の音による誤推定や無検出部分は存在する．

3.2 メロディ推定器

3.2.1 概要

音源分離によって主人公の謡以外の音を抑圧した音に対してメロディを推定する．自己相関から得られる基本周波数 (F0) によって求める代表的なメロディ推定手法では，生の音声時間波形の自己相関を用いるため，F0 を求めたい目的音以外の音が対象のファイルに混じっている場合，それらの音に強く影響されてしまう．音声のみに対して自己相関によって F0 を求めたグラフと音声に他の音が混じった音に対して自己相関によって F0 を求めたグラフを以下の図 7 に示す．

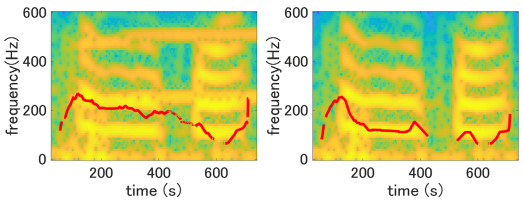

図 7. 左:雑音混じりの音声，右:クリーンな音声

雑音成分の影響で，正確な F0 が抽出できていないことが確認できる．本研究では音源分離後に適用するが，音源分離後でも目的音のみを分離できているわけではなく目的音以外の音が混じっていることが考えられるため，音声時間波形の自己相関によってメロディを求めるべきでない．本研究では Melodia と CNN,LSTM によるメロディ推定器を比較する．

3.2.2 CNN,LSTM によるメロディ抽出

従来研究では, 伴奏が混じった音から歌の部分のメロディを推定するため, それらの混合音を入力としている. 本研究では音源分離後の音声でメロディ推定 CNN を学習することによる精度を確かめるため, 音源分離後の謡部分が入力に対応する. 音源分離後の音の時間周波数情報から直接メロディを推定する CNN の構成を以下の図 8 に示す.

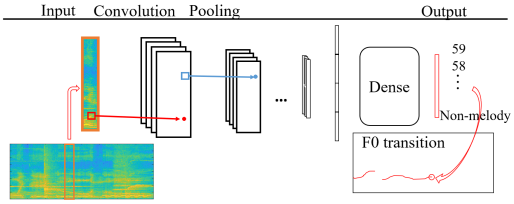

図 8. CNN を用いたメロディ推定器の構成

入力データとして音源分離後の謡のスペクトログラム, ラベルデータには謡部分のみのメロディに対応した値を cent に変換して設定し分類器として学習する. 周波数の値からセントへの変換式は以下の式 5 である [13]. LSTM によるメロディ推定器においても同様の入出力対応である.

$$\mathbf{Cent}(f) = 1200 * \log_2 \frac{f}{55.0} \tag{5}$$

なお, 正解ラベルに対応するメロディの決定には, 既存のメロディ推定器を用いる [3]. CNN によるメロディ推定器はスペクトログラム上の局所的な倍音構造のパターンを学習する. 従来研究 [5] では, 雑音に影響されて微妙に正解ピッチとズレが生じることがあると報告されている. LSTM によるメロディ推定器の特徴は, シーケンスの時間関係を柔軟に学習することであり, メロディラインの大きな変動も前後の情報で考慮し, メロディ推定ができることである [8]. メロディ推定の前処理として音源分離を施す本研究では, 滑らかなメロディラインにおける突発的なピッチの欠如を回避できることを期待し導入する.

4 DATA AUGMENTATION

深層学習モデルにおいては, データの量は精度に大きく影響する. データを増大させることで, よりロバストにできる. 一方で, 入手可能な能の謡や囃子の音源は少ない. そこで, データを増量する処理を施す.

予め, 持ち得るすべての能の謡と囃子の音源を同じ長さにカットしておく. ひとつの謡のセグメントに対していくつかの組み合わせで囃子を足し合わせる. つまり, ひとつの謡に対して複数の組み合わせの混合音を準備することができる. 音源分離の学習データとする際は, 多くの種類をカバーするために, すべての混合音群から学習データ量分, ランダムに混合音を抽出する. 能の謡の演目は絶対的に少なく, 歌い方に特徴もあるため, このようにすることで, 多くのバリエーションの混合音に対するマスクを学習できることになる.

5 評価

以下の 5 項目について評価する.

- Melodia を混合音に対して直接適用
- メロディ推定 CNN を混合音に対して直接適用
- DNN 音源分離によって分離した音に対して Melodia を適用
- U-Net 音源分離によって分離した音に対して Melodia を適用
- DNN 音源分離によって分離した音に対してメロディ推定 CNN を適用
- U-Net 音源分離によって分離した音に対してメロディ推定 CNN を適用
- U-Net 音源分離によって分離した音に対してクリーンな謡で学習したメロディ推定 CNN を適用
- U-Net 音源分離によって分離した音に対してメロディ推定 LSTM を適用

本研究では, メロディを分析できるようにすることが目的あり, 謡っているフレームかどうかは使用者が音を聴き, メロディのグラフを見て判断すれば良いと考えている. そこで Raw Pitch Accuracy(RPA) と Overall Accuracy(OA) の尺度で評価を行う [14][15]. RPA は正解音声区間におけるピッチ正解フレーム数である. OA は, 正解音声区間におけるピッチ正解フレーム数に加えて, 正解無音区間において正しく無音区間と検出されたフレーム数を合わせた尺度である. 従来研究に基づき, 正解の前後 50cents は正解ピッチとして算出する.

5.1 使用するデータ

音源分離 DNN,U-Net ともに, 実際に収集した独吟と囃子の音源を用いる. 含まれる謡の流儀は 5 種類, 人数は約 30 人である. 混合音は 4 章で示したように, 独吟に囃子を混ぜて作成する. また, メロディ推定 CNN, LSTM の学習データには, 音源分離部で使用しなかったデータで混合音を作成し, 音源分離に適用した後の音を用いる. 音源分離 DNN の学習データ量は約 11 時間, U-Net の学習データ量は約 105 時間, メロディ推定 CNN, LSTM の学習データ量は約 5 時間 30 分である. 評価には, 30 秒ごとにランダムに抽出した 60 ファイル分のデータとした.

6 実験条件

DNN 音源分離部では, 音声のつながり具合を学習させるために, 512 点でフーリエ変換した 257 点の振幅スペクトル情報を, 対象のフレームとその前後 3 フレーム分, 1 方向に連結させ, DNN の入力とする. つまり, 入力層は (257×3) + 257 + (257×3) = 1799 ノードとなる. 中間層は 4 層あり, 入力側から 512-512-512-512 のノードを持つ. 出力層では, 対象の 1 フレーム分のノードを持つ. よって出力層のノードは 257 ノードとなる. 損失関数には, 正解マスクと推定されたマスクの二乗誤差を最小化する式を導入している. 正解マスクには, ソフト時間周波数マスクを用いる. 機械学習フレームワークは Chainer を用いる. U-Net 音源分離の構造は, 3.1.2 章で示した通りである.

CNN の構造は, [5] で実装しているネットワークとハイパーパラメータ自動最適化ツールである Optuna を基に構成する. 入力は対象フレームの前 7 フレーム, 後ろ 8 フレームを連結した時間周波数情報としている. したがって, サイズは 160×16 となる. 2 つの畳み込み層と 2 つのプーリング層を持つ. 始めの畳み込み層のカーネルサイズは 5×5 である. 始めの畳み込み層は 15 個の特徴マップに対応する 15 カーネルを含んでいる. また, 2 つ目の畳み込み層はサイズ 5×5 で 20 カーネルを持つ. プーリングはサイズ 2×2 の平均プーリングである. 最後のプーリング層では全結合層に入力するために 1 次元にしている. 1 層目の全結合層は, 300 ノード, 2 層目の全結合層は 800 ノード, 3 層目の全結合層は 600 ノードの隠れ層を持ち, Softmax 関数を活性化関数としている. 最終出力も Softmax 関数である. Optimizer は Adam を使用している. LSTM の構造は, メロディ推定に LSTM を用いている従来研究 [8] と Optuna によって決める. LSTM は用途として回帰問題に使用されることが多いが, 本研究ではクラスタリング問題として実装する. 入力は対象フレームの前 12 フレーム, 後ろ 13 フレームを連結

した時間周波数情報である．LSTM 層は 1 層で, 1500unit を持つ．最終出力の活性化関数は CNN と同様 Softmax 関数である．Optimizer は RMSprop を使用している．メロディ推定 CNN, LSTM ともに, ラベルデータは, 正解周波数をセントに変換し, 従来研究に基づき 50 ノードごとにまとめて 1 ノードとしている．なお, ラベルデータ生成時, 能の謡のピッチ幅に合わせ, 最低周波数を 73Hz, 最高周波数を 400Hz として計算している．学習の簡単のため, メロディ推定器の入力スペクトログラムは正規化処理を施すのが一般的である．学習データ全体での正規化と, 固定長にカットされたファイル単位での正規化を比較する予備実験を行った結果, ファイル単位で正規化を施す方が良い結果が得られたため, ファイル単位での正規化を行い, 入力スペクトログラムとする．

DNN 音源分離では対象フレームの前後 3 フレームをコンテキストとしているため, 評価の際は音声の前後 3 フレームは評価に含めない．また, メロディ推定 CNN では, 対象フレームの前 7 フレーム, 後ろ 8 フレーム, LSTM では対象フレームの前 12 フレーム, 後ろ 13 フレームをコンテキストとしているため, 評価の際は音声に対してこのコンテキスト分は含めない．

7 結果と考察

7.1 章では, 5 章で示した客観評価尺度に関する結果と考察を示す．7.2 章では, メロディ推定器の入力データのコンテキストの検討を行った結果を示す．7.3 章では, 実際の能の音源に対してメロディを推定した結果を示す．

7.1 結果

混合音に対して直接 Melodia を適用した結果 (1), メロディ推定 CNN を混合音に対して直接適用した結果 (2), DNN 音源分離後の音に対して Melodia を適用した結果 (3), U-Net 音源分離後の音に対して Melodia を適用した結果 (4), DNN 音源分離後の音に対してメロディ推定 CNN を施した結果 (5), U-Net 音源分離後の音に対してメロディ推定 CNN, LSTM を施した結果 (6),(8), U-Net 音源分離によって分離した音に対してクリーンな謡で学習したメロディ推定 CNN を適用した結果 (7) を表 1 に示す．

表 1. 実験結果 (%)

	RPA	2. OA
1. Melodia	72.3	76.5
2. CNN	79.8	78.8
3. DNN+Melodia	48.6	59.2
4. U-Net+Melodia	91.4	**91.3**
5. DNN+CNN	58.2	69.2
6. U-Net+CNN	90.4	88.2
7. U-Net+Clean CNN	82.2	84.5
8. U-Net+LSTM	**91.7**	87.4

1. と 3. を比較すると, 1. の方が良い結果が得られている．1. と 3. の RPA と OA の差に着目すると, 3. の方が差が大きく, OA の方が良いことがわかる．このことから, 音源分離を施したことで無音区間を無音区間として分析できていると言える．3. と 5. を比較すると, どちらの尺度においても 5. の方が良い結果であることがわかる．音源分離後の雑音なども混じった音でメロディ推定 CNN を学習したことが貢献していると考えられる．メロディ推定の前に DNN 音源分離を施している 3.5. と U-Net 音源分離を施している 4.6. を比較すると, DNN 音源分離を施した謡に対するメロディ推定性能よりも U-Net を施した謡に対するメロディ推定性能の方が良い結果であることがわかる．また, U-Net 音源分離に関する評価に一貫して, DNN 音源分離に関する評価ほどの OA と RPA の差はないことが確認できる．このことから, 高解像度でマスクを推定できる U-Net による音源分離を施すことで, 無音区間のみならず, メロディ区間においても高精度でメロディを推定できると言える．6. と 7. を比較すると, 6. の方が 3 ポイント以上良い結果が得られた．この結果から, U-Net 音源分離をメロディ推定の前処理として施す場合でも, 音源分離後の謡でメロディ推定 CNN を学習することで, ロバストなメロディ推定が可能であることがわかる．4. が全ての結果の中で総合的に最も良い結果であった．スペクトログラム上の倍音構造が重要な Melodia を適用する前処理として, 高解像度でマスクを推定する U-Net による音源分離を施したことで高性能なメロディ推定が実現できた．6. と 8. を比較すると, OA は CNN の方が良いが, RPA に関しては, コンテキストをより柔軟に考慮できる LSTM の方が 1 ポイント以上向上する結果が得られた．音声区間のみでは, LSTM が最も良い結果を示した．図 9 に CNN と LSTM による出力メロディの例を示す．赤い部分に着目すると, ラベルデータのフレーム数をメロディ推定器と合わせる都合上, 除かれてしまった謡い出しのフレーズが LSTM では推定できている．客観評価尺度の算出上では, メロディの誤検出として算出されているが, この謡い出しは, 図 1 にも見られるように, 能においては謡本と異なることが多く, 重要なメロディの特徴である．時間関係を柔軟に学習したことでこのような部分が推定できる LSTM が, 能を対象にした場合は最も適していると考えられる．

学習データとして能の謡と囃子を大量に集め, 音源分離の学習データとした．能の演目は 200 程度と少なく, 囃子の楽器も 4 種類に限られる．加えて, 西洋音楽に比べて独特の倍音構造があるため, 能のための音源分離器を構成することができたことで, 良い性能が得られたと考えられる．

囃子方には, 楽器音に加えて囃子方自身による掛け声も含まれる．掛け声が存在する区間でも概ね正確にメロディが抽出できたことから, 囃子方の掛け声にも謡とは違った時間周波数特性があるといえる．

7.2 メロディ推定器の入力データコンテキストの比較

最終的なメロディ推定器の入力データのコンテキストは, CNN では, 対象フレームの前 7 フレーム, 後ろ 8 フレームとし, LSTM の場合, 対象フレームの前 12 フレーム, 後ろ 13 フレームとした．メロディ推定器の入力データコンテキストを決定するにあたり, 前 7 フレーム, 後ろ 8 フレームの場合と, 前 12 フレームと後ろ 13 フレームでの比較を行った．ここでは, 比較に関する考察を述べる．

CNN, LSTM それぞれで比較実験を行った全ての結果を以下の表 2 に示す．なお, 実験に用いた U-Net の学習時間は 104 時間であり, メロディ推定器の学習データ量はそれぞれ約 4 時間分である．

表 2. コンテキストの検討結果 (%)

	RPA	OA
CNN(7,8)	88.4	88.3
CNN(12,13)	86.7	85.9
LSTM(7,8)	87.9	86.3
LSTM(1213)	91.4	88.2

表 2 から, LSTM は, 時間構造を柔軟に学習するため, コンテキストを長くすることで, より良い精度が得られた．一方で, CNN は, 少ないコンテキストの方が良い結果となった．コンテキストを増やした結果, 2 層の畳み込み層の処理後の点数が増えたことで, 全結合層の入力のノード数が増え, 精度が悪化したことが考えられる．ここで, 3 層目の 2 次元畳み込み層として, カーネルサイズ 3×3 の畳み込み層と 2 点の平均プーリングを加える．結果, **RPA:89.1%, OA:86.9%** が得られた．RPA は向上し, OA は悪化する結果となったが, 前 7 フレーム, 後ろ 8

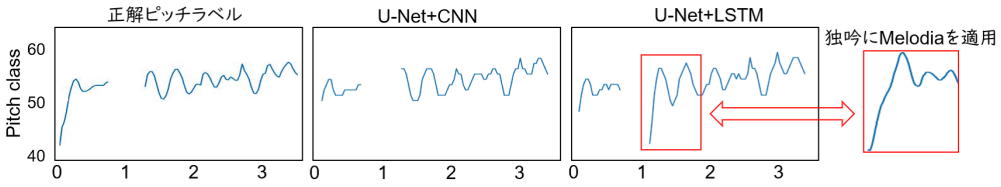

図 9. CNN と LSTM による出力メロディの比較. 左から正解ピッチラベル, U-Net+CNN,U-Net+LSTM の出力, 独吟に対して Melodia を適用したメロディを示している.

フレームの結果からの大きな改善は見られなかった. この結果から, CNN の場合, 入力データのコンテキストと, ネットワーク構成のバランスが重要であることがわかる.

7.3 実際の舞台の音源に適用

本研究では, 独立な独吟と囃子の音を混ぜて結果を算出した. ここでは, 本手法が実際の音源にも有効であることを示すために, 実際の音源に対しても適用した. 実際の舞台での音源に直接 Melodia を適用したメロディ遷移と, 実際の舞台での音源に U-Net 音源分離を施し, Melodia を適用したメロディ遷移のグラフを以下の図 10 に示す.

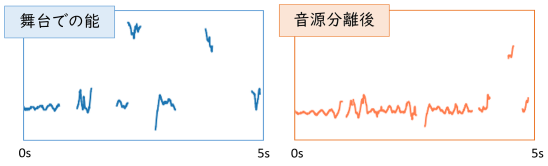

図 10. 実際の舞台での音源に適用

Melodia を直接適用したグラフでは正確に分析できていないが, 音源分離を施すことで, 正確に分析できていることが音を聴くとわかる.

8 結論と今後の展望

本研究では, メロディ推定器の前処理として音源分離を施す手法を実装した. DNN, U-Net の二つの音源分離と, Melodia, CNN, LSTM によるメロディ推定器を比較した結果, 総合的には U-Net による音源分離 +Melodia が最も良い性能が得られた. 音声区間のみにおいては LSTM が最もよく, RPA91.7% が得られた. また, 実際の出力メロディを比較することで, 能を対象にする場合は LSTM によるメロディ推定器が最も適していることが確認できた. 能の演目が少なく, 今回大量に能のデータを収集できたことで, 良いメロディ推定精度が得られた. 加えて多くの組み合わせを考慮する Data augmentation によってこの良い結果が得られたと考えられる.

また, 実用性を論証するために, 実際の舞台にて収集した能の音源に対しても適用し, 可視化した. 音を聴くことで, メロディの遷移が概ね謡のみのメロディになっていることが確認できた.

今回は仮想的に独吟と囃子の音を混ぜて混合音を作成し, 学習, 評価に用いた. 予備実験として実際の能の音源に対しても実行し, 分離, メロディ推定ができることが確認できた. 囃子や独吟のみの音源は近くに設置したマイクで収録される. 一方で, 実際の能の音源は, そのように近くで録音することは困難であることが多く, ある程度距離のある位置で録音される. 独吟のみ, 囃子のみの学習データに対して実際の能を録音する場を模擬した空間特性を畳みこむことで, さらなる音源分離性能向上が期待できる. さらなるロバスト性能向上に向けて, メロディ推定 CNN と Melodia の併用も検討できる. メロディ推定 CNN は, 目的音以外の音に対してロバストである. 一方 Melodia では, 音源分離後でも目的音の倍音構造が薄まってしまうと, 正確に推定できないこともあり得る. 音源分離後の音で学習したメロディ推定 CNN をピッチ候補 (幅) 抽出として使用し, その幅内で Melodia を使用することで, さらにロバストで正確なメロディ抽出が期待できると考える.

参考文献

[1] “囃子”, 新版 能・狂言事典, JapanKnowledge Lib, https://japanknowledge.com, (参照 2019-07-24)

[2] “能楽”, Wikipedia, https://ja.wikipedia.org/wiki/能楽.

[3] J. Salamon et al., “Melody extraction from polyphonic music signals using pitch contour characteristics,” IEEE Transactions on Audio, Speech, and Language Processing, vol. 20, no. 6, pp. 1759-1770, 2012.

[4] G. E. Poliner et al.,“Melody transcription from music audio: Approaches and evaluation,” IEEE Trans. on Audio, Speech and Language Process., vol. 15, no. 4, pp. 1247-1256, 2007.

[5] Hong Su et al., “Convolutional neural network for robust pitch determination,” in Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on. IEEE, 2016, pp. 579-583.

[6] Hsin Chou et al., “A HYBRID NEURAL NETWORK BASED ON THE DUPLEX MODEL OF PITCH PERCEPTION FOR SINGING MELODY EXTRACTION” In Proc. ICASSP, 2018.

[7] Ming-Tso Chen et al., “CNN BASED TWO-STAGE MULTI-RESOLUTION END-TO-END MODEL FOR SINGING MELODY EXTRACTION,” In Proc. ICASSP, 2019.

[8] H. Park et al.,“Melody extraction and detection through LSTM-RNN with harmonic sum loss,” In IEEE ICASSP, pages 2766-2770, 2017.

[9] Andreas Jansson et al.,“Joint singing voice separation and f0 estimation with deep u-net architectures,” in 2019 27th European Signal Processing Conference (EUSIPCO). IEEE, 2019, pp. 1-5.

[10] E. Gómez et al.,“Predominant fundamental frequency estimation vs singing voice separation for the automatic transcription of accompanied flamenco singing,” 13th International Society for Music Information Retrieval Conference (ISMIR 2012) (2012)

[11] P.-S.Huang et al., “Deep learning for monaural speech separation”, in Proc.IEEE Int. Conf.Acoust.,Speech,Signal Process. (ICASSP), 2014,pp.1562-1566.

[12] A. Jansson et al.,“Singing voice separation with deep u-net convolutional networks,” in Proc. 18th Int. Soc. Music Inf. Retrieval Conf., 2017, pp. 23-27.

[13] “Melody Extraction in Python with Melodia,” http://www.justinsalamon.com/news/melody-extraction-in-python-with-melodia

[14] C. Raffel et al., “mir eval: a transparent implementation of common MIR metrics,” in Proc. of the 15th ISMIR, Taipei, Taiwan, 2014.

[15] Justin Salamon et al., “Melody extraction from polyphonic music signals: Approaches, applications and challenges,” IEEE Signal Processing magazine, vol. 31, no. 2, pp. 118-134, Mar. 2014.