

音楽のコード認識技術とその応用

サービスイノベーション部 まつおか ほうせい わたべ みずき
松岡 保静 渡部 瑞季

近年、人工知能技術の発展により、音声対話などのシステムが実用化されつつある。ドコモでは、この音声対話システムに絶対音感や音楽的頭脳を組み込み、音楽を理解できるエージェントの実現を目的として、演奏の判定に必要な音響認識技術を開発した。この技術により、楽曲のコード進行の認識やアドリブ演奏の評価などが可能になった。本稿では、開発したコード認識技術とそれを利用したアドリブ演奏評価機能について解説する。

1. まえがき

音楽が生体や心理に与える影響に関して、生理学、心理学、医学、看護学、音楽療法など、多岐にわたる領域での研究が行われており、リラックス効果やストレス緩和効果などの科学的なデータも報告されている[1][2]。

特に音楽の演奏をすることは、脳の活性化にもつながることが実証され、今後の高齢化社会における認知症対策などにも重要な役割をもつと考えられている。

そこでドコモでは、1人でも音楽の演奏を楽しめるようなエージェントの開発を目的とし、演奏の判定に必要な音響認識技術を開発した。開発にあたって特に楽曲のコード認識技術に注目し、そこに

ディープ・ニューラル・ネットワーク（DNN：Deep Neural Network）^{*1}を適用することでコード認識の精度向上を図った。

さらに、このコード認識技術を応用したアドリブ演奏評価機能も開発した。本機能では、ユーザは自分の好きな曲をかけながら、それに合わせてアドリブを弾くことで、エージェントが曲に合っているかどうかを判定し、反応してくれる。このように、ユーザの演奏を理解できるエージェントを実現することで、音楽に対する楽しさや練習のモチベーションを向上させることができると考えられる。

本稿では、楽曲や楽器の演奏音の音程を理解するために必要となるクロマベクトルの概要と、本技術でのクロマベクトルの抽出方法について解説する。また、開発したコード認識技術における認識方法や、

^{*1} ディープ・ニューラル・ネットワーク（DNN）：機械学習の手法の1つであるニューラルネットワークの中間層を多層にしたネットワーク。

DNNを利用した認識方法について解説し、その精度についても報告する。さらにこのコード認識技術を利用したアドリブ評価機能について解説する。

2. クロマベクトル

2.1 概要

演奏音の音程認識において、単音であれば、f0抽出^{*2}の技術（自己相関法^{*3}など）を用いて、簡単に認識できるものもあるが、ギターやピアノなどの複数の音程が混ざった演奏音となると、f0抽出では、単音の音程しか認識できない場合や誤った音程を認識する場合が多いため困難である。そこで複数の音程の音や和音（コード）を解析するためには、クロマベクトルという特徴量^{*4}を用いることが多い [3]。

クロマベクトルとは、12音階およびオクターブ違いも含めた音階ごとの周波数の振幅強度を特徴量としたベクトルである。ここでの振幅強度とは、ある区間に、特定の周波数の信号がどれくらい多く含まれているかを示すものである。各音階の周波数は決まっており、例えばA（ラ）の音は440Hz、E（ミ）の音は660Hzとなる。周波数が2倍になると1オクターブ上の音程となり、1/2になると1オクターブ下の音程となる。このように周波数の比で音程が決まるため、12音階の音程は、周波数が $2^{1/12}$ 倍ごとに半音上がることになる。この各音階の周波数ごとに振幅強度を計算したものがクロマベクトルであり、これによりどの音階の音が強いかがわかるため、強い音階の組合せからコードを推定することなどが可能になる。

2.2 クロマベクトルの計算方法

(1) フーリエ変換による算出

ある音階の振幅強度を求めるためには、その音階の周波数付近の振幅強度の平均値を算出する必要がある。そこで音響（信号）から周波数成分を抽出す

るために、フーリエ変換^{*5}を用いることが多い。フーリエ変換による周波数解析では、直交周波数を用いて行う。この直交周波数の間隔は、解析する区間の時間長を T とすると、 $1/T$ (Hz) となるため、周波数分解能を高めるためには、ある程度の時間フレーム長が必要となる。特に低音域では、微小の周波数の変化で音程が変わるため、隣接する音程の周波数の差よりも細かい周波数分解能でのフーリエ変換が必要になる。そのため、速いメロディの分析が困難になる。

図1(a)にフーリエ変換による周波数解析を示す。フーリエ変換では、時間フレーム長が T 秒だとすると、サンプル数が n 点の場合、基底周波数^{*6}は $1/T$, $2/T$, $3/T \dots n/T$ となる。サンプル数とは、ある時間区間の信号を離散的に分割して量子化する分割数である。低音域のラの音 (55Hz) まで解析できるようにするためには、1つ上の音階のシの音 (58Hz) との差が3Hzなので、 $1/T$ が3Hzより小さくなるような周波数分解能が必要となる。したがって T は、333ms以上必要となり、333msより速い時間で音程が変わるメロディを解析することが困難になってくる。

(2) 本技術による算出

図1(b)に本技術での周波数解析を示す。本技術では、フーリエ変換の基底周波数を音階の周波数にし、フーリエ変換と同様に下記の演算式で振幅強度を算出する。

$$p(f) = \sum_{k=0}^n \cos\left(\frac{2\pi f k}{SF}\right) - i \sum_{k=0}^n \sin\left(\frac{2\pi f k}{SF}\right) \quad (1)$$

SF はサンプリング周波数で、音階の周波数 f の振幅・位相情報を $P(f)$ とし、 $P(f)$ の2乗を周波数 f の振幅強度とする。サンプリング周波数とは、信号波形を1秒間に何回サンプリングするかを示す値である。cos部分は、余弦波との相関をとり、sin部分は正弦波との相関をとっている。通常のフーリ

*2 f0抽出：倍音が含まれる波形信号において、最も低い基準周波数を解析すること。

*3 自己相関法：自己の波形信号同士をずらしながら相関をとり、どのくらいずれたタイミングで相関が高くなるかを検出して、基準周波数を推定する方法。

*4 特徴量：データから抽出される、そのデータを特徴づける値の

こと。

*5 フーリエ変換：信号の中に含まれる周波数成分とその割合を抽出する処理。

*6 基底周波数：周波数解析における離散的な周波数の単位。

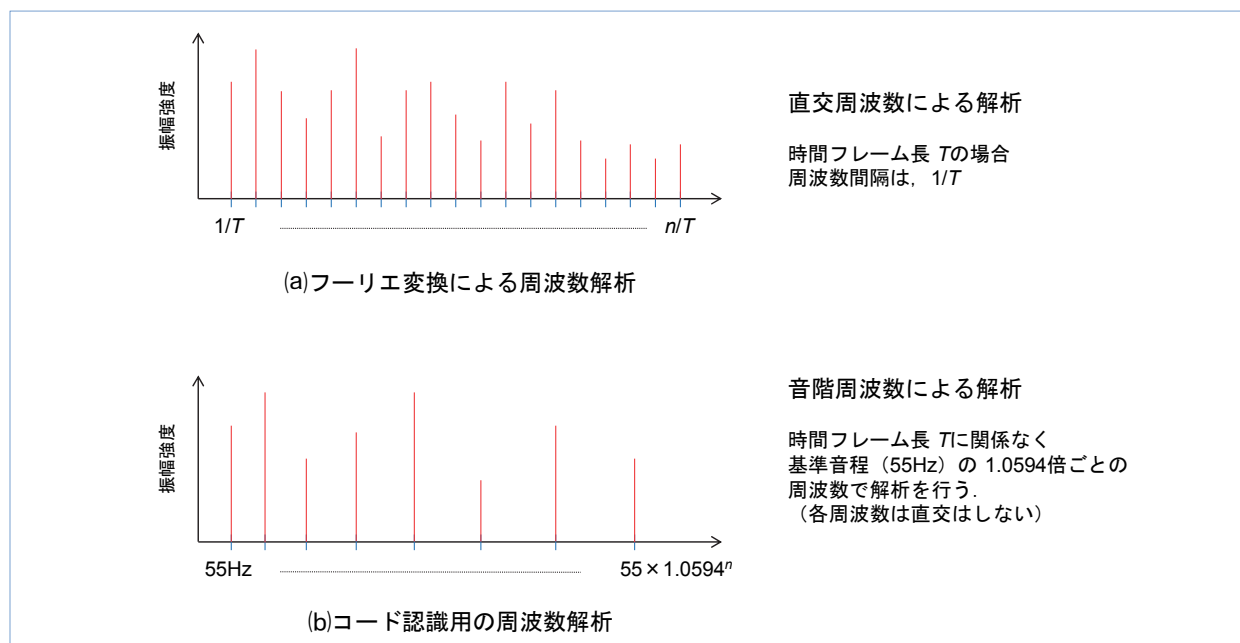


図1 周波数解析方法

エ変換とは異なり、各周波数同士が直交条件を満たしていないため、互いに干渉するというデメリットはあるが、それでも各音階の周波数の成分を解析できるため、音階的な特徴を出しやすい。また、任意の時間フレーム長のデータを解析できる。したがって、速いメロディは短い時間で解析し、遅いメロディは長い時間で解析することも可能になる。

音階は、オクターブ違いを同類とすると12種類のため、オクターブ違いの同じ音階の振幅強度を足し合わせた12次元^{*7}のクロマベクトルを利用して計算時間を短縮することもできる。また、オクターブ違いの差を見るために、クロマベクトルの次元数を増やすこともできる。4オクターブの音階を解析すれば48次元のクロマベクトルとなる。

2.3 クロマベクトルによる音楽のコード認識

このクロマベクトルだけでも音楽のコード認識はある程度可能である。単純な3和音などのコードで、メジャーコードとマイナーコードだけであれば、12

音階ごとに2種類のコードなので、合計24種類のコードが存在する。各コードの構成音は決まっているため、コードの構成音を1、その他を0としたベクトルとクロマベクトルの内積をとり、24種類のコードの中で最も大きい値のコードを出力することで、コード認識器が実現できる。

単一の楽器の演奏音であれば、これでも十分な性能である。しかし、複数の楽器で打楽器や歌もまざった楽曲のコード進行を認識するとなると、認識精度が十分ではなくなってくる。そこで、本技術では、クロマベクトルをDNNに適用し、楽曲のコード進行を精度よく認識できるようにする。

3. 楽曲のコード認識技術

楽曲のコード認識技術とは、CDなどに録音されている楽曲のデータを解析し、コード譜を作成できる技術である。楽曲にもいろいろな種類があるが、本稿では、いわゆるバンド形式（ドラム、ベース、

*7 次元：DNNにおける入力ベクトルの要素数。

ギター、鍵盤、ボーカルなどが混在)で演奏された楽曲を対象とする。本技術では、このような楽曲データのコード解析を精度よく実現するために、DNNを利用して、既存の楽曲データとコード譜を学習し、精度の向上を試みる。

図2に、本技術によるコード認識の手順を示す。

(1)楽曲データ入力 (図2①)

まず、コード認識器に楽曲データを入力する。楽曲データとしては、CDなどに記録されているフォーマットを想定し、ステレオの音源を入力とする。

(2)ビート^{*8}検出 (図2②)

コード譜を作成するためには、ビートの検出により小節の区切りを検出する必要がある。ドラムや楽器音の強弱からそれを行う。これはステレオ音源をそのまま利用し、音源の振幅の強弱の変化から推定する。

(3)ボーカルキャンセル (図2③)

コード認識のためにボーカルや打楽器などを消す

ことを目的として、ステレオ音源のセンターに定位している音源をキャンセルする。これは、単純に左チャンネルの信号に右チャンネルの信号の逆位相を足し合わせれば、センターに定位している音はキャンセルできる。ギターや鍵盤などの音はどちらかの方向に定位している場合が多いので、伴奏だけの演奏に近い音になる。コード認識は、この伴奏音に近くなった音を用いて行う。

(4)ビートごとのコード認識 (図2④)

コード認識では、前述のようにクロマベクトルからコードのパターン認識を行うことになるが、ここでDNNを用いることにする。図3に楽曲の波形データからDNNを用いてコードを認識する手順を示す。

(a)ビートごとの特徴量マップ生成

まず、検出したビートごとに時間領域^{*9}の区間として区切る。各ビートの時刻から次のビートの時刻までの区間をコード認識する。その際クロマベクトルの算出は80ms単位で計算することとする。したがって、1つのビートの時間

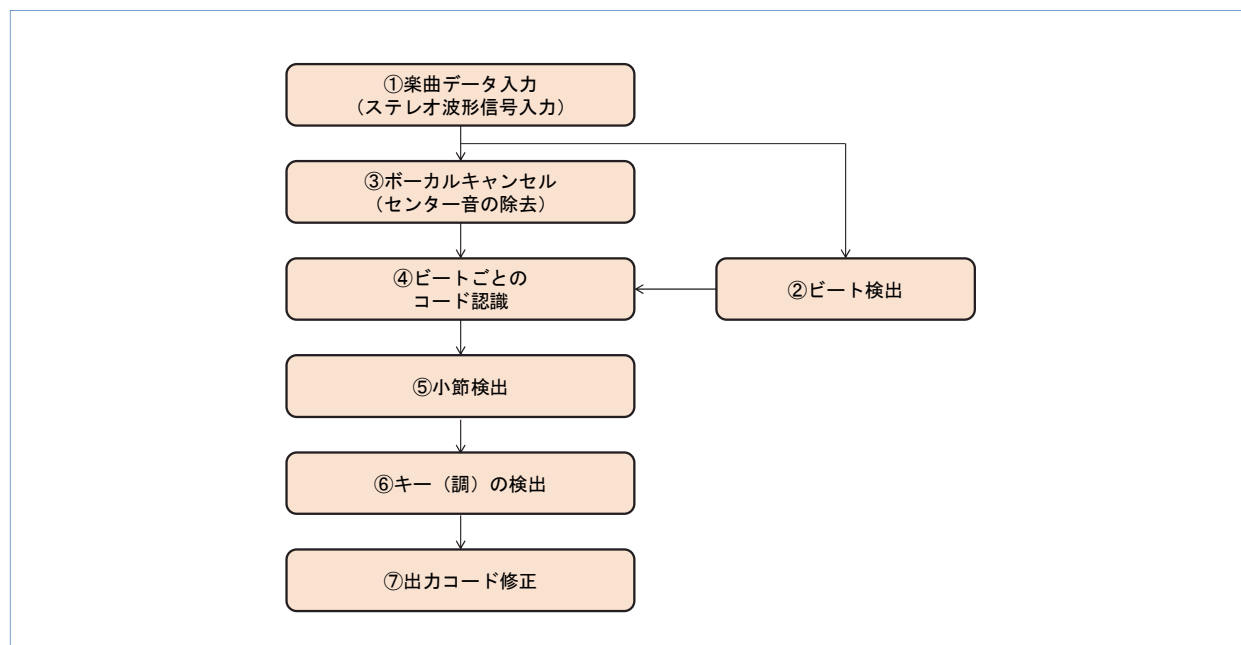


図2 楽曲のコード解析手順

^{*8} ビート：音楽における4分音符の拍。

^{*9} 時間領域：信号などの解析において、その信号が各時間においてどのくらいの成分をもっているかを示すのに用いられる。時間領域の信号をフーリエ変換することで周波数領域の信号に変換することができる。

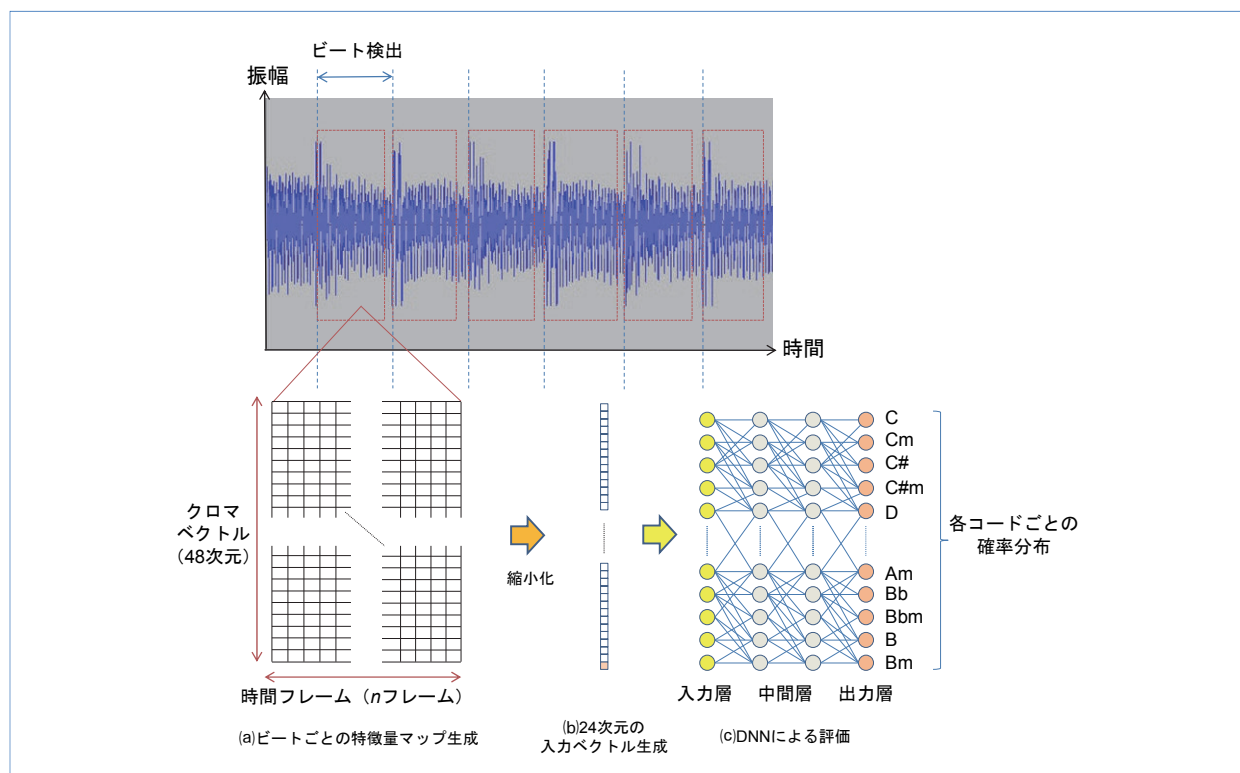


図3 コード認識方法

間隔が500msの場合は、80ms×6フレーム分のクロマベクトルを算出する。また、1フレームのクロマベクトルは4オクターブ分の音程を解析した48次元のベクトルとする。つまり、1つのビート間隔に n フレームの領域があれば、 $48 \times n$ の特徴量マップが生成できる。

(b)24次元の入力ベクトル生成

この特徴量マップをDNNの入力としても良いが、検出されたビートごとの時間区間が異なるため、また特徴量のサイズが固定ではなく、かつ冗長でもあるため、CNN（Convolutional Neural Network）^{*10}のように、特徴量マップを縮小化し、24次元の固定長の入力ベクトルとする。このようにして生成した24次元のベクトルをDNNの入力ベクトルとする。これは、CNNの畳込み層^{*11}やプーリング層^{*12}に相当す

る処理とも考えられる。

(c)DNNによる評価

この入力ベクトルをDNNの入力層の値とし、中間層を2層、出力層はsoftmax関数^{*13}で24種類の確率分布となるようにして、DNNのネットワークを形成する。なお、本技術を4和音のセブンスコード^{*14}やディミニッシュコード^{*15}に対応させることもできるが、本稿では性能向上の効果を見るため、単純な3和音コード24種類で評価を行う。

学習データとしては、CDの音源を用い、44,100Hzサンプリング、16ビット量子化、ステレオのwaveファイル^{*16}と、ビートごとのコードを記述したメタファイル^{*17}を用いて学習を行う。

ビートごとに特徴量マップを生成し、縮小化して

^{*10} CNN：ニューラルネットワークに入力する特徴量に対してフィルタなどの前処理を行い、入力ベクトルを形成するニューラルネットワーク。

^{*11} 畳込み層：入力する特徴量マップに対してフィルタなどをかけて主要な特徴を強調する役割をもつ層。

^{*12} プーリング層：入力する特徴量に対して冗長な特徴量を平均化

ないしは選択し、特徴量を圧縮する役割をもつ層

^{*13} Softmax関数：正規化関数の1つであり、出力値全体の合計が1となるように正規化するため、確率分布として利用できる。

^{*14} セブンスコード：4和音の代表的なコードであり、ルート音、3度、5度の3和音のコードに加えて、7度の音を加えたコード。

24次元の入力ベクトルを生成するところまでは、学習プロセスでも評価プロセスでも同様である。学習の楽曲データとしては、バンド形式で演奏された楽曲30曲を用いてDNNの学習を行った。評価用の楽曲データとしては、学習に用いた楽曲データとは別に用意した楽曲5曲を用いてコードの認識を行った。コード認識には、前述のクロマベクトルのみの方法で認識した場合と、DNNも用いた方法で認識した場合とで正解率を比較した。比較結果を図4に示す。正解率は楽曲データにも依存するが、クロマベクトルのみでコード認識するよりもDNNまで用いて認識する方が、正解率が平均して12%程度向上した。今回は学習データとして30曲を用いたが、より多くの学習データを用いれば正解率のさらなる向上の可能性もある。

(5)小節^{*18}検出 (図2⑤)

ビートごとのコード認識の後には、小節の区切りの検出を行う。これは、コードの切替りが小節の区切りで変わることが多いことを利用する。具体的には、ビートごとにコード認識した出力結果を並べて、一般的な拍子である4分の4拍子を前提とし、コードチェンジの多い拍子（ビート）を小節の区切りとし

て識別する。

(6)キー（調）の検出 (図2⑥)

次に曲の調を判定する。これは最後にコード認識の出力結果を修正する際に必要となる。ハ長調（Cメジャー）のダイアトニックコードを図5の上段に示す。ダイアトニックコードとは、その調でよく使われるコードであり、ハ長調であれば、ドレミファソラシの音階で各音階の音に3度上と5度上の音を追加したコードの集合となる。このダイアトニックコードの出現率が高い場合には、曲の調はCメジャーである可能性が高くなる。このように、Cメジャー、C#メジャー、Dメジャーと各調のダイアトニックコードの出現率を計算し、出現率の高いダイアトニックコードの調をその楽曲の調と判定する。

(7)出力コード修正 (図2⑦)

最後に、認識コードの修正を行う。

DNNを用いてコード認識をしてもコードの正解率は7～8割程度である。誤ったコード認識結果を少しでも修正するために、ダイアトニックコードとセカンダリードミナントコードに基づいた重み付けを行って、最終的なコード判定を行う。

図5下段にニ長調（Dメジャー）のダイアトニッ

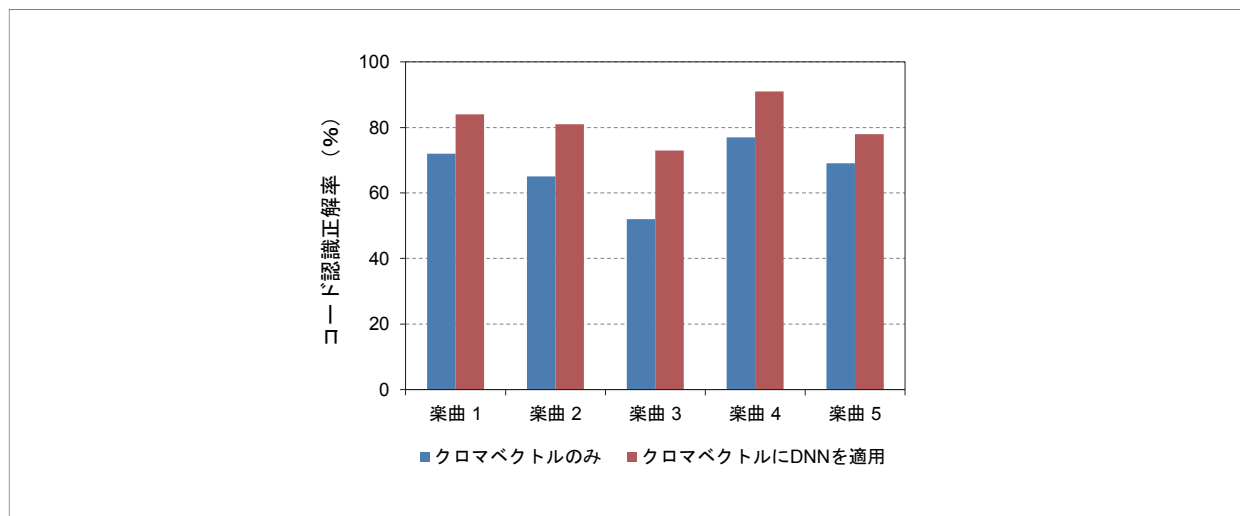


図4 コード認識正解率

*15 ディミニッシュコード：ルート音に対して、短3度と減5度を加えたコード。

*16 waveファイル：音声の波形を保存するファイル形式。サンプリングしたデータをそのまま格納する。

*17 メタファイル：本稿では、音楽コンテンツのコード進行やリズムなどのデータを記載されている情報ファイルを示す。

*18 小節：音楽における時間区間の単位。4分の4拍子であれば、4拍で1小節となる。

【キーがCメジャーの場合】

セカンダリドミナントコード

【キーがDメジャーの場合】

セカンダリドミナントコード

図5 ダイアトニックコードとセカンダリドミナントコード

クコードを示す。二長調（Dメジャー）の場合は、ドとファが＃（シャープ）になるため、ハ長調とは使われるコードが違ってくる。基本的にダイアトニックコードだけで構成されている楽曲が多いが、次に使用頻度が高いコードがセカンダリドミナントコードである。これは、ダイアトニックコードの各コードに対して5度上のセブンスコードになり、図5に青字で記載している。図5の左から1番めのコードに関しては、5度上のコードは通常のダイアトニックセブンスコードに含まれるため、あえてセカンダリドミナントコードとしない。また、7番めのコードに関しては、5度上のセブンスコードのルート音^{*19}（F＃7）が曲調のスケール（C, D, E, F, G, A, B）になく、あまり用いられないため、セカンダリドミナントコードとしない。

このセカンダリドミナントコードにも重みを付けてコード認識結果を修正する。重みの計算は、DNNのコード認識結果によって得られた確率分布に、これらの重みを掛け算して、最終的に最も高い

コードを認識結果とする。これによりコードの正解率はさらに5～10%程度向上する。

このようなプロセスで出力したコード認識結果によって、楽曲のコード譜が作成できる。100%合っているコード譜は難しいが、ある程度参考のできるコード譜の作成は可能である。

4. アドリブ演奏評価機能

音楽におけるアドリブとは、楽譜通りに弾くスタイルではなく、ある伴奏に合わせて自由に弾くスタイルと言える。ユーザーが手軽にアドリブを楽しめる方法の1つとしては、好きな音楽を流しながら、その楽曲に合わせて自由にメロディを弾いてみることである。ここでは、前述のコード認識技術を応用して、ユーザーが好きな楽曲を流しながらアドリブ演奏をした場合の演奏評価機能について解説する。

まず、ユーザーが好きな楽曲を流す際に、その楽曲のコード認識を行う。それに伴い、小節の区切りや

^{*19} ルート音：和音を構成する音の中で、基準となる音階の音。最も低い音である場合が多い。

曲の調も識別する。演奏評価システムは、楽曲を流しながら、現在再生している小節とそのコードを把握しておく。そして、ユーザの演奏音をクロマベクトル化し、演奏音の音程を検出する。演奏音の音程が、小節のコードの構成音になっていれば、得点を高くする。また、コードの構成音でなくても、曲の調のメジャースケールの音であれば、少しだけ得点を高くする。どちらの音でもない音を1ビート以上伸ばしてしまったら、得点を減点する。基本的にはこのような採点方法でアドリブの演奏を得点化して評価を行う。

本技術の応用として実装したアドリブ演奏評価システムでは、楽曲を流しながらユーザのアドリブ演奏を解析し、小節ごとに得点が高ければ音声で褒める、得点が低ければ音声で注意する、という機能を実装した。これによりユーザは、アドリブ演奏しながらリアルタイムに評価されるという緊張感を得ることができる。

5. あとがき

本稿では、楽曲のコード認識技術として、クロマベクトルとDNNを用いた手法を解説した。また、応用例として、アドリブの演奏評価システムについても紹介した。今後はより音楽的な解析機能を追求して、音楽的なエージェントを実現したいと考えている。

文 献

- [1] M. Yamamoto, S. Naga and J. Shimizu: "Positive musical effects on two types of negative stressful conditions," *Psychol Music*, Vol.35, Issue 2, pp.249-275, Apr. 2007.
- [2] H. Nakayama, F. Kikuta and H. Takeda: "A Pilot Study on Effectiveness of Music Therapy in Hospice in Japan," *J Music Ther*, NC46.2, pp.160-172, Jul. 2009
- [3] C. Harte and M. Sandler: "Automatic Chord Identification Using a Quantized Chromagram," in *Proc. of Audio Eng. Soc.* May 2005.