

Chord-Cube: 和音距離計量空間による 楽曲可視化・ナビゲーションシステム

今井 樹 倉林 修一

慶應義塾大学環境情報学部 〒252-0882 神奈川県藤沢市遠藤 5322

E-mail: {t10109ti,kurabaya}@sfc.keio.ac.jp

あらまし 本稿では、音楽データベースを対象として、楽曲におけるコード進行を分析し、楽曲のセクション別に計量された類似度を3次元空間上に可視化する事により、楽曲間の動的かつ直感的なナビゲーションの実現方式を示す。本研究の特徴は、コードを構成音に分解し、音楽理論における五度圏表を用いて構成した24次元の構成音空間に写像する事により、コード進行におけるコード間の意味的な移動距離を計量する点にある。このコード進行の距離モデルにより、楽曲内のコード進行がもたらす印象の変化量を計量し、直感的な楽曲間の類似性計量を実現している。本稿では、プロトタイプシステムとして、ユーザーが選択した楽曲を基準に3つの評価軸で構成された3次元空間内に他の楽曲をプロットし、直感的な楽曲間ナビゲーションを行うシステムの実装方法について示す。本システムでは、膨大な数のデータベース内の楽曲をセクション別に分析・類似度算出し、任意の基準曲とデータベース内の距離を、3次元空間の座標として設定することにより、ユーザーは任意の楽曲を基準とした楽曲間の比較が可能となる。

キーワード 音楽, 推薦, 可視化

Chord-Cube: Music Visualization & Navigation System with a Chord Metric Space

Tatsuki IMAI and Shuichi KURABAYASHI

Faculty of Environment and Information Studies, Keio University, 5322 Endo, Fujisawa, Kanagawa, 252-0882, Japan

E-mail: {t09334yk,kurabaya}@sfc.keio.ac.jp

Abstract This paper presents new music visual-navigation system that provides a customizable 3-dimensional visualization engine, code-cube, for rendering semantic distance between music. Unique feature of this 3-dimensional visualization is that it is a content-based music analysis to shows users to recognize multiple relationships between music. For example, this visualization mechanism uses introductory-melody, continued-melody, and salient-melody as axis in order to render the content-based relevance with a timeline structure. On this visualized images, this system provides navigation of songs in this 3-dimensional space. This system input music as an origin point from users to create a 3D cube that locates many music according to the similarity to the origin music. As an application of this system, this visualization system enable to create new 3D space by multiple axis based on artists that selected by user for showing user to recognize comparison between a song and multiple artists.

Keyword Music, Recommendation, Visualization

1. はじめに

今日、あらゆる情報の電子化が加速的に進む中、音楽データもインターネット上での売買が主流になりつつある。日本レコード協会の国内の調査によれば、インターネット上での音楽データの売り上げは音楽売上全体の37.4%を占めているほか、インターネット上で購入可能な楽曲数は2007年の時点で200万曲以上にまで上っている。このような状況ではユーザーが全く新しい楽曲を探し出すのは難しい。また、電子化された楽曲情報のネットワークを通じた流通により、一般ユーザーが保有する楽曲数は膨大なものとなっており、ユーザーが必要な楽曲を直ちに見つけ出すことが困難な

状況となっている[1]。特に音楽のような時間軸に沿って変化するメディアデータ特有の問題として、ユーザーが楽曲の内容を把握するためにはその楽曲を視聴する必要があるため、ジャンル名、アーティスト名といったメタデータからユーザーは楽曲の内容を理解することはできない。

ユーザーが楽曲を試聴する際に感じる感性（センチメント）を反映した楽曲検索が必要とされている。ユーザーの感性に応じて嗜好が決定する音楽のようなメディアデータを対象とした効果的な検索・推薦システムには実際の音楽データを分析した情報抽出が重要である。楽曲の特徴抽出のための音楽分析を行う研究と

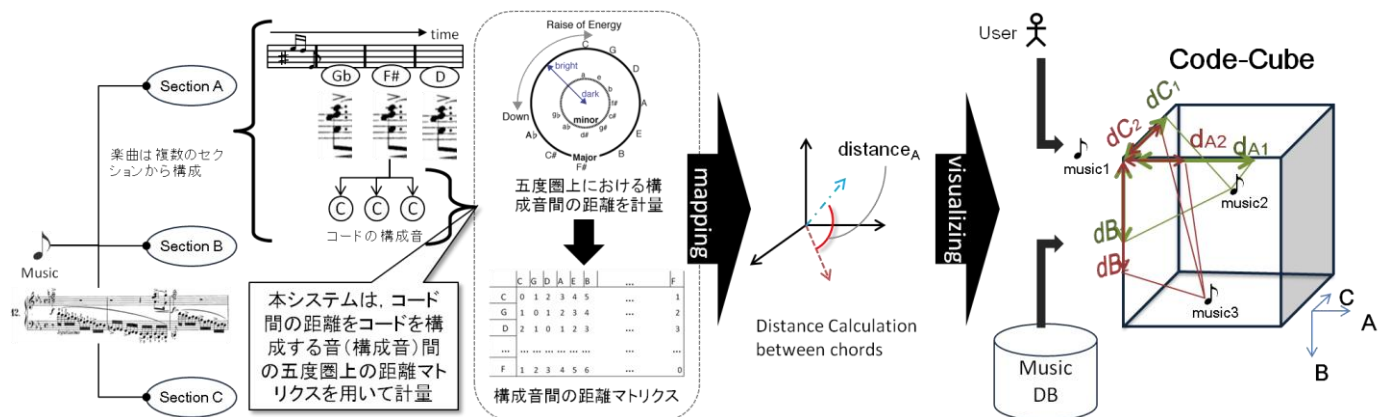


図 1:和音距離計量空間による楽曲可視化・ナビゲーションシステム Chord-Cube の概要

しては、周波数を分析する方式[2], 入力されたパワースペクトルを分析する方式[3], ユーザーが入力したハミングを分析する方式[4][5][6]などがある。[4][5]はデータベース内の楽曲を MIDI ファイルによって分析する方式である。従来の視聴支援システム等では、抽出した特徴からそれぞれの目的に応じてある楽曲についての分析結果を提示するが、楽曲間の比較を行うことは難しかった。ユーザーが指定した任意の楽曲について他の楽曲との比較を行うことができれば、ユーザーの新しい楽曲の発見性は飛躍的に高まると考えられる。

そこで、本稿では、楽曲の背景的な音として、全体の雰囲気や流れを決定する重要な要素であるコード進行に着目し、コード進行の変化に応じて楽曲間の類似性を計量・可視化することにより楽曲間の動的なナビゲーションを行う音楽検索システムを提案する。同様にコード進行を用いて楽曲の類似度を算出する研究に web 上のデータからコード進行のパターンを抽出する研究がある[7]。本方式の特徴は、コード進行における、それぞれのコード間の意味的な距離を定義した音楽理論上のモデルである五度圏を用い、コード間の距離を計量するコード・ベクトル空間 (Chord Vector Space) を構築する点にある。このコード・ベクトル空間は、楽曲を構成するコード進行を特徴量とした 24 次元の空間であり、楽曲内に登場する任意のコード進行と、他のコード進行の意味的な距離を計量する事が出来る。さらに、JPOP に固有の構造である、A メロ、B メロ、サビなどのセクション構造ごとにコード進行を分析し、分析した結果を、3D を用いて可視化する事により、直感的な音楽ナビゲーションを実現する。(図 1)

本システムは、3 つ以上の音が重なることで構成されるコード進行をその構成音に分解し、その構成音の一楽曲中での出現回数と五度圏上での距離との相関を計量し、求められた 24 次元の値によってコード・ベクトル空間を構築する。このコード・ベクトル空間は、任意のコード進行間の類似性を、コード進行の意味的な距離に応じて計量する事が出来る。コード進行は出

現位置によってそれが視聴者に与える印象が異なるため、音楽の感性分析においてコード進行をセクション毎に比較することは重要である。

本システムは、コード・ベクトル空間により算出された類似度を楽曲間の距離として表す。各セクションを軸として構成される立体において原点に任意の楽曲を設定し、各楽曲の距離に応じて、立体中に楽曲をプロットする事により、基準の楽曲と比較対象楽曲の類似度を、空間の原点と立体中の点という関係として表すことで可視化する。本システムによってユーザーは任意の比較対象を用いて、直感的に新しい楽曲の検索を行う事が出来るようになる。

2. 楽曲の類似度をセクション別に算出する楽曲推薦・ナビゲーションシステム

本節では、セクション毎の類似度に応じて楽曲間の距離を可視化し、動的なナビゲーションを可能にする検索システム code-cube のシステム概要、データ構造、及び、基本機能について説明する。

2.1. システム概要

本システムは、コード・ベクトル空間によって楽曲のコード進行に応じた比較演算を行う。本システムはコードの構成音の頻度と距離から楽曲内におけるコードの意味合いの強さを算出することで楽曲の特徴を算出する。セクション毎に与えられた楽曲のコード進行情報からコードの構成音を抽出し、それぞれの出現回数を算出する。構成音の出現回数と五度圏上に定義されたコード間の距離の積を求め、総和を得ることで、24 次元の構成音空間における特徴量ベクトルを求める事が出来る。このベクトルをセクション毎に算出し、セクション別の比較を実現する。

2.2. データ構造

本節では、本システムのデータ構造を述べる。

2.2.1. コード進行

本システムの分析対象データとして、楽曲データであるコード進行の基本構造を示す。コード進行は3つ以上の構成音からなる楽曲の印象を決定づける音の要素である。ほとんどの西洋音楽の基礎となっている和声という西洋音楽の概念では、コード進行の配置と組み合わせを基に曲を構成する。そのため、コード進行を分析することが、楽曲分析において重要である。

本システムではコード進行によるセクションの類似度計量が可能である。コード進行の構成音がその楽曲で持つ意味の大きさを計量するために、その頻度以下の手順で得る。与えられたコードからコード構成音テーブル(図2)によって構成音を得る。このテーブルは行を12種類の構成音、列を約50種類のコードとして定義されるテーブルで、ある構成音が、そのコードを構成している場合要素には1が、そうでない場合は0が格納される。例えば、CというコードはC,E,Gという3つの音から構成され、それらC,E,Gの要素に1がそれ以外のセルには0が格納される。本システムは与えられたコードの構成音に分解し、その登場回数を構成音頻度テーブルに格納する。

	C	C#	D	D#	E	F	F#	G	G#	A	A#	B
C	1	0	0	0	1	0	0	1	0	0	0	0
C6	1	0	0	0	1	0	0	1	0	0	0	0
C#7	0	1	0	0	0	1	0	0	1	0	0	0
...
Bm	0	0	1	0	0	0	1	0	0	1	0	0

図2 コード進行構成音テーブル

2.2.2. 構成音間距離マトリクス

本システムは、音楽理論における五度圏表に基づいた構成音同士の距離を用いて構成した24次元の構成音空間に写像することにより、コード進行におけるコード間の意味的な移動距離を計量する。調性の五度圏(Circle of Fifth: 図3)は、全24種類の調性間の近さや類似性といった関係を定義したモデルである[8]。本システムでは、この五度圏を調性の変化によって引き起こされる印象の変化を計量するメトリクスとして利用する。

本システムでは構成音間の距離を五度圏表上の距離を基に格納した構成音間距離マトリクスを定義する。(図4)マトリクスの行・列の両方に1オクターブの12個の音を設定し定義される。五度圏上で同位置にある音同士の距離を0、隣の位置の音との距離を1と定義し、ひとつ離れるごとに距離が1増える。五度圏表は12音で一周するため、最も離れた音までの距離は6となる。求められた距離を6で割ることによって正規化した値を各要素に格納する。このマトリクスを元に距離の計算を行う。また、五度圏は環状に配置され

た調性の移動の種類によって印象が変化するという特徴を持つ。例えば、時計周りに調整が移動した場合はエネルギーが上昇し、反時計まわりではエネルギーが減少する他、短調(minor)への移動は暗い印象となり、長調(Major)への移動は明るい印象になる。これらの印象変化をそれぞれ距離に反映した比較演算を行うために全16種類のマトリクスが定義する。(図4は印象変化を反映しない同調同士の距離マトリクス。上が長調間、下が短調間)

	C	C#	D	D#	E	F	F#	G	G#	A	A#	B
C	0	0.83	0.33	0.50	0.67	0.17	1	0.17	0.67	0.50	0.33	0.83
C#	0.83	0	0.83	0.33	0.50	0.67	0.17	1	0.17	0.67	0.50	0.33
D	0.33	0.83	0	0.83	0.33	0.50	0.67	0.17	1	0.17	0.67	0.50
D#	0.50	0.33	0.83	0	0.83	0.33	0.50	0.67	0.17	1	0.17	0.67
E	0.67	0.50	0.33	0.83	0	0.83	0.33	0.50	0.67	0.17	1	0.17
F	0.17	0.67	0.50	0.33	0.83	0	0.83	0.33	0.50	0.67	0.17	1
F#	1	0.17	0.67	0.50	0.33	0.83	0	0.83	0.33	0.50	0.67	0.17
G	0.17	1	0.17	0.67	0.50	0.33	0.83	0	0.83	0.33	0.50	0.67
G#	0.67	0.17	1	0.17	0.67	0.50	0.33	0.83	0	0.83	0.33	0.50
A	0.50	0.67	0.17	1	0.17	0.67	0.50	0.33	0.83	0	0.83	0.33
A#	0.33	0.50	0.67	0.17	1	0.17	0.67	0.50	0.33	0.83	0	0.83
B	0.83	0.33	0.50	0.67	0.17	1	0.17	0.67	0.50	0.33	0.83	0

図4 構成音距離マトリクス(長調-長調)

	a	e	b	f#	c#	g#	e#	b#	f	c	g	d
a	0	0.17	0.33	0.50	0.67	0.87	1	0.87	0.67	0.50	0.33	0.17
e	0.17	0	0.17	0.33	0.50	0.67	0.87	1	0.87	0.67	0.50	0.33
b	0.33	0.17	0	0.17	0.33	0.50	0.67	0.87	1	0.87	0.67	0.50
f#	0.50	0.33	0.17	0	0.17	0.33	0.50	0.67	0.87	1	0.87	0.67
c#	0.67	0.50	0.33	0.17	0	0.17	0.33	0.50	0.67	0.87	1	0.87
g#	0.87	0.67	0.50	0.33	0.17	0	0.17	0.33	0.50	0.67	0.87	1
e#	1	0.87	0.67	0.50	0.33	0.17	0	0.17	0.33	0.50	0.67	0.87
b#	0.87	1	0.87	0.67	0.50	0.33	0.17	0	0.17	0.33	0.50	0.67
f	0.67	0.87	1	0.87	0.67	0.50	0.33	0.17	0	0.17	0.33	0.50
c	0.50	0.67	0.87	1	0.87	0.67	0.50	0.33	0.17	0	0.17	0.34
g	0.17	0	0.17	0.33	0.50	0.67	0.87	0.67	0.33	0.17	0	0.17
d	0.17	0.33	0.50	0.67	0.87	1	0.87	0.67	0.50	0.33	0.17	0

図4 構成音距離マトリクス(短調-短調)

異なる調同士の移動では、暗い印象を与える移動の距離を大きくし、明るい印象を与える移動は小さく値をとることで、その印象を反映した計算を行う。それを反映したマトリクスが印象変化距離マトリクスである。印象変化を距離に反映するマトリクスの要素は、

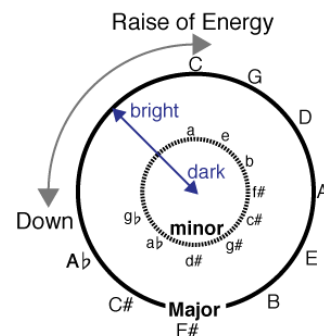


図3 五度圏表

同調同士の移動と同様に距離を格納したマトリクスの要素から、印象を強く反映させる方向の移動の距離を0.5減らし、印象を弱く反映させる方向の移動の距離を0.5足した上で正規化を行う。距離を小さく定義することで類似度が高く算出され、任意の印象について推薦されやすくすることができる。図5に明るい印象の変化の類似度を高める際のマトリクスを示す。(図5上が長調から短調、下が短調から長調への移動の距離マトリクスである。)これらの距離マトリクスを用いることで精密な距離の演算が可能であると考えられる。

	a	e	b	f#	c#	g#	e ^b	b ^b	f	c	g	d
C	0.083	0.25	0.416	0.583	0.75	0.916	1.083	0.916	0.75	0.583	0.416	0.25
D	0.25	0.083	0.25	0.416	0.583	0.75	0.916	1.083	0.916	0.75	0.583	0.416
E	0.416	0.25	0.083	0.25	0.416	0.583	0.75	0.916	1.083	0.916	0.75	0.583
F	0.583	0.416	0.25	0.083	0.25	0.416	0.583	0.75	0.916	1.083	0.916	0.75
G	0.75	0.583	0.416	0.25	0.083	0.25	0.416	0.583	0.75	0.916	1.083	0.916
A	0.916	0.75	0.583	0.416	0.25	0.083	0.25	0.416	0.583	0.75	0.916	1.083
B	1.083	0.916	0.75	0.583	0.416	0.25	0.083	0.25	0.416	0.583	0.75	0.916
C ^b	0.916	1.083	0.916	0.75	0.583	0.416	0.25	0.083	0.25	0.416	0.583	0.75
D ^b	0.75	0.916	1.083	0.916	0.75	0.583	0.416	0.25	0.083	0.25	0.416	0.583
E ^b	0.583	0.75	0.916	1.083	0.916	0.75	0.583	0.416	0.25	0.083	0.25	0.416
F ^b	0.416	0.583	0.75	0.916	1.083	0.916	0.75	0.583	0.416	0.25	0.083	0.25
G ^b	0.25	0.416	0.583	0.75	0.916	1.083	0.916	0.75	0.583	0.416	0.25	0.083

図5:印象変化距離マトリクス(短調)

	C	G	D	A	E	B	G ^b	D ^b	A ^b	E ^b	F ^b
a	0	0.083	0.25	0.416	0.583	0.75	0.916	0.75	0.583	0.416	0.25
e	0.083	0	0.083	0.25	0.416	0.583	0.75	0.916	0.75	0.583	0.416
b	0.25	0.083	0	0.083	0.25	0.416	0.583	0.75	0.916	0.75	0.583
f#	0.416	0.25	0.083	0	0.083	0.25	0.416	0.583	0.75	0.916	0.75
c#	0.583	0.416	0.25	0.083	0	0.083	0.25	0.416	0.583	0.75	0.916
g#	0.75	0.583	0.416	0.25	0.083	0	0.083	0.25	0.416	0.583	0.75
e ^b	0.916	0.75	0.583	0.416	0.25	0.083	0	0.083	0.25	0.416	0.583
b ^b	0.75	0.916	0.75	0.583	0.416	0.25	0.083	0	0.083	0.25	0.416
f	0.583	0.75	0.916	0.75	0.583	0.416	0.25	0.083	0	0.083	0.25
c	0.416	0.583	0.75	0.916	0.75	0.583	0.416	0.25	0.083	0	0.083
g	0.25	0.416	0.583	0.75	0.916	0.75	0.583	0.416	0.25	0.083	0
d	0.083	0.25	0.416	0.583	0.75	0.916	0.75	0.583	0.416	0.25	0.083

図5 異調間距離マトリクス(長調)

2.2.3. コード・ベクトル

本システムは、楽曲の特徴をコードの構成音と構成音の五度圏上における距離の関係から算出された行列で表す。本稿ではこの行列をコード・ベクトル(CV)と呼ぶ。構成音は鍵盤上の12音であるので、CVは12行×12列の行列である。CVを算出する関数を次のように定義する。

$$f_{CV}(d, e) := \left(\sum_{i=1}^{12} d_{[i,1]} \cdot e_{[i]}, \sum_{i=1}^{12} d_{[i,2]} \cdot e_{[i]}, \dots, \sum_{i=1}^{12} d_{[i,12]} \cdot e_{[i]} \right)$$

ここで、dは構成音の距離、eは構成音の頻度の値である。構成音間距離マトリクスと各構成音の登場回数

の積から一つの構成音に対する他の構成音の重みを求める。全ての構成音に関してそれぞれの構成音との相関量が格納されたベクトルを生成し、各セクションのコード・ベクトル(CV)と定義する。

2.3. コード・ベクトル抽出機能

本節では、楽曲の物理的特徴量をコード進行に応じて比較を実現するコード・ベクトルの生成方法について実例を挙げて説明する。

本システムは、次の手順を実行する。

- Step-1: [コード分解]

本システムは、セクション別に与えられたコード進行をその構成音に分解する。与えられたコードをコード進行構成音マトリクスと照合し、該当する構成音の登場回数をそれぞれ要素として配列に格納する。これを構成音頻度テーブルと定義する。本節では嵐の「サクラ咲ケ」のAメロの分析結果を例に説明する(図6)。

C	C#	D	D#	E	F	F#	G	G#	A	A#	B
14	0	11	0	11	5	1	18	0	7	1	10

図6:「サクラ咲ケ」Aメロの構成音頻度テーブル

- Step-2: [ベクトル抽出]

Step1で得た構成音頻度テーブルと構成音距離マトリクスの各行毎に掛け、積を求める。次に列毎に値の総和を求めて得た配列をコード・ベクトルと定義する(図7)。このサンプルのコード進行は長調間の移動のみなので、要素は12個となっている。

$$CV = (23.86, 15.33, 15.21, 21.48, 30.23, 39.39, 51.34, 58.82, 59.29, 52.9, 43.99, 35.22)$$

図7:「サクラ咲ケ」Aメロのコード・ベクトル

- Step-3: [距離計算]

本システムは、基準となる楽曲(CV₁)と比較対象の楽曲(CV₂)を対象として、Step2で生成されたセクション別のコード・ベクトル間の内積値を算出する関数を提供する。この関数を次のように定義する。

$$f_{distance}(CV_1, CV_2) := \sum_{i=1}^{12} CV_{1[i]} \cdot CV_{2[i]}$$

2.4. 可視化機能

本説ではコード・ベクトル抽出機能で得られた距離を用いて可視化を行う方法を説明する。

本システムはユーザーが指定した楽曲とデータベース内の楽曲を比較する。ユーザーが指定した楽曲music1の各セクションをS1a, S1b, S1c, データベース

内のある楽曲 music2 の各セクションを s2a, s2b, s2c と定義する. コード・ベクトル抽出機能により, S1a と S2a の距離 d1, S1b と S2b の距離 d2, S1c と S2c の距離 d3 が求められる. 楽曲中の 3 つの各セクションを座標軸として構築される 3 次元立体空間のなかで, Music1 を原点, 座標(0,0,0)と設定し, 上記の方法で求められた距離を music2 の座標(d1,d2,d3)とすることによって本システムはこの立体中に楽曲間の距離を可視化する. この立体を code-cube と呼ぶ.

2.5. ナビゲーション機能

本システムは楽曲から楽曲へと辿って楽曲検索を行うナビゲーション機能を有する. このナビゲーション機能は立体中への可視化によって実現する. 本システムは立体中にプロットされた楽曲群の中からある 1 曲をユーザーが新たに基準の楽曲として設定することで, 新しく code-cube を構築する機能を持つ. ナビゲーション機能により, 今までの楽曲推薦システムではできなかった新しい推薦が可能となる.

2.6. 個人化機能

本システムは, 2.2.2 で述べた 16 種類の構成音距離マトリクスの設定に応じて高い個人化機能を備えることができる. 印象変化とエネルギー変化をそれぞれ反映した距離計算, 双方を反映した距離計算という様に複数のマトリクスを組み合わせることで, それぞれ特有の類似度の算出が可能であると考えられる. また, 類似度を高めるための距離設定を自由に行えるのでこれをユーザーに応じてその尺度変更することによって高い個人化機能を備える事が出来ると考えられる.

3. プロトタイプシステム

本節は, プロトタイプシステムの実装について述べる. 実装には JavaScript 言語を用いた. 可視化における立体の描画は JavaScript で 3D 描画を可能にする Three.js ライブラリを用いた. 本システムは音楽メディアプレーヤーと併用することを想定しており, ユーザーがメディアプレーヤーで再生中の曲をシステムに与えることで, データベース内の楽曲, 選択中の楽曲を分析する. ユーザーが指定した楽曲を原点とし, 他の曲との距離を立体中に可視化されたものをユーザーに表示する. また, ユーザーは画面中のスライダーを用いてこの立体を任意の尺度で表示することができる. 立体中には無数に楽曲が表示されるが, 拡大することによって 1 曲単位での類似の表示が可能である. (図 8)

関数はコード・ベクトル算出機能を持つ関数とコード・ベクトル比較機能を持つ関数の 2 種類を有する.

楽曲情報の入力に対して, コード・ベクトル算出関数は, あらかじめ定義されたコード進行データベースかたコード情報を引き出し, 2.3 で述べた手順に従ってコード・ベクトルを算出する. それと同時にその他の楽曲のコード・ベクトルも算出し, その値を比較関数へと渡す.

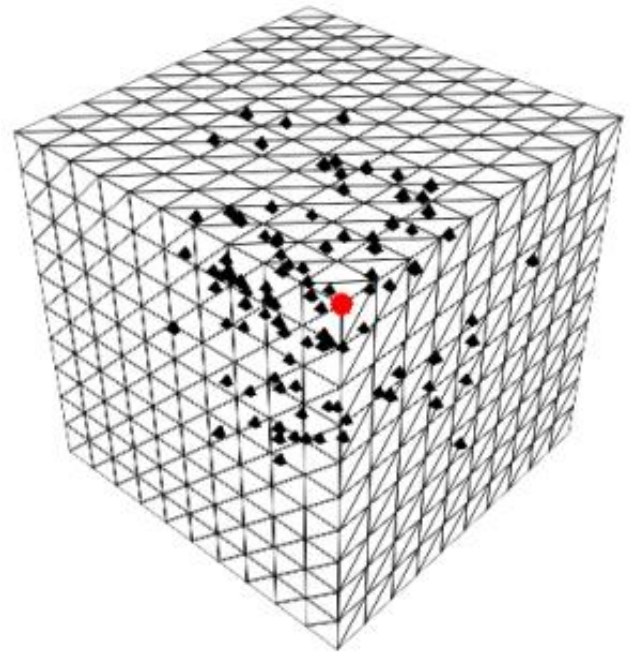


図 8 : Chord cube プロトタイプの実装

立体の頂点に位置する赤い球が基準楽曲を意味し, 立体中に分布した黒い点はコード・ベクトルによる比較を元にプロットされた楽曲群である.

立体的に距離を表現し, 距離を可視化する.

4. 評価実験

本システムにおいて, 与えられたコード進行から楽曲の類似度比較を実現するコード・ベクトル空間を用いた距離計量の有効性の検証を行った実験結果を示す.

4.1. 実験概要

質問一つにつき, 基準の 1 楽曲とそれに対する 5 曲の楽曲群を与える. 被験者は基準の楽曲に似ていると感じた順に 5 曲のランキングをつける. 得られた回答について, 1 位の楽曲に 5 ポイント, 2 位の楽曲に 4 ポイント, という様にポイントを加算し, 被験者全体の回答を合計し, 最終的なランキングを決定する. この実験結果と, 本システムの類似度抽出によるランキングを比較し, コード・ベクトルによる比較の有効性を示す. なお, 今回の実験に用いる構成音間距離マトリクスは, 印象変化, エネルギー変化の要素を反映させないものとする.

質問 1 基準楽曲：	絢香「melody」
比較楽曲群：	浜崎あゆみ「identity」 Superfly「Alright!」 レミオロメン「雨上がり」 コブクロ「神風」 Dreams Come True「連れてって 連れてって」

質問 2 基準楽曲：	浜崎あゆみ「GREEN」
比較楽曲群：	絢香「I believe」 嵐「明日の記憶」 Superfly「愛に抱かれて」 倭田來未「愛のことば」 B'z「PRAY」

神風	26.23
雨上がり	25.12
Alright!	27.67
連れてって 連れてって	26.65
Identity	28.89

・ B メロ

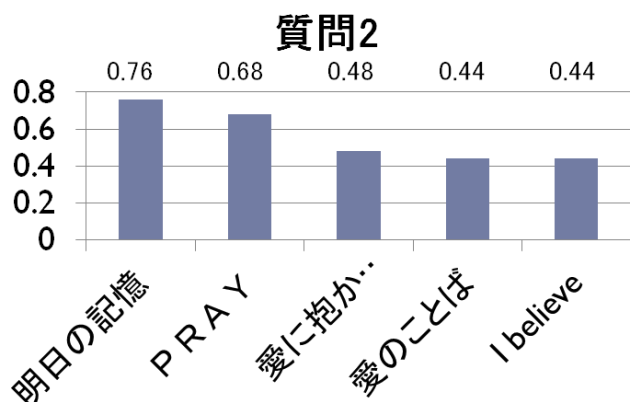
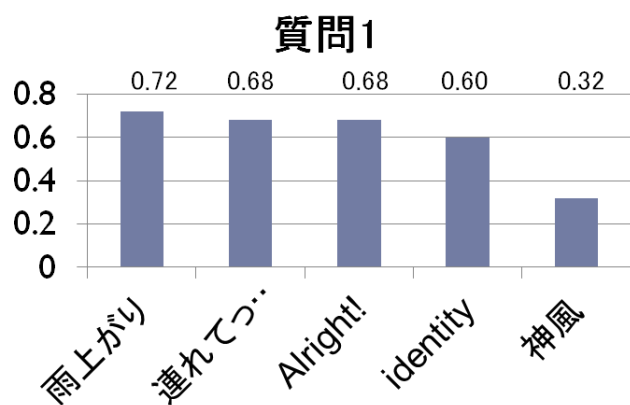
神風	27.54
雨上がり	27.12
Alright!	16.53
連れてって 連れてって	28.29
Identity	28.11

・ サビ

神風	15.97
雨上がり	25.56
Alright!	26.46
連れてって 連れてって	26.11
Identity	27.37

4.2. 実験結果

5人の被験者から回答が得られ、その合計は以下に示すものになった。(数値はポイントを満点の25で割った値を表す.)



本システムのコード・ベクトルによる比較によって算出された相関量はセクション別に以下の通りである.

(数値は本システムで計量された相関量である.)

質問 1.

・ A メロ

質問 2.

・ A メロ

I believe	26.23
明日の記憶	25.12
愛に抱かれて	27.67
愛のことば	26.65
PRAY	28.89

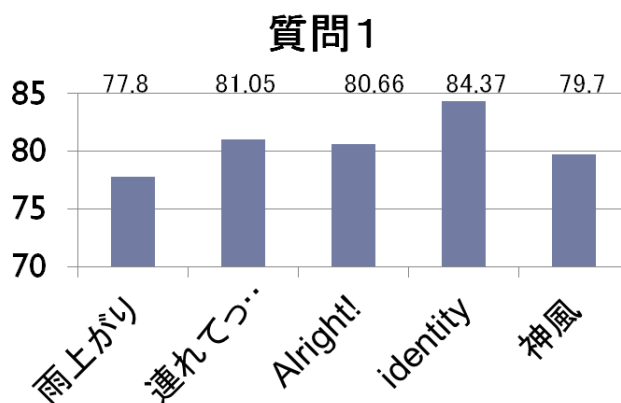
・ B メロ

I believe	27.54
明日の記憶	27.12
愛に抱かれて	16.53
愛のことば	28.29
PRAY	28.11

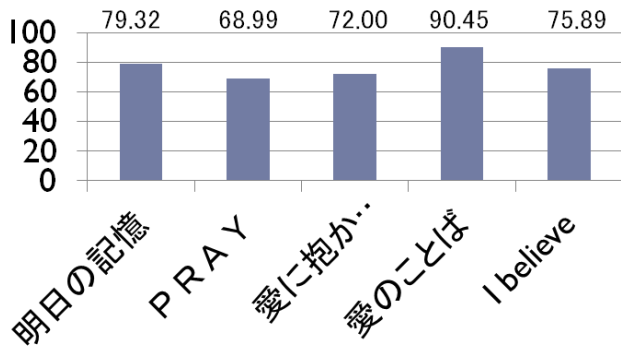
・ サビ

I believe	15.97
明日の記憶	25.56
愛に抱かれて	26.46
愛のことば	26.11
PRAY	27.37

これを合計すると、本システムによって導き出されるランキングは以下の通りになる.



質問2



4.3. 考察

質問1の基準楽曲は明るくポップな印象をもつテンポの軽い楽曲であり、質問2は重く、壮大な雰囲気のあるバラードである。

質問1について、本システムの方式によって算出されたランキング上位3曲中2曲は回答と同じ楽曲が得られた一方、回答で1位だった楽曲の順位が低くなった。質問2では、同様に回答で上位になった楽曲の類似度が低く算出された他、回答で順位の低かった楽曲の類似度が高く算出された。

質問1の場合、類似が高く算出された楽曲「identity」は電子音とエレキギターの重厚な音を組み合わせが多く使われているテンポの早い楽曲である。リズムを刻むピッキングという奏法で軽やかな印象を与える基準楽曲とテンポは近いが聞いた印象としては遠い印象を受ける。しかし、コードの構造が近いものであった場合、類似度は高く算出される。そのため、印象は遠いが、類似度が高く算出されることでこの落差が生じたと考えられる。類似度が低く算出された楽曲「雨上がり」はその逆で、こちらは、ギターなど楽器の音を生かした早めのテンポの楽曲であり、基準楽曲に近い印象を受けるが、コードの構成は違いがみられるものであった事が、算出結果と回答の差異の原因として考えられる。

質問2で類似度が低く算出された「PRAY」、類似度が高く算出された「愛のことば」、「I believe」についても同様のことが考えられる。基準楽曲と似た荘厳な印象を与える「PRAY」のコード自体の距離は遠いものであり、「愛のことば」、「I believe」は共に基準楽曲とは異なり音の重厚感は感じられない楽曲であるが、コードの構成は似ている事が原因と考えられる。

これらのギャップは構成音間距離マトリクスに印象変化やエネルギー変化の要素を加え、構成音間距離マトリクスの設定をより詳細に行うことで、より精密な比較が可能になると考えられる。

5. まとめ

本稿では、音楽を対象として、楽曲におけるコード進行による分析・比較を可能にする楽曲ナビゲーションシステムを提案した。本システムは和音による距離計量空間であるコード・ベクトル空間を用いて楽曲の距離計量を楽曲のセクション別に行うことでコード・ベクトルを抽出し、コード・ベクトルを比較することによって和音による距離計量を実現した。

本システムの実現方式として、五度圏に準拠した距離マトリクスを定義することにより、利用者が楽曲に抱く印象を反映した類似度算出を実現するシステムの構成を示した。

本方式は楽曲内のセクションだけでなく、アーティストやアルバムなど、あらゆる楽曲要素を比較材料として設定することで、様々な尺度による比較で利用者が幅広い検索を行う事が可能となる。

参考文献

- [1] 後藤真孝, 平田圭三 音楽情報処理の最近の研究 日本音響学会誌 60 巻 11 号(2004), pp.675-681
- [2] E.Gómez, J.Bonada.: Tonality visualization of polyphonic audio, Proceedings of International Computer Music Conference 2005.
- [3] 後藤真孝 SmartMusicKIOSK:サビ出し機能付き音楽試聴機 情報処理学会論文誌 vol.44 No.11 (2003)
- [4] A.Ghias, J.Logan, D.Chamberlin and B.C. Smith: Query by humming: Musical information retrieval in an audio database. Proc.ACM Multimedia 95, 231-236, 1995.70
- [5] T.Sonoda, T.Ikenaga, K.Shimizu and Y.Muraoka: The design method of a melody retrieval system on parallelized computers. Proc. WEDELMUSIC 2002, 66-73, 2002.
- [6] R.B.Dannenberg, W.P.Birmingham, G.Tzanetakis, C.Meek, N.Huand B.Pardo: The MUSART testbed for query-by-humming evaluation. Proc. ISMIR 2003, 41-47, 2003
- [7] 長澤 慎子, 渡辺 知恵美, 伊藤貴之: Web から入手したデータに基づくコード進行を利用した楽曲類似度の提案と楽曲視聴支援システムの開発, 電子情報通信学会 第19回データ呼応学ワークショップ, (DEWS2008), 2008
- [8] 今井 小帆里, 倉林 修一, 清木 康: 連続メディアデータ分析・可視化機構を有する音楽データベースシステムの実現方式

