

音楽情報処理 最前線!

■ 第20回

未来の音楽の楽しみ方、作り方はどう変わるのか？
コンピュータは音楽を理解できるようになるのか？
コンピュータを使って音楽を研究する「音楽情報処理」
という研究分野が、世界的に注目を集めています。
本連載では、そうした最先端の研究事例を紹介していきます。

市販楽曲を自分好みの曲調に変えられる MusicFactorizer

「この曲、このメロディーがこうだったらもっとイケてるのに・・・」
「この曲、こんな曲調で聞いてみたいな・・・」などと感じたことはないだろうか？
そんな望みをかなえてくれるかも知れない技術、それが MusicFactorizer です！

楽曲を書き替える

購入した市販楽曲の、例えばコード進行・楽器構成・演奏スタイル・キー・リズム・メロディーなどを自由に書き換えて自分好みの曲に作り変えたいと思っただろうか。このようなことがもし可能になれば、音楽鑑賞中に自分の気分に合わせて曲調を変えてみたり、例えば楽曲制作の場面では、ゼロから曲を作り始めるのではなく、まず作りたいイメージの楽曲を準備して、その曲をベースに少しずつ作り変えていくことで手軽に作曲ができたりするようになるかも知れない。そうなれば、楽曲制作やアレンジに要するスキルの数値は格段に低くなるはずであるから、大衆にとってもはや楽曲制作はすべてプロにお任せするものではなくなり、自己表現の手段としてより身近なものとなってくるはずだ。このように書くとも誰もが簡単に考えつきそうな音楽サービスに聞こえるかも知れないが、ではなぜ今までこのようなサービスが実用化されてこなかったのか。それは技術的に大変難しい問題をクリアする必要があったからだ。今回ご紹介するのは、その難しい問題を面白い発想で解決してくれる技術である。

技術的に何が難しかったのか

では、何が難しいのか。それをまず説明しよう。例えば楽器音が一つしか鳴っていないのであれば、音の高さを変えるのは比較的簡単だ。要は、カセットテープを早送り(遅送り)するのと同じ

ように、波形編集ソフトなどを使って音の波形を時間方向に伸び縮みさせればよい。しかし、楽器音が二つ以上重なってくるとはそう簡単にはいかなくなる。複数鳴っている楽器音のうち、どれか一つの楽器音の高さだけを変えたいと思っても、同じ要領で波形を伸び縮みさせたのでは、残りの音もすべて同じように音の高さが変わってしまうからだ。音の高さのみならず、音色や音の長さについても個別に操作しようとするところと全く同様の問題に直面する。従って、混合音の中の個々の楽器音を個別に操作できるようにするためには、個々の楽器音の波形を分離する必要が出てくるわけだが、これが実に難しい。2と8を足すと10になるのは当たり前だが、10が2と8を足したものであるとは限らない(4と6を足したのもだったかも知れない)のと同じで、混合音の波形を作り出すのは簡単でも、混合音から混ぜる前の元の波形に分解するのは大変難しいのだ。

音楽を「音のパーツ」と「設計図」に分解する

楽器音の分離は、あらかじめ個々の楽器音のサンプルが手元であれば比較的容易に行えるが、あらゆる楽器音のサンプルを事前に登録しておくのはとても面倒だし現実的でない。今回ご紹介する MusicFactorizer には、楽器音サンプルが手元にない状況でも楽器音分離を可能にする強力な技術が組み込まれている。その技術コンセプトは、「赤ちゃんがいろいろな音を耳にするうちに一

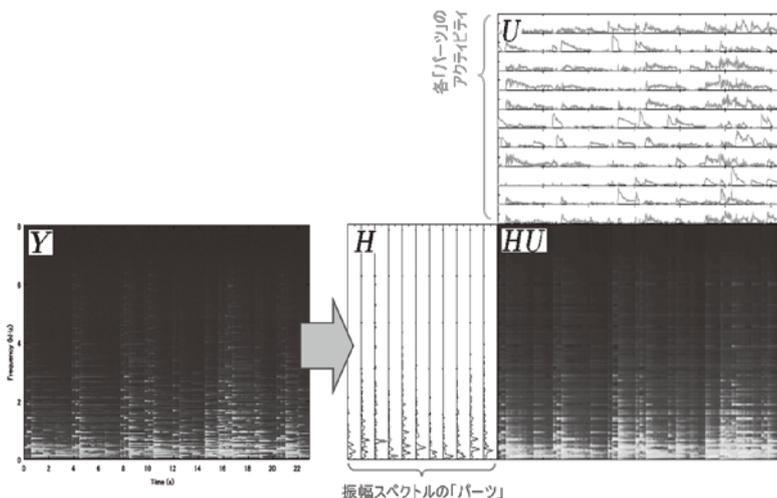


図1 非負値行列因子分解による
スペクトログラムのパーツベース表現

音響信号の時間周波数成分を非負値の行列 Y と見なして、非負値行列因子分解 [1] と呼ぶ方法で二つの非負値行列 H と U の積に分解 (Factorize) してみる。すると、行列 H の各列ベクトルは音響信号に含まれる特定楽器の特定音階に対応したスペクトルとなる。一方、行列 U の各行ベクトルはそれぞれの楽器音スペクトルがどの時刻にどの程度の強さでアクティブしているかを表した情報となる。MusicFactorizer では、さらに U をいくつかのパーツの重ね合わせで表わした階層的なモデルでスペクトログラムをモデル化している。この工夫により、一つ一つのノートがどの時刻からどの時刻まで及んでいるかも併せて推定可能となる。詳細は文献 [3] を参照されたい。

亀岡弘和

(かめおかひろかず)

2007年 東京大学大学院情報理工学系研究科博士課程修了。
情報理工学博士。現在、日本電信電話株式会社
NTTコミュニケーション科学基礎研究所 研究員。

音響信号を対象とした統計的信号処理に関する基礎研究に取り組んでいる。

「音楽情報科学研究会」へ 参加してみませんか？

情報処理学会 音楽情報科学研究会(SIGMUS)は、コンピュータと音楽とが関わり合うあらゆる場面を活動対象とする学際的研究会で、年5回の研究発表会を開催しています。研究会に会員登録すると、研究発表会の参加費が無料になるだけでなく、過去の全研究発表会の論文のダウンロードなどの特典があります。研究会の登録方法や研究発表会の開催に関する最新情報などは <http://www.sigmus.jp/> をご覧ください。

一つ一つの音を聞き分けることができるようになっていく学習過程]にたとえることができる。イメージはこうだ。まず、世の中の音を、いくつかの音の「パーツ」から成っているものだと考えて、初めて耳にした音に対してとりあえず現時点で記憶されている音のパーツを適当な音量バランスで混ぜ合わせたもので当てはめてみる。つまり、自分の知る限りの知識で世の中の音を認識しようとしていることになる。もし、それらの音のパーツではどう頑張ってもうまく当てはめられないようなら、記憶内の音のパーツを微調整して再度当てはめてみる。これを繰り返していくと、最終的に、世の中の音を構成している音のパーツと、それらの当てはめ方、すなわち「音の設計図」を同時に得ることができる。一つの楽曲に対してこの一連のプロセスを行うと、楽曲中の一つ一つのノートが音のパーツとして取り出され、それらがどのタイミングで鳴っているかが音の設計図として書き出される。つまり、楽曲の波形をMIDIのような形態で記述できたことになるわけだ。このように、音楽を、音のパーツと設計図に「因数分解(Factorize)」するのがMusicFactorizerなのだ。この技術は、非負値行列因子分解と呼ぶ技術(図1参照)をヒントにしている。

「MusicFactorizer」で 何が実現できるか？

このように音のパーツと設計図が得られれば、それぞれを個別に操作して楽曲を作り変えることが可能になる。例えば、音のパーツを別のもの(別の楽器音や別の音の高さなど)に置き換えて、演奏を再構成すれば、曲調や楽器構成を変えることができたりするだろう。また逆に、設計図の方を別のものに置き換えれば、同じ楽器構成で別の曲に変えることができるはずだ。MusicFactorizer(図2)を用いた楽曲の加工例を[2]にいくつか挙げているので、興味のある読者はぜひ一度チェックしてみたい。また、技術の原理に興味のある読者は下記参考文献[3]を一読されたい。

▼参考文献

- [1] D.D. Lee, and H.S. Seung, "Algorithms for nonnegative matrix factorization," Proc. NIPS'00, pp. 556-562, 2000.
- [2] <http://www.br1.ntt.co.jp/people/kameoka/MusicFactorizer/index.html>
- [3] 亀岡弘和、ルルー・ジョンソン、大石康智、柏野邦夫, "Music Factorizer: 音楽音響信号をノート単位で編集できるインタフェース," 情報処理学会研究報告, 2009-MUS-81-9, 2009.

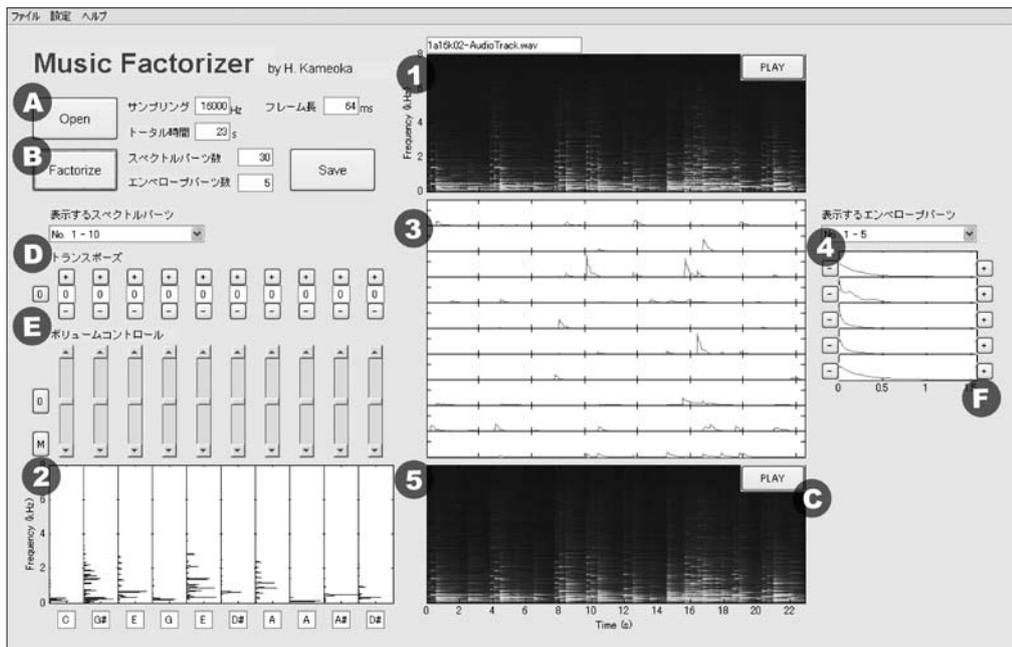


図2 MusicFactorizerの表示画面

- A 音響信号を読み込み、短時間フーリエ変換(STFT)により時間周波数解析を行ってスペクトログラムを出力する。
- B 観測信号から音のパーツと設計図を獲得するアルゴリズムを実行する。
- C D, E, Fの操作を介して加工されたスペクトログラムから音響信号を合成し、再生する。
- D トランスポーズ機能(各ノートを+/-ボタンにより100cent(半音)単位でトランスポーズできる。)
- E ボリュームコントロール機能(各ノートの音量をスライダにより自由に調節できる。)
- F サステイン・リリースタイムコントロール機能(各ノートのサステインないしリリースタイムを+/-ボタンにより調節できる。)