ニューラルネットワークによる
音楽の自動変換

陶山大輝[1 2 3]

2021年1月7日

[1]東京大学教養学部後期課程学際科学科総合情報学コース
[2]学籍番号：08-192021
[3]指導教員：金子知適准教授

概 要

本研究ではニューラルネットワークを用いて音色を変換することを目標とする。

目 次

第１章　用語の説明

本章では、音楽用語の説明を行う。

1.1　音

弾性体 (空気) 中を伝播する弾性波により起こされる音波が聴覚により感じられるもののことである。また、音波に周期性があり明確な音程を持つ音として聞こえる場合は楽音と呼ばれる。

1.2　楽音の三要素

楽音は高さ, 大きさ, 音色の三つの要素 (音の三要素) から成り立っているとされる。図 1 の通り、高さは音波の振動数により決まり、大きさは音波の振幅により決まる。

1.2.1　楽音の音色

音の高さと大きさが同じであるが異なった音として知覚される時の属性のことである。図 2 では音の高さと大きさが同じギターの音とハープの音の波形を示しており、この波形の違いが音色の違いとなる。

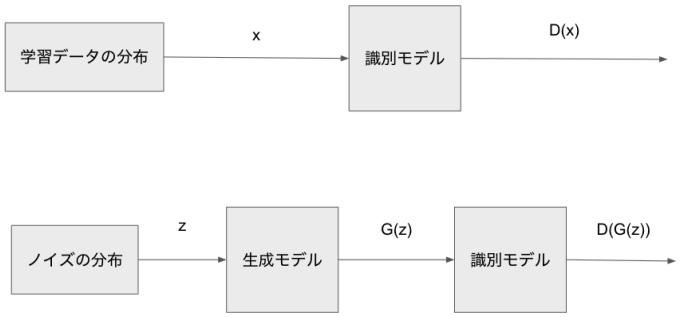
第2章　音楽の既存研究

　長期的な構造と短期的な構造のいずれもを変換しようとしており、その多くでは長期的な構造を音楽性を持ったまま変換するのが難しい (←要参考文献)
　長期的な構造は別のアルゴリズムを使うのが適切では？(文章の生成などの自然言語処理に近い？)
　短期的な構造のみを変えることに注目したい→音色による変換

第3章　背景

3.1　多層パーセプトロン

3.2　GAN

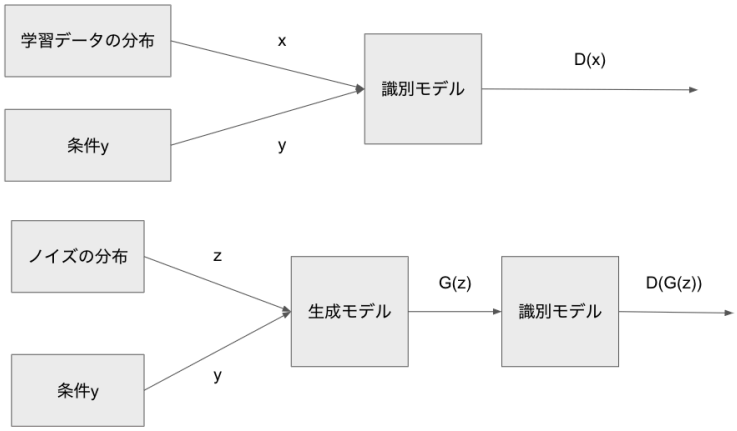


GAN (敵対的生成モデル) とは生成モデルと識別モデルが競合して学習を行うディープラーニングのモデルである。生成モデルは訓練データの分布を捉えようとし、識別モデルは訓練データである確率を推定する。

具体的には、訓練データ x の分布を p_{data}, 生成モデルの入力のノイズ z の分布を p_z, ノイズ z を元にデータを生成する関数を G, 生成モデルのデータではなく訓練データである確率を返す関数を D とした時、式 3.1 を生成モデルは最小化し識別モデルは最大化をすることを目標として学習を行う [1]。

$$\mathbb{E}_{\boldsymbol{x}\sim p_{\mathrm{data}}(\boldsymbol{x})}[\log D(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{z}\sim p_{\boldsymbol{z}}}(\boldsymbol{z})[\log(1 - D(G(\boldsymbol{z})))] \tag{3.1}$$

3.3 CGAN



CGAN(条件付き敵対的生成モデル) とは条件付きの GAN である。具体的には y という条件下で式 3.2 を生成モデルは最小化し識別モデルは最大化をすることを目標として学習を行う [2]。

原著論文では訓練時に MNIST の数字のラベルを条件として与えることで特定の数字のラベルの画像を生成するモデルを実装している。

$$\mathbb{E}_{\boldsymbol{x}\sim p_{\text{data}}(\boldsymbol{x})}[\log D(\boldsymbol{x} \mid \boldsymbol{y})] + \mathbb{E}_{\boldsymbol{z}\sim p_z(\boldsymbol{z})}[\log(1 - D(G(\boldsymbol{z} \mid \boldsymbol{y})))] \tag{3.2}$$

3.4 Pix2pix

Pix2pix とはある条件下で画像間の変換を行う GAN である。先程の CGAN を元に U-Net と PatachGAN を組み合わせたネットワークになっている。U-net とは…,PatchGAN とは…。(ここは後々調べて書く)

具体的には、画像 x を条件として式 3.5 を生成モデルは最小化し識別モデルは最大化をすることを目標として学習を行う [3]。

$$\mathcal{L}_{cGAN}(G, D) = \mathbb{E}_{x,y}[\log D(x, y)] + \mathbb{E}_{x,z}[\log(1 - D(x, G(x, z)))] \tag{3.3}$$

$$\mathcal{L}_{L1}(G) = \mathbb{E}_{x,y,z}\left[\|y - G(x, z)\|_1\right] \tag{3.4}$$

$$\mathcal{L}_{cGAN}(G, D) + \lambda\mathcal{L}_{L1}(G) \tag{3.5}$$

第4章　提案手法

4.1　データ形式

音のファイル形式としてはWAVを用いる。一般的にはMP3やMP4などの非可逆圧縮形式が広く用いられるが、WAVは波形データを直接保持しており本論文ではWAVを扱いやすいと考えたためである。

音のデータそのものだけでなく量子化ビット, サンプリング周波数, チャンネル数, サンプリング数のメタ情報を得ることができる。

また、サンプリング周波数を44100Hz, 量子化ビットを16bit, チャンネル数を1, サンプリング数を44100に固定して今回の実験を行った。

4.2　データセット

楽譜作成ソフトのMuseScoreを利用して国際の階名表記でA0からC8に含まれる半音をwav形式で88音生成した。また、この88音は一般的な88鍵のピアノで出すことのできる全音階の音であり、人間が音程として聞き分けることのできる限界の音域としてこの音域を選んだ。

また、楽器としてはエレキギターとハープを選んだ。

4.3　判定器

(1) 音程が維持されているか
(2) 変換されて音色が変換されているか
波形とスペクトログラム？

4.4　音色の変換

音色の変換を音楽で行う際に (音の特徴を学習する) が、以下の三つの点を解決するのが難しいと考えられる。また、以下の三つを解決することで、単音における音色の変換を音楽に適用することができる。

4.4.1　楽器の重ね合わせ

楽器ごとに音色が異なるので、楽器ごとの音波に分解して音色変換を行うことが良いと考えられる。なお、楽曲の作成時に楽器ごとに分離したデータ (パラデータ) で保存しておけば、直接楽器ごとの音波を利用できる。

4.4.2　時間方向の音の繋ぎ方

時間方向での音の繋ぎ方は都合の良いように分割していくことでなんとかなるのではないか…？、分割する (1 つの音の判定を行う) のは難しい…？、自己回帰？

4.4.3　音の重ね合わせ

単位時間の楽器の音に注目した時、楽器ごとの音波に分離したとしても和音のようにその単位時間で複数の種類の音が鳴っている場合も難しいと考えられる。

とりあえず試しでも良いので実験を行いたい

第5章　まとめ

謝辞

関連図書

[1] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks, 2014.

[2] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets, 2014.

[3] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-to-image translation with conditional adversarial networks, 2018.