ニューラルネットワークによる
音楽の自動変換

教養学部後期課程学際科学科総合情報学コース
陶山大輝
学籍番号：08-192021
指導教員：金子知適准教授

目 次

第1章　初めに

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。

1.1　着想

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2章　用語の説明

本章では音楽用語の定義及びその説明を行う。

2.1　音

音とは、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。

2.2　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。

音の高さ

音の高さは音波の周波数により決まる。一般には、フーリエ変換によりスペクトル分析を行った際の最も低い周波数成分の音波の周波数 (基音) を音の高さと呼ぶ。また、図 2.1 は図 2.2 で示される音波のスペクトルであり、基音が 264Hz となる。

音の大きさ

音の大きさは音波の振幅により決まる。図 2.3 では同じ楽器から出る同じ高さの音の音波を示しているが、振幅の大きい後者の方が音の大きさは大きい。

音の音色

音の高さと大きさが同じであっても異なった音として知覚される時の違いを音色と呼ぶ。図 2.4 は上側はギターの音波, 下側はハープの音波で同じ高さかつ同じ大きさであるが、このような音波の波形の違いが音色の違いを作り出す。

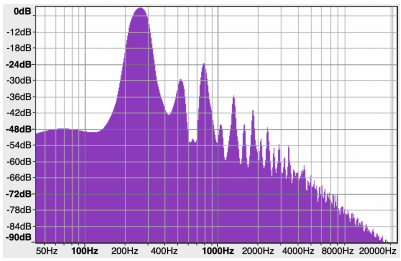

図 2.1: ハープの音波のスペクトル

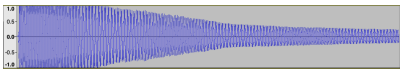

図 2.2: ハープの音波の波形

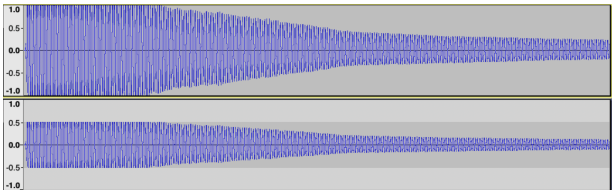

図 2.3: 音の大きさの異なるハープの音波

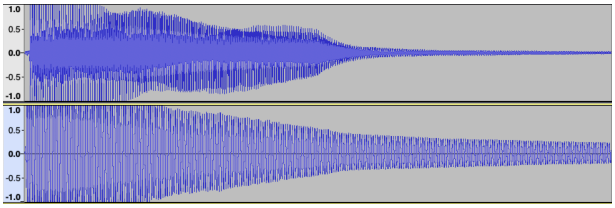

図 2.4: ギターとハープの音色の比較

第3章　背景

本章では、Multilayer perceptron (MLP) 及びその応用例である Generative Adversarial Networks (GAN) の説明を行った後に GAN を画像のスタイル変換に応用した Pix2pix を紹介する。

3.1　MLP

MLP は入力層と出力層を持つニューラルネットワークの一つであり、式 3.1 として定式化することができる。

$$\boldsymbol{y} = f_n(W_n(f_{n-1}(W_{n-1}\cdots(f_1(W_1(\boldsymbol{x})))))) \tag{3.1}$$

ここで、$\boldsymbol{x}$ は MLP への入力の教師データ,$\hat{\boldsymbol{y}}$ は MLP の出力,$\boldsymbol{y}$ は MLP の出力の教師データ,n は層の総数,f_i は i 番目の層の活性化関数,W_i は i 番目の層の重みの行列である。

また、損失関数 $L(\hat{\boldsymbol{y}}, \boldsymbol{y})$ を小さくする方向に重みの更新が行われることで学習は進むが、任意の活性化関数が微分可能である時は誤差逆伝播法により高速に重みの更新を行うことが可能である。

3.2　GAN

GAN [1] は MLP の応用例であり、生成モデルと識別モデルが競合して学習を行う。生成モデルは自身の出力が学習データであると識別モデルに推定させること、識別モデルは学習データと生成モデルの出力のどちらであるかを識別することを目指して学習する。

また、生成モデルの目的関数は式 3.2 であり、識別モデルの目的関数は 3.3 である。

$$\arg\min_{\theta_G} \mathbb{E}_{\boldsymbol{z}}[\log(1 - D(G(\boldsymbol{z};\theta_G);\theta_D))] \tag{3.2}$$

$$\arg\max_{\theta_D} \mathbb{E}_{\boldsymbol{x}}[\log D(\boldsymbol{x};\theta_D)] + \mathbb{E}_{\boldsymbol{z}}[\log(1 - D(G(\boldsymbol{z};\theta_G);\theta_D))] \tag{3.3}$$

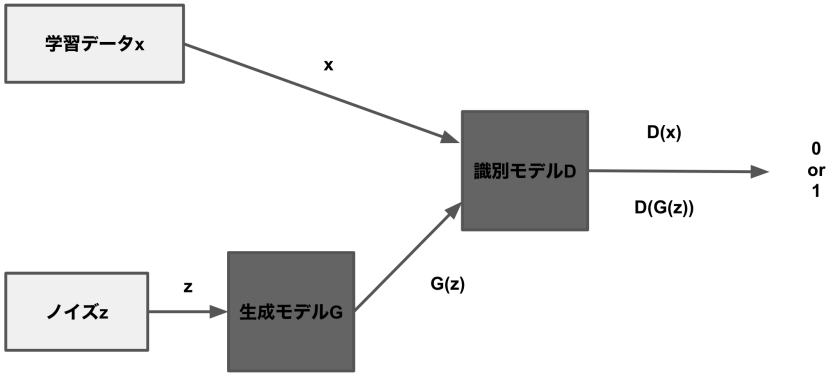

図 3.1: GAN のネットワークの図

ここで、$\boldsymbol{x}$ は学習データ,$\boldsymbol{z}$ は生成モデルへの入力のノイズ,$G(\boldsymbol{z};\theta_G)$ はノイズ $\boldsymbol{z}$ を入力とする生成モデル,$D(\cdot;\theta_D)$ は識別モデル,θ_G は生成モデル G のパラメータ,θ_D は識別モデル D のパラメータである。

3.3 Pix2pix

Pix2pix [2] はある条件下で画像間の変換を行う GAN である。図 3.3 にあるようにピクセルの対応関係を変えずにスタイル変換を行うことができる。

また、生成モデルの目的関数は式 3.4 であり、識別モデルの目的関数は 3.5 である。このような条件付きの GAN を Conditional GAN [3] と呼ぶ。

$$\arg\min_{\theta_G} \mathbb{E}_{\boldsymbol{x},\boldsymbol{z}}[\log(1 - D(\boldsymbol{x}, G(\boldsymbol{x}, \boldsymbol{z}; \theta_G); \theta_D))] + \mathbb{E}_{\boldsymbol{x},\boldsymbol{y},\boldsymbol{z}}[\|\boldsymbol{y} - G(\boldsymbol{x}, \boldsymbol{z}; \theta_G)\|_1] \tag{3.4}$$

$$\arg\max_{\theta_D} \mathbb{E}_{\boldsymbol{x},\boldsymbol{y}}[\log D(\boldsymbol{x}, \boldsymbol{y}; \theta_D)] + \mathbb{E}_{\boldsymbol{x},\boldsymbol{z}}[\log(1 - D(\boldsymbol{x}, G(\boldsymbol{x}, \boldsymbol{z}; \theta_G); \theta_D))] \tag{3.5}$$

ここで、$\boldsymbol{x}$ は変換先の学習データ,$\boldsymbol{y}$ は変換元の学習データ,$\boldsymbol{z}$ は生成モデルへの入力のノイズ,$G(\boldsymbol{x}, \boldsymbol{z}; \theta_G)$ はノイズ $\boldsymbol{z}$ を入力とする生成モデル,$D(\boldsymbol{x}, \cdot; \theta_D)$ は識別モデル,θ_G は生成モデル G のパラメータ,θ_D は識別モデル D のパラメータである。

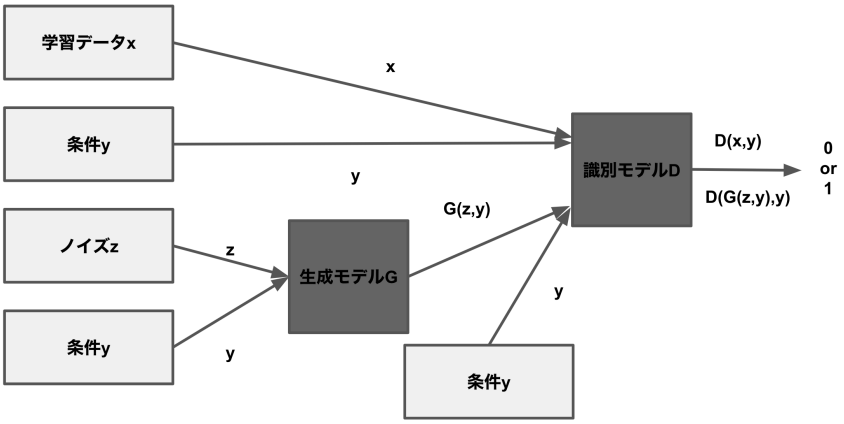

図 3.2: pix2pix のネットワークの図

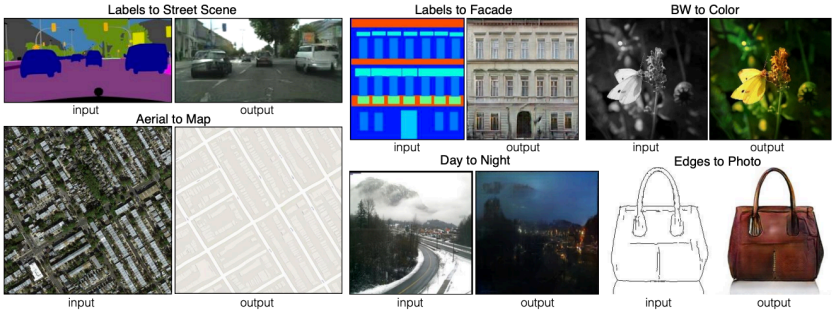

図 3.3: pix2pix のスタイル変換の例

3.3.1 生成モデルの構造

スタイル変換では基礎的な構造を保持したまま変換を行う必要があり、一般には encoder-decoder のネットワークが用いられる。Pix2pix の生成モデルでは、3.4 のように U-net[4] で用いられるスキップコネクションを持ったネットワークを用いている。

3.3.2 識別モデルの構造

Pix2pix の識別モデルでは、局所的な部分の識別の精度を高めるために PatchGAN という手法を用いている。これにより、パッチと呼ばれる小領域ごとで学習データであるかどうかの識別を行うことができる。

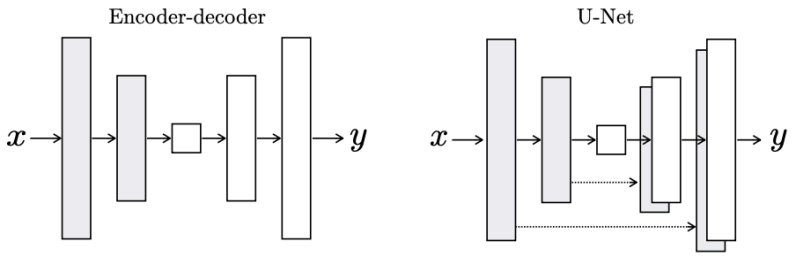

図 3.4: U-net のネットワーク

第4章　提案手法

本研究で使用したデータセット及び提案手法の説明を行う。

4.1　データセット

本研究のデータセットの作成方法及びその形式についての説明を行う。

4.1.1　データセットの作成方法

楽譜作成ソフトの MuseScore[1]により国際の階名表記で A0 から C8 までの半音を wav 形式で 88 音生成した。また、これらの音は 88 鍵のピアノで出すことのできる音であり、最も一般的な音域として今回の実験では選んだ。

そして、音色の変換を行う楽器としては十分に音色が異なると考えられるエレキギターとハープを選んだ。

4.1.2　データ形式

音のファイル形式としては非圧縮形式の WAV を用いる。MP3 や MP4 などの非可逆圧縮形式も一般には広く用いられるが、音波の波形データを直接保持しているために扱いやすい WAV を本論文で用いることにした。また、WAV の持つ波形データ以外のメタデータのうち本論文で扱うものについて以下で説明をする。

サンプリング周波数

サンプリング周波数とは、デジタル信号の単位時間あたりの標本化の回数のことである。本論文では 44100Hz に固定して実験を行う。

サンプリング数

サンプリング数とは、デジタル信号の標本化の合計の回数のことである。本論文では 44100 回に固定して実験を行う。

[1] https://musescore.org/

量子化ビット数

量子化ビット数とは、デジタル信号の大きさを表現するビット数のことである。本論文では 16bit に固定して実験を行う。

チャンネル数

チャンネル数とは、モノラルな音声の出力の総数のことである。本論文では 1 に固定して実験を行う。

4.2 手法

Pytorch による Pix2pix の実装[2]を参考に音楽で用いることができるように改変したものを実装では用いた。学習やテストの際のパラメータなどについては第 6 章に示す。

4.3 実験

4.3.1 生成モデルの表現力

4.3.2 生成モデルの汎化能力

[2] https://github.com/junyanz/pytorch-CycleGAN-and-pix2pix

第5章　まとめ

5.1　future work

5.2　音楽の変換

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。

5.2.1　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。

5.2.2　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？

5.2.3　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

5.2.4　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？

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

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

[4] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation, 2015.

第6章　付録

6.1　実験時のパラメータ