

卒業論文概要書

Summary of Bachelor's Thesis

Date of submission: 02/02/2021

学科名 Department	情報通信	氏名 Name	蓮沼 亮永	指導 教員 Advisor	渡辺 裕 ㊞
研究指導名 Research guidance	オーディオビジュアル 情報処理研究	学籍番号 Student ID number	1w173107-2 ^{CD}		
研究題目 Title	敵対的生成ネットワークによる線画形状のスタイル変換 Style Transfer for Line Drawings Using Generative Adversarial Networks				

1. まえがき

機械学習を用いた画像処理は、その応用の容易さから、様々な分野で利用されている。ここでは応用先の一つとして、漫画を取り上げる。漫画に機械学習を用いた既存研究には、漫画の自動彩色、ある漫画家風の画像生成などがある。本研究ではより実践的な漫画画像を生成するため、漫画の画風変換を試みる。

画像生成の研究には敵対的生成ネットワーク (Generative Adversarial Networks(GAN))[1] と呼ばれる画像生成のモデルが広く採用されている。GAN に多くの種類がある。今回は CycleGAN[2], U-GAT-IT[3], StyleGAN[4]による画風変換を検討する。

2. 関連技術

2.1. I2I-Translation(画像変換)

馬の画像をシマウマに変換するように、二つのドメイン間で変換を行うことを画像変換という。画像変換には、同じ位置、姿勢の馬とシマウマのペア画像が必要な場合があり、その場合データセットの用意が困難である。一方で GAN をベースにした CycleGAN, U-GAT-IT はペアでないデータセットから画像変換を行うことが可能であり、学習が容易な手法である。

2.2. StyleGAN

StyleGAN は画像変換手法ではなく、一種の画像生成手法である。Progressive GAN[5]をもとにしており、潜在空間を取り入れることで、生成画像の精度を上げている。多くの GAN はランダムな変数である潜在コードを持つ。潜在コードには対応した生成画像があるが、番号と画像の特徴に相関はない。

画像データセットは画像空間という分布上に配することができる。画像空間に対応するように潜在空間を学習させることで、生成画像と潜在空間に相関を持たせることができるのだ。ここでは StyleGAN

に改良を施した StyleGAN2[6]を用いることとする。StyleGAN2 では Generator の構造がシンプルになり、StyleGAN と比べ生成画像の精度が上がっている。

3. 検討手法

データセットには Manga109[7]を用いた。図1のように顔部分を切り抜き、正面を向いている画像を残すようにデータクリーニングを行った。

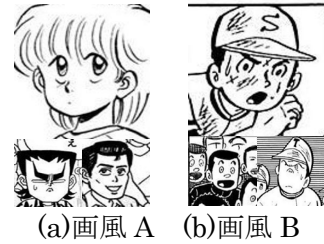


図1 データクリーニング後の Manga109

画風変換の手法として、CycleGAN, U-GAT-IT, StyleGAN2 の三つを検討した。CycleGAN, U-GAT-IT は、生成モデルから直接変換画像を出力可能である。StyleGAN2 では画風を変換するために GANSpace[8]を用いた。StyleGAN2 は画像生成手法であるため、単体では画風変換が出来ない。潜在空間を操作することで変換を行い、その操作のために GANSpace を用いる。

4. 実験

データセットには、画風 A, B の画像を約 600 枚ずつ用意した。生成画像の評価値は FID[9]である。FID はデータセットと生成画像の分布の差と生成画像の多様性を測定する。CycleGAN と U-GAT-IT は画風 A→B の生成画像と、画風 B のデータセット間の FID を測定した。StyleGAN2 では画風 B の生成画像とデータセット間の FID を算出している。図2に各 GAN の生成画像を、表1に FID を示す。

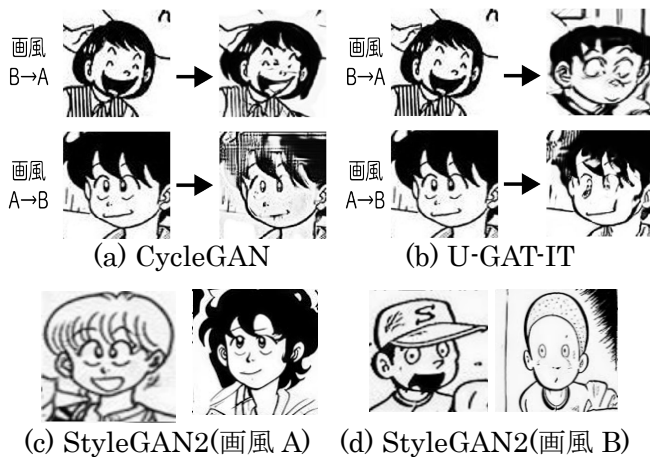


図 2 各 GAN の生成画像例

表 1 各 GAN における FID

	CycleGAN	U-GAT-IT	StyleGAN2
FID	61.01	79.03	20.17

StyleGAN2 は学習コストが高いため、既存の学習モデルを再学習させる転移学習という手法を用いた。今回はさらに、転移学習を安定させる FreezeD[10] と、画像枚数を増強する Adaptive Discriminator Augmentation(ADA)[11]を適用した。このように StyleGAN2+FreezeD+ADA で学習させたモデルに GANSpace を用い、画風変換した結果を図 3 に示す。



図 3 画風変換の結果

5. 考察

図 2、表 1 の結果から StyleGAN2 が画風変換に適していることが分かった。図 1 から CycleGAN はテクスチャの変換のみで、線画の変換が難しいことが確認できる。U-GAT-IT は画像の変換は行えているものの、絵がくずれてしまっている。FID は画像の認識のしやすさが重要であるため、U-GAT-IT の FID が一番低くなったと考えられる。StyleGAN2 の生成画像は学習したデータセットに近い画像が得られた。しかし GANSpace を用いた画風変換は、きれいな結果は得るまでには至らなかった。GANSpace は潜在空間内の変化の大きい軸を検出する。軸の特徴を指定するわけではないため、その軸にはいくつ

かの特徴が混ざり合うことになる。そのため図 3 の画風変換として採用した軸に、性別や髪型の要素が混ざり合っている。

また三つ以上の漫画データを学習させた場合も検討した。生成画像の画風を指定するには、それぞれの画風の持つ潜在空間上の部分空間を探す必要がある。しかし StyleGAN2 のもつ 512 次元上の空間を三つ以上の画風に分解することは困難であった。このことからより多くの画風を自在に生成するためには、潜在空間のより簡単な操作方法が必要だと考えられる。この他にも Manga109 の画像枚数が少ない点、FID が最適な評価手法でない点など改善の余地は多い。

6. まとめ

本研究では、CycleGAN、U-GAT-IT、StyleGAN2 による漫画の画風変換を比較した。実験からは StyleGAN2 の潜在空間を操作することで、いい結果が得られることが分かった。自由にいくつもの画風を変換するには、データセットの増加やより詳細な潜在空間の操作を検討すべきである。

参考文献

- [1] I.J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, "Generative Adversarial Nets", arXiv:1406.2661, Jun, 2014,
- [2] J. Zhu, T. Park, P. Isola, A. A. Efros, "Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks", arXiv:1703.10593, Mar 30, 2017
- [3] J. Kim, M. Kim, H. Kang, K. Lee, "U-GAT-IT: Unsupervised Generative Attentional Networks with Adaptive Layer-Instance Normalization for Image-to-Image Translation", arXiv:1907.10830, Jul 25, 2019
- [4] T. Karras, S. Laine, T. Aila, "A Style-Based Generator Architecture for Generative Adversarial Networks", arXiv:1812.04948, Dec 12, 2018
- [5] T. Karras, T. Aila, S. Laine, and J. Lehtinen, "Progressive growing of GANs for improved quality, stability, and variation. CoRR, abs/1710.10196, 2017
- [6] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, T. Aila, "Analyzing and Improving the Image Quality of StyleGAN", arXiv:1912.04958, Dec 3, 2019
- [7] K. Aizawa, A. Fujimoto, A. Otsubo, T. Ogawa, Y. Matsui, K. Tsubota, H. Ikuta, "Building a Manga Dataset 'Manga109' with Annotations for Multimedia Applications", arXiv:2005.04425, May 9, 2020
- [8] E. Härkönen, A. Hertzmann, J. Lehtinen, S. Paris, "GANSpace: Discovering Interpretable GAN Controls", arXiv:2004.02546, 6 Apr 2020
- [9] S. Mo, M. Cho, J. Shin, "Freeze the Discriminator: A Simple Baseline for Fine-Tuning GANs", arXiv:2002.10964, Feb 25, 2020
- [10] C. Bowles, L. Chen, R. Guerrero, P. Bentley, R. Gunn, A. Hammers, D. A. Dickie, M. V. Hernández, J. Wardlaw, D. Rueckert, "GAN Augmentation: Augmenting Training Data using Generative Adversarial Networks", arXiv:1810.10863, Oct 25, 2018
- [11] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, S. Hochreiter, "GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium", arXiv:1706.08500, Jun 26, 2017

2020 年度 卒業論文

敵対的生成ネットワークによる

線画形状のスタイル変換

Style Transfer for Line Drawings Using Generative
Adversarial Networks

提出日 2021 年 2 月 2 日

指導教員 渡辺 裕 教授

早稲田大学 基幹理工学部

情報通信学科

1W173107-2

蓮沼 亮永

目次

第1章	序論.....	1
1.1	研究背景.....	1
1.2	研究目的.....	1
1.3	本論文の構成.....	2
第2章	関連研究.....	3
2.1	まえがき.....	3
2.2	Generative Adversarial Networks.....	3
2.2.1	概要.....	3
2.2.2	CycleGAN.....	3
2.2.3	U-GAT-IT.....	4
2.2.4	StyleGAN と StyleGAN2.....	5
2.3	GAN の関連技術.....	6
2.3.1	FreezeD.....	6
2.3.2	Adaptive Discriminator Augmentation.....	7
2.3.3	GANSpace.....	8
2.4	むすび.....	9
第3章	検討手法.....	10
3.1	まえがき.....	10
3.2	データセット.....	10
3.3	Image-to-Image Translation を用いた画風変換方法.....	11
3.3.1	CycleGAN を用いた変換.....	11
3.3.2	U-GAT-IT を用いた変換.....	11
3.4	StyleGAN を用いた生成.....	11
3.4.1	StyleGAN のモデルの学習.....	11
3.4.2	生成画像の操作.....	12
3.5	評価手法について.....	13
3.6	むすび.....	13
第4章	実験.....	14
4.1	まえがき.....	14
4.2	実験の概要.....	14
4.3	各 GAN の結果と I2I-Translation の問題点.....	15

4.3.1 各 GAN の生成画像とその評価	15
4.3.2 CycleGAN の結果と考察	16
4.3.3 U-GAT-IT の結果と考察	17
4.4 StyleGAN での画風変換	18
4.4.1 潜在空間の検討手法	18
4.4.2 潜在空間の考察	19
4.5 むすび	20
第5章 結論と今後の課題	21
5.1 結論	21
5.2 今後の課題	21
謝辞	22
参考文献	23
図一覧	25
表一覧	26

第1章 序論

1.1 研究背景

機械学習を用いた画像処理は、その応用の容易さから、様々な分野で利用されている。ここ数年はアート分野でもその力を発揮しており、AI の画家が個展を開いた例もある。このような画像処理技術の応用先の一つとして、漫画を取り上げる。

世界のコミックスの中で最大の市場規模を持つ日本の漫画は、現在も勢い衰えることなく数々の人気作品を生み出し続けている。この人気を裏付けるように漫画家の数は年々増加傾向にある。昨年には AI 漫画家のデビュー作ともいえる、制作の随所に機械学習を用いた短編漫画が人気雑誌に掲載された。

漫画制作においては、用紙にペンで描く場面が多く想定されるが、実際はデジタルで制作を行っている現場がほとんどである。アナログな下書きをスキャンして、デジタルでペン入れをする場合もあるが、最初から全工程を漫画制作ソフトにゆだねている作家が大勢を占めている。制作ソフトの手振れ補正、編集可能なトーンなどは非常に重宝され、中には 3D モデルを参考に下書きを描く場合もある。このように日本の漫画制作にとってデジタル技術はきつても切り離せないものであり、デジタル技術の向上が漫画の発展に貢献しているのは確かである。

1.2 研究目的

本研究では、漫画の画像データをもとに、二つ以上の漫画間の画風変換を行うことを目的とする。

既存の研究では、敵対的生成ネットワーク(Generative Adversarial Networks(GAN)) [1] という画像生成技術を用い、特定の漫画家やアーティストの画風を再現することに成功している。しかし、生成できる画像は非常に限定的であり、自分の思い描く絵を生成するという段階までは程遠い。単一の画風ではなく、生成画像の画風の幅が広がれば、その分想定される利用用途は増えると考えられる。そのため、この研究では画風の調整について検討をする。

学習モデルの生成画像が、いくつかの画風を自在に行き来できるようであれば、思い浮かべる絵に近い画像が生成できる。このような処理が実現できれば、機械学習による漫画画像の生成は、漫画制作の助力となる技術により近づけると考えられる。

1.3 本論文の構成

本論文の構成を以下に示す。

第 1 章は本章であり、本研究の背景、目的について述べる。

第 2 章では、本研究で用いる Generative Adversarial Networks (GAN) と、その周辺技術について述べる。

第3章では、GANを用いた漫画の画風変換手法を紹介する。

第4章では、提案手法を用いた実験、その結果と考察について述べる。

第5章では、本研究の結論と今後の課題について述べる。

第2章 関連研究

2.1 まえがき

本章では、本研究で用いる技術について述べる。まず、敵対性生成ネットワーク (Generative Adversarial Networks (GAN)) [1] を用いた画像生成手法について、そして今回使用した CycleGAN [2], U-GAT-IT [3], StyleGAN [4] の詳細を記載する。さらに GAN の補助的な技術として、レイヤーの固定、データセットの拡張、GAN の潜在空間について扱う。

2.2 Generative Adversarial Networks

2.2.1 概要

Generative Adversarial Networks (GAN) [1] は、2014 年に I.J. Goodfellow らによって提案された、生成モデルを訓練するためのフレームワークである。

全体は Generator と Discriminator から形成される。Generator は画像の生成部分であり、学習データにより近い画像を生成する。Discriminator は学習データの画像と Generator の生成画像を見比べ、生成画像が本物であるかを見極めを行う。学習が進むと互いに精度が上がり、学習データに近い画像が生成できるようになる。

このように二つのモデルを敵対的に学習させることで、Generator は学習データの分布を学習することになる。下の図 2.1 に GAN 全体の構造を示す。

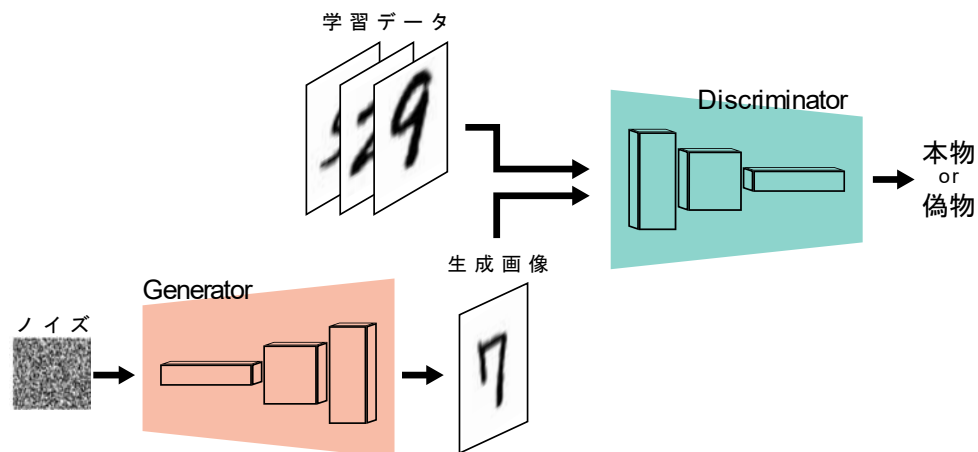


図 2.1 Generative Adversarial Networks (GAN) の構造

2.2.2 CycleGAN

J.Y. Zun らによって 2017 年に提案された CycleGAN [2] は、ある画像をソースドメイン X からターゲットドメイン Y へ教示なしで変換する手法である。CycleGAN は二つの写像 G, F が構成される。写像 G はドメイン X から Y への変換を行い、Generator $G(x)$ の分布をドメイン Y の分布へ近づける。写像 F は対照的にドメイン Y から X への変換を行う。

CycleGAN 全体の構造を見ると、Generator は六つ、Discriminator は四つである。写像 G , F の二つの画像生成のシステムを同時に学習させ、安定した画像変換を行う。次の図 2.2 に CycleGAN の一つの写像の学習フレームワークを示す。

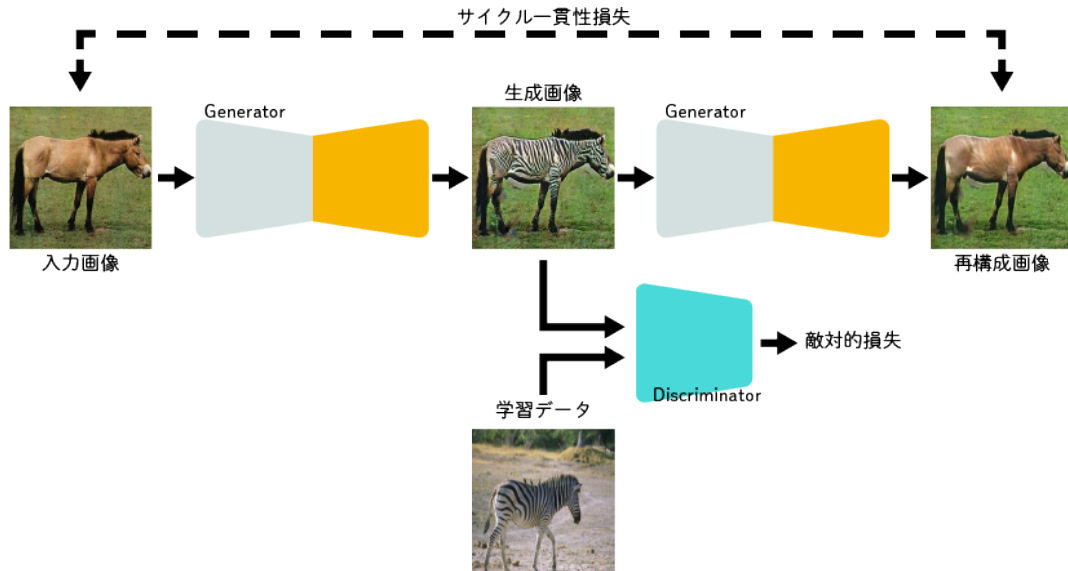


図 2.2 CycleGAN の学習フレームワーク

GAN では、出力から計算される損失関数を小さくするように各重みを調整し、モデルを学習させる。どの GAN でもそれぞれ妥当な損失関数を設計しており、CycleGAN では、サイクル一貫性損失と呼ばれる項を損失関数に取り入れている。この項は写像 G , F それぞれにつき、変換後の画像から元の入力画像を復元したものと、入力データとの比較を行い、その差を小さくすることを期待している。Generator と Discriminator から計算される敵対的損失をそれぞれ $L_{GAN}(G, D_Y, X, Y)$, $L_{GAN}(F, D_X, Y, X)$ とする。サイクル一貫性損失を $L_{cyc}(G, F)$ とすると、目的関数は相対的な重みを制御する λ を用いて、式(2.1)で計算される。

$$L(G, F, D_X, D_Y) = L_{GAN}(G, D_Y, X, Y) + L_{GAN}(F, D_X, Y, X) + \lambda L_{cyc}(G, F) \quad (2.1)$$

2.2.3 U-GAT-IT

U-GAT-IT[3]は 2019 年に J.Kim らによって提案された、教示なしの画像変換手法である。既存の GAN と異なる部分は、Attention モジュールと新しい正規化関数 AdaLIN を組み入れている点である。

U-GAT-IT は、Generator と Discriminator の両方において、画像データの重要な部分を抽出する機能、Attention モジュールを組み込んでいる。この補助機能により画像の形を変換することを促進している。また Adaptive Layer-Instance Normalization (AdaLIN) という新しい正規化手法によって、学習時のハイパーパラメータや、モデルの構造を変えることなく、

画像の形とテクスチャの変換強度を調整している．Attention モジュールは Generator と Discriminator の両方に組み込まれ，それぞれの精度を向上させている．U-GAT-IT の Generator の構造を図 2.3 に示す．

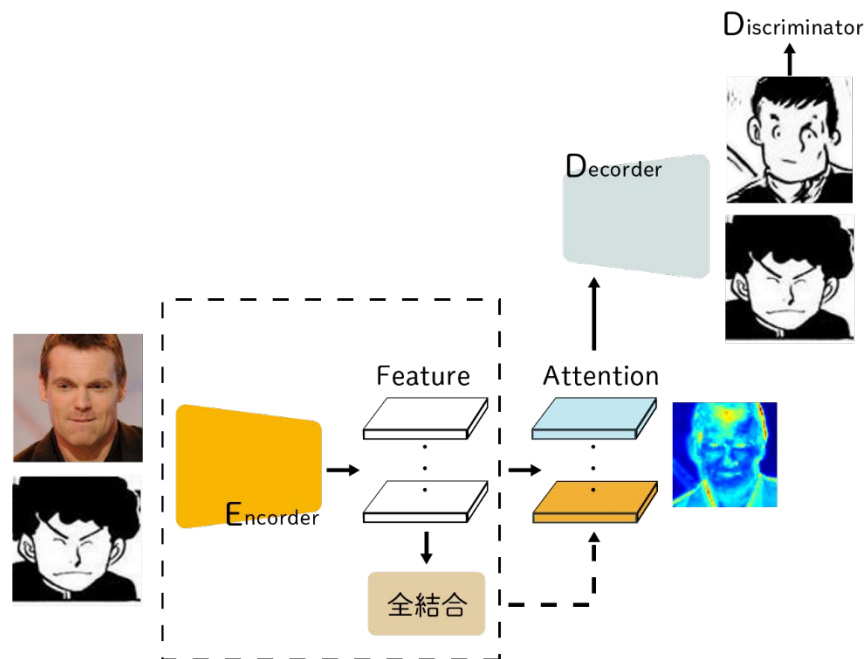


図 2.3 U-GAT-IT の Generator の構造

CycleGAN と比較すると評価値 KID[13]が良好であり，テクスチャのみならず，形状の変化も伴う結果が得られている．人の顔画像をアニメ調に変換する処理では，リアルな人の顔がアニメキャラクターのような目や髪に変換されている．このことから漫画での変換結果にも期待ができると考えられる．

2.2.4 StyleGAN と StyleGAN2

T.Karras らによって 2019 年に提案された StyleGAN[4]はスタイル変換の技術を GAN へ取り入れた生成手法である．StyleGAN は画像生成の手法であり，Image-to-Image Translation，つまり画像変換の手法である CycleGAN や U-GAT-IT とはアプローチが異なる．Generator と Discriminator を画質ごとに段階的に学習させる Progressive GAN[5]が基礎にあり，ノイズや潜在空間の探索といった改良点により，精度の良い画像を生成する．

2020 年にはこれを改良した StyleGAN2[6]が提案され，生成画像の細部のクオリティが向上した．StyleGAN と比べると Generator の部分に多くの改良点があるが，大まかなネットワークの構造は同様である．StuleGAN2 の構造を図 2.4 に示す．

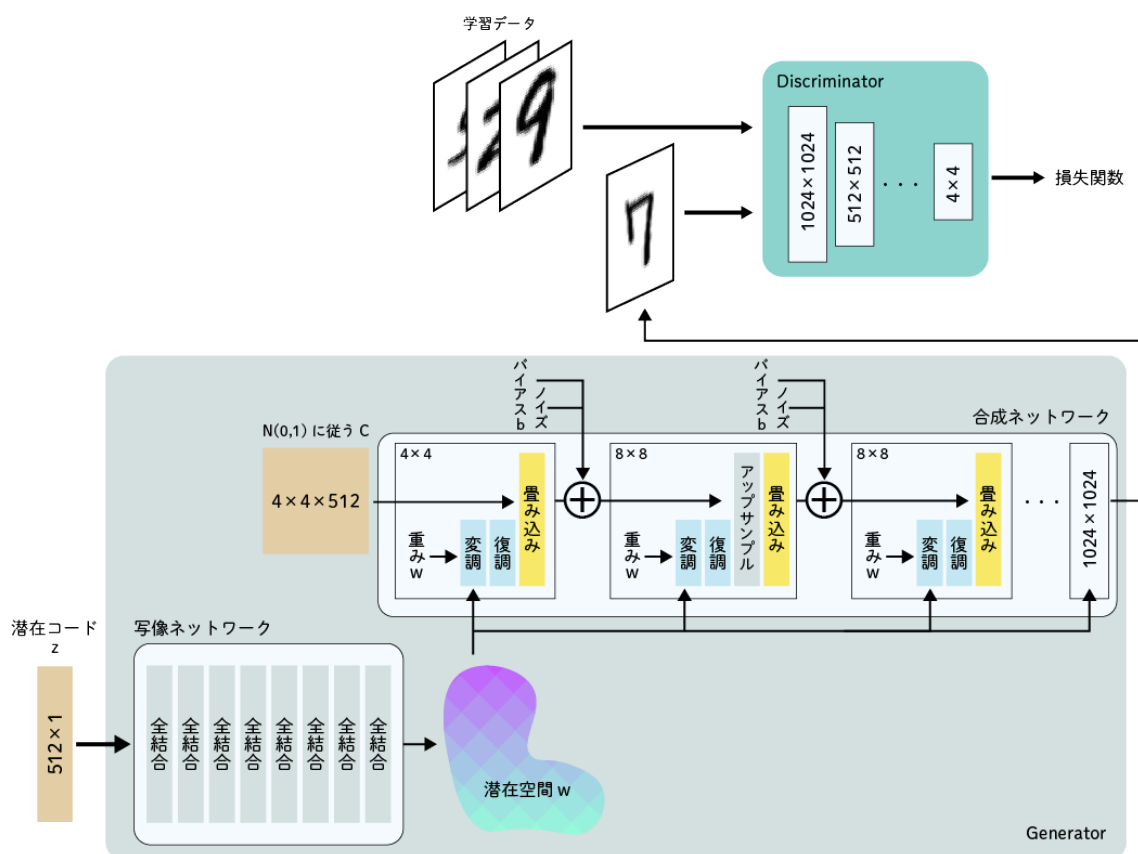


図 2.4 StyleGAN2 の構造

Generator は写像ネットワークと合成ネットワークから構成されている。写像ネットワークは学習データの潜在空間を推定し、形成された潜在空間をスタイルとして扱い、合成ネットワークの各解像度で畳み込みに適用している。CycleGAN や U-GAT-IT は数百枚～千枚前後の画像があれば学習に事足りたが、StyleGAN の学習には数万枚の画像が必要となる。学習時間も数週間にわたり、クオリティが高い代わりに、学習コストも高いのが特徴である。そこで学習コストを減らす方法はいくつか考案されており、そのうち二つを次の 2.3 に示す。

2.3 GAN の関連技術

2.3.1 FreezeD

FreezeD[7]は S.Mo らによって提案された、ファインチューニングの GAN での改良手法である。

GAN を学習する上で、データセットの画像枚数が少ない場合、Discriminator の過学習など問題が発生する。過学習が起きると、Discriminator は意味のある分類を行わなくなり、学習が発散する。転移学習はこれを防ぐための手法である。図 2.5 に転移学習の手法を示す。

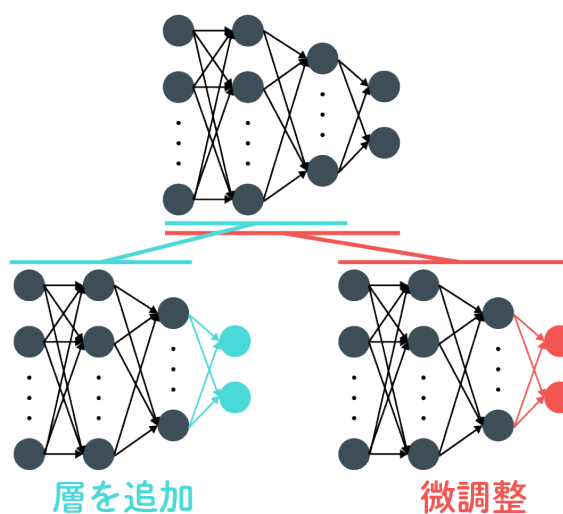


図 2.5 転移学習の手法

画像枚数が少ない場合，学習済みのモデルから転移学習を行うことで，少ないデータセットであっても十分に学習を行うことが可能となる．ファインチューニングはこの転移学習の一種である．モデルの重みを再学習して調整する手法であるが，ファインチューニングにも過学習の問題は度々付きまとう．Discriminator はいくつかのレイヤーで構成される．FreezeD はシンプルに Discriminator の数レイヤーを固定するという手法である．基本的にファインチューニングでは Discriminator と Generator の両方を再学習させるが，これが結果的に過学習を導いていると考え，再学習するレイヤーを一部に絞っている．FreezeD では Discriminator の入力側に近い数レイヤーを固定し，出力に近い数レイヤーのみをファインチューニングする．全体のファインチューニングと比べて，過学習が起こりづらいことが分かっており，学習を安定させる目的で FreezeD を用いる．

2.3.2 Adaptive Discriminator Augmentation

上でも述べた通り，過学習を防ぎ，良い結果を出すにはデータセットの数は非常に重要な要素である．Adaptive Discriminator Augmentation(ADA)は学習データの拡張を指し，T. Karras らによる論文[8]で提案された．

データセットの拡張とは，具体的には画像の左右，上下反転や，シフト，回転や，色の変換などいくつかの画像に対する操作である．どれも画像に少しの変化を加えることで，学習データをかさ増ししている．しかし単純な拡張には問題がある．データセット全体に回転を加える拡張を行った場合，生成画像にも回転した画像が現れるのだ．これを拡張の「漏れ」と呼ぶ．そのため拡張手法は，漏れを起こすことなく過学習に陥らない，適度な拡張の強度が求められる．ADA は学習の過程で動的に強度を調整することを提案している．データセットが求められる数より少ない場合，ADA を使うことで漏れのない拡張を適応する．その結果，生成画像の精度を向上させ，同時に過学習を防ぐことも可能となる．

2.3.3 GANSpace

E.Arkonen らによる GANSpace[9]は, GAN の生成画像を意図的に操作する方法である. GAN の Generator は, シード値によって対応する画像が生成される. そのため生成画像の細かな特徴, 例えば性別やライティングを滑らかに調整することはできない.

ただ StyleGAN のように, シード値や潜在コードに加え, 潜在空間を持っている GAN では生成画像の操作が可能である. GANSpace は, 潜在空間や特徴空間を PCA(主成分分析)により解析し, その調整を補助している.

StyleGAN の場合は, 512 次元の潜在空間を持っている. 各解像度の畳み込みで潜在空間を用いてスタイルを学習させる. 学習データにはそのデータの持つ分布, 画像空間が存在している. Generator の学習が進むにつれ潜在空間は画像空間の分布を学習していく. つまりこの潜在空間は画像空間とある程度は対応関係にあることが分かる. 潜在空間を分析することで, 512 次元の空間上で性別や向きなどを調整できる次元, またはベクトルを検出する. 見つけたベクトルの操作により, 生成画像の編集が可能になる. この潜在空間を仮に 2 次元空間上で表すと図 2.6 のようである.

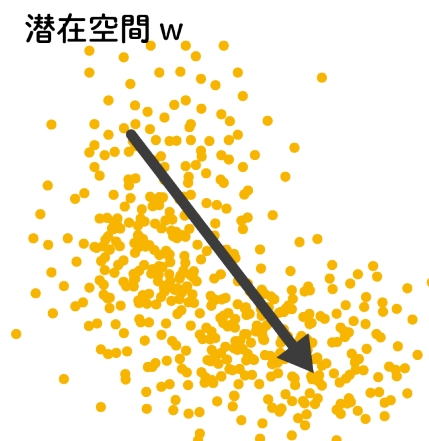


図 2.6 潜在空間内のベクトル

従来は生成画像を手動でラベル付けし, 潜在空間を分析する手法が主流であった. それに対して GANSpace は, ラベル付けなしで潜在空間を分析することが可能である. 学習済みモデルの潜在空間を PCA で分析し, 潜在空間の中で変化の大きい軸を見つける. そして画像を生成する際に, 見つけた軸方向に変化を加えることで, 生成画像の調整が可能になる. しかしこの手法では特定の意味成分を指定して調整することはできない. 従って最終的に求める軸方向を一つ一つ手探りで探すこととなる. 見つけた軸方向がそれぞれどのような意味を持つのかを生成画像から確認することで, 調整したいベクトルを探索する.

2.4 むすび

本章では, 本研究の関連技術について述べた. 画像生成手法として CycleGAN, U-GAT-IT, StyleGAN について, そしてそれらを補助する FreezeD と GAN Augmentation, GANSpace についてそれぞれ述べた.

第3章 検討手法

3.1 まえがき

本章では、GAN を用いて、二つの漫画間で画風変換を行う手法について述べる。

3.2 データセット

本研究では、日本の漫画を集めたデータセットである Manga109[10]を用いた。Manga109 はその名前にもあるように、109 冊の漫画作品から構成されている。その収容作品は 1980 年代から 2010 年代までの幅広いジャンルにわたっており、顔、人物、セリフなどのアノテーションがつけられている。このアノテーションの中に、登場人物の顔を囲むように設定されているものがあり、今回はそれを用いて顔画像の切り抜きを行った。このアノテーションの範囲は顎から眉上あたりの狭い範囲である。FFHQ[4]や CelebA などのデータセットで目にする顔画像の雰囲気になら近づけるため、切り抜く幅を左右上下少しずつ拡大する調整を行った。さらに切り抜いた画像の中から、正面を向いている画像を抽出した。これらの Manga109 に対する処理をデータクリーニングと呼ぶ。データクリーニングの際に取り除いた画像は、画質の低いもの、横を向いているもの、コマの切れ目で顔全体が収まっていないものなど学習の質を下げる画像である。

図 3.1(a)に用いたデータセットの例を、(b)にデータクリーニングの過程で省いた画像の例を示す。

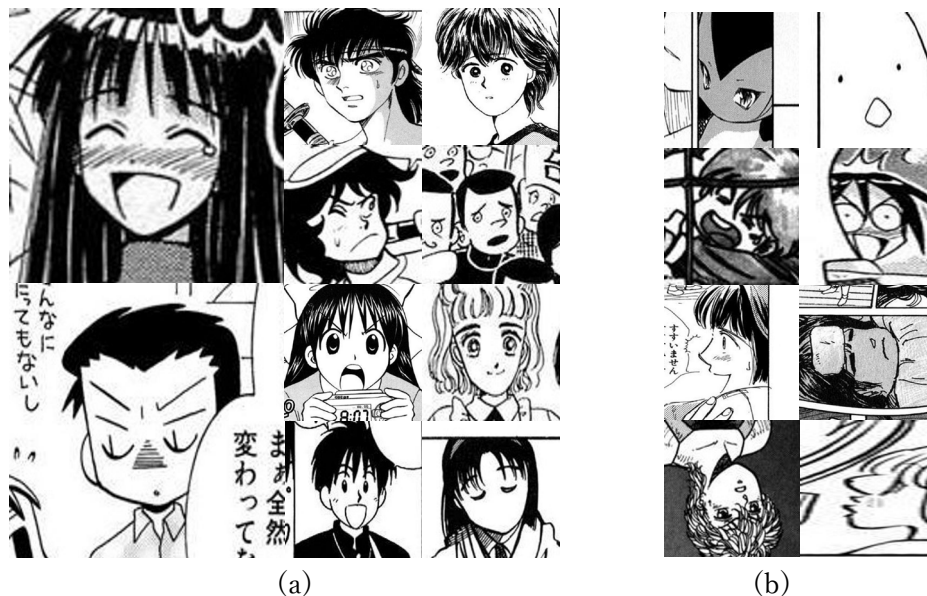


図 3.1 Manga109 のデータクリーニングの様子

(a) データクリーニング後の画像, (b) データクリーニングの際に省いた画像

3.3 Image-to-Image Translation を用いた画風変換

3.3.1 CycleGAN を用いた変換

データクリーニング後の Manga109 を用い、CycleGAN で二つの漫画間のスタイル変換を行う。CycleGAN は 2.2.2 で紹介したように、二つのデータセット間で各ドメインを変換する手法である。学習時間は 2 日程度であり、必要なデータセットはそれぞれ 1000 枚前後である。

今回は二つの漫画からそれぞれ顔画像 600 枚ほどを切り抜き、CycleGAN を実行した。ある画風で描かれた顔画像に、もう一方の漫画のドメイン、つまり画風を適応可能かの検討である。

3.3.2 U-GAT-IT を用いた変換

CycleGAN で扱う画像のスタイルは、基本的にテクスチャに限っている。そのため、形の変換を行うスタイルの変換には向いていない。U-GAT-IT はそれを克服するために、Attention モジュールや AdaLIN を用いている。これらの機能により、人の顔画像をアニメ風にするなど、形の変化を伴う画像のスタイル変換が可能である。

ここで U-GAT-IT の問題点を挙げるとすれば学習コストである。公式の実装では、学習データを合計で 3400 枚用意し、Nvidia の V100(32GB)で 4 日間学習を行っている。しかしながらそのような実装環境を整えるのは非常に困難である。今回は 1 枚の P100 で実行した。U-GAT-IT の処理を軽くして学習を行ったが、学習が収束するまで実行できなかった。iteration 数でみると、公式での学習の 1/15 ほどが限界で、それでも学習結果はある程度収束し、他の GAN と比べると十分な時間学習を行ったため、今回はその結果を用いて比較することとした。

3.4 StyleGAN を用いた生成

3.4.1 StyleGAN のモデルの学習

StyleGAN は上記の CycleGAN や U-GAT-IT と違い、画像のスタイル変換手法ではなく、シンプルな画像の生成手法の一つである。なので StyleGAN では、二つのドメイン間の変換を学習させるのではなく、まず二つのドメインを含めた画像生成モデルを学習させることとなる。その上で、異なるスタイルを混ぜる Style Mixing や、生成画像の細かな調整を行う潜在空間の操作を行うのである。そうすることで最終的には、画風の変換画像を得られることになる。

StyleGAN においてモデルを一から作るには、数万～数十万の画像、性能の良い複数の GPU で、1 週間以上学習を行う必要がある。しかし当然そのような大量の漫画データや、GPU を用意することはできない。そこで限られた環境下で StyleGAN2 を訓練するための補助的な手法として、転移学習、データセットの拡張、再学習時の一部レイヤーの重みを固定の三つを用いた。U-GAT-IT では講じる策もなく、学習をかなり早めに切り上げるこ

とになった。StyleGAN は本来 U-GAT-IT よりもさらに学習コストの高い GAN であるが、この転移学習などの手法により、かなり学習を進めることができた。

転移学習は、学習済みのモデルに対し、少ないデータセットで再学習させ、必要な学習コストを抑える手法である。再学習をさせる際、元のモデルのドメインと、再学習のモデルのドメインが近いほど、転移学習が安定して行える。今回は漫画の顔画像の学習を行うため、人間の顔画像を生成するモデルを、元の学習済みモデルとして用いた。StyleGAN2 で学習された人の顔画像の生成ではいくつかあり、ここでは FFHQ で学習されたモデルを使用した。

データセットの拡張は 2.3.2 で述べた手法である。左右反転や、回転など、いくつかの処理を加え、データセットの数を重増しを行う。Manga109 にデータクリーニングを行うと、全体で数万、一つの漫画について言えば数百から数千程度の顔画像しか集まらない。数万の画像を求める StyleGAN には不十分であるため、データセットの拡張で画像枚数を増やす。

そして再学習時の一部パラメータの固定には、2.3.1 で述べた FreezeD を用いる。FreezeD は再学習時の Discriminator の重みを部分的に固定することで、学習を安定させ、画像枚数が少ないことに起因するモデルの発散を防ぐ。

根本となる StyleGAN についても、改良版である StyleGAN2 を用いた。今回用いた学習手法を一つにまとめるとすれば、StyleGAN2+ADA+FreezeD と表せる。

3.4.2 生成画像の操作

GAN では画像を生成する際に、初期値としてランダムなシード値を入力することが多く、これを潜在コードと呼ぶ。学習させたい画像データの持つ分布、画像空間を考える。GAN はシード値ごとに生成画像が決まるが、単純なシード値を、画像空間に対応付けることはできない。StyleGAN ではこの画像空間をなるべく捉えるため、潜在コードだけでなく、潜在空間を用いている。学習を進めると、実際にこの潜在空間は画像空間とらえていくため、StyleGAN では良い結果を望むことができる。GAN の生成画像をある程度意図したものにしたとき、画像空間との相関が分かりづらい潜在コードでは細かな調整ができない。しかし画像空間との対応関係が分かり易い潜在空間では、その関係を分析することで、生成画像を調整できる。

StyleGAN の写像ネットワークは 8 層の全結合層で構成されており、ランダムな潜在コードを 512 次元の潜在空間へマッピングしている。人の画像データが持つ画像空間は、女性の画像が持つ画像空間、男性の画像の持つ画像空間などそれぞれが交わり全体の空間を成している。様々な特徴に対応する画像空間があり、それに対応した潜在空間がある。つまり操作したい特徴の画像空間に対応した潜在空間を見つけることが、生成画像を操作するポイントとなる。

ここで漫画の画風変換を行う場合を考える。あるモデルに二つの漫画のデータセットの学習をさせる。二つの漫画に応じた画像空間があり、学習したモデルは対応する潜在空間を

持つ。片方の漫画の潜在空間と、もう片方の漫画の潜在空間を把握することで、任意の画風の漫画画像の生成が可能になる。この潜在空間の分析には、2.3.3 の GANSpace を用いる。手作業のラベル付けのいない GANSpace では、生成画像の変化の大きい次元を見つけることができる。その中から二つの漫画の潜在空間を行き来できる次元を探すことで、漫画の画風変換が可能となる。

3.5 評価手法について

GAN のモデルの良し悪しを評価するには、モデルのパラメータではなく、生成画像を見る。学習に用いたデータの分布と、生成画像の分布の距離を測定する。二つの分布の距離を測る手法としては、KL divergence や、その改良手法である JS divergence, また Wasserstein 距離などがある。しかし単に距離を計算するだけでは、モデルの多様性を評価することができない。そこで用いられるのが Inception Network を用いた、Inception Score(IS)である。

Inception Network は、画像認識のモデルである。画像が猫である確率、犬である確率などを、それぞれ測定する。生成画像全体に対して Inception Network を適用することで、モデルの生成画像の認識のしやすさと、多様性を評価するのだ。IS はある程度、人の判断と一致することも報告されている[11]。ただ IS では学習データのことがあまり考慮されていない。そこで提案されたのが Fréchet Inception Distance(FID)[12]である。学習データに対しても Inception Network を適用し、学習データのもつ Inception モデルの出力分布を得る。そして Inception Network を通した学習データの分布と、生成画像の分布の Fréchet 距離を Score として算出する手法である。

他の評価手法には、U-GAT-IT で使われていた KID[13]もある。しかしここでは、StyleGAN などで主に使われ、CycleGAN でも用いられることの多い FID で評価することにする。FID には Inception Network に依存している点などの問題はあるが、既存研究などと比較する点で有用でと考えられる。

3.6 むすび

本章では、本研究で検討する GAN を持ちいた漫画の画風変換の手法について述べた。

第4章 実験

4.1 まえがき

本章では，前章で提案した手法を用いて，漫画画像の生成及び，その画風変換を行う．まず使用するデータセットについて述べ，その後それぞれの GAN における生成画像を評価する．

4.2 実験の概要

データセットについては 3.3 で述べたようにデータクリーニング済みの Manga109 を用いた．Manga109 には多くの漫画のデータが含まれており，画風の変換を行う際には，二人の漫画家による漫画画像データを重点的に用いた．一人は新沢基栄氏であり，Manga109 に含まれている「ハイスクール!奇面組」と「ボクはしたたか君」の画像データを用いた．そしてもう一人は「しまってこうぜ!」が収録されている吉森みき男氏である．ここでは両氏の漫画をそれぞれ画風 A, B とする．Manga109 に含まれているデータ数が多かったこと，絵柄が安定しており，キャラクターを認識しやすいなどの点を考慮して両氏を選んだ．図 4.1 に学習に用いたデータセットを示す．

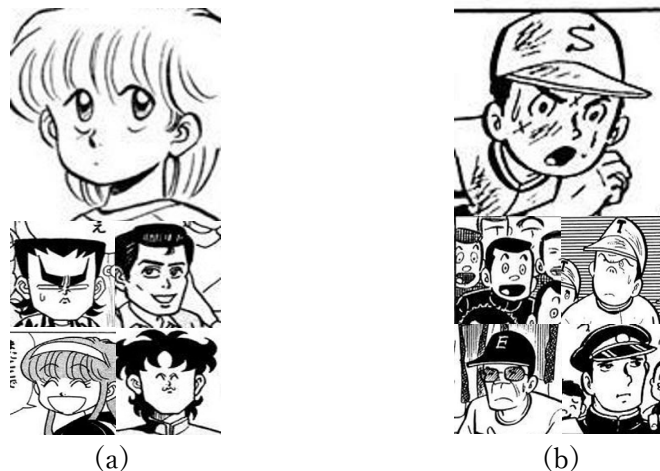


図 4.1 変換を試みた二つ画像セットの例

(a) 新沢基栄氏(画風 A)の漫画画像，(b)吉森みき男氏(画風 B)の漫画画像

CycleGAN, U-GAT-IT, StyleGAN の三つの GAN についてそれぞれ両氏の漫画データを学習させ，それぞれの結果を画像と評価指数を用いて比較する．先に述べておくと，CycleGAN と U-GAT-IT では望んだほどの結果は得られなかった．そして StyleGAN では前述の二つの GAN よりはるかに見た目の良い画像，評価の良い数値が出る結果となった．まずはこのことをそれぞれの生成画像と数値による評価で確認し，その後 CycleGAN と U-GAT-IT の問題点について述べる．StyleGAN についてはその画像変換の試みも含め，より詳しい実験を 4.4 に記載する．

4.3 各 GAN の結果と I2I-Translation の問題点

4.3.1 各 GAN の生成画像とその評価

それぞれの GAN で生成した画像を下に示す．CycleGAN と U-GAT-IT は，画像変換の手法であるため，双方向の変換の結果を載せる．StyleGAN については，画風の変換には潜在空間の操作が必要である．そのためここでは，それぞれの漫画に対応していると推測される潜在空間の生成画像を載せる．

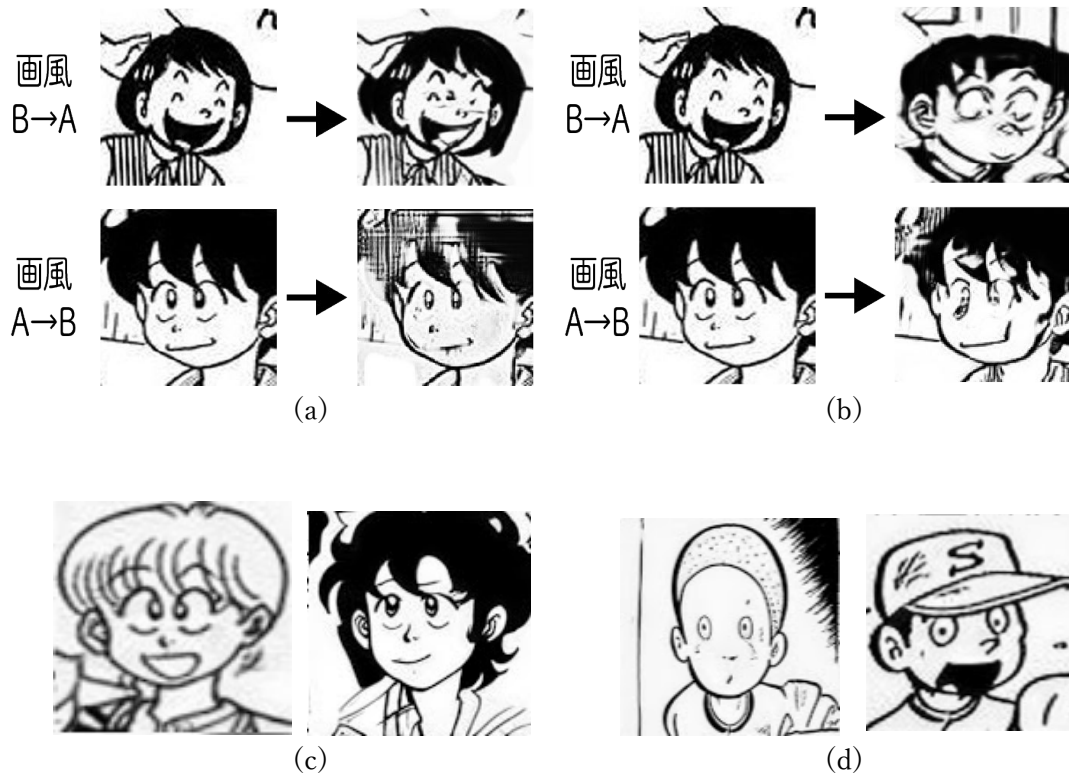


図 4.2 各 GAN の生成画像例

(a) CycleGAN の生成画像, (b) U-GAT-IT の生成画像

(c) StyleGAN の生成画像(画風 A), (d) StyleGAN の生成画像(画風 B)

CycleGAN は，変換をほとんど行えていない．よく観察すれば，多少の適切な変化がみられるものの，画風変換とは呼ぶことができない．U-GAT-IT は CycleGAN と比べれば変化という意味では大きいものの，絵が崩れでしまい、こちらも同様に画風の変換はできていない．StyleGAN は画風の変換を行っているわけではないため，単純に前の二つと比較することはできない．しかしその生成画像のクオリティは圧倒的に高く，そのことは表 4.1 の評価値でも確認できる．CycleGAN と U-GAT-IT においては，スタイル変換した画像と，ターゲットドメインの FID を計測した．StyleGAN では変換元も変換先の画像も生成できるため，変換先と仮定した方面風 B の漫画画像と，そのデータセットの FID を測定した．

表 4.1 各 GAN における FID

	CycleGAN	U-GAT-IT	StyleGAN
FID	61.01	79.03	20.17

この結果から漫画の画風変換，もとい漫画画像の生成には StyleGAN が適していることが分かった．それぞれの GAN の詳しい結果，そして StyleGAN の画風変換については次に記述する．

4.3.2 CycleGAN の結果と考察

CycleGAN はペア画像を用いずとも，比較的簡単に画像のスタイル変換が行える手法である．今回は 4.3.1 に記載の通り，良い結果は得られなかった．学習が妥当に進んでいたかを検討するために，一般的に知られている馬とシマウマを変換する場合と学習状況を比較した．その際の両者の学習パラメータを表 4.2 に示す．馬とシマウマの変換を horse2zebra，漫画の画風変換を manga2manga と表記する．

表 4.2 CycleGAN の学習パラメータの比較

	画像の総数	チャンネル数
manga2manga	1293	1
horse2zebra	2401	3

それぞれの epoch ごとの生成画像を図 4.3 に示す．

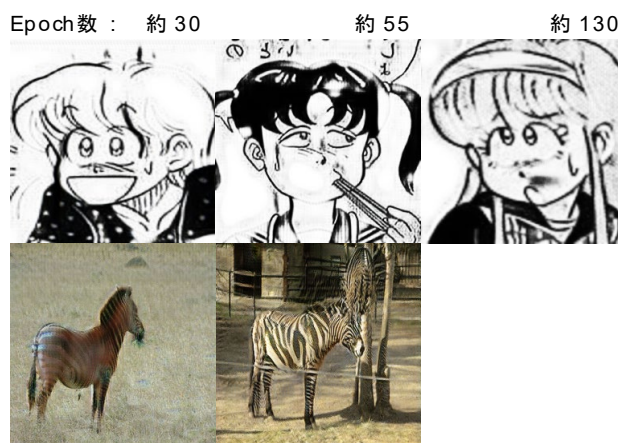


図 4.3 二つのモデルの学習段階ごとの生成画像の比較

表 4.2 にある通り，画像の総数が少ないため，学習が十分に行われないと考えられる．しかし図 4.3 を見ると，horse2zebra が epoch55 にしてかなり学習が進んでいるのに対し，manga2manga は全く学習が進んでいない様子である．Epoch130 でも様子は変わらず，画

像枚数に関わらず CycleGAN での漫画の画風変換が難しいことが分かる。CycleGAN の変換はテクスチャがメインであり、難しいことは予想されたが、図 4.4 の損失関数からその原因をより詳しく探る。

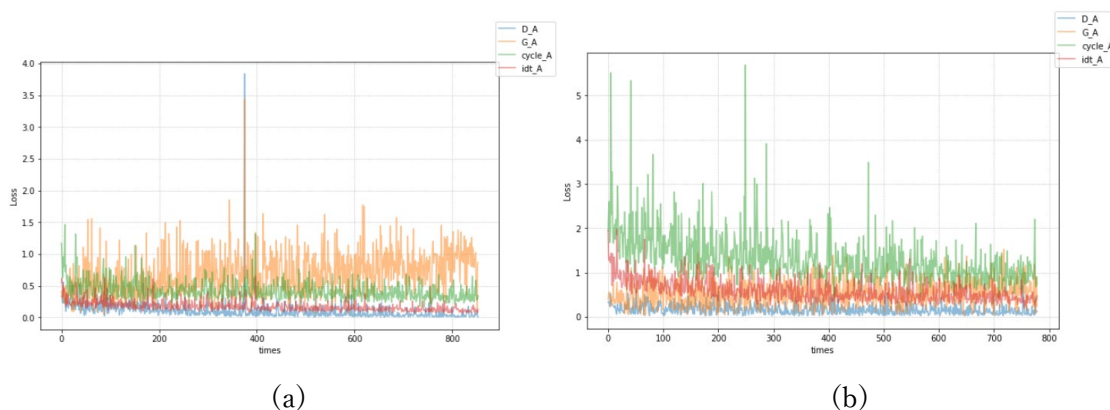


図 4.4 CycleGAN の損失関数の比較

(a) manga2manga の損失関数, (b) horse2zebra の損失関数

図 4.4(a)から分かる通り, manga2manga では損失関数がかなり早い段階で収束している。このことから学習が進んでいないのではなく、漫画同士のスタイルの差を認識できていないということが推測される。テクスチャの側面から見れば、どちらも白黒のイラストである点に変わりはなく、CycleGAN での変換は難しいと判明した。

4.3.3 U-GAT-IT の結果と考察

U-GAT-IT は CycleGAN と比べるとテクスチャに限らない変換を行える。しかし図 4.2 にあるとおり、望んだほど良い結果は得られなかった。学習時の各 iteration での生成画像を図 4.5 に示す。

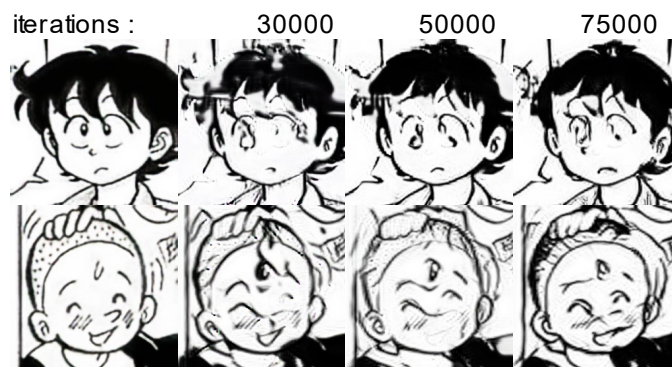


図 4.5 U-GAT-IT の学習段階ごとの生成画像

学習コストが高いため、十分な学習時間と、画像データを用意で着なかったことは確かである。しかしながらたとえ学習が十分に行われた場合でも、図 4.5 からはあまり良い結果を

想像することはできない. CycleGAN と比べれば変化は大きい, 画風変換と呼ぶには至らない. それ以前にキャラクタの造形に関する問題が大きいことが見て取れる. FID が良くなかった要因も, この画像自体の認識に難しさであると考えられる.

4.4 StyleGAN での画風変換

4.4.1 潜在空間の検討手法

図 4.2 の画像から, StyleGAN が CycleGAN, U-GAT-IT と比べ見栄えの良い画像を生成可能であると確認できる. しかし上でも述べた通り, StyleGAN は他二つの GAN と根本的な目的が異なる. StyleGAN はあくまで画像生成手法であり, 画像変換の実装には一工夫必要となる. 漫画画像を学習させた StyleGAN のモデルは, 学習させた漫画画像の生成を実行する. 画風 A, 画風 B の画像をそれぞれランダムなシード値から生成できるが, 入力画像に対しての変換や, 中間の画風の生成は StyleGAN 単体では困難である. そこで今回は 2.2.3 で述べた GANSpace を使用し, StyleGAN の生成画像の調整を実装した. GANSpace は, 潜在空間の持つ主要な軸を抽出する. 抽出した各軸は生成画像の特徴を操作可能で, その中から画風変換が行える軸を一つ一つ探していく. 図 4.6 に GANSpace で取得できた, 主要な軸の変化の様子を示す. 画像の左に軸の持つ特徴名を記載した. GANSpace では同一の特徴を持つ軸が複数見つかる. 画風変換の確認可能な軸も複数存在し, 図 4.7 にその内いくつかの例を示す.

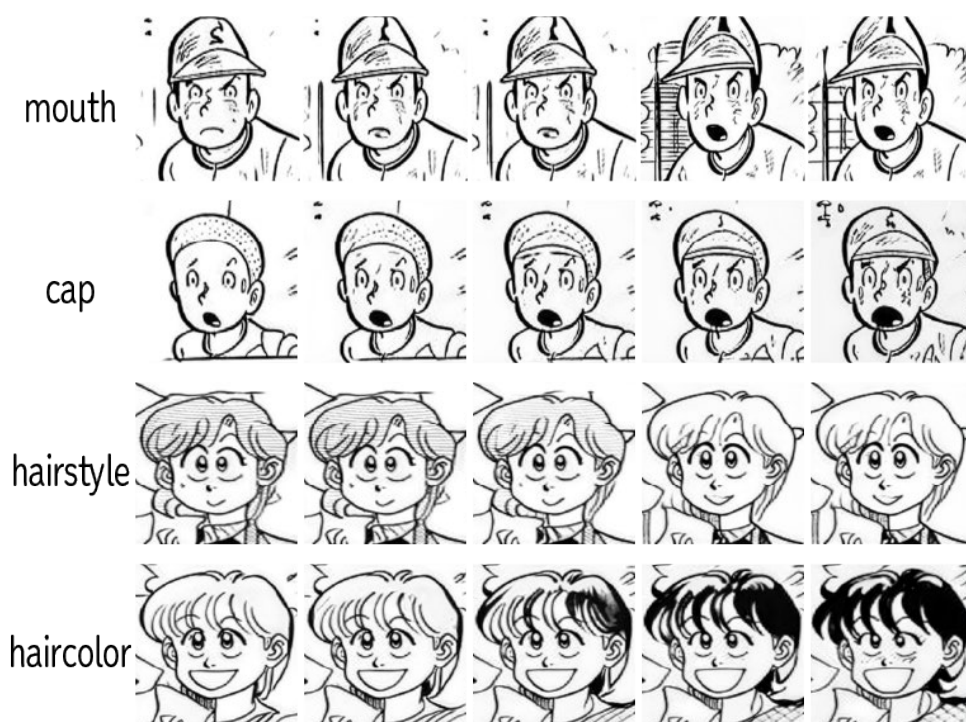


図 4.6 検出した軸方向の変化の様子



図 4.7 GANSpace による画風変換の結果

4.4.2 潜在空間の考察

図 4.6 で、口や帽子、髪色などが変化する様子を確認できる。上段の口の変化する画像では、同時に帽子の向きも変化している。他の画像も同じように、一つの軸方向に複数の特徴空間をまたいでいることが分かる。軸を探す際は、少しずつ変数を変化させ、適切なものを探した。特徴が明白な軸を見つけることは難しく、軸の探索には 1 時間程度を要した。図 4.7 の画風変換を見る。画風 B の学習データには男性キャラクターが多いため、画風の変換と同時に性別も変化している。次に、図 4.8 に 14488 枚の漫画データを学習させた場合の GANSpace の実装結果を示す。

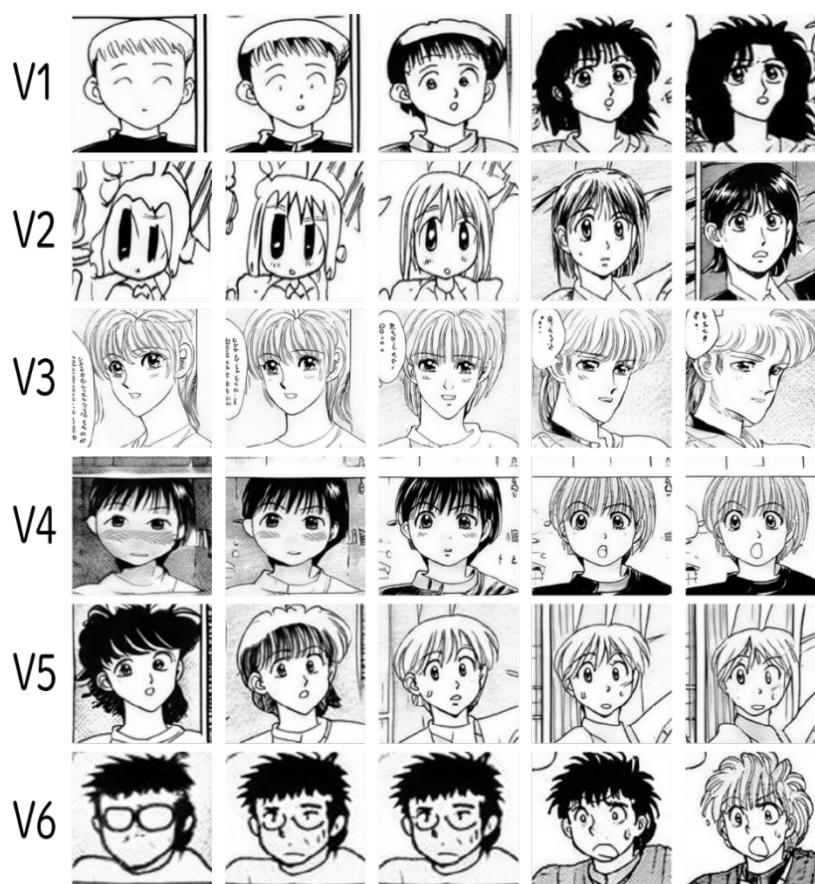


図 4.8 多数の漫画を学習させた際の GANSpace の結果

このモデルは、Manga109 のうち 31 の漫画、合計 14488 枚の画像を用いて StyleGAN2+FreezeD+ADA により学習を行った。二つの漫画と同様に画風変換が可能かを確かめることが目的である。生成画像の FID は 18.02 であり、学習は十分に収束している。二つの漫画で学習を行った場合に比べ画像枚数が多く、スコアに向上が見られる。このモデルに対しても GANSpace を用いて同様の方法で軸を探した。多くの漫画を学習した結果、変化する特徴が多くなり、様々な変化方向が検出された。V1～V6 を見ると、V3 は顔の向き、V4 は色と表情の変化と捉えられる。しかし V1、V2 のように画像の変化が大きく定義づけが難しい軸も多く見られた。キャラクターの画像を見ると、いくつかの画風が混ざり合っており、目と輪郭と髪型で別の画風が見受けられる。さらにデータセットの平均をとったような画風が多く見られ、どの漫画家風と定義づけることが困難であった。潜在空間は 512 次元であり、空間内でそれぞれの特徴が成す部分空間が混ざり合っている。学習した特徴が増えれば GANSpace で取り出した軸がまたぐ特徴の空間が増える。二つの画風を学習させた場合であればあまり多くの特徴をまたぐことはない。しかし 31 もの漫画を学習させた場合は三つ以上の画風をまたぐことは頻繁にあると推測される。三つ以上画風をまたぐと、変化が大きく、軸の定義づけが困難である。複雑な変化のある軸は、直観的な生成画像の操作が行えない点が問題となる。生成画像をより直観的に整理するため、複雑な潜在空間を一度きれいに分解できるような処理が求められると考えられる。

4.5 むすび

本章では、各提案手法の実行結果を確認した。そしてその評価を示し、それに対する考察を述べた。

第5章 結論と今後の課題

5.1 結論

本研究では、GAN を用いた漫画の画風変換手法をいくつか検討した。三つの GAN を比較した研究結果から、StyleGAN による漫画の画風変換が有効であることが分かった。

今回の学習結果と Metface[8]などの結果から推測するに、1500 枚程度の画像があれば十分に画風を学習できると考える。1500 枚のきれいな画像を集めるには、おおよそ漫画 3 冊分が必要である。つまり現状では、3 巻以上の漫画をスキャンし、正面の顔部分を切り抜く。そしてデータ拡張やファインチューニングを用いて StyleGAN のモデルを学習させ、GANSpace で潜在空間を操作を行う。そうしてやっと任意の画風の漫画画像を生成できるのだ。これはアニメ調のイラストでも同様であり、一つのキャラクターなら数百枚程度、画風なら 1500 枚程度必要である。

5.2 今後の課題

今回の画風変換は必要な工程も多く、精度も高いとは言えない。漫画分野で機械学習を応用していくには、より大きなデータセットが必要である。またこれはよく言われていることであるが、FID は多様化する GAN のモデルに対し、評価基準としてあまり最適でないことも課題である。理由は FID が Inception Network に依存していることである。Inception Network では漫画の画風を区別することはできないため、評価として最適ではないのだ。

そして潜在空間の操作についてもは、まだ発展の余地がある。GANSpace や、今回実装しなかったが SeFa[14]のようなラベルなしの手法は非常に便利である。しかし使える場面は限られている。生成画像や、入力した画像の画像空間から、潜在空間の逆算が可能であれば、より使える場面は増えるだろう。

StyleGAN に対して大量の画風を学習させると、互いの画風が 512 次元もある潜在空間上で混ざり合い、区別をつけられない。そのため描いた絵を好きな画風に寄せるとなれば、それぞれの画風のモデルを用意し、Style Mixing によって混ぜる手法が想定される。しかしこれは工数が多く、今回の手法のほうがまだ実践に近いと考えられる。今回の手法の延長線上で、任意の画風の漫画を用意に生成する場合、大量のモデルを行き来可能な手法か、潜在空間のより細かな分類手法が必要であると考えられる。

謝辞

本研究の実験環境や，研究の方向性について丁寧かつ熱心なご指導をいただいた渡辺教授に心から感謝いたします。

また，日頃からご意見やアドバイスをくださった研究室の皆様にご礼申し上げます。

最後に，私をここまで育ててくださった家族に感謝いたします。

参考文献

- [1] I.J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, "Generative Adversarial Nets", arXiv:1406.2661, Jun, 2014,
- [2] J. Zhu, T. Park, P. Isola, A. A. Efros, "Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks", arXiv:1703.10593, Mar 30, 2017
- [3] J. Kim, M. Kim, H. Kang, K. Lee, "U-GAT-IT: Unsupervised Generative Attentional Networks with Adaptive Layer-Instance Normalization for Image-to-Image Translation", arXiv:1907.10830, Jul 25, 2019
- [4] T. Karras, S. Laine, T. Aila, "A Style-Based Generator Architecture for Generative Adversarial Networks", arXiv:1812.04948, Dec 12, 2018
- [5] T. Karras, T. Aila, S. Laine, and J. Lehtinen. Progressive growing of GANs for improved quality, stability, and variation. CoRR, abs/1710.10196, 2017
- [6] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, T. Aila, "Analyzing and Improving the Image Quality of StyleGAN", arXiv:1912.04958, Dec 3, 2019
- [7] S. Mo, M. Cho, J. Shin, "Freeze the Discriminator: a Simple Baseline for Fine-Tuning GANs", arXiv:2002.10964, Feb 25, 2020
- [8] C. Bowles, L. Chen, R. Guerrero, P. Bentley, R. Gunn, A. Hammers, D. A. Dickie, M. V. Hernández, J. Wardlaw, D. Rueckert, "GAN Augmentation: Augmenting Training Data using Generative Adversarial Networks", arXiv:1810.10863, Oct 25, 2018
- [9] E. Härkönen, A. Hertzmann, J. Lehtinen, S. Paris, "GANSpace: Discovering Interpretable GAN Controls", arXiv:2004.02546, 6 Apr 2020
- [10] K. Aizawa, A. Fujimoto, A. Otsubo, T. Ogawa, Y. Matsui, K. Tsubota, H. Ikuta, "Building a Manga Dataset "Manga109" with Annotations for Multimedia Applications", arXiv:2005.04425, May 9, 2020
- [11] T. Salimans, I. J. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen. Improved techniques for training GANs. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, Advances in Neural Information Processing Systems 29, pages 2234–2242, 2016
- [12] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, S. Hochreiter, "GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium", arXiv:1706.08500, Jun 26, 2017
- [13] M. Bińkowski, D.J. Sutherland, M. Arbel, A. Gretton, "Demystifying MMD GANs", arXiv:1801.01401, Jan 4, 2018

- [14]Y. Shen, B. Zhou, “Closed-Form Factorization of Latent Semantics in GANs”,
arXiv:2007.06600, Jul 13, 2020

図一覧

図 2.1	Generative Adversarial Networks(GAN)の構造	3
図 2.2	CycleGAN の学習フレームワーク	4
図 2.3	U-GAT-IT の Generator の構造	5
図 2.4	StyleGAN2 の構造	6
図 2.5	転移学習の手法	7
図 2.6	潜在空間内のベクトル	8
図 3.1	Manga109 のデータクリーニングの様子	9
図 4.1	変換を試みた二つ画像セットの例	14
図 4.2	各 GAN の生成画像例	15
図 4.3	二つのモデルの学習段階ごとの生成画像の比較	16
図 4.4	CycleGAN の損失関数の比較	17
図 4.5	U-GAT-IT の学習段階ごとの生成画像	17
図 4.6	検出した軸方向の変化の様子	18
図 4.7	GANSpace による画風変換の結果	19
図 4.8	多数の漫画を学習させた際の GANSpace の結果	19

表一覧

表 4.1	各 GAN における FID	16
表 4.2	CycleGAN の学習パラメータの比較	16