＜ 修 士 論 文 ＞

画像生成モデルの適応的パラメータ入れ替えを用いたイラストの画風変換

滋 賀 大 学 大 学 院
デ ー タ サ イ エ ン ス 研 究 科
デ ー タ サ イ エ ン ス 専 攻

修了年度：２０２２年度
学籍番号：6021124
氏　　名：中江　剛之
指導教員：飯山　将晃
提出年月日：２０２３年１月 11 日

目 次

第 1 章　はじめに **2**

第 2 章　関連研究 **5**
2.1　敵対的学習 (GAN) 5
2.2　画像変換 5
2.3　StyleGAN2 の学習済みパラメータの入れ替え 6

第 3 章　深層学習による画風変換 **8**
3.1　画像生成ネットワーク StyleGAN と StyleGAN2 8
3.2　2 つのドメインの StyleGAN と Swapping 8
3.3　StyleGAN2 を用いた画像変換 11
3.3.1　生成器からの画像の復元 11
3.3.2　画像生成 12

第 4 章　提案手法 **13**
4.1　提案手法の概要 13
4.2　パラメータ比率の探索 14

第 5 章　実験 **16**
5.1　データセット 16
5.2　モデルの学習・ハイパーパラメータ 16
5.3　ユーザースタディ 17
5.4　アイデンティティの評価の議論 20
5.5　画風の評価の議論 21
5.6　合成品質の評価の議論 23
5.7　今後の課題 24
5.7.1　損失関数の再構築 24
5.7.2　一部のアイデンティティの損失 25

第 6 章　結論 **27**

謝辞 **28**

参考文献 **28**

付録 **33**

第1章　はじめに

画像変換は、画像のドメインを別の画像のドメインに変換する手法である [1]。例えば、実世界の風景の写真を入力として、その構造を保ったままモネが描いたような風景画のように変換する処理が挙げられる。これは実世界の風景写真というドメインを画像変換によってモネの描いた風景画というドメインに変換することであり、画風の変換と見なせる。近年ではこの画像変換を応用して、表情を変換するアプリケーション[1]や人間の顔をアニメ風に変換するアプリケーション[2] [2] などが登場し、エンターテインメント分野での活用が進んでいる。この活用は実世界の画像だけに限った話ではなく、イラストや漫画等にも需要がある [3,4]。例えば下書きのラフを仕上げの状態まで変換 [5] することで、イラストを描く手間を省いたり、描く際の参考資料にすること[3]も可能になる。

イラストや漫画を構成する要素として、画風とアイデンティティの要素が存在する。画風とは画像やイラスト内の色合いや線のタッチなどから構成されている表現である。例えば、図 1.1 の少年漫画と少女漫画を比較すると、目の形や髪の毛の書き方・絵のタッチの質感など様々な面で表現が異なっている。また図 1.2 のような同じ少年漫画同士でも同様のことが言え、作者や作品によって書き方の傾向が異なる。一方で人物やイラストにおけるアイデンティティとは個人を特定するための特徴の集合である。例えば、図 1.3 の二つの画像は同じ作品の男性キャラクターであるが、左右では同一人物とは言えない。この 2 枚の画像を比べると短髪と長髪という大きな違いがあり、画像の人物の特徴となっている。

図 1.1: 少年漫画と少女漫画の例 (左: © 咲 香里, 右: © あゆみ ゆい)

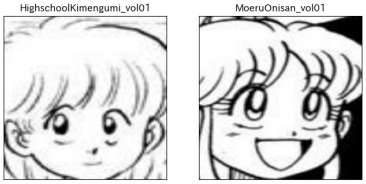


図 1.2: 同じ少年漫画での例 (左: © 新沢 基栄, 右: © 佐藤 正)

図 1.3: 同じ漫画のキャラクターの違い © 愛田 真夕美

[1] https://www.faceapp.com/
[2] https://toonify.photos/
[3] https://illustmimic.com/

人物画の画像変換においては、画像に映る顔の特徴を維持したまま画風を変換することでアイデンティティを維持することが重要になる。例えば顔写真から肖像画への画像変換では、被写体の人物が肖像画風になることを目的としているため、顔の特徴を維持したまま変換することが重要である。仮に変換時に被写体の顔の特徴が変わると描写されている人物が変化し、肖像画風にするという目的が達成されない。このため同様にイラスト同士での画像変換でも、画風とアイデンティティの要素が重要となる。

画像変換の従来手法 [1,6] では、人物の顔を変換する際、変換対象となった人物の顔が持つ髪の色や目の形状・表情などの特徴が変わり、アイデンティティを喪失する問題がある。例えば図 1.4 の左の画像は短髪の黒髪の少年の顔画像だが、右の画像はその髪の毛が伸びておりアイデンティティが失われている。この問題の対応として多くの顔画像を教師データとした学習 [6] がある。

図 1.4: アイデンティティを損失している画像変換の例 © 愛田 真夕美

しかしこの対応策を漫画やイラストに適用するのは難しい。多くの画像を使う方法に関しては、漫画やイラストは個人により画風が異なるため描ける人が限られている。これは画風を模倣することで解決するが人による手間がかかるためデータ数を増やしにくい。加えて作品によって作者が既に亡くなり作風のデータをこれ以上増やせない場合もある。他にもイラストなどの顔は作者によって、図 1.5 のように顔の形状・目の形状・表情の表現法等が人間の顔と比べて異なる部分が多いため、統一したサンプルが集まりにくいといった問題点もある。このため一般的な人間の顔画像と違い取得できるデータ数が限られているイラストなどでは、少ないサンプル数でアイデンティティを維持しつつ画風を変換する必要がある。

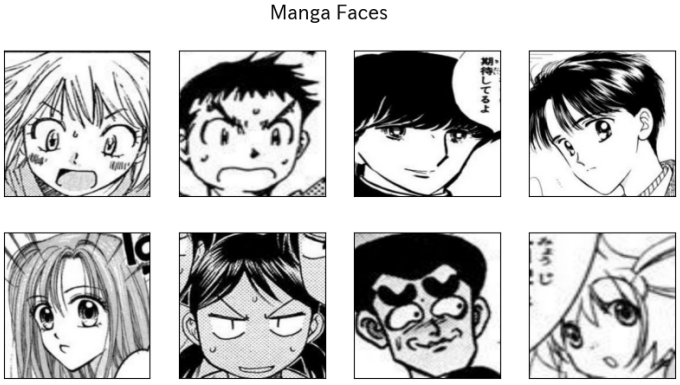


図 1.5: 様々な漫画の顔 © 咲 香里 © 愛田 真夕美 © あゆみ ゆい © 浅月 舞 © 新沢 基栄 © 桜野 みねね

少量の学習データを用いた画像変換として、画像生成モデルである StyleGAN2 [7] のパラメータを入れ替えることで画像変換を実現する手法 [2] がある。この手法は StyleGAN2 の画像生成をする多くの層からなる生成器の構造を利用した画像変換である。変換したい画像の生成器を二つ用意し、この生成器の層のパラメータを入れ替えることでそれぞれの生成画像の特徴を引き継いだ変換画像を生成できる。例えば少年漫画を少女漫画に変換する場合、少年漫画の画像生成器と少女漫画の画像生成器のパラメータを、層ごとに入れ替えることで、少年漫画と少女漫画の特徴が混ざった画像を生成することができる。この層の入れ替えでは入れ替える層に応じて生成画像の特徴の強度を調整することができる。この方法では、少ない教師データでアイデンティティを維持しつつ画風を変換することは出来るが、図 1.6 の様に画像ごとにアイデンティティを維持しつつ画風を変換できる最適な層の入れ替え位置が異なるという問題がある。このため多くの画像を変換する際にパラメータを入れ替える層の位置を統一すると、画像によってアイデンティティが崩壊したり、画風が変わらない問題が発生する。

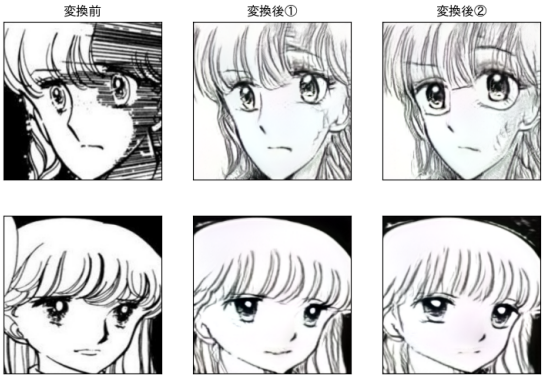


図 1.6: 層の入れ替え位置によってアイデンティティが変わる例 (© 愛田 真夕美)
上部の変換 1 と変換 2 ではメガネが付与されるが、下部では変換 1 と変換 2 では目が少しだけしか変わらない

本研究では StyleGAN2 のパラメータを入れ替える方法を用いて、画像のアイデンティティを維持しつつ、少量の画像での画風変換を行う手法を提案する。提案手法では、グリッドサーチを用いてパラメータを入れ替える比率を自動で調整し、各画像に最適な画風の変換を見つける。

第2章　関連研究

2.1　敵対的学習 (GAN)

敵対的学習 (GAN) [8] は、潜在変数と呼ばれる乱数の集合からサンプルを生成する生成器と、入力されたサンプルが本物か偽物かを分類する識別器の 2 つを用い、お互いを競い合わせ学習する手法である。これにより学習するデータセットの分布に従うようなサンプルを生成できる。この手法を用いた画像生成は、生成器が潜在変数から画像を生成し、識別器が入力する本物の画像と生成画像の識別の真偽性を競い合わせることで、精巧な画像を生成出来るのが特徴である。

GAN を用いた様々な画像生成手法が提案されている。DCGAN [9] は畳み込みニューラルネットワークを用いて生成画像の品質を上げることに成功した。しかし DCGAN のような生成器の構造は高解像度の画像生成は難しく、学習の安定性の課題が存在する。これに対して Prooressive GAN [10] では、画像生成の学習を解像度ごとに段階化することで問題に対処する。初期段階では低解像度の学習から始め、低解像での画像生成が安定した段階で、学習する解像度を上げる。この過程をくり返し高解像度の画像生成に成功した。そこから発展した StyleGAN [11] では各解像度の画像の生成過程で AdaIN [12] を用いることで、高品質な画像を生成することが可能になる。さらに画像を生成する元となる潜在変数を Mapping Network に入力することで、潜在変数の値を変動させることによる画像操作が直感的になり画像合成の可能性が広がった。しかし後に生成時に水泡状のアーチファクトが出現する課題が出現し、その課題を改良した StyleGAN2 [7] が提案された。この手法はアーチファクトの削除だけでなく、潜在変数の変動による画像の変化が滑らかになり、画像操作の柔軟性が改良された。

本研究では StyleGAN2 のアーキテクチャをベースに画像変換を行う。また学習も StyleGAN2 で提案された方法と同じようにして画像の生成を行う。

2.2　画像変換

深層学習を用いた画像変換は、変換元となるソースドメインと呼ばれる画像群と、変換先であるターゲットドメインと呼ばれる画像群を、深層学習を使ったネットワークと敵対的学習を用いて変換する手法である。例えば、ソースドメインと呼ばれる画像群 A と、ターゲットドメインと呼ばれる画像群 B のデータセットがあり、このドメイン間の関係を変換器のネットワークを用いて変換する。そして本物の画像と変換画像を識別器に入力することで敵対的学習を行い、変換のマッピング関数を学習する。このような変換手法は、ターゲットドメインの画像をソースドメインに変換することも可能である。

pix2pix [13] ではペアのある画像同士の変換に成功している。pix2pix は条件付き敵対生成ネットワーク (cGAN [14]) の仕組みを利用している。しかしこの手法は色が塗られていない靴と色が塗られている靴のような、ペアのある画像群を教師データとして用いる必要がありデータの収集には手間がかかる。この課題に対処した手法として CycleGAN [1] がある。CycleGAN は一度ドメイン A からドメイン B に変換した画像を、A に再変換したものが元の画像と同一となるような Cycle Consistency Loss を追加することで、ペアのない画像同士でも変換を可能にした。しかし CycleGAN は人間の顔からアニメ顔の変換ような、輪郭や目などの形状を含めた変換では低い性能を示している。この課題を解決する手法は多く提案されている [15, 16] が、本研究で最も関連する研究は U-GAT-IT [6] である。U-GAT-IT は画像を変換するモデルと識別器だけでなく、さらに補助分類器を追加することで形状変換に成功した。しかしこの手法は人間の顔からアニメ画像に変換するのに多くの枚数の画像を使用し学習している。少量の枚数で学習を行うと過学習を起こし、変換時に変換先の画像群の画像と似たものを出力する問題がある。

これらの画像変換には様々な応用研究が存在する。アイデンティティを維持しつつ変換を行う研究としては、Face-ID-GAN [17] がある。アイデンティティを維持するために、さらに人間の顔から個人を特定する分類器を導入し、人間の個人を特定するラベルや、キーポイントを追加情報として学習に組み込むことで変換に成功した。これはイラストと異なり様々なラベルが教師データとして追加されているため、ラベルが付与されにくいイラスト同士の変換では適用できない。本研究に関連する MangaGAN [18] では、人間の顔から特定の作品の漫画の顔に変換することに成功した。この研究では顔のパーツごとに様々な変換器・生成器を用意することで人間の顔から漫画画像へのアイデンティティを保った変換を可能にした。しかしこの手法は、顔をパーツごとに分ける必要があり、変換先の画像群によっては新たにパーツを分けラベル付けを行う必要があるためデータの作成に大きな手間がかかる。また漢字のフォントを変換する研究 [19] では、変換対象の漢字の構造を抽出した後、その構造をフォントに合わせた書体に変え、フォントを仕上げるといった 3 段階のモデル構造によって変換を行っている。この研究では書体の幹となる部分を一度抽出してから漢字に合わせた変換を行うため、他分野へ応用するために幹のような構造のデータを取得する必要がある。漢字のフォントを変換する研究は他にもあるが [20, 21]、これらも漢字のフォントの変換に特化した研究のためイラストへの応用は難しい。

このためこれらの研究では、データ数が少なくラベルやアノテーションが少ないようなイラスト同士でアイデンティティを維持する変換は難しい。CycleGAN や U-GAT-IT の画像変換では、イラスト同士の変換時にアイデンティティを維持せずに変換を行う問題がある。この問題が発生する原因は変換器の学習時に変換先のドメインの分布に出力を合わせるためである。

2.3 StyleGAN2 の学習済みパラメータの入れ替え

StyleGAN2 [7] は画像を生成する手法であるが、生成器のパラメータの一部を、別データで再学習した別の生成器のパラメータと入れ替える [2] ことで生成画像の変換が可能になる。この手法の場合、StyleGAN2 の生成モデルの構造とパラメータの関係性を利用す

ることで、画像の構造を決める役割とスタイルの強度を決める役割の二つを手動で調整できるため、適切に調整することでアイデンティティを維持しつつ画風を変えることが可能になる。また StyleGAN2 の再学習時に生成される画像が徐々に変わる現象を応用した変換手法もある。これは Stylegan2 が再学習前に生成できた画像が徐々に変わり最終的に生成不可となる前に学習を打ち止めると、再学習前に生成した画像の特徴と、再学習時に学習した画像の特徴を引き継いだ画像を生成できる仕組みを応用したものである。

この StyleGAN2 の再学習を応用した画像変換の研究 [22] も存在する。例えば AgileGAN [23] では、再学習の早期停止と指定した画像を生成する invert をエンコーダーとすることによって変換の性能を改善・高速化し、顔画像の変換を少ないサンプルで早く変換することを可能にした。他にも再学習時に使用する学習済みモデルの低解像度の層のパラメータをフリーズする [24] ことで、パラメータの入れ替え時に発生する構造間の矛盾を解消し、画像変換の性能が向上した。しかし再学習による変換を行うこれらの研究はアイデンティティの面には焦点を当てず、変換時の画風の調整によってはアイデンティティを失う可能性もある。またこれらの画像変換では画像によって画風とアイデンティティの変化が異なるため、手動で重みを入れ替えるだけでは全ての画像を最適に変換するのは難しい。

本研究では、再学習した StyleGAN2 の生成器の各解像度の学習済みパラメータを入れ替えることによって、少量データ同士の変換を可能にする。さらにアイデンティティを維持しつつ画風を変換できる最適な比率の探索を画像ごとに行い、多くの画像で最適な変換の発見を目的とする。また本研究の手法は学習時に顔パーツを分離するような手間や、追加のラベル付けをする必要がないというメリットもある。

第3章　深層学習による画風変換

深層学習による画風変換は敵対的学習を用いて変換器を学習するのが一般的であるが、本研究ではStyleGAN2の画像生成器の学習済みパラメータ入れ替えを利用した画像変換手法について解説する。

3.1　画像生成ネットワークStyleGANとStyleGAN2

StyleGAN [11] は敵対的画像生成ネットワークと同じく潜在変数を入力することによって、画像を生成する手法である。この手法の特徴として、多層構造による高解像度の画像生成が可能なことと、層ごとに生成画像の特徴を分離できる点が挙げられる。

このStyleGANの特徴に大きく関与している多層構造はProgressive GAN [10] から派生している。この手法は高解像度の画像生成を行うために、図3.1のような方法で段階的に生成する画像を大きくする工夫が施されている。具体的には低解像度の画像生成の学習から始め、学習が一定以上成功した段階で生成モデルに更に大きな解像度の生成層を追加し、そしてその生成層の学習を繰り返すことで大きな解像度の画像を生成できる。

Progressive GANの特徴を引き継いだStyleGANでは、層の位置に応じて画風と画像の構造の影響力が変わる特徴を持っている。これは低い解像度の層であるほど生成される画像の構造に影響を与え、高い解像度であるほど画像の細かい部分であるテスクチャに影響を与える特徴がある。

本研究で使用するStyleGAN2 [7] はStyleGANの特徴を引き継ぎながら、いくつかの課題を解消した手法である。このためStyleGAN2も、潜在変数からの画像生成や多層構造による画像の特徴の分離、解像度層による生成画像構造・テスクチャの制御が可能である。

3.2　2つのドメインのStyleGANとSwapping

この解像度の層に応じて画像の制御が可能な特徴を応用して、学習済みモデルのパラメータの入れ替えで画像変換をするSwappingと呼ばれる手法 [2] が提案された。

これを実行するためには、ソースドメインの画像を学習した生成器AのパラメータG_Aと、ターゲットドメインの画像を学習した生成器BのパラメータG_Bが必要である。この2つの生成器G_AとG_Bのどちらかは、生成器のパラメータの類似性を維持するためにもう一方の生成器のパラメータから再学習する。例えば少年漫画を少女漫画風に変換する場合、少女漫画を生成する学習済みモデルは、少年漫画を生成する学習済みモデルのパラメータから再学習を行う。この生成器G_AとG_Bどちらも潜在変数から画像を生成できるが、図3.2のG_AとG_Bのように二つの生成器に同じ潜在変数を入力しても出力される画

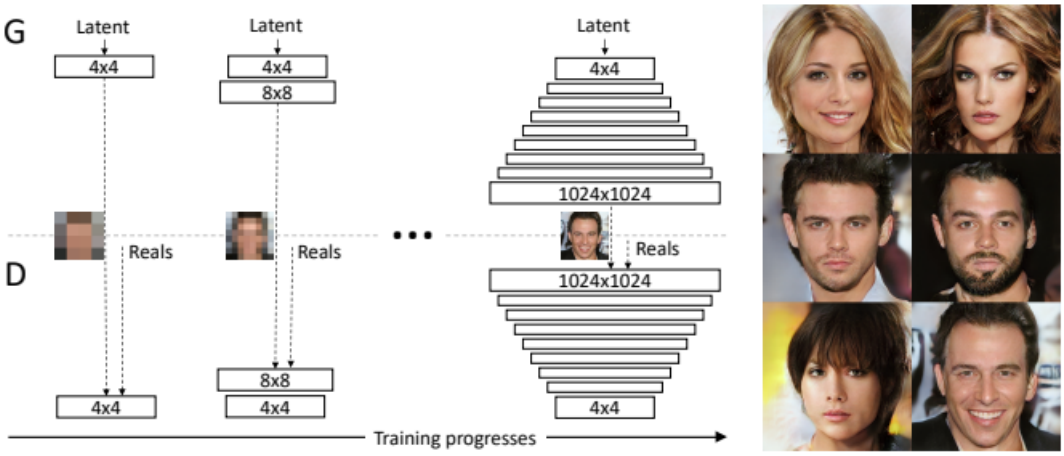


図 3.1: Progressive GAN のアーキテクチャ 図は関連論文 [10] から引用

像は異なる。しかしこの生成器のパラメータを各解像度の層ごとに入れ替えることでできる新しい生成モデル G_{swap} は、生成器 A の画像と生成器 B の画像の特徴を組み合わせた右の図 3.2 の様になる。

StyleGAN2 も StyleGAN と同じく層の解像度が低いと生成画像の構造を重視し、層の解像度が高い程生成画像のテスクチャを重視する。このためソースドメイン A からターゲットドメイン B への変換を行う G_{swap} を作成する場合は、一般的に G_A の低解像度の層が G_{swap} の低解像度の層のパラメータになり G_B の高解像度の層が G_{swap} の高解像度の層になる。これにより画像の大まかな形状はソースドメイン A に由来し、細かいテスクチャはターゲットドメイン B に由来する新たな画像が生成される。反対にターゲットドメイン B からソースドメイン A への画像変換を行う G_{swap} を作成したい場合は、G_B の低解像度層のパラメータが G_{swap} の低解像度層になり、G_A の高解像度層の層が G_{swap} の高解像度層になる。この入れ替えによって生成画像の大まかな形状はターゲットドメイン B に登場する画像だが、細かいテスクチャはソースドメイン A に由来する画像となる。このような形で行われる画像変換を Swapping と以降は呼ぶ。

Swapping は図 3.2 のように特定の解像度の層のパラメータとして生成器 A・B のどちらか一方のパラメータを用いるものだが、2 つのドメインの生成器のパラメータを混合して新しい生成モデル G_{mix} を作成することも出来る。例えば図 3.3 のように G_{mix} の特定の層 (図 3.3 では 32×32 の層) の生成器 A のパラメータ G_A を 25 %、生成器 B のパラメータ G_B を 75 %とした重み付き和とすることもできる。この重みの調整により細かく画風とアイデンティティの調整が可能になる。本研究ではパラメータを混合する方法で画像変換を行い以降ではこれを Mixing と呼ぶ。

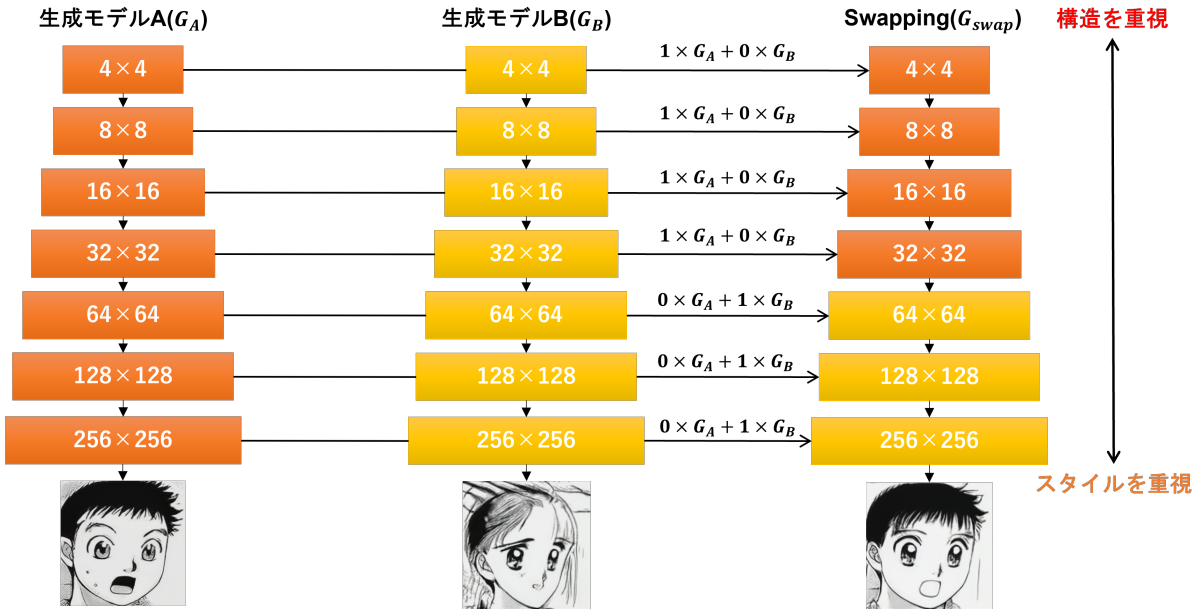


図 3.2: Swapping のイメージ図
図は少年漫画 (© 咲 香里) を少女漫画風 (© あゆみ ゆい) に変換した場合

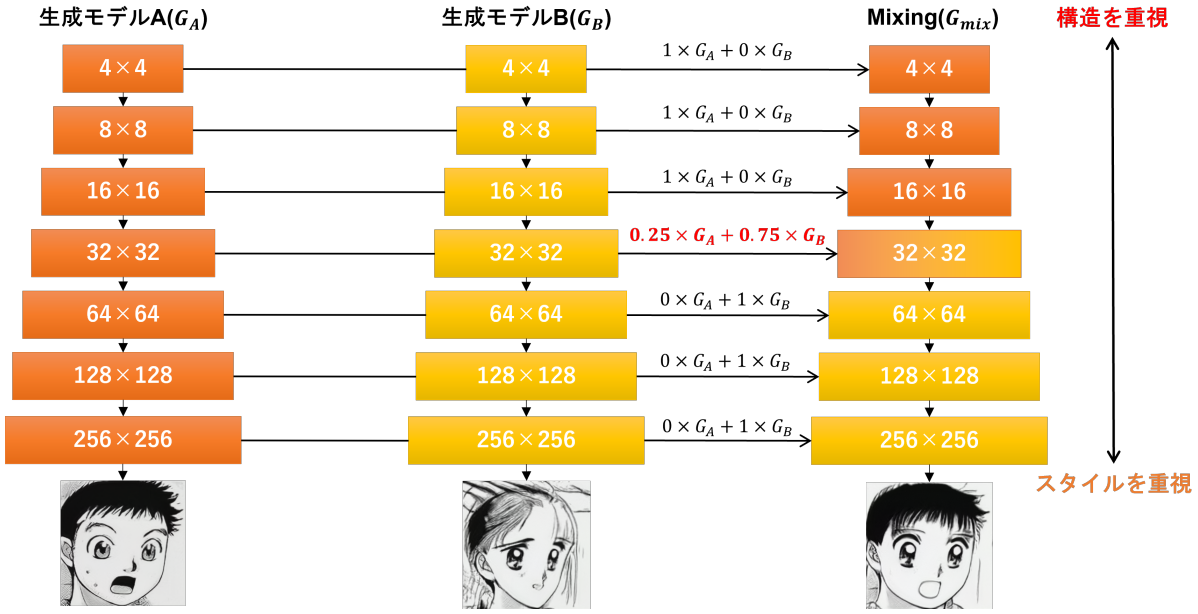


図 3.3: Mixing のイメージ図
図は少年漫画 (© 咲 香里) を少女漫画風 (© あゆみ ゆい) に変換した場合。32×32 層の G_B の割合を増やしたことで、Mixing の画像が Swapping よりも少し少女漫画寄りになっている。

3.3 StyleGAN2 を用いた画像変換

3.3.1 生成器からの画像の復元

上記の Swapping･Mixing によって画風の変換が可能になるが、上述のように StyleGAN2 は潜在変数と呼ばれる乱数の集合を生成モデルに入力して画像生成を行う手法のため、目的の画像を明示的に生成するには様々な乱数の集合を探索する必要がある。加えて StyleGAN2 では画像生成時に各層にノイズを付与して多様な画像を生成するため、潜在変数以外にも適切なノイズの付与が必要となる。この潜在変数の探索を手動で行うと膨大な手間と作業が必要なため、これを自動化する方法として invert [25] がある。これは潜在変数 w とノイズ n を誤差逆伝播法で最適化することで目的の画像 x を復元できる。具体的には、潜在変数とノイズから生成される画像 $G(w, n)$ と目的の画像 x 間の違いを損失として、誤差逆伝播を用いて潜在変数を最適化することで目的の画像を生成する潜在変数を見つけるものである。

損失関数は画像間の知覚的な類似性を測る LPIPS(Learned Perceptual Image Patch Similarity) [26] と、StyleGAN2 の画像生成時に各解像度に付与されるノイズに関する正則化項を加えたものである。

LPIPS は L_{image} で表され、目的の画像 x と探索対象の潜在変数 $\tilde{w}$ とノイズ n_i(i は層 i に入力する乱数) から生成された画像 $g(\tilde{w}, n_0, n_1, ...)$ 間の知覚的な差を測る関数である。知覚的な違いは VGG [27] 等の深層学習モデルに画像を入力し、そこから得られた中間出力の差を 2 乗し総和したものである。

$$L_{image} = D_{LPIPS}[x, g(\tilde{w}, n_0, n_1, ...)] \tag{3.1}$$

ノイズに関する正則化項は $L_{i,j}$ で表される。この数式は画像生成時に投与するノイズ n_i と、そのノイズを水平・垂直方向に 1 ピクセルずらしたものの積の結果をノイズのサイズで正規化した自己相関係数の二乗和の正則化項である。StytleGAN2 のノイズは生成層の各解像度の大きさに合わせて付与されるため、64×64 の解像度層では 64×64 サイズのノイズが付与される。

$$L_{i,j} = \left(\frac{1}{r_{i,j}^2} \cdot \sum_{x,y} n_{i,j}(x, y) \cdot n_{i,j}(x-1, y) \right)^2 + \left(\frac{1}{r_{i,j}^2} \cdot \sum_{x,y} n_{i,j}(x, y) \cdot n_{i,j}(x, y-1) \right)^2 \tag{3.2}$$

i がノイズの解像度の番号、j がノイズのサイズのプーリングの規模 ($(j \times 2)^2$ でプーリングを行う、$j = 0$ の時プーリングなし)、r がノイズのサイズ、x, y がノイズの座標を表す。

これらから出力される二つの数式の値を最小化することで目的の画像を生成する潜在変数の探索が可能になり、以下の数式で表される。

$$L_{total} = L_{image} + \alpha \sum_{x,y} L_{i,j}(x, y) \tag{3.3}$$

3.3.2 画像生成

StyleGAN2 のパラメータを入れ替える方法の画像変換では、まず変換したい画像のドメインの生成器 G_{base} を用いて invert の処理を行い、目的の画像を生成する潜在変数 $\tilde{w}$ を得る。次に、変換したい画像のドメインの生成器 G_{base} と変換先のドメインの生成器 G_{style} を用意し、パラメータを入れ替え G_{swap} というモデルを得る。最後に invert で獲得した潜在変数 $\tilde{w}$ を、パラメータ入れ替えを行った生成器 G_{swap} に入力し画像を生成する。以上の処理により StyleGAN2 を用いた画像生成でも画像変換が可能になる。この処理では G_{swap} によって画像を変換したが、パラメータの混合による生成モデル G_{mix} でも変換が可能である。

第4章　提案手法

この章では生成器のパラメータの混合でできる生成器 G_{mix} を用いて画像変換を行う手法について解説する。またこれ以降、生成器のパラメータを混合して行う画像変換を Mixing、パラメータを入れ替えて行う画像変換を Swapping と呼ぶ。

4.1　提案手法の概要

本研究で提案する Mixing の概要を図 4.1 示し、以下の手順リストに記載する。パラメータ比率の探索は以下の手順の 3. と 4. の部分が該当し、最適な変換となるパラメータ比率を探す。

1. 変換したい画像 x を入力
2. 変換したい画像 x を生成するモデル G_{base} で invert を行い、x の画像を復元できる潜在変数 $\tilde{w}$ を探索・獲得する
3. invert で獲得した潜在変数 $\tilde{w}$ を Mixing されたパラメータの生成器 G_{mix} に入力する
4. G_{mix} から生成された画像 $G_{mix}(\tilde{w}, n_0, n_1, ...)$ を 4.2 節で定義する損失関数に入力し損失を計算する
5. 3. と 4. のパラメータの探索を繰り返し、損失関数が最小となるパラメータ比率の変換画像を決定する

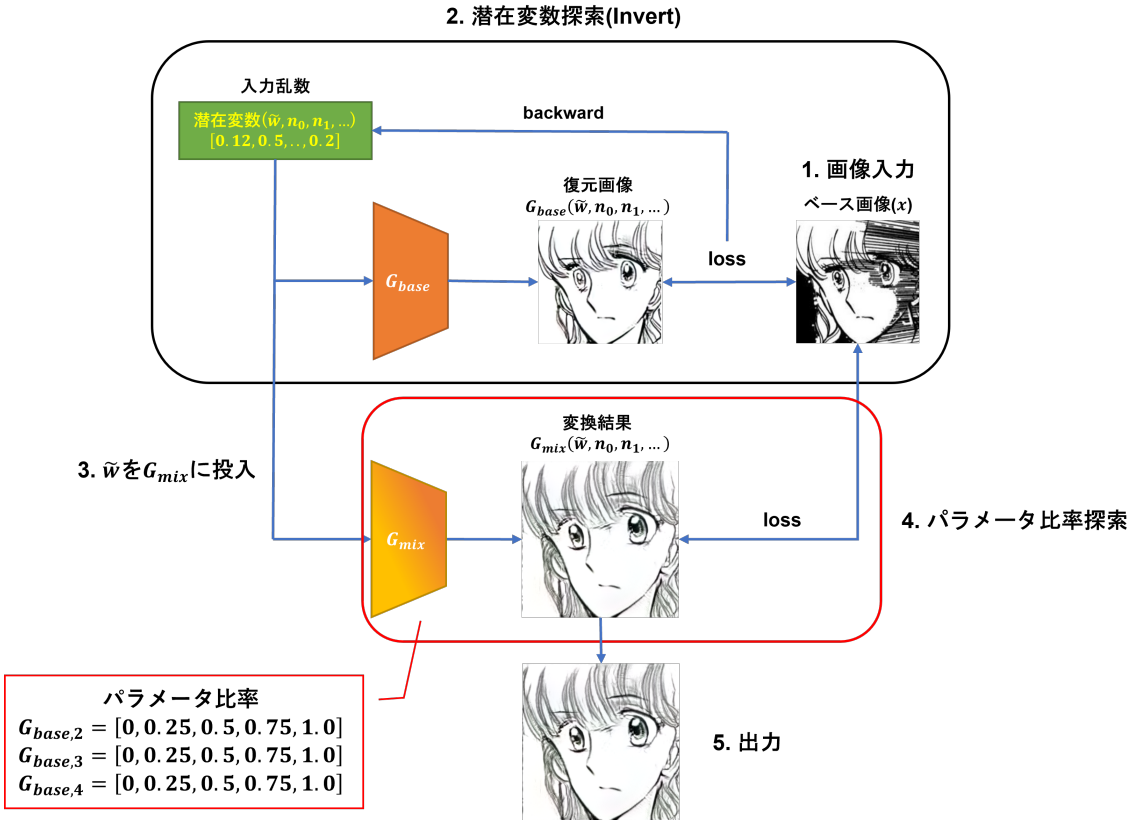


図 **4.1:** Mixing の概要 ©愛田 真夕美

4.2 パラメータ比率の探索

本研究では、アイデンティティを維持した画像変換を目的としているため、アイデンティティと画風に注目して各層のパラメータ比率を探索することが必要になる。このため、アイデンティティと画風の評価を行う損失関数を定義する。

アイデンティティの評価を行う損失関数は、画像の知覚的な構造を見る LPIPS を使用する。LPIPS は 3.3.1 節の invert の説明で提示した LPIPS と同一であり、変換前の画像 x と特定のパラメータの比率で変換した画像 $G_{mix}(\tilde{w}, n_0, n_1, \ldots)$ 間で比較を行い損失が計算される。この値が小さい程変換前の画像と変換後の画像の構造が類似するため、変換前のアイデンティティを維持していると判断する。

$$L_{identity} = D_{LPIPS}[x, G_{mix}(\tilde{w}, n_0, n_1, \ldots)] \tag{4.1}$$

画風に関しては特定の画風を評価する明確な指標が存在しないため、既定の数式ではなくイラストなどの作品を分類する学習済みの分類器を利用して評価を行う。この分類器は入力された画像の作品が何かを分類するモデルであり、学習は変換したいドメインのデータセットで行った。分類器を C として、ImageNet のデータセットで学習したモデルを漫画の作品で再学習を行った Xception [28] を採用した。モデル C に変換画像 $G_{mix}(\tilde{w}, n_0, n_1, \ldots)$ を入力し作品ラベルを推論することで、そこから画風に関する尤度が得られる。この尤度

と変換目標ラベル $label$ との間の誤差で画風変換の評価を行う。この損失関数は変換画像を入力した分類器から得られる尤度が、変換目標のドメインのラベルに近くなると値が小さくなるように設計されており、平均二乗誤差を用いて以下の様に計算される。

$$L_{style} = (C(G_{mix}(\tilde{w}, n_0, n_1, \ldots)) - label)^2 \tag{4.2}$$

例えば、変換画像を推論したモデルの尤度の出力が $label = [0.9, 0.1]$ で、変換目標のラベルが [0,1] だった場合、計算される損失は 0.81 となる。

これらを踏まえてパラメータ探索は $L_{identity}$ と、画像分類の出力と変換目標のラベルとの誤差 L_{style} を計算した MSE の値を合わせたものを評価関数として行われ、以下の数式で表される。

$$L_{total} = \alpha L_{identity} + \beta L_{style} \tag{4.3}$$

α と β はハイパーパラメータである。

またこの各層のパラメータ比率の探索をグリッドサーチで行う。各層のパラメータ比率は p_i で表され、i は生成モデルの層の位置を表す。i 層ごとに $p_i = [0, 0.25, 0.5, 0.75, 1.0]$ の 5 つの比率候補が存在する。この比率 p_i は変換元の画像を生成するモデルの i 層目のパラメータ $G_{base,i}$ の比率を指し、以下の数式で表される。

$$G_{mix,i} = p_i \times G_{base,i} + (1 - p_i) \times G_{style,i} \tag{4.4}$$

例えば比率の探索で 0.25 が選ばれた場合、$G_{base,i}$ が i 層では 25 %使用されることになる。一方で変換先のドメインの画像を生成するモデルのパラメータ $G_{style,i}$ は、$p = 1 - 0.25 = 0.75$ つまり 75 %が G_{style} のパラメータとなる。このため 0 が選択された場合、$G_{base,i}$ のパラメータは使用せず、$G_{style,i}$ のパラメータのみを使用し、反対に 1 が選択された場合、$G_{base,i}$ のパラメータのみを使用する。

しかし 256×256 の画像生成モデルでこの探索を行うと、探索対象が 7 層になり $5^7 = 78125$ 通りの膨大な探索量になるため、探索量を削減すべく探索する層を限定する。パラメータ比率を探索する層は 3 層目 (16×16)・4 層目 (32×32)・5 層目 (64×64) の 3 層で、それ以外の低解像度層 $(4 \times 4 \cdot 8 \times 8)$ は G_{base} のパラメータを 100 %、高解像度層 $(128 \times 128 \cdot 256 \times 256)$ は G_{style} のパラメータを 100 %に固定する。

第5章　実験

5.1　データセット

本研究の実験では Manga109 [29,30] と呼ばれる漫画データセットを使用した。このデータセットは日本の様々な漫画作品 109 種類が収録されており、アノテーションとして漫画のコマ・吹き出し・キャラクターの全身や顔のアノテーションが付与されている。実験に利用した作品は、「プリズム・ハート」と「Oh!われら劣等生徒会」の 2 作品である。この 2 作品のジャンルは「少女」で統一されているが、描かれている年代が異なっている。この実験は変換対象を顔とし、顔を目と鼻と口と髪の毛そしてその他アクセサリーで構成されているものと定義した。顔画像を抜き出すために、Manga109 に付与されている図 5.1 の赤枠の部分にあるアノテーションを利用した。しかし顔部分のアノテーションは、アイデンティティの一部となる髪の毛が含まれないため、髪の毛のパーツが含まれるようにアノテーションの y 軸の長さを 1.4 倍にした。またこのアノテーションは正面以外の顔のデータもあるため、顔のパーツが全て含まれない場合もある。このため手作業で正面を向いていない顔画像と顔のパーツである目と鼻と口が全て揃っていない画像を削除した。この条件で顔画像を抽出した結果、「プリズム・ハート」では 384 枚・「Oh!われら劣等生徒会」では 528 枚の顔画像が得られ、これらを学習データ・評価データとして利用した。

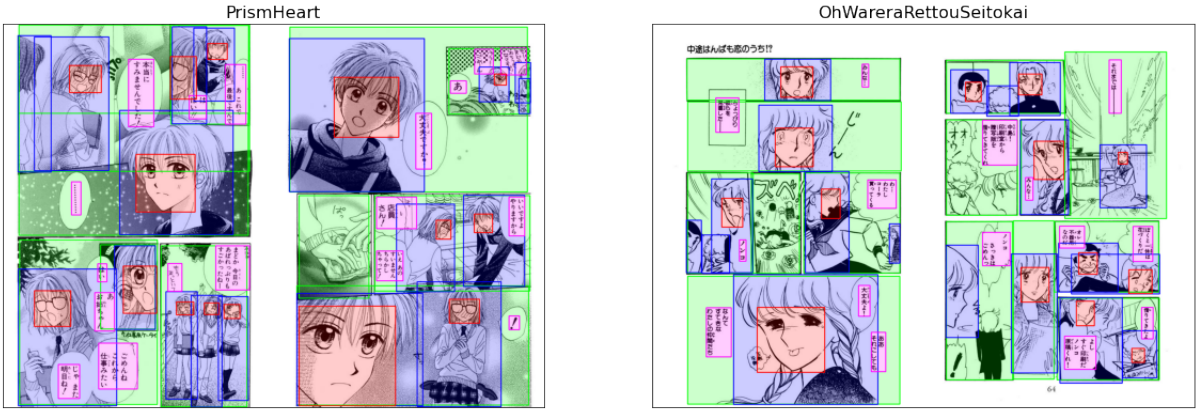


図 5.1: 「プリズム・ハート (© 浅月 舞)」と「Oh!われら劣等生徒会 (© 愛田 真夕美)」のデータ例とアノテーション例

5.2　モデルの学習・ハイパーパラメータ

本研究では先行研究との比較のために、CycleGAN・U-GAT-IT・Swapping と提案手法 Mixing との比較を行う。Swapping と Mixing は StyleGAN2 の学習済みモデル G_{base} と

G_{style} のパラメータを入れ替え・混合したモデルを用いて画像変換を行う方法であり、その学習手順を以下に記載する。

1. Manga109 データセットの全ての顔データを用い StyleGAN2 の学習法に基づいて 150000 バッチ学習する。

2. 1. で学習したモデルを「プリズム・ハート」・「Oh!われら劣等生徒会」のデータセットでそれぞれ 10000 バッチの再学習を行い二つのモデルを作成する。このとき「プリズム・ハート」で再学習したモデルを $G_{base,P}$、「Oh!われら劣等生徒会」で再学習したモデルを $G_{base,O}$ とする。

3. 2. で獲得した生成器を更に 2000 バッチ分再学習する。ここで $G_{base,P}$ については「Oh!われら劣等生徒会」のデータで再学習し $G_{style,PO}$ を得る。$G_{base,O}$ については「プリズム・ハート」のデータで再学習し $G_{style,OP}$ を得る。

上記手順で得られるモデル $G_{base,O}$ と $G_{base,P}$、$G_{style,PO}$、$G_{style,OP}$ の学習は各々のデータセット内の全ての画像データを用いて行う。このためデータセットは学習用・評価用のデータセットに分割していない。

2. で得られた生成モデル $G_{base,O}$ と $G_{base,P}$ と、3. で得られた生成モデル $G_{style,PO}$ と $G_{style,OP}$ で Swapping・Mixing を行い変換する。変換には目的の画像を生成器に復元させる潜在変数を探す invert が必要であるが、この探索は [7] の invert と同じ設定で実行した。パラメータの入れ替えを行う Swapping では「プリズム・ハート」から「Oh!われら劣等生徒会」の変換を行う場合 4×4・8×8・16×16・32×32 の層は $G_{base,P}$ のモデルのパラメータを利用し、64×64・128×128・256×256 の層は $G_{style,PO}$ のパラメータを利用し変換を行う。一方「Oh!われら劣等生徒会」から「プリズム・ハート」の変換を行う場合は、4×4・8×8・16×16・32×32 の層は $G_{base,O}$ のパラメータを利用し、64×64・128×128・256×256 の層は $G_{style,OP}$ のパラメータを利用し変換を行う。提案手法 Mixing の比率の探索についても、混合する学習済みモデルのパラメータの組み合わせは Swapping と同一である。Mixing の比率探索時の損失関数のハイパーパラメータを $\alpha=1,\beta=0.333$ とした。

CycleGAN・U-GAT-IT は [1,6] の実験方法を変えずに学習を行い、画像変換を実行した。しかし CycleGAN と U-GAT-IT の学習と評価では、データセットを一度分割して学習用と検証用に分けてから行っている。このため学習用はモデルの学習にのみ利用し、検証用はモデルの評価にのみ利用されるため、学習に使用できるデータ量が Swapping や Mixing と比べると少なくなる。この学習するデータ量を同じにするために CycleGAN・U-GAT-IT では共に全ての画像データを学習データとして使用し、変換も学習データでの画像で行った。

5.3 ユーザースタディ

本研究はアイデンティティを維持しつつ画風を変換することを目的としている。このためアイデンティティの維持に関する評価と、画風の評価、さらに画像変換の品質の評価を行う。しかしこれらの評価は定量的にはできないため、ユーザースタディによる定性的な

評価を行う。本研究のユーザースタディは、アイデンティティの評価 10 問・画風の評価 10 問・変換品質の評価 10 問の 3 セクションで構成されており、合計 30 問のアンケートとなっている。

アイデンティティの評価のためのアンケートの例を図 5.2 に示す。4 手法の変換画像と変換元に類似するターゲットドメインの画像 (以降類似画像と呼ぶ)1 枚の計 5 枚の中から、変換前の画像の人物の特徴を最も捉えているものを一つ上部にある数字で選択してもらう。図 5.2 の左にある base_image が変換元の画像で、その右にある 5 枚の内 4 枚の画像群が CycleGAN・U-GAT-IT・Swapping・Mixing で変換した画像、残り一枚は base_image の類似画像である。それらの画像の上部に示されている数字が選択肢で、この中から変換前の画像の人物の特徴を最もとらえている変換画像を数字で一つ選択してもらう。類似画像として、base_image と変換先のドメイン画像群との間で算出される LPIPS の値が最も小さい画像を用いる。このアンケートの評価では、選択された変換画像を作成した手法が最もアイデンティティを維持した変換画像として優れることになる。また変換画像だけでなく類似画像を入れた理由は、画像変換の必要性を調べるためである。これは類似画像が変換性能が良いと思われた場合、類似キャラクターを用いるだけで画像変換が達成できることが言えるので、その場合は画像変換を用いる必要がないことがわかる。

図 5.2: アイデンティティ評価のアンケート例 © 浅月 舞 © 愛田 真夕美
1 は U-GAT-IT 2 は mixing 3 は類似画像 4 は CycleGAN 5 は swapping

画風の評価のためのアンケートの例を図 5.3 に示す。4 手法の変換画像と変換元の画像 1 枚の計 5 枚の中から、左の 6 枚の画像群と同じ作者が描いたと思われるものを一つ選択してもらう。図 5.3 の左の画像群としてターゲットドメインの画像群の 6 枚のサンプルをランダムに表示する。図 5.3 の右に変換画像とその変換元の画像を表示し、左の画像群と同じ作者が描いていそうなものを 5 枚の中から 1 枚を上部にある数字で選んでもらう。このアンケートでは被験者が選択した変換画像が最も画風を変換していると言え、その変換画像を作成した手法がその画像群の中で最も画風を変更できると評価される。また変換画像だけでなく変換元の画像も比較したのは、「プリズム・ハート」と「Oh!われら劣等生徒会」で画風の違いが存在しないかを確かめるためである。これは変換元の画像が良いと思われた場合、変換元の画像が変換先の画像と思われるため「プリズム・ハート」と「Oh!われら劣等生徒会」で画風の違いが存在しなくなり、アンケートを取る意義を確認できる。

図 5.3: 画風評価のアンケート例 © 浅月 舞, © 愛田 真夕美
1 は変換元の画像 2 は U-GAT-IT 3 は mixing 4 は swapping 5 は CycleGAN

この二つの評価では、各セクション 10 問の内前半 5 問が「プリズム・ハート」、後半 5 問が「Oh!われら劣等生徒会」となっている。アンケートの問題の詳細を以下の表に記載する。「問題: 共通」とはどのユーザーでも同じ画像の問題が出現することを意味し、

問題	作品	問題に表示する画像
1	プリズム・ハート	共通
2	プリズム・ハート	共通
3	プリズム・ハート	共通
4	プリズム・ハート	ユーザーごとに異なる
5	プリズム・ハート	ユーザーごとに異なる
6	Oh!われら劣等生徒会	共通
7	Oh!われら劣等生徒会	共通
8	Oh!われら劣等生徒会	共通
9	Oh!われら劣等生徒会	ユーザーごとに異なる
10	Oh!われら劣等生徒会	ユーザーごとに異なる

「問題: ユーザーごとに異なる」はユーザーによって問題となる画像が異なることを意味する。どのユーザーでも同じ画像の問題を出題するのは、画像による変換のクオリティとユーザーによる評価のばらつきを防ぐためである。アンケートに選ばれる画像は無作為に選択されるが、1 作品につき選択される変換元の画像の枚数が 5 枚と少ないため、チェリー・ピッキングを行う可能性がある。このため一部の問題ではユーザーごとに異なる変換元の画像が出現するようにした。またこの 2 つのアンケートではアイデンティティと画風の両立を評価するために、設問に選ばれた変換元の画像はセクション 1・セクション 2 で同一のものとなる。つまりアイデンティティの問題 1〜10 で出題された画像は、画風の問題 1〜10 でも出題される。

変換品質のためのアンケートの例を図 5.4 に示す。本物の画像と生成画像をランダムで掲載し本物の画像と思われるものを全て選択してもらう。例えば図 5.4 で 1 と 2 が本物と思われた場合、1 と 2 の画像を選択する。この評価では本物の画像 10 枚、比較手法で変換した画像 40 枚の合計 50 枚を準備し、無作為にシャッフルしたものを、1 問につき 5 枚表示しその中から本物の画像と思われるものを選択する形式にした。このため、1 問 5 枚の画像の中に本物の画像がない場合もあり、複数枚が本物の画像の場合もある。このアンケートで変換画像が選択された場合、その手法が本物に近い精巧な変換をしていると判

断する。このため変換画像が選択される数が多い手法が画像変換の質として優れると言える。この評価では、前 2 セクションのアンケートに登場した本物の画像とそれに関連する変換画像の影響を受けないようにするため、前 2 セクションで使用した画像以外からランダムに画像を抽出した。またこのセクションは全てのユーザーで同一の問題となっており、ユーザーごとに異なる画像が登場せず、出現する画像の順番も同じになっている。

図 5.4: 変換品質評価のアンケート例 © 浅月 舞, © 愛田 真夕美
1 は本物の画像 2 は本物の画像 3 は mixing 4 は mixing 5 は U-GAT-IT

これらのアンケートを 30 人に取り評価を行った。

5.4　アイデンティティの評価の議論

アイデンティティの評価を全て集約した結果を、図 5.5 に示す。これを見ると Mixing や Swapping のような StyleGAN2 を用いた変換手法がアイデンティティを維持できると回答した人がほぼすべてを占めている。しかし Mixing より Swapping の方がアイデンティティの維持に優れると回答した人が多いため、Swapping より Mixing の方がアイデンティティの維持が劣ることがわかる。この結果の要因として、Mixing によって選択された変換元 G_{base} の比率が、Swapping よりも小さいことが原因だと考えられる。図 5.6a と 5.6b はアンケートに使用した変換画像のパラメータ比率を平均した結果である。5.2 節で述べたように Swapping の 16×16 と 32×32 の層のパラメータの使用比は G_{base} が 100 % である。一方で Mixing では使用比が全体で 50 % 前後と Swapping の 100 % と比べ約半分と少ないため、画風の変換が強くなりアイデンティティが Swapping より消失しやすくなっている。一方で CycleGAN や U-GAT-IT はほとんどの回答で選択されていないため、アイデンティティの維持に関しては StyleGAN2 関連の手法 Swapping や Mixing の方が有効であることが示された。

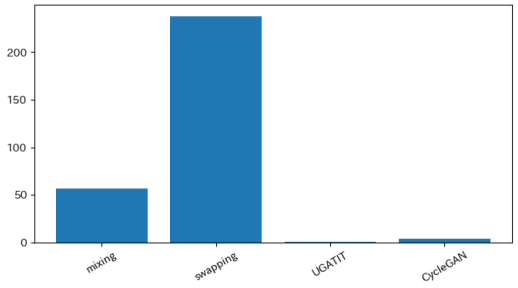
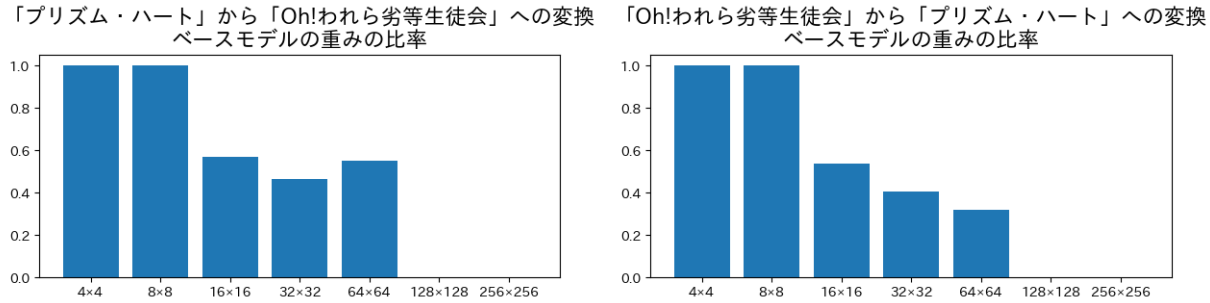


図 5.5: アイデンティティ評価の結果グラフ



(a) (b)

図 5.6: Mixing による各層の G_{base} 比率の平均

5.5 画風の評価の議論

画風の評価を全て集約した結果を、図 5.7 に示す。これより CycleGAN や U-GAT-IT の手法によって変換された画像が、変換先のドメインの作者が描いていそうだと回答した人が多いため、画風の変換という観点では有効であることが示された。しかし CycleGAN や U-GAT-IT のような画像変換の手法では 1 章の図 1.4 のように変換元のキャラクターを変換先のドメインに登場するキャラクターに置き換えるという問題があった。

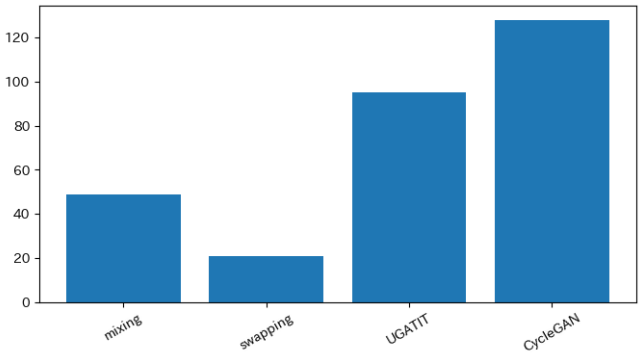


図 5.7: 画風評価の結果グラフ

「同じ作者が描いたと思われるもの」による問題の判定は、画風だけでなくキャラクターによっても判断される。例えば「Oh!われら劣等生徒会」に登場する短髪の男性を「プリズム・ハート」の作風に変換することを考える。そして「Oh!われら劣等生徒会」に登場する短髪の男性を変換した結果を図 5.8 に掲載する。この図 5.8 から明らかのように、中央の提案手法 Mixing による変換画像 (Mixing_Image) ではアイデンティティを保たれているが、右端の CycleGAN による変換画像 (CycleGAN_Image) ではアイデンティティが保たれていない。一方で変換先の作風である「プリズム・ハート」に登場するキャラクターの例を図 5.9 に示す。図 5.9 に示されるように、「プリズム・ハート」には短髪の男性は存在しない。そのため、被験者はアイデンティティが保たれた短髪の男性を同じ作者が描いたものと判断するのではなく、短髪ではない CycleGAN の結果の方を同じ作者が描いたものと判断する傾向にある。しかし本研究ではアイデンティティを維持しつつ画風を変換

することが目標であるため、キャラクターそのものを変換した場合、目標を達成したということは出来ない。

図 5.8: 短髪の男性の変換例 © 愛田 真夕美

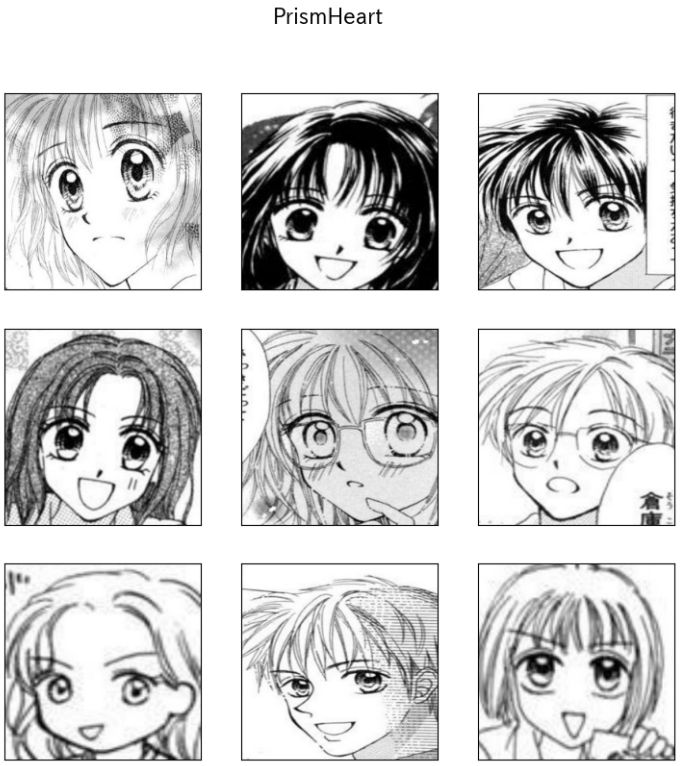


図 5.9: 「プリズム・ハート」に登場するキャラクター例 © 浅月 舞

一方で、アンケートでは Swapping や Mixing の変換手法も少ないが画風が同じものとしてある程度選択されている。Swapping や Mixing は変換先のドメインのパラメータを低解像度の層まで設定する (図 5.10 の右端の画像は 8×8 から 256×256 層まで G_{style} のパラメータに設定した。) と、図 5.10 のように左端の元画像が右端の画像のようにキャラクターが変化する現象が発生する。これを過剰な入れ替えや混合と呼ぶ。一方でバランスよくパラメータを混ぜ合わせた Mixing ではキャラクターを変換せずに画風を変換できている。このため Swapping や Mixing のような変換手法ではアイデンティティと画風のトレードオフの関係を操作できると言える。このトレードオフの関係を踏まえると、Mixing や Swapping は被験者に「同じ作者が描いたもの」であると選択された数が少ないが、CycleGAN や

U-GAT-IT におけるアイデンティティの評価と違い画風でも一定の評価がされており、アイデンティティを保ちつつも画風を変換できている。また Swapping と Mixing の二つを比較すると、Swapping より Mixing の方が「同じ作者が描いたと思われるもの」と回答した数が多く、提案手法 Mixing の方が Swapping より画風の変換性能が高い。これは 5.4 節のアイデンティティの評価の議論から言えるように、Mixing は Swapping よりも変換先の重みを多く利用した結果である。

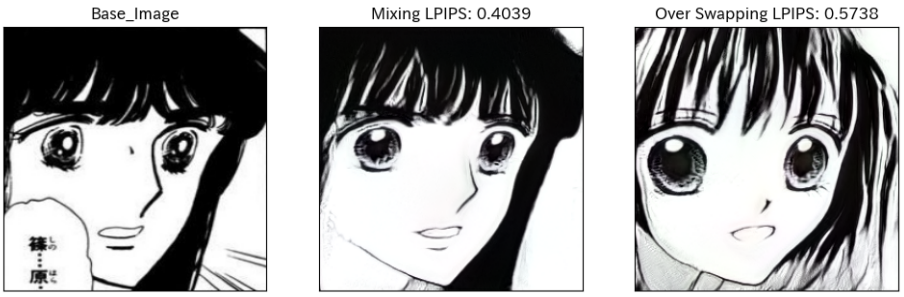


図 5.10: 過剰な Swapping の例 over_swapping では base_image のキャラクターが変わっている ⓒ 浅月 舞 ⓒ 愛田 真夕美

5.6 合成品質の評価の議論

合成品質の評価を全て集約した結果を、図 5.11 に示す。これを見ると、本物の画像より品質の高くなった手法はないが、最も本物の画像と思われた手法は Mixing でその次に、Swapping、U-GAT-IT、CycleGAN と続く形となった。Swapping や Mixing のような StyleGAN2 を用いた変換の方が品質が高いため、画像変換で本物らしさとしての品質を求める場合 StyleGAN2 によるパラメータの入れ替えが良いことが考えられる。また Swapping より提案手法 Mixing の方がわずかに評価が高い理由として、図 5.12 を見ると Mixing の方がわずかに頭の部分や目の部分でノイズが少ないため、提案手法の方が変換品質の評価が高いと考えられる。

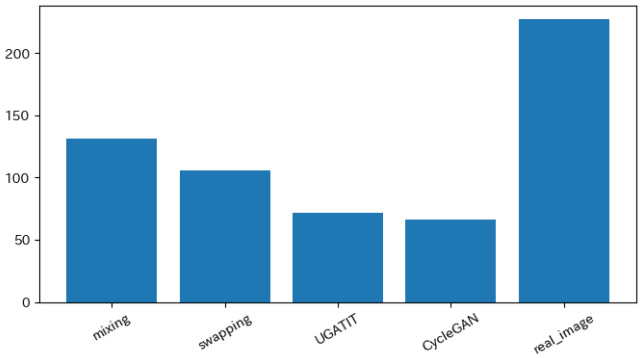


図 5.11: 合成品質評価の結果グラフ

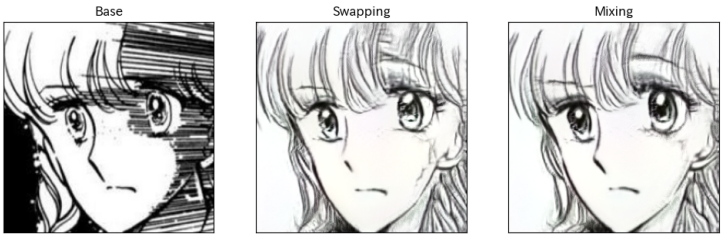


図 5.12: Swapping と Mixing の生成画像の比較 © 愛田 真夕美

5.7 今後の課題

5.7.1 損失関数の再構築

この結果を踏まえた今後の課題として、最適なパラメータ比率の決定するにあたってグリッドサーチの代わりに誤差逆伝播法による最適化により比率を探索することが挙げられる。グリッドサーチによる探索では画像サイズ 256×256 でも 7 層分の探索となり、探索数が $5^7 = 78125$ 通りと探索数が膨大になる。本研究ではその削減のために比率探索する層を 16×16〜64×64 の中間層に限定したが、本来は 4×4 のような初期層や 256×256 のような最終層もパラメータ比率を探索することでさらに変換の性能が良くなる可能性がある。グリッドサーチによる探索の代わりに最適化によるパラメータ比率の探索ができれば、多くの層で比率が探索可能になり、比率の細かい調整ができるため変換の可能性を広げられ、最適な変換の解を見つけやすくなると考えられる。

しかし本研究で提案した損失関数では、最適化によるパラメータ比率の探索ができない。最適化を行うと図 5.13 のような挙動を見せ初期値から大きく変動せず、最適化が有効な手段にならないからである。このため、どのような初期値からでも結果が同じように最適化できる損失関数を再構築することが今後の研究で重要な課題となる。

特に再構築を行うべき損失関数は漫画やイラストの画風の評価を行う損失関数である。本研究の画風の損失関数は作品の分類から出力される尤度を用いて画風を判断しているが、この分類では画風を判断するためにその作品に登場するキャラクターそのものから画風を判断している場合がある。この場合画風の定義である色合いや線のタッチなどを評価しているとは言えない。これを改善することで、比率探索だけでなく更なるアイデンティティと画風の変換のバランスの改善や変換品質の改善の可能性が期待できる。

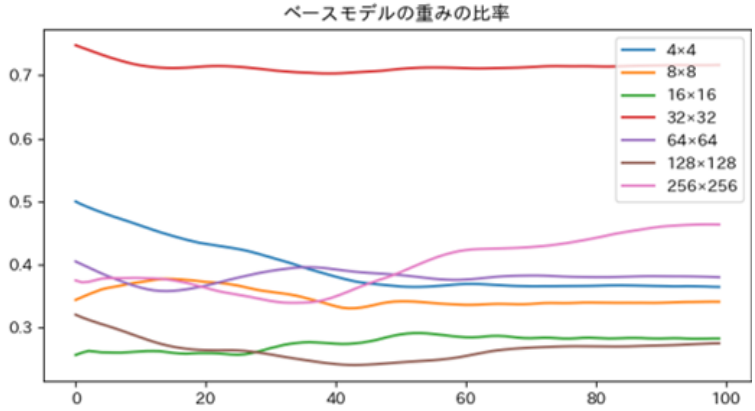


図 5.13: ランダムに比率を初期化し最適化した場合の比率の変化例

5.7.2 一部のアイデンティティの損失

他にも現状の StyleGAN2 による画像変換では一部のアイデンティティを失う問題点がある。これは画像変換時に、メガネや帽子と言ったアクセサリーが図 5.14 のように消失する問題である。

この問題は図 5.15 のように「鉢巻きをつけている」ような少量しかないデータの特徴を生成器が復元できないことと、変換元のドメインで登場する属性が変換先のドメインで出現しないことが原因に挙げられる。

前者の問題は限られたデータの情報を残すために「メガネ」等の属性ラベルを付与するなどで画像以外の追加情報を付与し、生成器の損失関数にその属性の維持を強制するペナルティを追加することで解決できると考えられる。後者の問題も前者の解決策を利用できるが、変換先のドメインでその属性が登場しない場合アイデンティティを維持した変換が上手くいかない可能性がある。これは G_{base} から生成されるキャラクターと再学習後のモデル G_{style} から生成されるキャラクターの関係性を一致させる方法 [24] が有効になる可能性がある。関係性を一致させることで変換元のドメインにしか登場しないパーツが、変換先でも類似したものとして登場しやすくなり属性を維持できる可能性がある。これらを一致させることで少ない特徴の維持と変換品質の向上が期待できる。

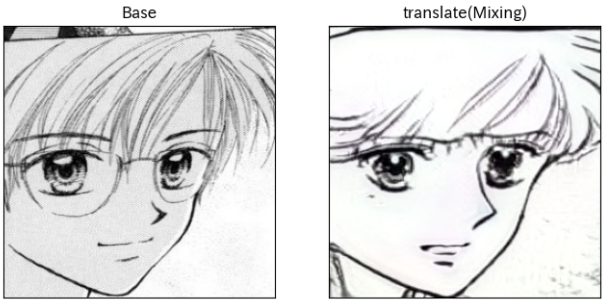


図 5.14: 変換時にメガネを失う例 © 浅月 舞

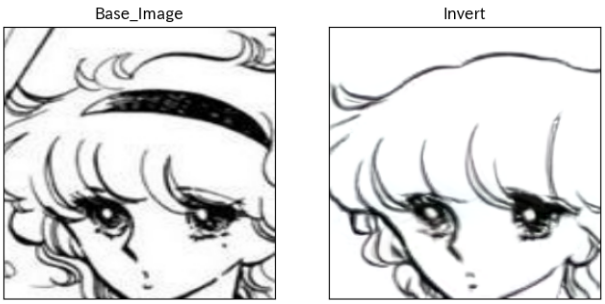


図 5.15: アクセサリーが消失する例 © 愛田 真夕美

また Mixing や Swapping は変換時間が CycleGAN や U-GAT-IT より長い。Mixing や Swapping では GPU A6000 でも変換に約 1 分かかる一方で、CycleGAN や U-GAT-IT では 1 秒もかからず変換出来る。この問題は StyleGAN2 の invert が大きな原因となっている。これを解決するために先行研究 [23] では VAE [31] などで画像を潜在変数に変換するエンコーダーを事前に作成し、それから潜在変数を作成することで潜在変数の探索時間を削減する方法がとられている。

第6章　結論

本研究ではStyleGAN2のパラメータの入れ替えによる画像変換手法を提案した。提案手法では画風を変換しつつアイデンティティ維持できる最適なパラメータの入れ替え・混合を探索し、アイデンティティを維持した画風変換を画像ごとに行った。実験結果より提案手法は、アイデンティティの維持では既存手法Swappingに劣るものの、画風の変換では既存手法Swappingより良い評価を得ることができ、変換の品質の評価も高いことがわかった。今後の課題としては、損失関数の見直しが挙げられる。

謝辞

本論文を作成するにあたり、研究指導をして下さった飯山将晃教授、Manga109 のデータセット提供して頂きました相澤・山崎・松井研究室、アンケートに回答してくださったみなさまに感謝の意を表します。

参考文献

[1] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In Proceedings of the IEEE international conference on computer vision, pp. 2223–2232, 2017.

[2] Justin NM Pinkney and Doron Adler. Resolution dependent gan interpolation for controllable image synthesis between domains. arXiv preprint arXiv:2010.05334, 2020.

[3] Yang Chen, Yu-Kun Lai, and Yong-Jin Liu. Cartoongan: Generative adversarial networks for photo cartoonization. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 9465–9474, 2018.

[4] Lvmin Zhang, Xinrui Wang, Qingnan Fan, Yi Ji, and Chunping Liu. Generating manga from illustrations via mimicking manga creation workflow. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5642–5651, 2021.

[5] シモセラエドガー, 飯塚里志. ラフスケッチの自動線画化技術 (特集漫画・線画の画像処理)–(漫画・線画の補正処理). 映像情報メディア学会誌= The journal of the Institute of Image Information and Television Engineers, Vol. 72, No. 3, pp. 338–341, 2018.

[6] Junho Kim, Minjae Kim, Hyeonwoo Kang, and Kwanghee Lee. U-gat-it: Unsupervised generative attentional networks with adaptive layer-instance normalization for image-to-image translation. arXiv preprint arXiv:1907.10830, 2019.

[7] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 8110–8119, 2020.

[8] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. Communications of the ACM, Vol. 63, No. 11, pp. 139–144, 2020.

[9] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv preprint arXiv:1511.06434, 2015.

[10] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. arXiv preprint arXiv:1710.10196, 2017.

[11] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 4401–4410, 2019.

[12] Xun Huang and Serge Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In Proceedings of the IEEE international conference on computer vision, pp. 1501–1510, 2017.

[13] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1125–1134, 2017.

[14] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. arXiv preprint arXiv:1411.1784, 2014.

[15] Aaron Gokaslan, Vivek Ramanujan, Daniel Ritchie, Kwang In Kim, and James Tompkin. Improving shape deformation in unsupervised image-to-image translation. In Proceedings of the European Conference on Computer Vision (ECCV), pp. 649–665, 2018.

[16] Matthew Amodio and Smita Krishnaswamy. Travelgan: Image-to-image translation by transformation vector learning. In Proceedings of the ieee/cvf conference on computer vision and pattern recognition, pp. 8983–8992, 2019.

[17] Yujun Shen, Ping Luo, Junjie Yan, Xiaogang Wang, and Xiaoou Tang. Faceidgan: Learning a symmetry three-player gan for identity-preserving face synthesis. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 821–830, 2018.

[18] Hao Su, Jianwei Niu, Xuefeng Liu, Qingfeng Li, Jiahe Cui, and Ji Wan. Mangagan: Unpaired photo-to-manga translation based on the methodology of manga drawing. In Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 35, pp. 2611–2619, 2021.

[19] Yiming Gao and Jiangqin Wu. Gan-based unpaired chinese character image translation via skeleton transformation and stroke rendering. In proceedings of the AAAI conference on artificial intelligence, Vol. 34, pp. 646–653, 2020.

[20] Bo Chang, Qiong Zhang, Shenyi Pan, and Lili Meng. Generating handwritten chinese characters using cyclegan. In 2018 IEEE winter conference on applications of computer vision (WACV), pp. 199–207. IEEE, 2018.

[21] Yangchen Xie, Xinyuan Chen, Li Sun, and Yue Lu. Dg-font: Deformable generative networks for unsupervised font generation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5130–5140, 2021.

[22] Jialu Huang, Jing Liao, and Sam Kwong. Unsupervised image-to-image translation via pre-trained stylegan2 network. IEEE Transactions on Multimedia, Vol. 24, pp. 1435–1448, 2021.

[23] Guoxian Song, Linjie Luo, Jing Liu, Wan-Chun Ma, Chunpong Lai, Chuanxia Zheng, and Tat-Jen Cham. Agilegan: stylizing portraits by inversion-consistent transfer learning. ACM Transactions on Graphics (TOG), Vol. 40, No. 4, pp. 1–13, 2021.

[24] Jihye Back. Fine-tuning stylegan2 for cartoon face generation. arXiv preprint arXiv:2106.12445, 2021.

[25] Antonia Creswell and Anil Anthony Bharath. Inverting the generator of a generative adversarial network. IEEE transactions on neural networks and learning systems, Vol. 30, No. 7, pp. 1967–1974, 2018.

[26] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 586–595, 2018.

[27] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556, 2014.

[28] François Chollet. Xception: Deep learning with depthwise separable convolutions. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1251–1258, 2017.

[29] Yusuke Matsui, Kota Ito, Yuji Aramaki, Azuma Fujimoto, Toru Ogawa, Toshihiko Yamasaki, and Kiyoharu Aizawa. Sketch-based manga retrieval using manga109 dataset. Multimedia Tools and Applications, Vol. 76, No. 20, pp. 21811–21838, 2017.

[30] Kiyoharu Aizawa, Azuma Fujimoto, Atsushi Otsubo, Toru Ogawa, Yusuke Matsui, Koki Tsubota, and Hikaru Ikuta. Building a manga dataset "manga109" with annotations for multimedia applications. IEEE MultiMedia, Vol. 27, No. 2, pp. 8–18, 2020.

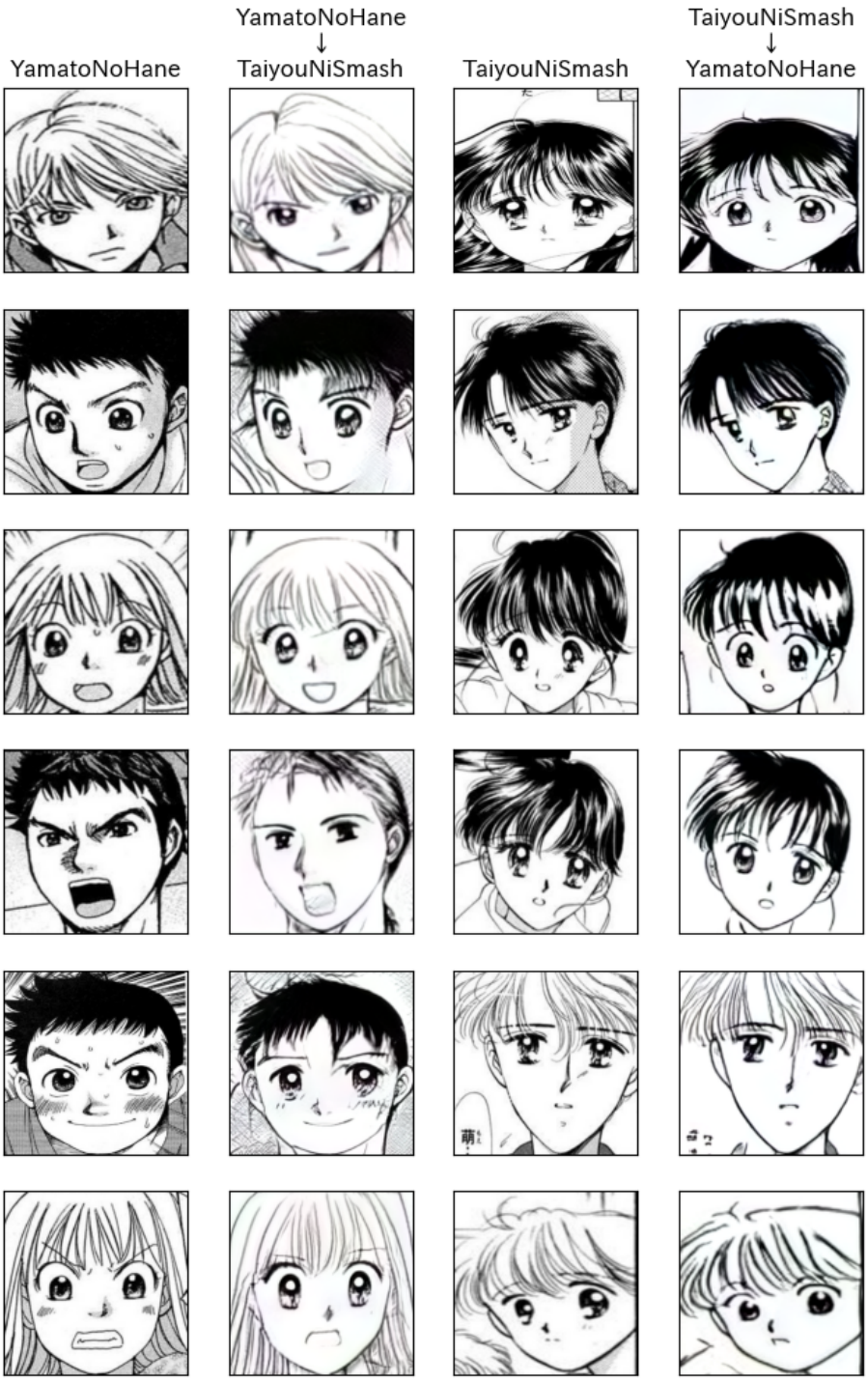
[31] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114, 2013.

[32] Yingtao Tian, Tarin Clanuwat, Chikahiko Suzuki, and Asanobu Kitamoto. Ukiyo-e analysis and creativity with attribute and geometry annotation. In Proceedings of the International Conference on Computational Creativity, 2021.

[33] Tero Karras, Miika Aittala, Janne Hellsten, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Training generative adversarial networks with limited data. Advances in Neural Information Processing Systems, Vol. 33, pp. 12104–12114, 2020.

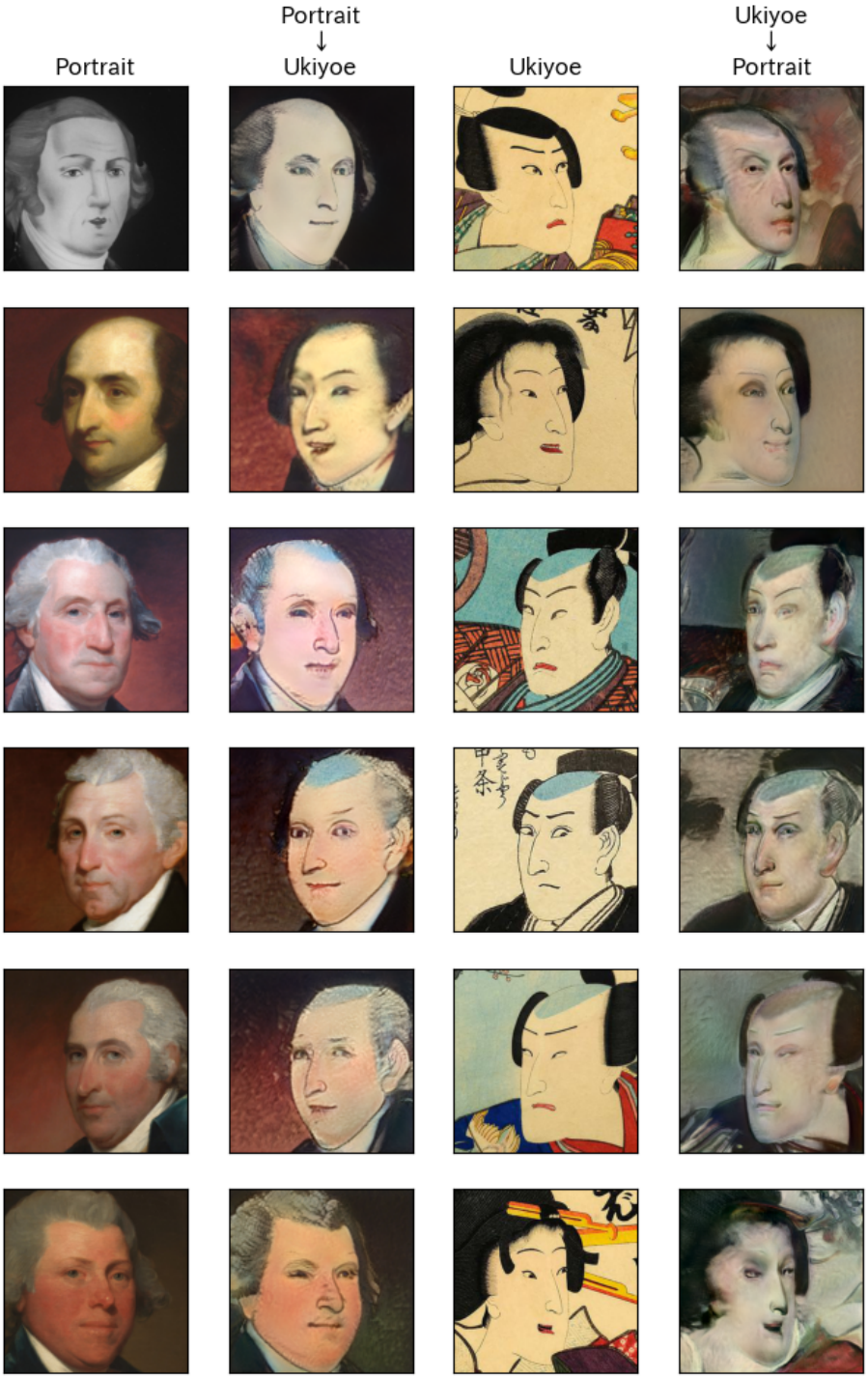
付録

その他比較例 ©浅月 舞 ©愛田 真夕美



Mixing による「やまとの羽根」と「太陽にスマッシュ！」の変換 ©咲 香里 ©あゆみ ゆい

Mixing による「プリズム・ハート」と「ハイスクール！奇面組」の変換 ©浅月 舞 ©新沢 基栄



Mixing による浮世絵 [32] と肖像画 [33] の変換