筑波大学大学院博士課程

システム情報工学研究科修士論文

実写背景画像のアニメ風変換のための ニューラルネットワークによる 自動画風変換手法の評価

李 驍

修士（工学）

(コンピュータサイエンス専攻)

指導教員 金森 由博

２０１８年　３月

概要

アニメやゲームの制作では、風景画像や屋内画像など、様々な背景画像が使われている。その数は一作品あたり数百枚で、アーティストがペイントソフトなどを用いてゼロから手作業で制作しており、専門知識と高度な技術が必要とされる上に時間がかかる。背景画像制作のコストを削減するひとつの方法として、写真からアニメ風背景画像を半自動で生成することが考えられる。入力が写真であるため、ゼロから背景画像を作るよりも簡単かつ効率的だと考えられる。しかしこの手法では、複数の画像処理ソフトウェアを用いて対象画像ごとに各ソフトウェア上でフィルタの適用範囲やパラメータなどを調整することが必要であり、大量の背景画像を処理すると、時間がかかる。そこで、入力写真のシーンに合った画像処理を、完全に自動で推定・適用する枠組みが望ましいと考えられる。

本研究では、1 枚の実写背景画像からアニメ風背景画像への全自動変換を実現することで、制作の効率化を目標とする。そして自動化の可能性を検討するため、アニメ風背景画像における、既存の画風自動変換手法はどこまでできるか、また何か足りないかについて、それらの手法を調査・整理し、実験で評価する。

本研究で検討するアプローチは、過去にアーティストが制作したアニメ風背景画像を基に、1 枚の新たな入力写真に適切な画像変換を施し、背景画像を生成する、というデータ駆動型の手法である。特に Deep neural network (DNN) を利用した style transfer という画風変換手法が顕著な成果を収めている。本論文は、この画風変換手法に注目し、既存の 5 つの画風変換手法を用いて実験を行い、結果を考察する。実験のデータは、実写背景画像とアニメ風背景画像のペアからなるデータセットである。また実験結果を用いて、どの手法の結果がアニメ風背景画像に近いと実写背景画像に近い、2 つの基準に基づく、アンケート調査を行った。アンケート調査の結果から見ると、異なるシーンによって、それぞれの手法の結果が最もアニメ風背景画像に近いと判断されることがある。これにより、実写背景画像からアニメ風背景画像への自動変換タスクに対して、シーンによって、異なる対応ができる柔軟性があるシステムが望ましいと考えられる。

目次

図目次

表目次

第1章　　序論

1.1　研究の背景

1.1.1　　アニメ風背景画像の応用

日本では、平面に描かれた絵を使用するセルアニメーション（アニメ）が標準である。アニの制作工程では、キャラクター制作とアニメ風背景画像制作の二つのプロセスがある。アニメ風背景画像では、風景画像や屋内画像などの異なるシーンでは異なる背景画像が使われるため、一作品ごとに大量の背景画像（数百枚以上）が必要になる。またアニメ風背景画像はアニメだけではなく、ノベルゲームの中でも使われている。例えば、図１-1 に背景画像とキャラクターを組み合わせて作成したノベルゲームの例を示す。

図 1-1　背景画像とキャラクターを組み合わせて作られたベルゲームの例

画像の出典:「君の名は。」© 2016「君の名は。」制作委員会.

アニメ風背景画像は、作品によって様々なスタイルがある。例えば、図 1-2 に示すように、「ちびまる子ちゃん」という作品（左）のような漫画風スタイルの背景画像と右、「君の名は。」

漫画風スタイルの背景画像

実写的なアニメ風背景画像

図 1-2　異なるスタイルのアニメ風背景画像. 画像の出典: 左「ちびまる子ちゃん」© 1995 フジテレビと右「君の名は。」© 2016「君の名は。」制作委員会.

という作品（右）のような実世界のシーンに近く、写実的なスタイルの背景画像がある。本研究では、写実的なスタイルのアニメ風背景画像に注目し、研究を行う。

1.1.2　　アニメ風背景画像の制作

アニメ風背景画像の制作プロセスは図 1-2 に示した 3 つの段階で構築される。まず鉛筆で専門の紙の上に背景コンテンツの線画を描く。次は紙上に描かれた線画をスキャナーに取り込み、パソコン上で線画に着色する。最後に、陰影や質感など様々な表現効果を加える。また紙上ではなく線画を直接液晶タブレット内のペイントソフトで描くこともある、パソコンと制作・管理用ソフトの性能を向上することで、アニメ制作の効率が向上し、商業用アニメの主要な制作手法となった。しかし、今までこの 3 つの段階はすべてアーティストが手描きで行われている。これは人海戦術的な方式でありながら、技術が必要であり、時間的なコストが高いという問題がある。またアニメーションの市場需要の増加に伴って、アニメ風背景画像の需要量も増え続けている。一方で、労働集約的なアニメ風背景画像の制作の現場は、労働力不足と制作費用の減少などが問題になっており、これまでの制作手法にも効率的な限界が見えてきている。この問題を解決するため、より効率的なアニメーションの制作環境が必要だと考えられる。

線画作成　　線画の着色　　陰影や質感の追加

図 1-3　アニメ風背景画像の制作フロー.

近年、アニメ風背景制作のコストを削減するひとつの方法として、写真からアニメ風背景画像を半自動で生成する方法が登場している。その例として、図 1-4 に示す Photo-Dramatica という写真加工技術がある。この方法では、画像編集ソフトを用いて、写真に色変換、領域フィルタリング、ハイライト付加とぼかしなど単純な複数の画像処理を適用して、アニメ風の画像を生成する。入力が写真であるため、線画と着色に関わる様々な作業が不要になり、ゼロから背景画像を作るよりも簡単かつ効率的だと考えられる。しかしこの手法では、複数の画像処理ソフトウェアを用いて対象画像ごとに経験則に基づいた作業が必要である。具体的には各ソフトウェア上でフィルタの適用範囲やパラメータなどを調整する必要がある。現在、労働集約的なアニメ制作の現場は、慢性的な労働力不足と制作費用の減少が問題となっていることからも、完全に自動で入力写真からアニメ風画像を生成することが望ましいと考えられる。

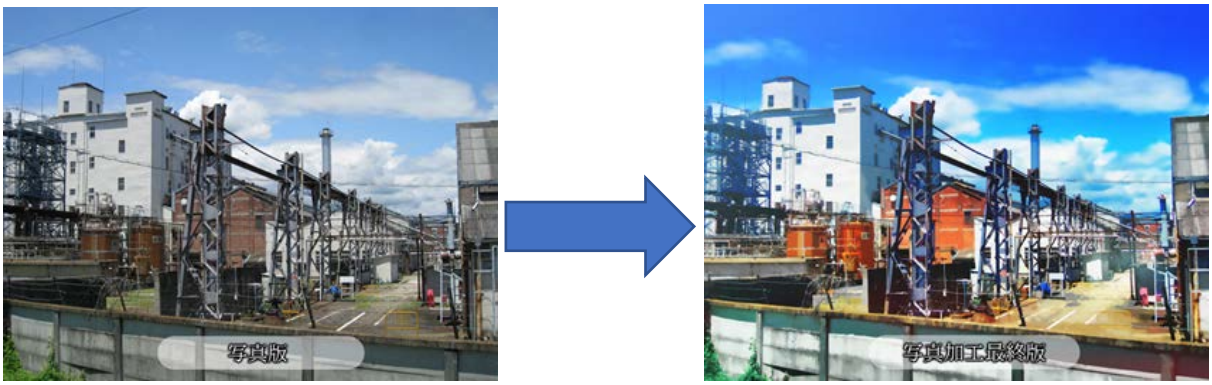


写真　　加工された画像

図 1-4　PhotoDramatica に基づいた加工例.

1.2　研究の目的

本研究の目的は、1 枚の実写背景画像からアニメ風背景画像への自動変換手法を確立し、アニメ用背景画像の効率的な制作システムを構築する。図 1-5 に理想的なシステムを示す。ここで、実写背景画像 1 枚(左)を入力し、自動計算によってアニメ風背景画像(右)が出力される。本研究では、研究の端緒として、画像のスタイルや雰囲気を自動的に変換する従来の画風変換手法を調査・整理し、アニメ風背景画像への応用が可能であるかを検討・評価する。本研究では、既存手法の結果を考察することによって、本タスクに興味を持つ他の研究者にとって研究開発の参考になることができると考えている。また既存の画風変換手法の結果が自然かどうかの評価は難しく、目的ごとの評価が必要だと考えられるため、本研究は既存手法をまとめ、評価方法の一例として挙げられる。

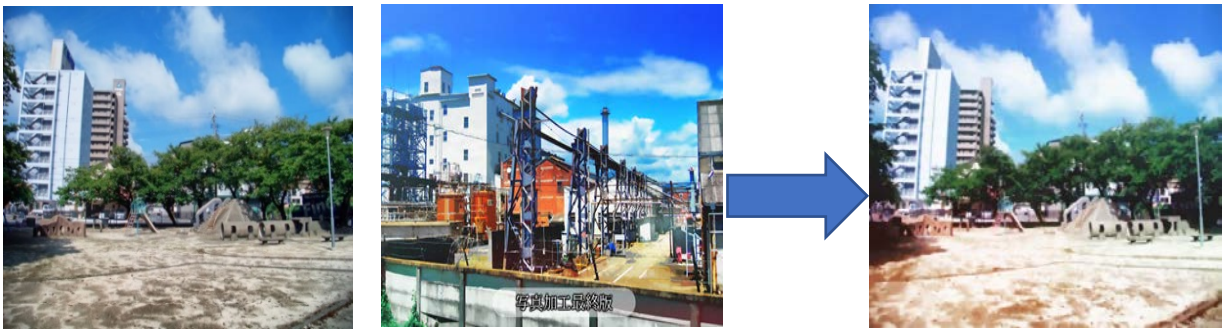
入力　　出力（イメージ）

図 1-5　アニメ風背景画像の全自動生成イメージ.

1.3　アプローチと評価手法

1.3.1　　データ駆動型のアプローチ

本研究で検討しているアプローチは、過去にアーティストが制作した背景画像のデータセットを基に、1 枚の入力写真に適切な画像変換を施し、アニメ風背景画像を生成する、データ駆動型の手法である。特に Style Transfer という画風変換手法を注目している。このアプローチの仕組みを図 1-6 に示す。コンテンツ画像とスタイル画像の 2 枚を入力して、コンテンツ画像にスタイル画像のような色合いや雰囲気を転写する。現在まで、5 つの画風変換手法を実験し、結果を考察した。もう一つは図 1-7 に示すスタイル画像なしのアプローチであり、事前に多くの実写背景画像とアニメ風背景画像のペアデータを用意し、学習モデルを用いて、実写背景画像からアニメ風背景画像への変化パターンを学習する。このアプローチは豊富な情報に基づいて画風を変換できるが、事前に画素の対応関係があるペアデータを用意することが困難である。また学習に多くの時間がかかると考えられる。本研究は、主にスタイル画像ありのアプローチを注目し、実験を行う。



コンテンツ画像　　スタイル画像　　出力（イメージ）

図 1-6　Style Transfer のアプローチの仕組み.

1.3.2　　評価手法

本研究では、5 つの既存の画風転写手法と、実写背景画像とアニメ風背景画像のペアで構成される 2 種類のデータセットを用いて、実験を行う。そして、それらの結果画像を比較し、複数の基準に基づき評価を行う。実験結果の詳しい内容は第 4 章で述べる。また、どの手法の結果が本タスクに対して、一番良い結果だと感じるかについて、アンケート調査を行う。アンケート調査の内容について、設問毎に実写背景画像、アニメ風背景画像と 5 つの手法の結果画像を並べて、ユーザが 5 つの手法の結果にランクを付ける。ユーザテストは、どの結果が良いかについての投票だけではなく、ユーザの投票の統計データを用いて、文献[31]による統計手法で、手法毎の平均値を計算する。ユーザテストの詳しい内容は第 5 章に説明する。アンケートのすべての設問の統計結果を付録に記載する

1.4　本論文の構成

本論文は本章を含め全 6 章と付録で構成される。第 1 章では序論として、本研究の研究背景や研究目的を説明する。第 2 章に本研究と関連した研究をまとめ、紹介する。第 3 章では本研究の検討対象になった既存手法の詳細内容と特徴を説明する。第 4 章では既存手法を用いた実験結果について、第 5 章ではその結果に対する評価とユーザテストについて示す。第 6 章では本論文のまとめと今後の展望について示す。なお、第 4 章の結果は、アンケート調査で用いた画像と統計データの一部であるため、すべての結果画像と統計データは付録に掲載する。

1.5　本論文の用語

ここでは本論文で用いる用語について説明する。

コンテンツ画像: 画風を変換する際に、画風や雰囲気を変換したい画像。
スタイル画像: 画風を変換する際に、近づきたい画風や雰囲気に対して、参照になる画像。
聖地巡礼: アニメの舞台となった土地や建物など、ファンにとって思い入れのある場所が「聖地」と呼ばれる。こうした「聖地」を実際に訪れ、憧れや興奮に思いを馳せることを、「聖地巡礼」と称される。

第2章　関連研究

本研究では、データ駆動型の画風変換手法を、スタイル画像あり・なしの 2 種類に分類する。またスタイル画像ありの手法の中に、deep neural network (DNN)を利用する手法がある。近年、DNN の発展に伴って、画像処理に大きく影響を与えている。DNN を多くの画像処理タスクに適用することにより、人間と同程度かそれ以上の高精度を実現できるようになった。そのような DNN モデルの中でも、「畳み込みニューラルネットワーク（Convolutional Neural Network, CNN)」は、視覚野の特徴抽出の仕組みをモデル化したもので、画像解析において高い性能を発揮している。CNN は、畳み込み演算（Convolution）による画像特徴量の抽出とプーリング（Pooling）と呼ばれるダウンサンプリング処理を行い、何層にもわたって積み上げられたネットワークから構成される。人間の手を介さずネットワークの学習を通して画像特徴量を自動抽出できるようになったことで、既存手法を著しく上回る精度を実現できるようになった。その中に、CNN を使って、コンテンツ画像の物体の形状や構造を保ったまま、コンテンツ画像にスタイル画像のテクスチャや質感などのスタイルを転写するという画風変換手法が発表されており、顕著な成果を収めている。本章では、画風変換に関する研究の中で本研究に関連するものを紹介する。

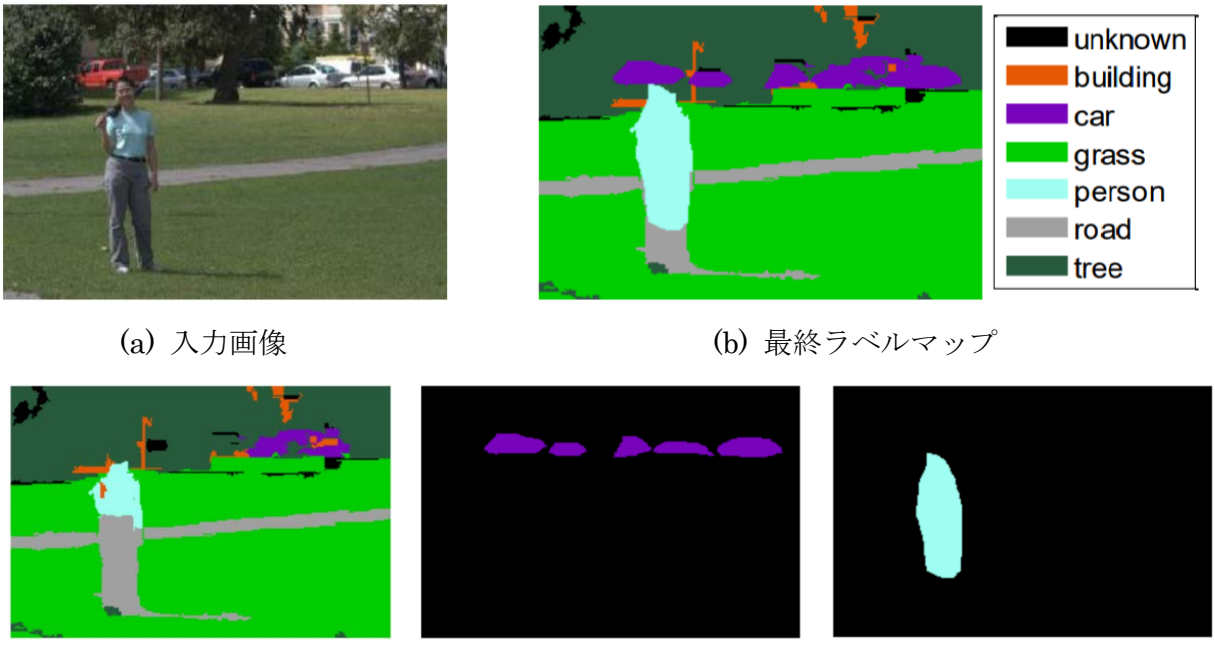
まず、2.1 節では、スタイル画像の入力が必要なく、1 枚のコンテンツ画像のみを入力とし自動的に画風を変換する研究を紹介する。2.2 節では、コンテンツ画像とスタイル画像を入力し、画風を変換する研究を紹介する。

2.1　スタイル画像なしの手法

アーティストが画像編集ソフトを用いて、写真にフィルタリングや色変換などのエフェクトを適用することで、写真のスタイルを変えることができる。しかし、一枚ずつ手作業でエフェクトを写真にかけるのは非常に時間がかかる。Yan らは、プロのアーティストが施した画像のエフェクトをエフェクトごとに学習し、自動的に入力画像に適用する手法[3]を提案した。画像のエフェクトは、画像中の物体や意味的内容に依存し、空間的に変化する。彼らの手法では、画像の領域分割後の領域ごとに自動で画風変換を施す。さらに画像を superpixel に分割し、画像の大域的特徴量、局所コンテキスト特徴量、画素単位の特徴量に基づいてエフェクトを学習する。Superpixel とは似た傾向を持つ画素をひとまとめにした領域である。変化の激しい画素単位の色変化をモデル化すると出力画像にノイズが現れやすいため、superpixel 単位で、色の変換関数を Lab 色空間における一次式あるいは二次式として表現し、その係数行列を学習する。

学習に用いる特徴量について説明する。大域特徴量には既存研究で示された 6 つの大域特徴を用い、計 207 次元の特徴量を得る。画素単位の特徴量は 3×3 画素の色の平均値と正規化された座標である。局所コンテキスト特徴量の計算にはまず、図 2-1 に示す意味的ラベルマップの構築が必要である。ラベルマップの構築手順は次の通りである。まず既存のシーン領域分割手法を用いて、入力画像(図 2-1(a))における意味ラベルを推定する（図 2-1(c)）。しかしこれだけではシーン中の物体の検出精度が高くないため、これとは別途、既存の物体検出手法を用いて、シーン中の物体を検出する（図 2-1(d)および(e)）。そして後者の信頼度の高いラベルで前者の意味ラベルを上書きする。さらに、そのままではラベルのノイズが多いので、superpixel 単位でラベルを投票してノイズを除去し、図 2-1(b) に示したラベルマップを出力

する。そして画素ごとにその画素の周囲を階層的に調べてラベルのヒストグラムを作成し、局所コンテキスト特徴量とする。Yan らの手法では、データセットに様々な前処理が必要である。しかし、前処理の情報が公開されていないため、作者が提供したデータセットしか使えず、適用できるスタイルが限られる。これにより、本研究の目的であるアニメ風スタイルへの変換ができないため、検討対象から除いている。



(a) 入力画像　(b) 最終ラベルマップ

(c) シーン解析結果　(d) 車検出結果　(e) 人物検出結果

図 2-1　意味的ラベルマップの構築．図は文献［3］より引用．

Taigman らは Deep Convolutional Generative Adversarial Networks(DCGAN)を用いて、教師なし学習で顔写真を似顔絵に変換する手法[4]を提案している。人の顔は複数のドメインで構築されると見なす。例えば、目や鼻などのパーツである。異なるドメイン間で類似点を見つけて、対応関係を作る。そして一つのドメインから別のドメインに要素を変換する関数を学習する。この手法は顔写真を似顔絵に変換する手法であるが、実写写真からアニメ風背景画像への変換も可能だと考えてられる。またドメイン変換を学習するため、ペアになる訓練データはたくさん必要があるが、実際にこれらのペア画像がなかった場合が多いである。ペア画像がなくても、学習を行うため、Zhu らは circle consistency 損失を GAN に導入した手法[1]を提案した。さらに、これまでの手法は一枚入力画像に対して、1 つの出力結果をしか得ることができないが、決められたタスクについて、同時にすべて可能になる出力結果を得るため、Zhu らは隠れた中間層を出力画像と連結した手法[2]を提案した。

画風変換の手法ではないが、Long らは畳み込み（convolution）と逆畳み込み(deconvolution) モデルからなる Fully convolutional networks (FCN) モデルを用いて、画素ごとのラベルを予測する手法［5］を提案している。彼らの FCN モデルは汎用的であるため、出力を変えればアニメ風背景変換にも利用できると考えられる。Long ら[5] の FCN モデルは、元々は入力画像を意味的な単位で領域分割する semantic segmentation のために提

案された。既存の CNN モデルにおいて画素単位で意味ラベルを推定するには、superpixel を用いるなど前処理や後処理が必要であった。それに対し、FCN モデルは end-to-end、つまり生の入力画像からこのモデル以外の前処理や後処理なしに、画素単位で最終的なラベルを出力できる。

FCN モデルでは、クラス分類に用いられる既存の CNN モデルを利用する。具体的には、学習済み CNN モデルの畳み込み層をそのまま FCN モデルの畳み込み層とし、その後ろに入力画像の解像度にアップサンプリングする逆畳み込み層を加える。そして、このモデルを fine-tuning して領域分割結果を出力する。さらに推定精度を向上させるため、浅い層と深い層を組み合わせた skip 構造を導入している。モデルの全体像を図 2-2 に示す。図 2-2 中の垂直線は畳み込み層、格子状の四角形はプーリング層、s はストライドの大きさを表す。

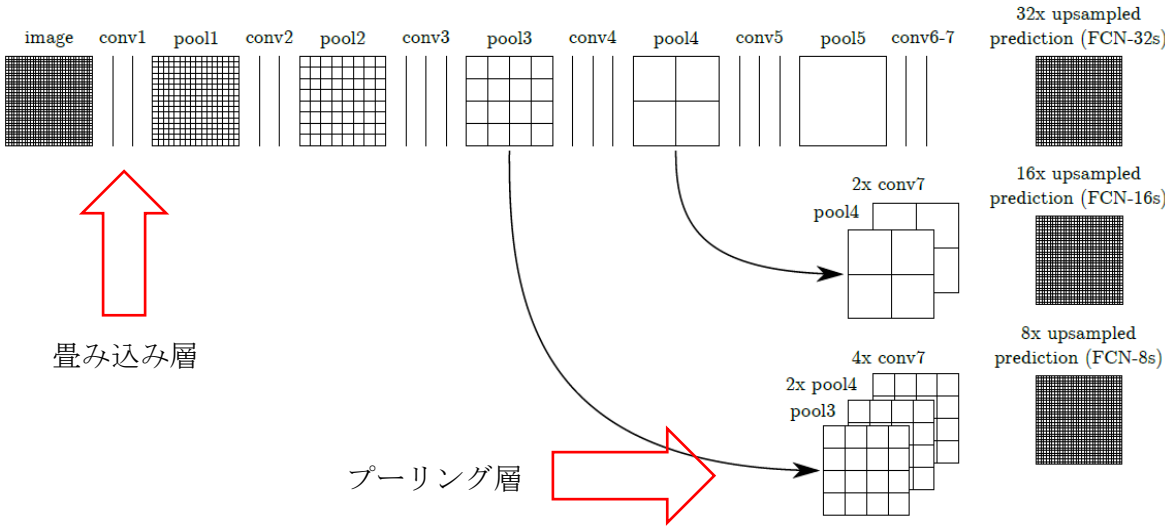


図 2-2　FCN モデルの Skip 構造．文献［5］より引用した図に加筆．

スタイル画像なしの手法では、実写背景画像からアニメ風背景画像への変換を学習するため、画素を厳密に対応している実写背景画像とアニメ風背景画像のペアデータが必要あると考えられる。このようなデータセットを用意するのは困難であるため、本研究では主にスタイル画像ありのアプローチを注目している。

2.2　スタイル画像ありの手法

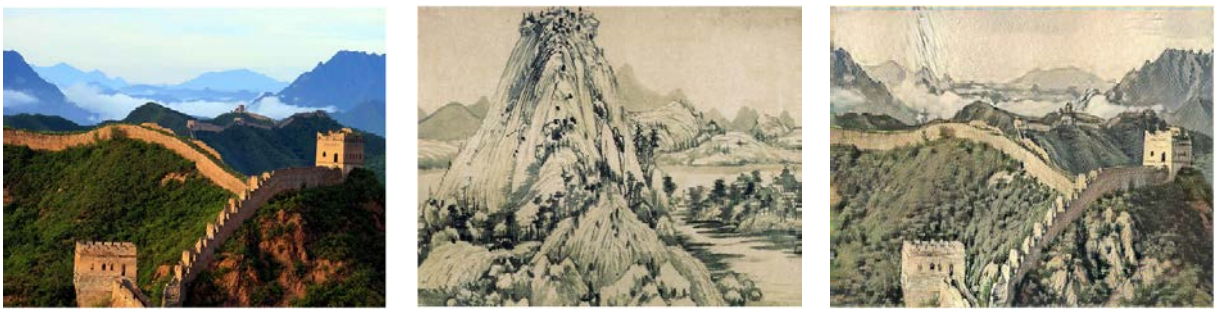
スタイル画像ありの代表的な画風転写手法として、Hertzmann ら［6］の image analogies という画風転写手法が挙げられる。この手法は図 2-3 に示したように、スタイル画像ペアの A と A’の 2 枚の画像から、A を A’へ変換するようなフィルタを学習し、画素単位で学習した変換フィルタを入力画像 B に適用して結果画像 B’を得る。この手法では、入力データとしてスタイル画像ペア A、A’と変換したい画像 B の 3 枚画像が必要である。Chang らは、画像を領域分割し、領域単位で入力画像とスタイル画像の対応を取った上で画風を転写することで、Hertzmann らの手法[6]を改良した[7]。また、Chang らはスタイル画像のデータベ

ースを構築することで、ユーザが事前にスタイル画像を用意しなくても、データベース内の画像をスタイル画像として画風を転写できるようにした。さらに最近、Hertzmann らの手法[6]をベースとし、Liao らは意味的な領域の対応関係を考慮し、CNN から画像の特徴を抽出する手法[35]を提案した。

A　　A’　　B　　B’

図 2-3 Hertzmann らの image analogies 手法. 図は文献 [6] より引用.

この数年の間に、画風変換において CNN ベースのアプローチが提案され、非常に良好な結果が報告されている。CNN は層が進むにつれて、タスクにとって重要な特徴量を強調するように情報処理が進んでいくと言われている。物体認識タスクについて訓練したネットワークの深い層では、多少色合いや質感が変わっても同じ物体とみなしてほしいため、形状の情報が強まり、色合いや質感の情報が弱まる傾向がある。この弱まっている部分を、別の画像の色合いや質感に置き換えることで、画像の形状情報を残しまま、画風を変換することが可能である。CNN による画風変換の代表的な手法として Gatys ら[8]の手法が挙げられる。ユーザがコンテンツ画像とスタイル画像を一枚ずつ入力すると、コンテンツ画像の形状情報を残しつつ、スタイル画像のスタイルに自動変換して出力する。この手法を利用すると、例えば、図 2-4 に示すように、写真をアーティストが手描きした絵のようなスタイルに変換できる。この手法が、Neural Style Transfer 分野の先駆けとなり、アカデミアの世界に注意を集めて、現在では多くの改善や発展した手法が提案されている。本研究の一つの貢献として、それらの手法をまとめて、評価する。本研究では、これら既存手法の特徴に応じて、画像の最適化に基づく手法とモデルの最適化に基づく手法に分類する。前者は、ネットワークモデルを最適化するのではなく、直接的に画像を最適化する。後者は、画像生成を行うネットワ



コンテンツ画像　　スタイル画像　　出力結果

図 2-4　 Gatys らの手法による画風変換の例. 図は文献 [34] より引用.

ークモデルを最適化し、都度画像を最適化する必要がない。2.2.1 節で画像の最適化に基づく手法の研究を紹介し、2.2.2 節でモデルの最適化に基づく手法の研究を紹介する。また既存の画風変換手法をベースとして、少し改善した手法を 2.2.3 節に紹介する。

2.2.1 画像の最適化に基づく手法

本節では、画像の最適化に基づく関連研究を紹介する。このアプローチでは、生成画像の初期値をランダムなノイズとし、生成画像中の物体の形状がコンテンツ画像に近く、生成画像のスタイルが参照画像に近くなるように、画素値を更新する。この時、生成画像とコンテンツ画像間、生成画像とスタイル画像間それぞれで損失を定義し、これらを最小化するため逆伝搬法を用いる。Gatys らは画像の画風情報を表現するため、グラム行列という概念を導入し、生成画像のグラム行列を参照画像のグラム行列に近づけることによって、スタイルの転写を実現した。ここのグラム行列は同じ中間層の各チャネル間の相関を計算したものである。この手法では、グラム行列間の誤差関数を画像間のスタイル損失関数として定義し、最小化している。この他にも様々なスタイル損失関数が提案されており、代表的な手法として、Maximum Mean Discrepancy(MMD)をベースにした Li ら手法[9]と Markov Random Fields (MRF)をベースにした Wang らの手法[10]が挙げられる。MMD はカーネル平均場を用いたノンパラメトリックな分布間距離を表し、検定に用いられる。

MMD をベースした手法

MMD は、2 つの分布がヒルベルト空間上の特徴の平均値を求めることにより、2 つの分布の類似性を検出するため、よく使われる行列である。Li らはコンテンツ画像とスタイル画像を二つの分布と見なし、画風変換がこの二つの分布の配向プロセスと等価であることを証明した。また Li らの手法[9]は画像の再構築とテクスチャ合成のアルゴリズムをベースとし、画風を変換した画像を生成することができる。画像の再構築のプロセスは既存の Mahendran ら[11]の画像から画像の特徴情報を表現する手法の反転になることも考えられる。したがって、画像を再構築する時に画像の特徴を表現する情報だけ反映される。Gatys ら[8]と Li らの手法[9]は、VGG モデルの中間層から画像の特徴情報を再構築することによって、CNN がコンテンツ画像からの意味的なコンテンツ情報とスタイル画像からのテクスチャや質感などの画風情報を抽出することができると示す。生成画像はコンテンツとスタイルの二つの要素が構成される。コンテンツ要素はコンテンツ画像とスタイル画像から抽出した異なる表現情報にペナルティ関数をかけて、生成画像のコンテンツ情報を更新する。スタイル要素は生成画像とスタイル画像の間にテクスチャなどの特徴情報をマッチングする。これにより、彼らの手法は画像の再構築とテクスチャ合成を結合した手法とも考えられる。またスタイル要素のプロセスは既存のテクスチャ合成手法[12]の画像からテクスチャ情報を捉えていると見なすことができる。

VGG モデルは画像認識タスクのために訓練されたモデルである。VGG モデル以外の画像認識用のモデルを用いて、画風を変換することもできる。例えば、ResNet モデルを用いた手法[13]が提案されている。また、total variation にノイズ除去項を追加することによって、生成画像の品質を向上できる。Gatys ら[8]の手法は不安定があり、手動でパラメータを調整することが必要である。例えば、CNN の上に平均値と分散が異なる特徴領域は同じグラム行列を持つ可能性がある。この不安定性を解決するため、Risser らは、Gatys らが提案した損失関数にヒストグラムの損失項を追加した[14]。ヒストグラムの損失は損失関数を最適化する

時に特徴領域の全体のヒストグラムを保存することで、特徴領域の平均値と分散も保存することができる。また Gatys ら手法が手動のパラメータ調整を必要とするのに対し、Risser らは自動的にパラメータを調整する手法を提案した。Gatys らの手法[8] は画風変換において目覚ましい成果を上げているが、彼らの手法は画風を変換する時に、画像の意味的な領域情報を考慮していない。この問題に対応するため、Yin らは画像の領域を分割し、画風を変換する時に意味的な領域情報を考慮した手法[15]を提案した。さらに Yin らの手法[15]をベースに、空間的な対応関係と高次元の特徴情報を制約する Chen らの手法[16]が提案され、生成画像の品質を向上した。Gatys らの手法[8]では空間的な歪みが多く発生し、手描きのような画像が生成される。そこで、より写実的な画像を生成するため、Luan らは意味的な領域情報を考慮すると同時に、画風変換操作を色空間上のみで実行し、画像の空間的な歪みを抑制した[17]を提案した。次の節には、MRF をベースした手法を紹介する。

MRF をベースした手法

MRF はマルコフ確率場モデルとも呼ばれ、統計手法に基づいて、画像復元、領域分割、テクスチャ解析など、画像を表現するため、よく用いられるモデルの一つである。このモデルは画像の局所的なパッチに対して、確率的に最も関連するパッチが存在することを仮定している。MRF をベースした画風変換手法は局所的なパッチを操作することによって、画像の画風を変換する。Wand らが初めて MRF を画風変換に導入し、MRF をベースした画風変換手法[10]を提案した。Wand らは、Gatys らの手法がスタイル損失を計算する時に画素毎の特徴量の相関だけを捉え、空間レイアウトを制約していないことを指摘した。これによって、Gatys らの手法の結果は写実的なスタイルについて、視覚的な妥当性が足りないである。そして解決方法として、Wand らは MRF を用いて、スタイル損失関数を再定義することにより、パッチベースの画風変換手法を提案した。また Wand らの手法[10]をベースに、さらに発展させた Champandard の手法[18]が提案された。Champandard らは入力画像の意味的な領域情報を考慮した分割マップを用いて、画風を変換する。分割マップの作成方法については、既存のラベル割り当て手法で自動的に作る方法とユーザが手動で領域分割マップを作る方法の 2 つが提供された。パッチベースの手法として、Chen らはパッチ毎にスタイル画像の特徴量をコンテンツ画像の特徴量と入れ替えることで、画風を変換する手法[19]を提案した。この手法では、パッチ毎の特徴量を用いて計算するため、Gatys らの画素毎に計算する手法より、画像の出力時間が短くなる。

Gatys らの手法[8]をベースにした改善手法

Novak らは Gatys らの手法のスタイルの情報に加え、画像の空間構成情報を捉える、新しいスタイル表現手法[25]を提案した。またフレームワーク、ネットワーク、CNN の層を変更し、パラメータの調整など複数の実験設定で実験を行い、結果を考察した。Gatys らの手法では、複雑なスタイルのパターンの転写が困難であるという問題点がある。この問題を解決するため、Novak らは CNN の異なる層間でのグラム行列を用いる改善やより多くの層の特徴情報を使う改善などを行った[26]。

最近、Gatys ら自身でも画像の色、スケール、空間情報の 3 つ知覚要素をコントロールすることによって、柔軟性が高い画風変換手法を提案している[27,28]。空間情報のコントロールについて、ガイダンスチャンネル Tr をコンテンツ画像とスタイル画像に導入し、それぞれの領域に値[0,1]を与える。そしてコンテンツ画像のガイダンスチャンネル Tr 値が 1 の領域

にスタイル画像のガイダンスチャンネル Tr 値が 1 の領域のスタイルを転写する。図 2-5 に例を示す。コンテンツ画像とスタイル画像の右上の図はガイダンスマップを表す。まず、ガイダンスマップに基づき、コンテンツ画像の空の領域をスタイル画像 2 の空の領域に、コンテンツ画像の建物の領域をスタイル画像 1 の建物の領域に対応付ける。そして対応している領域間だけで画風変換を行い、2 つのスタイル画像のスタイルを同時にコンテンツ画像に適用する。また輝度チャネルだけで、画風変換を行い、色の転写をコントロールすることができる。具体的には、ユーザがコンテンツ画像の色を保ったままで、スタイル画像のテクスチャをコンテンツ画像に転写することができる。スケールのコントロールについて、down-sampling と up-sampling を行う時に生成画像のブラシサイズのスケールを制御することによって、異なるスタイルを結合することができ、新たなスタイル特徴を獲得することができる。

Gatys らの手法[8]のもう一つ問題点として、画像のデプス情報を考慮していないため、生成画像の奥行き感が損失する点が挙げられる。Liao らは目的関数にデプス損失関数を追加し、デプスを考慮した画風変換手法[29]を提案している。デプス推定について、既存のデプス推定手法[30]のアルゴリズムを用いた。また Gatys らの手法[8]の画風変換結果は、コンテンツ画像の物体や構造が欠けるとスタイル画像の構造が出る場合が多い。この問題点を改善するため、Li らが Laplacian 損失を損失関数に導入し、コンテンツ画像の物体エッジ情報を考慮した手法[31]を提案している。

図 2-5　空間をコントロールした画風変換. 図は文献［27］より引用.

2.2.2 モデルの最適化に基づく手法

前節に述べた画像の最適化に基づく手法は良い結果を得ることができるが、一枚の画像の生成に時間が掛かる。これに対し、モデルの最適化に基づく手法は画像の生成時間が非常に短く、高速化した画風変換手法とも言える。画風変換を高速化するためのキーアイディアとして、事前に異なるスタイルのスタイル画像毎に大量の画像データセットを用いて、ネットワークをトレーニングする。その後、訓練したモデルにコンテンツ画像を入力し、順伝搬の計算を行うことで、画風変換された画像を生成する。

高速な画風変換手法として、Johnson らの手法[20]が挙げられる。Johnson らが提案したシステムは画像生成ネットワークと損失ネットワークの２つのネットワークで構成される。画像生成ネットワークには Radford らのネットワーク[21]を用いた。目的関数について、Johnson らは Gatys らの手法[8]の目的関数をベースし、ノイズ除去項を追加する。ほぼ Johnson らと同時、Ulyanov らはテクスチャネットワークによる高速化手法[22]を提案した。Ulyanov らの手法は Johnson らの手法と似ているが、multi-scale 構造の画像生成ネットワークを使点が Johnson らの手法と異なる。また Wand らは前節に述べた MRF をベースした手法[10] をベースに、Markovian feedforward network をトレーニングする高速化手法[23]を提案した。モデルの最適化に基づく手法は feed-forward ネットワークをトレーニングすることによって、画像変換の高速化を実現した。しかし、この手法はスタイルが異なるスタイル画像に対して、その都度トレーニングをする必要があるため、事前準備の時間的コストが高い。また 1 つのトレーニングしたネットワークは 1 種類の画風しか変換できないため、手法の柔軟性が低い。この問題の解決案として、複数のスタイルを同時に学習する Dumoulin らの手法[24]が提案されている。Dumoulin らの手法は絵画が時に同じストロークで、異なる色を用いて描かれるという経験則に基づき、複数のスタイル画像が同じに計算を共用し、別々の feedforward network をトレーニングするというアプローチをとっている。このアプローチにより、Dumoulin らは同じタイプの複数のスタイルを学習するため、条件付きインスタンスの正規化をベースとし、条件付きの画風変換ネットワークを提案した。スタイル画像のスタイル毎に正規化を行い、アフィン変換のパラメータを調整することによって、スタイルに条件を付ける。

Ulyanov らの手法[22]をベースにした改善手法

Ulyanov らは画風を変換する時に batch normalization を instance normalization に変更した手法[32]を提案している。また Julesz texture ensemble を用いて、偏りのないサンプリングを行い、生成画像の多様性を向上した手法[33]が提案されている。

本研究では、実写背景画像のアニメ風変換の自動化を検討しており、この目的の下では、Gatys ら[8]、Li ら[9]、Chen ら[19]、Johnson ら[20]および Luan ら[17]の 5 つのスタイル画像ありの手法を実験し、本タスクの実現可能性を検討・評価した。これらの既存手法についての詳細は第三章で説明する。2.1 節に述べたスタイル画像なしのアプローチは図 2-6 に示すように、事前に多くの実写背景画像とアニメ風背景画像のペアデータを用意し、学習モデルを用いて、実写背景画像からアニメ風背景画像への変化パターンを学習する。このアプロ

ーチは豊富な情報に基づいて画風を変換できるが、事前に画素の対応関係があるペアデータを用意することが困難である。また学習に多くの時間がかかると考えられる。本研究は、主にスタイル画像ありのアプローチを注目し、実験を行う。

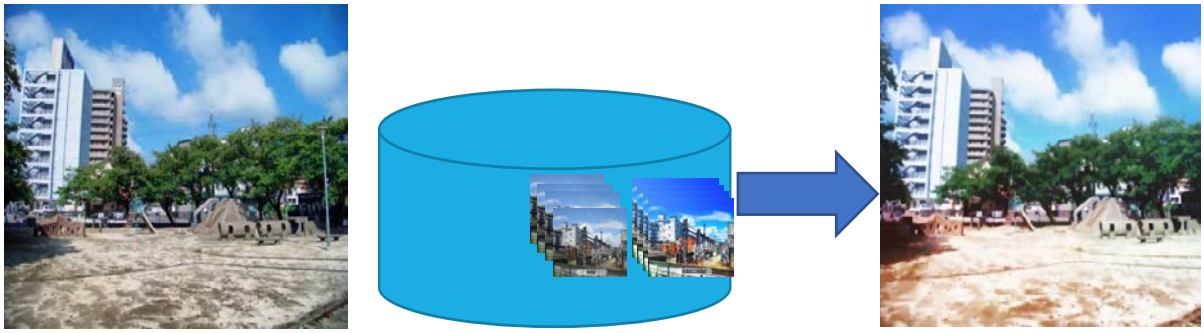


図 2-6　スタイル画像なしのアプローチの仕組み.

第3章　検討対象の既存手法

3.1　Gatys らの手法 [8]

CNN によるスタイル画像ありの代表的な手法として Gatys ら[1]の手法が挙げられる。Gatys らは学習済の CNN モデルを使って画風を変換する手法を提案した。この研究のキーアイディアは画風の情報をグラム行列によって表現したことである。ここでグラム行列は、同じ中間層の各チャネル間の相関を計算した行列である。例えば、グラム行列は赤と緑の相関など、画像全体でどんな色が使われているかという情報を表す。生成画像とスタイル画像間のグラム行列の差をスタイル損失 $\mathcal{L}_{\text{style}}$、ネットワークの深い層での生成画像とコンテンツ画像間の差をコンテンツ損失$\mathcal{L}_{\text{content}}$とし、これらの線形和で損失関数を以下のように定義する。

$$\mathcal{L}_{\text{total}} = \alpha\mathcal{L}_{\text{style}} + \beta\mathcal{L}_{\text{content}} \tag{1}$$

ここで、 α と β はコンテンツ損失とスタイル損失の重みである。$\mathcal{L}_{\text{content}}$はある層のコンテンツ画像の特徴量と生成画像の特徴量を比較する。$\mathcal{L}_{\text{style}}$はある層のスタイル画像の特徴量と生成画像の特徴量を比較し、マッチングする。この損失関数を最小化するために勾配降下法で最適化を行い、画像を更新する。層毎のスタイル損失$\mathcal{L}_{\text{style}}$の計算式を以下に示す。

$$\mathcal{L}_{style}^{l} = \frac{1}{4N_l^2M_l^2}\sum_{i=1}^{N_l}\sum_{j=1}^{N_l}\left(G_{ij}^{l} - A_{ij}^{l}\right)^2 \tag{2}$$

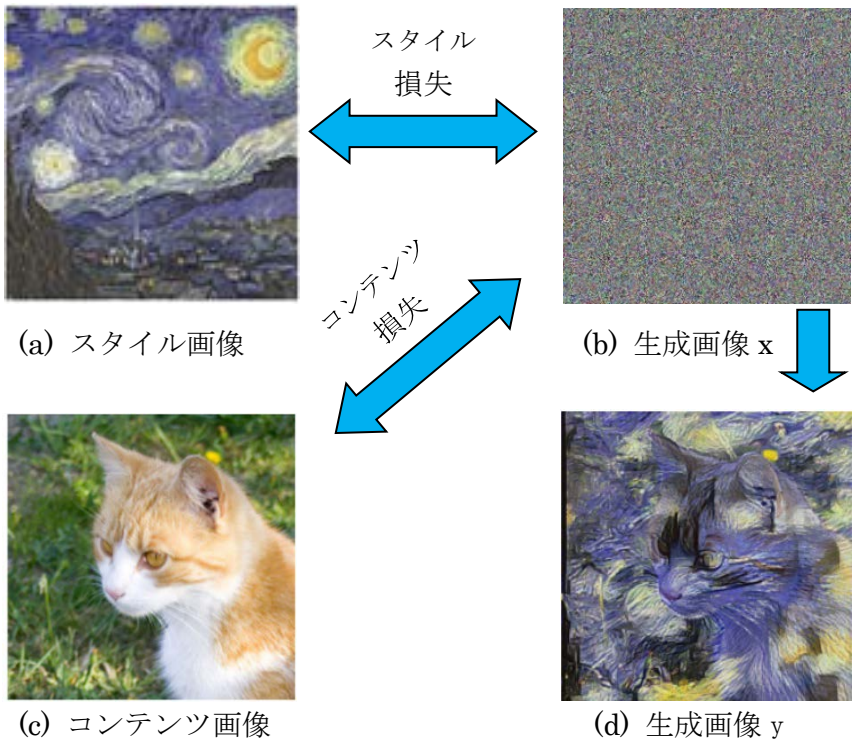


図 3-1　Gatys らの手法 [8] の概要.

ここで、N_lは第 l 層の特徴マップ数、M_lは特徴マップの高さと幅の積、G^l は出力画像のグラム行列、A^l はスタイル画像のグラム行列を表す。画像生成の流れを図 3-1 に示す。はじめに、図 3-1 の(b)のように一様乱数で生成画像 x を初期化し、スタイル損失とコンテンツ損失が小さくなるよう、勾配降下法によって生成画像 x を更新し、(d) の生成画像 y を出力する。これによって、コンテンツ画像の詳細形状を残しつつ、スタイル画像のスタイルに変換することができる。

3.2 Li らの手法 [9]

Li らは Gatys らの手法をベースとし、画風転写をドメイン適応と見なした [9]。ドメイン適応とは、あるドメインで学習した情報を、目標ドメインに転移する手法である。また、スタイルの損失関数の最適化が最大平均差異 MMD の最小化に等しいことを示し、式（2）のスタイルの損失関数を以下のように再定式化した。

$$\mathcal{L}_{style}^{l} = \frac{1}{4N_l^2} MMD^2\left(\mathcal{F}^l, S^l\right) \tag{3}$$

ここで、$\mathcal{F}^l$は出力画像の特徴集合、S^lはスタイル画像の特徴集合である。各画像位置の特徴量を独立なサンプルと見なし、コンテンツ画像の特徴量とスタイル画像の特徴量を重ね合わせることで画風を変換している。また図 3-2 に示したように、MMD に異なるカーネル関数 Linear、Poly、Gaussian と BN-loss (Batch Normalization)を適用することで、様々な結果を得ることができる。これは、カーネル関数ごとに異なる高次元空間へ写像されるため、捉えられた特徴も異なるからだと考えられる。

コンテンツ画像

スタイル画像

図 3-2 Li らの手法 [9] の結果. 図は文献 [9] より引用.

3.3　Johnson らの手法［20］

Gayts らの手法［8］のような既存の画風転写手法は、スタイル転写時にその都度最適化が必要なため、計算時間が長い。Johnson ら［20］の手法は画風転写を高速化するため、転写時に最適化計算をするのではなく、スタイルを変換するネットワークを学習する。Johnson ら［20］が提案した手法の概要を図 3-3 に示す。画風転写ネットワークf_Wと損失関数ネットワーク$\emptyset$ の 2 つのネットワークで構成される。画風転写ネットワークf_Wは複数の畳み込み層と Residual Block を組み合わせたネットワークで、コンテンツ画像にスタイル画像のスタイルを適用して変換を行う。損失関数ネットワーク$\emptyset$ はコンテンツ損失関数l_{feat}^{ϕ}とスタイル損失関数l_{style}^{ϕ}が構成されて、学習済みの VGG-16 モデルで、f_wの学習に用いる損失を計算するためのネットワークである。図 3-3 について、xは入力画像、y_sは目標とするスタイル画像、y_cは目標とするコンテンツ画像、$\hat{y}$は生成画像である。

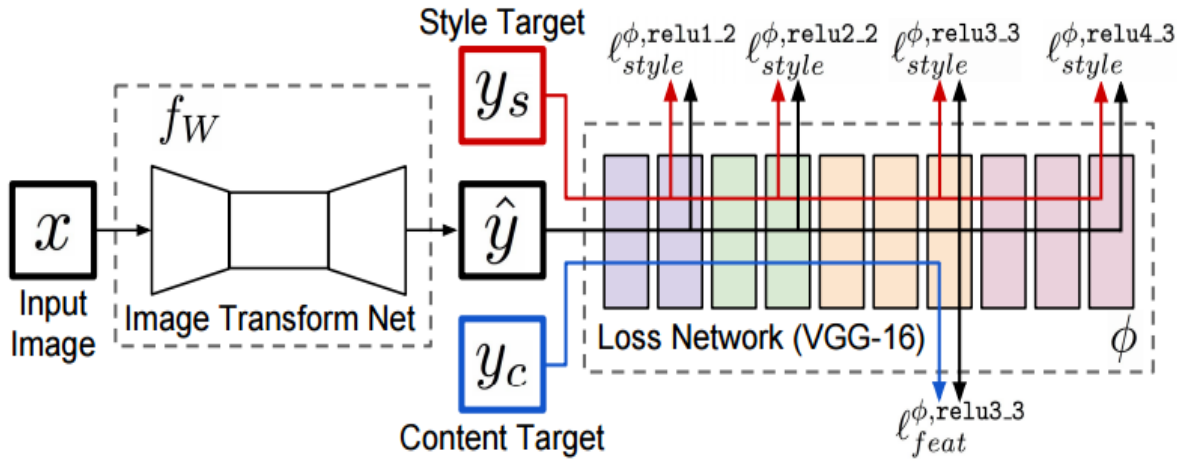


図 3-3　Johnson らの手法概要．図は文献［20］より引用．

スタイルを転写する前に、まず Microsoft COCO データセットに含まれた 8 万枚のコンテンツ画像と 1 枚のスタイル画像を損失関数ネットワーク ϕ に入力して、スタイルを変換するネットワークを学習し、学習済スタイルモデルを生成する。また損失ネットワークで、損失関数l_{feat}^{ϕ}とl_{style}^{ϕ}は目標としたコンテンツ特徴とスタイル特徴の差を測り、l_{feat}^{ϕ}とl_{style}^{ϕ}の重みつき和を減らすように画風転写ネットワークf_Wを更新する。これによって、画風転写時に、一回きりの順伝播計算ですむため、高速にスタイル変換を行うことができる。その結果、動画に変換を適用し、リアルタイムの画風変換が可能となった。また損失関数の構築について、Johnson らは Gatys らの手法[8]の損失関数をベースとし、ピクセル損失とノイズ除去項を追加した。ピクセル損失として生成画像$\hat{y}$-y_s間と$\hat{y}$-y_cの間のユークリッド距離を用いた。また、ノイズ除去項として$\hat{y}$に Total Variation 正則化$l_{TV}(\hat{y})$を用いた。ノイズ除去に

より、Johnson らの結果は Gatys らに比べて滑らかな結果となった。。

3.4 Chen らの手法 [19]

Gatys らの手法 [8] は、損失の計算時に複数の層のピクセル毎の特徴を用いるため、時間がかかる。これに対し Chen らは、一つ層に対し、指定したパッチサイズで特徴を抽出する手法 [4] を提案した。また、前節で述べた Johnson らの手法 [3] は、転写にかかる時間は短いが、スタイル毎に画風転写ネットワークf_Wを学習し直す必要があった。これに対し Chen らの手法では、様々なスタイルへの対応を維持しつつ、高速化を行っている。Chen らは図 3-4 に示したように、コンテンツ画像(下)とスタイル画像(上)の特徴量をパッチ単位で抽出し、正規化相互相関関数を用いて、マッチングを行った。つまり、コンテンツ画像のパッチに対して、最も類似するスタイル画像のパッチを探索し、パッチ毎にスタイル画像の特徴をコンテンツ画像の特徴と入れ替えることで、画風を変換する。具体的には 4 つの手順がある。まずはコンテンツ画像とスタイル画像の活性化領域からそれぞれのパッチセット$\{\phi_i(C)\}_{i\in n_c}$と$\{\phi_j(C)\}_{j\in n_s}$を抽出する。ここでn_cとn_sはパッチセットの数を表す。次は正規化相互相関を用

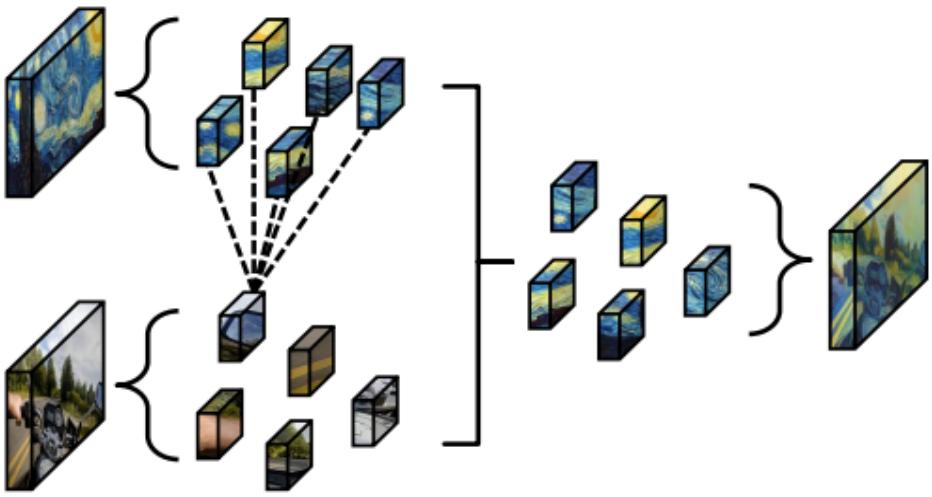


図 3-4 Chen らの手法のパッチマッチング．図は文献 [19] より引用．

いて、コンテンツパッチと最も相関したスタイル画像のパッチを探し、入れ替える。最後にパッチが入れ替えたコンテンツ画像の活性化領域を用いて、完全なコンテンツ領域を再構築する。一つの層に対しパッチ単位で最適化を行うため、従来の Gatys ら[8]の手法より、計算時間が短くなる。また Johnson らの手法のようにスタイルを学習する必要がなく、任意のスタイル画像のスタイルをコンテンツ画像に転写できる。パッチサイズの大きさを調整することにより、出力画像のスタイルをコンテンツ画像とスタイル画像のどちらに近づけるか制御できる。

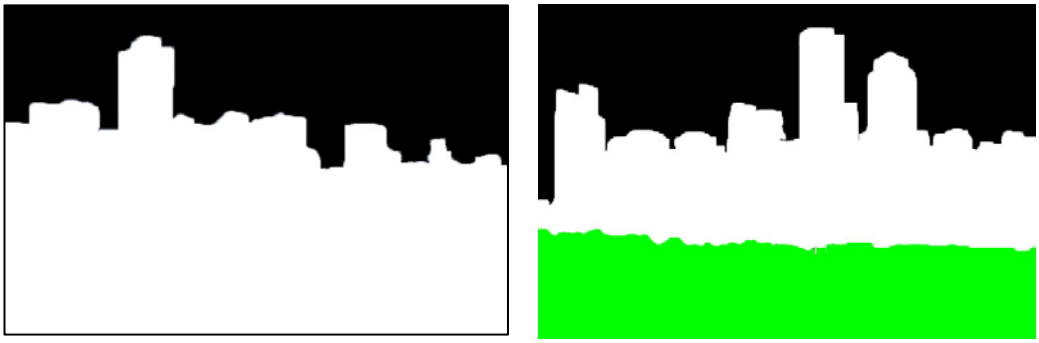
3.5　　Luan らの手法 [17]

画風変換を利用して、写真のスタイルを変換することができる。例えば、写真の時間帯（夜や昼）、写真の季節（春や秋）、アーティストによるエフェクトなどが挙げられる。しかし、既存の画風変換手法を用いて、写真のスタイル変換する際には、いくつかの問題点がある。図 3-5 に示すように、生成した画像が歪み、絵の質感に見え、写実的でなくなってしまう。

コンテンツ画像　　スタイル画像　　生成画像

図 3-5 既存手法を用いた結果. 図は文献［27］より引用.

コンテンツ画像　　スタイル画像



コンテンツ画像の分割マップ　　スタイル画像の分割マップを

図 3-6 Luan らの手法の入力画像. 図は文献［17］より引用.

また、空に建物の明かりが転写されており、空と建物という意味的に無関係な領域間で形状特徴の転写が行われている。

Luan らの手法[17]は、画像の空間的な歪みを防ぐため、転写の操作を色空間だけで行うことで、写実的な画像を生成している。またコンテンツ画像とスタイル画像の領域の意味的な対応を考慮し、形状特徴を転写することができる。入力データである、コンテンツ画像とスタイル画像とそれぞれの意味的領域分割マップを図 3-6 に示す。画風変換を同じ色の領域の間だけに行う。画像の歪みを防くため、Gatys ら手法[8]をベースに、目的関数に正則化項を導入し、色空間で局所の色のアフィン変換を行う。アフィン関数（一次関数）を用いて、出力画像の特徴領域毎に対応するコンテンツ画像の RGB 値をマッピングする。正則化項$\lambda\mathcal{L}_m$を追加した Luan らは目的関数を以下のように定義した。α、βとλは重みである。

$$\mathcal{L}_{total} = \sum_{l=1}^{L} \alpha_l \mathcal{L}_c^l + \Gamma \sum_{l=1}^{L} \beta_l \mathcal{L}_{s+}^l + \lambda \mathcal{L}_m \tag{4}$$

また入力画像にマスクチャンネルを追加し、領域の対応を考慮し、損失関数を計算する。Luan らはスタイル損失関数を以下のように定義した。

$$\mathcal{L}_{style}^l = \sum_{c=1}^{C} \frac{1}{2N_{l,c}^c} \sum_{ij} \left(G_{l,c}[O] - G_{l,c}[S]\right)_{ij}^2 \tag{5}$$

C はマスクのチャンネル番号、$\mathrm{G}[O]$は出力画像のグラム行列、$\mathrm{G}[S]$はスタイル画像のグラム行列を意味する。また、入力として必要な領域分割マップは、既存のシーン領域分割手法を用いて求めている。この時、コンテンツ画像と参照画像の分割マップは対応領域が同じ色である必要がある。例えば、図 3-6 に示したように、コンテンツ画像とスタイル画像の分割マップの中に同じ建物のところに白い色を使う。もう一つは分割マップに使える色は事前に決める必要がある。図 3-7 に Luan らの結果と Gatys らの結果の比較を示す。c)は(a)(b)中の矩形領域の拡大図であり、矩形の色で対応付けを行っている。また、(c)の各列について、左からコンテンツ画像、Gatys らの結果画像、Luan らの結果画像である。Gatys らの結果(中央列)と Luan らの結果(右列)をそれぞれコンテンツ画像(左列)と比較すると、Luan らの結果がよりコンテンツ画像の形状を保持しており、画像の空間的な歪みを抑制できていることがわかる。

(a)Luan らの結果　(b)Gatys らの結果　(c)拡大した領域

図 3-7 領域を拡大した比較. 図は文献［17］より引用.

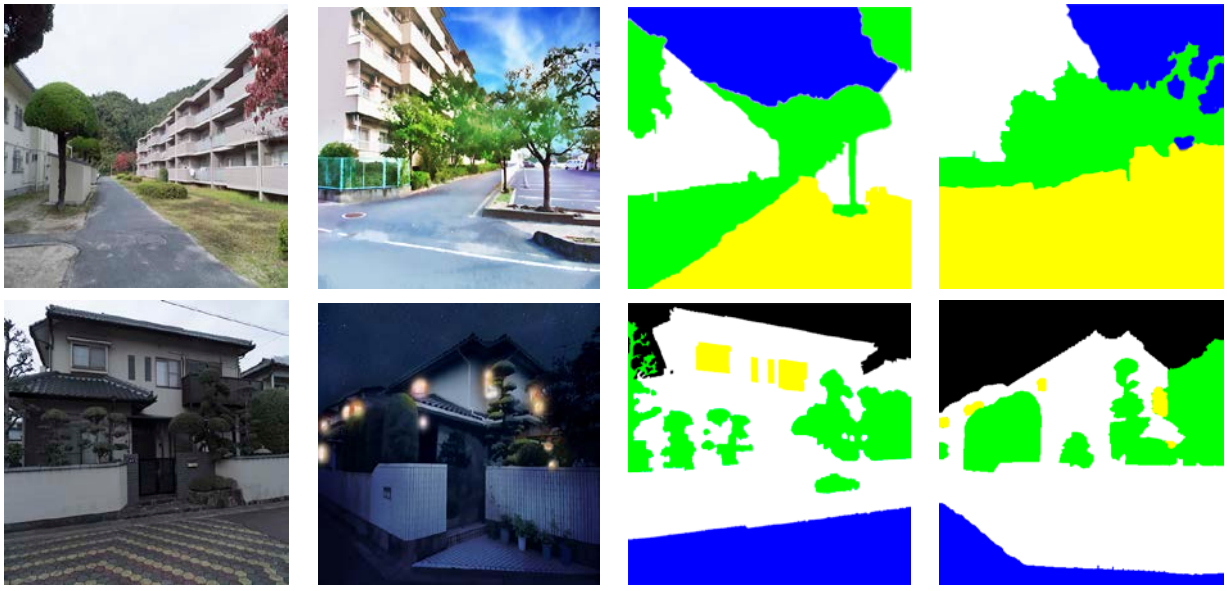
第4章　　実験結果

本研究では、前節で述べた既存手法を用いて、実写背景画像のアニメ風変換を行い、本タスクにおける各手法の評価を行う。実験データには、PhotoDramatica の作者から提供された、実写背景画像とアニメ風背景画像のペアからなるデータセットと、アニメ 10 作品の実写背景画像とアニメ風背景画像のペアからなるデータセットを用いた。画風変換手法として、前節で述べた 5 つの手法を実験した。開発言語は python であり、ライブラリとして chainer、Torch、MXNet を用いた。

4.1　　Phtotodramatica のデータセットについて

4.1.1　　実験データ

実験に使用するデータについて、2 つの例を用いて、説明する。Photodramatica のデータセットの画像を図 4-1 に示し、上の昼シーンと下の夜シーンがある。すべての手法について、コンテンツ画像とスタイル画像は、列（a）の実写背景画像と列（b）のアニメ風背景画像を使う。また Luan らの手法は意味的な領域分割マップが必要あり、(c) と（d）の列の分割マップを使う。領域分割マップについて、Luan らのウェブページで、いくつかの自動分割手法が挙げられているが、それら手法は精度が不十分であり、そのまま使うことができない。そこで、本研究では、画像処理ソフト Photoshop のクイック選択ツールを利用して、手作業で領域分割マップを作成した。また、コンテンツ画像と参照画像の分割マップについて、同じ意味の領域には同じ色を用いる。例えば、一行目の（c）と（d）の分割マップについて、黄色の領域が地面、緑の領域が木、青い領域が空、白い領域が建物を意味する。



(a) コンテンツ画像　(b)スタイル画像　(c)分割マップ　(d)分割マップ

図 4-1　Photodramatica のデータセットの入力画像.

4.1.2　　実験結果と評価

Photodramatica の昼シーンと夜シーンのデータを用いた 5 つ手法の出力結果を図 4-2 と図 4-3 に示す。(a) から (e) は、検討対象手法の出力結果、(f) は図 4-1 (a) のコンテンツ画像に対して、PhotoDramatica の作者が手作業で作成した正解画像である。まず図 4-2 の昼シーンの結果について、5 つの手法の結果はコンテンツ画像の形状を残したまま、スタイル画像の色合いや質感を転写できていることが確認できる。その中で、(d) の Chen らの出力画像は色合いや質感の変化が最も小さく、コンテンツ画像に近いと感じられる。これはパッチ単位で特徴量を抽出するため、スタイルを表す情報量が少ないからだと考えられる。また (a) の Gayts らの結果について、空に建物のようなテクスチャが確認できる。これはスタイル画像のスタイル損失を計算する時に、無関係な領域の形状特徴が転写されてしまったためだと考えられる。また、(b) と (c) の結果について、地面に空の青いテクスチャが出現した。これはスタイルを変換する時に領域の意味的な対応を考慮していないためだと考えられる。(e) の結果について、Luan らは意味的な領域分割マップを用いて、画風変換時に領域の対応関係を考慮しているため、Gayts らの結果 (a) のように無関係な領域の形状特徴が転写されることがなかった。また Luan らはコンテンツ画像の Laplacian 特徴と出力画像の Laplacian 特徴が近くなるようにしているため、他の手法に比べ、出力画像にコンテンツ画像のエッジや輪郭線などが良く保存されている。

(a) Gatys et al.　　(b) Li et al.　　(c) Johnson et al.

(d) Chen et al.　　(e) Luan et al.　　(f) 正解画像

図 4-2　Photodramatica の昼シーンの結果.

次に、図 4-3 の夜シーンの結果について述べる。ここで、Johnson らの手法は Gatys の手法の約 700 倍のスピードで、画像を出力できる。Johnson らは Gatys らの損失関数にピクセル損失とノイズ除去項を加えているため、Johnson らの結果 (c) は Gatys らの結果 (a) よ

り、ノイズ感が弱いように見える。また (a) と (c) の結果は不自然に明るい部分が散見される。これはスタイル画像の局所的な特徴である窓の光が大域的な特徴と見なされ、画像の全体に転写されたためだと考えられる。図 4-1 (a) のコンテンツ画像に図 4-1 (b) のスタイル画像の局所的な特徴である窓の光を転写する場合、Luan ら以外の手法では、上手く変換できなかった。Luan らの結果 (e) は窓の光の特徴を転写できているが、正解画像 (f) の理想的な光と比べて、まだ差がある。これは、Luan らの手法でコンテンツ画像のエッジや輪郭を保つようにしたためであり、参照画像の特徴が弱くなっているためであると考えられる。また、(b) の Li らの結果ではコンテンツ画像の地面のテクスチャがなくなっている。これは MMD カーネル関数を用いた損失計算では、地面の特徴量が足りなかったためだと考えられる。

これらの結果から、本タスクに対して、領域の対応関係を考慮することで、良い画風変換の結果が得られる。また領域の対応関係を取るだけではなく、スタイル画像の局所的な領域と大域的な領域を分けて、それぞれに画風変換を行う必要があると考えられる。本タスクの場合では、時にコンテンツ画像の特定の局所領域で特徴量の一部を使用しないという改良が考えられる。例えば、正解画像 (f) では、窓の光によって、図 4-1(a)下のコンテンツ画像の窓のエッジが目立たなくなっている。これに対し Luan らの結果では、エッジや輪郭を保持しすぎ、光のスタイルが弱くなっている。そのため、窓の領域ではエッジを保つための Laplacian 特徴量を使用しないことが有効であると考える。

(a) Gatys et al.　(b) Li et al.　(c) Johnson et al.

(d) Chen et al.　(e) Luan et al.　(f) 正解画像

図 4-3　Photodramatica の夜シーンの結果.

4.2　　アニメ 10 作品のデータセットについて

4.2.1　　実験データ

実際のアニメ作品の中で使われたアニメ風背景画像に対して、検討対象の手法が有効であるかを確かめるため、インターネット上で、アニメ 10 作品の実写背景画像とアニメ風背景画像のペアを集め、実験を行った。データの収集方法として、聖地巡礼のウェブサイトと Google 画像検索から入手する。聖地巡礼とは、アニメの舞台となった土地や建物など、ファンにとって思い入れのある場所を「聖地」とし、その「聖地」を実際に訪れることを指す。アニメで使われたアニメ風背景画像はあまり公開されていないが、多くのファンたちがアニメの舞台となった実世界の場所を訪ね、写真を撮り、そのシーンに基づいたアニメ風背景画像と一緒に聖地巡礼ウェブサイトに公開している。例えば、図 4-4 に示した画像は聖地巡礼のペア画像である。本研究では、これらのペア画像を利用する。またアニメの背景画像に対して、実世界のシーンがない場合は、Google 画像検索でそのアニメ風背景画像のコンテンツと一番似ている実写背景画像を用いた。表 4-1 と 4-2 に本研究で使った 10 アニメ作品の名前とそれぞれ実験したデータ数を示す。

実写背景画像　　アニメ風背景画像

図 4-4　聖地巡礼のペア画像. 画像の出典: 左は「舞台探訪まとめ Wiki」ウェブページ（http://seesaawiki.jp/w/lsh_er/）©2017 舞台探訪まとめ Wiki , 右は「とある魔術の禁書目録」©2010 鎌池和馬/アスキー・メディアワークス/PROJECT-INDEX.

表 4-1　アニメ作品の名前とデータ数

アニメ作品の名前	fate/stay night	G 線上の魔王	雲のむこう、約束の場所	とある魔術の禁書目録	バケモノの子
ペア数	4	4	3	3	3

表 4-2　アニメ作品の名前とデータ数

アニメ作品の名前	俺の妹がこんなに可愛いわけがない	サマーウォーズ	あの夏で待ってる	君の名は。	結城友奈は勇者である
ペア数	3	4	4	3	4

4.2.2　実験結果の評価について

本研究では、アニメ 10 作品の実写背景画像とアニメ風背景画像のペアを用いて、5 つの検討対象手法について、実験を行った。ここで、どの手法の結果が一番アニメ風背景画像に近いと感じるか、人によって感じ方が異なる可能性があるため、出力結果を用いて、ユーザテストによる評価を行った。ユーザテストについて、表 4-1 と表 4-2 の作品データの中に作品毎にランダムで 2 つの画像を選び、アンケート調査を行う。2 つのタイプの質問に対して、ユーザにどの結果が良いと感じるかを選んでもらう。詳細の実験結果と評価は、アンケート調査の統計結果と合わせて、第 5 章で説明する。

第5章　　評価手法とユーザテスト

5.1　　ユーザテストの設定

本研究では、5 つの手法のアニメ 10 作品に対する結果を用いて、アンケート調査によるユーザテストを行う。アンケートの制作には文献[37]の Tencent questionnaire という中国のアンケート調査サイトが提供しているツールを利用した。アンケートでは、本タスクにとって重要だと考えられる 2 つの基準に基づき、2 つのタイプの設問を設けている。一つは 5 つの手法の結果画像について、どれが最もアニメ風背景画像のスタイルや雰囲気に近いかである。もう一つは 5 つの手法の結果画像について、どれが最も実写背景画像の物体や構造に近いかである。ユーザは設問毎に 5 つの手法の結果画像を良い順に並べる。設問の例を図 5-1 に示す。一番上の行が実写背景画像（コンテンツ画像）とアニメ風背景画像（スタイル画像）であり、その下に 5 つの検討対象手法の結果を並べている。アニメ 10 作品から選んだ 20 セットのデータを用いて、2 つのタイプの設問に対してそれぞれに 20 設問を設けている。またアンケート調査のデータの有効性を保つため、ユーザに見せる結果画像の並び順を設問毎にシャッフルしている。設問の最後には、ユーザの年齢と性別の情報を入力させている。最終

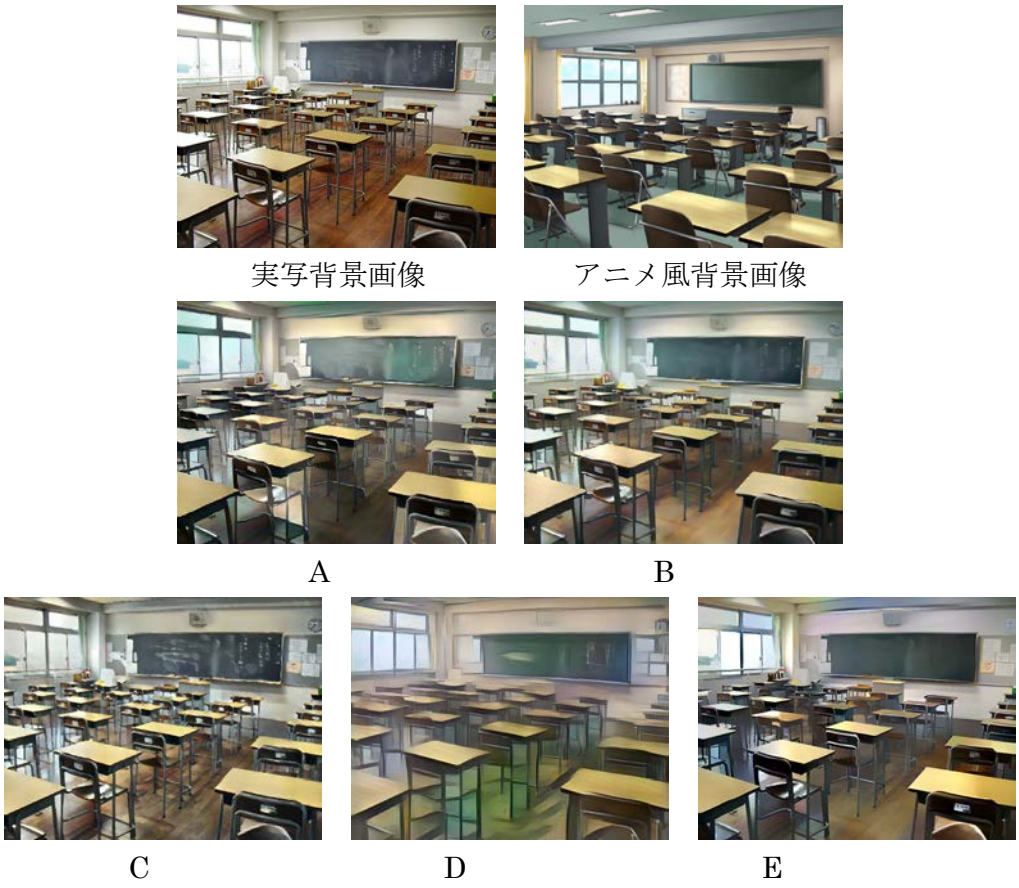


図 5-1　アンケートの設問の例（設問 2）．実写背景画像の出典:「東京ロケーションボックス」ウェブページ（http://www.locationbox.metro.tokyo.jp/catalog/school/005305.php）© 2018 TOKYO METROPOLITAN GOVERNMENT, アニメ風背景画像の出典:「G 線上の魔王」©2006 AKABEi SOFT2.

的に、男性 24 名と女性 18 名の合計 42 名のユーザからアンケートデータを集めることができた。ユーザの出身国は、日本、中国、タイ、スペイン、ドイツなど様々である。ユーザの平均年齢は約 27 歳で、アンケートの平均所要時間は 37 分 17 秒となった。

表 5-1　Rank の統計結果（設問 2）.

選択肢	Rank1	Rank2	Rank3	Rank4	Rank5	Rank 平均値
A Gatys et al.	8	16	9	6	3	3.47
B Li et al.	8	8	9	14	3	3.1
C Johnson et al.	6	10	9	6	11	2.86
D Chen et al.	1	2	6	11	22	1.79
E Luan et al.	19	6	9	5	3	3.79

5.2　評価指標の導入

図 5-1 のアンケートの結果を、表 5-1 に示す。この表では、一番右の列を除き、各手法を Rank1~5 に投票した人数を表している。一番右の列の Rank 平均値とは、文献[34]で提案された評価指標であり、各行から計算される。

Jing ら[34]は既存の画風変換手法をまとめ、結果を評価するため、結果画像に Rank を付けるという形でユーザテストを行った。そしてユーザテストのデータの評価指標として、Rank ごとの重み付け平均を提案した。本研究では、Jing らが提案した評価指標を導入する。Rank ごとの重みは、Rank1 が 5、Rank2 が 4、Rank3 が 3 というように、Rank がひとつ下がると重みが 1 減るように設定している。Rank 平均値の計算式は以下のように定義される。

$$\frac{\text{Rank1} \times 5 + \text{rank2} \times 4 + \text{rank3} \times 3 + \text{rank4} \times 2 + \text{rank5} \times 1}{42} \tag{6}$$

ここまでは、一つの設問に対する Rank 平均値の求め方を説明した。複数の設問に対する Rank 平均値は、設問毎に Rank 平均値を求めた後、手法ごとにそれらの平均を取ることで計算する。実際には、タイプで設問を 2 つのグループに分け、5 つの手法ごとに Rank 平均値の平均を計算した。求めたすべての平均値に四捨五入で近似値を取る。設問 1 から 20 まではタイプ 1 のアニメ風背景画像に近いの設問で、設問 21 から 40 までは実写背景画像に近いの設問である。

5.3 ユーザテストの結果と考察

設問のタイプを分けて手法ごとの平均値を表 5-2 に示す。これから 2 つのタイプの設問に対して、5 つの手法のの結果を考察する。

表 5-2 手法ごとの平均値.

手法名	設問タイプ 1	設問タイプ 2
Gatys et al.	3.07	2.97
Li et al.	2.43	2.09
Johnson et al.	3.37	3.80
Chen et al.	2.26	2.43
Luan et al.	3.86	3.72

5.3.1 設問タイプ 1（アニメ風背景画像に近い）

表 5-2 に示した設問タイプ 1 の手法ごとの平均値について、Luan らの手法が最も高く、最もアニメ風背景画像に近いと考えられる。これに対し最も低かったのが Chen らの手法であり、最もアニメ風背景画像から遠いと考えられる。例えば、図 5-1 と表 5-1 に示した設問 2 の例でも Luan らの手法がランキング 1 位になっている。これは Luan らの手法が画像の領域の対応関係を考慮し、無関係の領域間でにテクスチャーが転写されないためだと考えられる。これに対し、Chen らの手法はパッチサイズで画風を変換しており、スタイル画像から得られるスタイルの特徴量が少ないため、図 5-1 の E のように、生成した画像のスタイルの変化が少なくなり、結果としてユーザの評価が悪いと考えられる。

表 5-3 設問タイプ 1 に対する Luan らの平均値と他の 4 の手法の平均値の t 検定結果.

手法名	Gatys et al.	Li et al.	Johnson et al.	Chen et al.
P(T<=t) 両側	7.02×10^{-6}	3.62×10^{-11}	6×10^{-3}	4.96×10^{-10}

有意: $P<5 \times 10^{-2}$

ランキング 1 位になった Luan らの平均値と他の 4 つの手法の平均値の差に有意差があるかを確かめるため、これらの手法の 20 設問の平均値を 5 つの標本と見なし、Excel の分析ツールを用いて、t 検定を行った。t 検定は、2 組の標本の平均の差が偶然的な誤差のか、意義があるの差を検定する時によく用いられる。Luan らの標本と他の 4 つの標本をそれぞれに t 検定を行い、結果を表 5-3 に示す。表 5-3 の 1 行目に手法名を表し、2 行目に t 検定の P 値を表している。t 検定のパラメータについて、対応がない 2 標本の分散が等しくないと仮定し、両側検定とした。表 5-2 について、平均値の点数の差が最も小さい、ランキング 2 位になった Johnson らの平均値とランキング 1 位の Luan らの平均値の t 検定の結果は、両側 P(T<=t) =0.006<0.05 となり、有意差があると判断できる。また、Luan らの平均値と他の手法の平均値の t 検定の結果を表 5-3 に示したように、全部有意差があると判断できる。

多くの設問で Luan らの結果がランキング 1 位となるが、すべての設問でランキング 1 位になるではない。極端な例として、表 5-4 に示す設問 17 の統計結果では、Chen らの手法の平均値がランキング 1 位となった。図 5-2 に示す設問 17 の画像から、実写背景画像のコンテ

ンツ特徴が残る Luan らの結果 E と比べて、Chen らの結果 D は実写背景画像の一部分のコンテンツ特徴が弱まっているように見える。この例から、実写背景画像からアニメ風背景画像に変換する際、シーンによっては、実写背景画像のコンテンツ特徴の一部分が弱まる方が絵のような質感になると考えられる。

表 5-4　Rank の統計結果（設問 17）.

選択肢	Rank1	Rank2	Rank3	Rank4	Rank5	Rank 平均値
A Gatys et al.	17	8	7	4	6	3.62
B Li et al.	2	4	10	17	9	2.36
C Johnson et al.	1	6	9	10	16	2.19
D Chen et al.	18	12	4	4	4	3.86
E Luan et al.	4	12	12	7	7	2.98

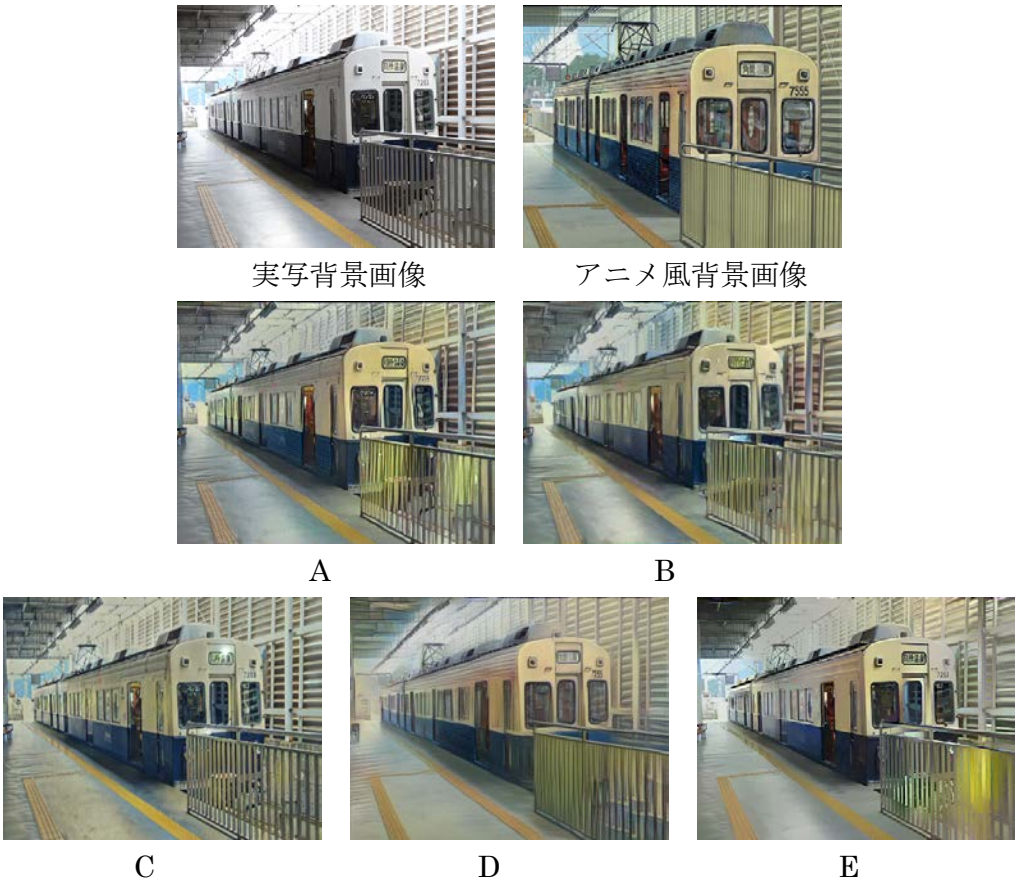


図 5-2　設問 17 の画像データ. 実写背景画像の出典:「舞台探訪まとめ Wiki」ウェブページ（http://seesaawiki.jp/w/lsh_er/）©2017 舞台探訪まとめ Wiki, アニメ風背景画像の出典:「サマーウォーズ」©2009 SUMMERWARS FILM PARTNERS.

5.3.2 設問タイプ 2（実写背景画像に近い）

表 5-2 に示した設問タイプ 2 の平均値（実写背景画像に近い）について、Johnson らの手法がランキン 1 位、Luan らの手法がランキン 2 位となった。実際に各設問の Rank 平均値を確認すると、Johnson らの手法が 1 位、Luan らの手法が 2 位になるケースが多かった。その中の一つの例として、表 5-5 に設問 30 の統計結果、図 5-3 に設問 30 の画像データを示す。図 5-3 の Johnson らの結果 C は他の結果画像より、ノイズ感を抑えることができ、実写背景画像のコンテンツ特徴がはっきり見える。Luan らの手法はコンテンツ画像の Laplacian の特徴量を出力結果に適用するため、出力結果 E がより実写的な画像に見える。またランキング 1 位になった Johnson らの平均値と他の 4 つの手法の平均値の差に有意差があるかを確かめるため、t 検定を行った。t 検定のパラメータについて、5.3.1 節に述べた設定と同様である。表 5-6 に t 検定の結果を示す。Johnson らと Luan らの平均値の t 検定の結果は、両側 P(T<=t) =0.644>0.05 となり、有意差がないと判断できる。また、表 5-6 に示した Johnson らの平均値と Gatys ら、Li ら、Chen らの平均値の t 検定の結果から、有意差があると判断できる。Li らの手法の平均値は表 5-2 においても表 5-5 においても最低となった。図 5-3 に示した設問 30 の例から、Li らの結果 B では、画像の歪みや無関係領域のテクスチャ転写がよく出でいる。これが原因でユーザの評価が最も悪かったと考えられる。

本章では、アンケートの一部分の結果だけを記載するため、すべての設問に対する統計結果を付録に記載した。

表 5-5 Rank の統計結果（設問 30）.

選択肢	Rank1	Rank2	Rank3	Rank4	Rank5	Rank 平均値
A Gatys et al.	2	11	14	12	3	2.93
B Li et al.	1	4	1	8	28	1.62
C Johnson et al.	24	12	3	1	2	4.31
D Chen et al.	6	5	5	17	9	2.57
E Luan et al.	9	10	19	4	0	3.57

表 5-6 設問タイプ 2 に対する Johnson らの平均値と他の 4 の手法の平均値の t 検定結果.

手法名	Gatys et al.	Li et al.	Chen et al.	Luan et al.
P(T<=t) 両側	1.74×10^{-5}	3.96×10^{-12}	1.86×10^{-6}	6.44×10^{-1}

有意: $P<5 \times 10^{-2}$

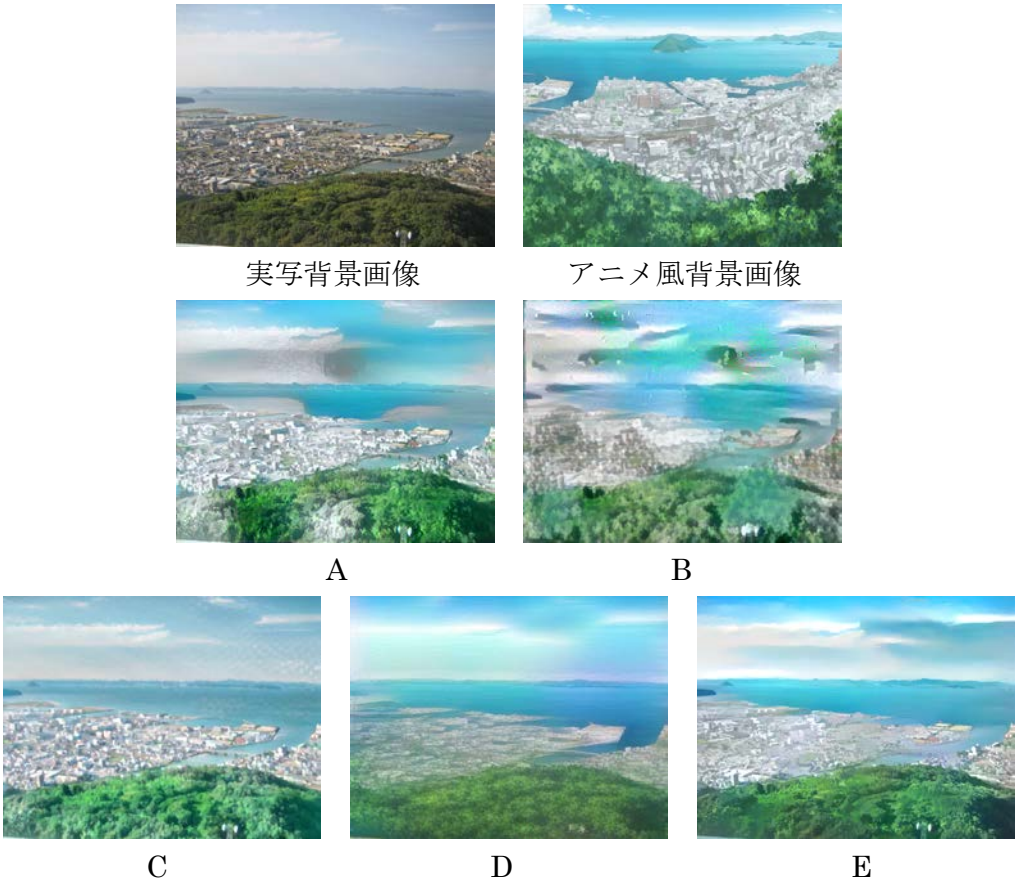


図 5-3　設問 30 の画像データ．実写背景画像の出典:「つればし」ウェブページ（http://tsurebashi.blog123.fc2.com/blog-category-111.html）©2017 つればし，アニメ風背景画像の出典:「サマーウォーズ」©2009 SUMMERWARS FILM PARTNERS.

第6章　まとめと今後の展望

本研究では、既存の画風変換手法をまとめ、手法ごとの特徴を紹介した。またその中の 5 つの CNN によるスタイル画像ありの画風変換手法を用いて、実写背景画像からアニメ風背景画像への自動変換を考察した。さらにアニメ 10 作品の実験結果を用いて、ユーザテストを行った。PhotoDramatica データの結果から、アニメ風背景画像の大域的な特徴（色合いや質感）を実写背景画像に転写できると同時に形状特徴が残っている。また検討した既存手法をアニメ風画像変換に用いる際の問題として、意味的な情報がないことにより、局所的な特徴の変換ができない問題とスタイル画像のテクスチャが不適切な領域に転写されてしまう問題がある。意味的な領域情報を利用した Luan らの手法 [5]はこの問題の発生を抑えることができ、他の 4 つの手法の結果と比べて、結果が良くなった。しかし、PhotoDramatica は画像の位置ごとに異なる処理を施した局所的な特徴と画像全体にかけた大域的な特徴がある。1 枚のスタイル画像だけの情報で正解画像を再現することは困難だと考えられる。また画風を変換する際に局所的な特徴と大域的な特徴を分ける必要があると考えられる。

アニメ 10 作品の結果から見ると、画像の歪みや無関係領域のテクスチャ転写が出る場合、ユーザの評価が最も悪かったことがよく出ている。生成画像のノイズ感を抑えることによって、ユーザの評価が良くなると考えられる。またシーンによって、実写背景画像の一部分のコンテンツ特徴を弱める方が絵のような質感になり、アニメ風背景画像に近いと考えられる。

今後の本タスクを実現するため、局所的な特徴と大域的特徴を分けて、画風変換を行うシステムが理想だと考えられる。局所的な特徴について、同じ意味を持つ領域の対応関係を考慮すべきである。またシーンによって、実写背景画像の一部分の特徴を省略する必要があると考え、システムにユーザが実写背景画像とコンテンツ画像のどちらの特徴をより反映することを制御し、対話的な仕組みが必要であり、柔軟性があるシステムが望ましいと考えられる。

謝辞

本研究を進めにあたり、本学大学院システム情報工学研究科の金森由博准教授をはじめとし、遠藤結城助教、三谷純教授には多くのご指導・ご協力をいただきました。ここに深く感謝の意を表しました。また非数値処理アルゴリズム研究室の皆様には多くの貴重なご意見をいただきました。皆様と過ごした 2 年間はとても楽しかったです。

さらにアンケート調査を協力して下さった皆様に心より御礼申し上げます。

それから、これまでの人生と日本への留学を支えてくれた両親に心から感謝いたします。

みなさまのお陰で、無事修士論文を書き上げる事が出来ました。ここに感謝をもって、謝辞を述べさせていただきます。

本当に有難う御座いました。

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 IEEE International Conference on Computer Vision (ICCV), 2017.

[2] Jun-Yan Zhu, Richard Zhang, Deepak Pathak, Trevor Darrell, Alexei A. Efros, Oliver Wang, Eli Shechtman. “Toward Multimodal Image-to-Image Translation”. In Advances in Neural Information Processing Systems 30 (NIPS), 2017.

[3] Zhicheng Yan, Hao Zhang, Baoyuan Wang, Sylvain Paris, and Yizhou Yu. Automatic photo adjustment using deep neural networks. ACM Trans. Graph., 35(2):11, 2016.

[4] Yaniv Taigman, Adam Polyak, Lior Wolf. Unsupervised Cross-Domain Image Generation. arXiv:1611.02200, 2016.

[5] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 3431-3440, 2015.

[6] A. Hertzmann, C. Jacobs, N. Oliver, B. Curless, D. Salesin. Image Analogies. In Proceedings of SIGGRAPH 2001, pages 327-340, 2001.

[7] I-Cheng Chang, Yu-Ming Peng, Yung-Sheng Chen, and Shen-Chi Wang. Artistic painting style transformation using a patch-based sampling method. J. Inf. Sci. Eng., 26(4):1443–1458, 2010.

[8] L. A. Gatys, A. S. Ecker, and M. Bethge. A Neural Algorithm of ArtisticStyle. arXiv:1508.06576, 2015.

[9] Yanghao Li, Jiaying Liu, Xiuming Zhang, Xiaodi Hou. Demystifying Neural Style Transfer. arXiv:1701.01036, 2017.

[10] C.Li and M. Wand. Combining markov random fields and convolutional neural networks for image synthesis. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 2479–2486, 2016.

[11] A. Mahendran and A. Vedaldi. Understanding deep image representations by inverting them. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 5188–5196, 2015.

[12] L. A. Gatys, A. S. Ecker, and M. Bethge. Texture synthesis using convolutional neural networks. In Advances in Neural Information Processing Systems, pages 262–270, 2015.

[13] Neural-style: Using other neural models, [online]. Available: https://github.com/jcjohnson/neural-style/wiki/Using-Other-Neural-Models [Accessed 4 Dec. 2017].

[14] E. Risser, P. Wilmot, and C. Barnes. Stable and controllable neural texture synthesis and style transfer using histogram losses. ArXiv e-prints, Jan. 2017.

[15] R. Yin. Content-aware neural style transfer. ArXiv e-prints, Jan. 2016.

[16] Y.-L. Chen and C.-T. Hsu. Towards deep style transfer: A content-aware perspective. In Proceedings of the British Machine Vision Conference, 2016.

[17] Fujun Luan, Sylvain Paris, Eli Shechtman, Kavita Bala. Deep Photo Style Transfer. arXiv:1611.02200, 2017.

[18] A. J. Champandard. Semantic style transfer and turning twobit doodles into fine artworks. ArXiv e-prints, Mar. 2016.

[19] Tian Qi Chen, Mark Schmidt. Fast Patch-based Style Transfer of Arbitrary Style. In IJCAI, 2016.

[20] J. Johnson, A. Alahi, and L. Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In ECCV, 2016.

[21] A. Radford, L. Metz, and S. Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. ArXiv e-prints, Nov. 2015.

[22] D. Ulyanov, V. Lebedev, A. Vedaldi, and V. Lempitsky. Texture networks: Feed-forward synthesis of textures and stylized images. In International Conference on Machine Learning, pages 1349–1357, 2016.

[23] C. Li and M.Wand. Precomputed real-time texture synthesis with markovian generative adversarial networks. In European Conference on Computer Vision, pages 702–716, 2016.

[24] V. Dumoulin, J. Shlens, and M. Kudlur. A learned representation for artistic style. ArXiv e-prints, Oct. 2016.

[25] Y. Nikulin and R. Novak. Exploring the neural algorithm of artistic style. ArXiv e-

prints, Feb. 2016.

[26] R. Novak and Y. Nikulin. Improving the neural algorithm of artistic style. ArXiv e-prints, May 2016.

[27] L. A. Gatys, M. Bethge, A. Hertzmann, and E. Shechtman. Preserving color in neural artistic style transfer. ArXiv eprints, June 2016.

[28] L. A. Gatys, A. S. Ecker, M. Bethge, A. Hertzmann, and E. Shechtman. Controlling perceptual factors in neural style transfer. ArXiv e-prints, Nov. 2016.

[29] R. Liao, Y. Xia, and X. Zhang. Depth-preserving style transfer. [online]. Available: https://github.com/xiumingzhang/depth-preserving-neural-style-transfer/blob/master/report/egpaper_final.pdf [Accessed 4 Dec. 2017].

[30] D. Zoran, P. Isola, D. Krishnan, and W. T. Freeman. Learning ordinal relationships for mid-level vision. In Proceedings of the IEEE International Conference on Computer Vision, pages 388–396, 2015.

[31] Shaohua Li, Xinxing Xu, Liqiang Nie, Tat-Seng Chua. Laplacian-Steered Neural Style Transfer. In Proceedings of the ACM Multimedia Conference, 2017.

[32] D. Ulyanov, A. Vedaldi, and V. Lempitsky. Instance normalization: The missing ingredient for fast stylization. ArXiv e-prints, July 2016.

[33] D. Ulyanov, A. Vedaldi, and V. Lempitsky. Improved texture networks: Maximizing quality and diversity in feed-forward stylization and texture synthesis. ArXiv e-prints, Jan. 2017.

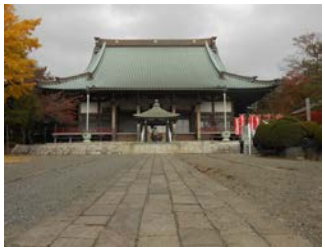
[34] Yongcheng Jing, Yezhou Yang, Zunlei Feng, Jingwen Ye, Mingli Song. Neural Style Transfer: A Review. arXiv:1705.04058, 2017.

[35] Jing Liao, Yuan Yao, Lu Yuan, Gang Hua, Sing Bing Kang. Visual Attribute Transfer through Deep Image Analogy. ACM Trans. Graph. (Proc. of SIGGRAPH) 36, 4, 2017.

付録

下には、アンケート調査で既存の 5 つ手法を用いたアニメ 10 作品の結果画像である。結果画像はアンケートの設問毎に並べ、20 設問がある。ここに設問 20 までの画像を記載したが、21 から 40 までの設問は同じくこの 20 設問のデータを用いた。設問の選択肢について、A に Gatys らの結果、B に Li らの結果、C に Johnson らの結果、D に Chen らの結果、E に Luan らの結果を示す。アンケートは URL: https://wj.qq.com/s/1617646/4ef8 でアクセスすることができる。またすべての設問に対するアンケート調査の Rank 平均値を画像の後ろに記載する。

設問 1.



実写背景画像

アニメ風背景画像

A

B

C

D

E

設問 2.

実写背景画像

アニメ風背景画像

A

B

C

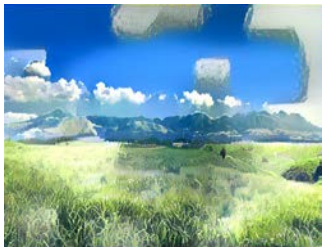
D

E

設問 3.

実写背景画像

アニメ風背景画像



A

B

C

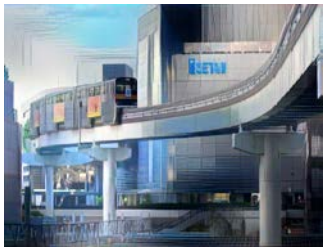
D

E

設問 4.

実写背景画像

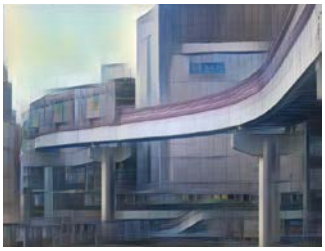
アニメ風背景画像



A

B

C



D

E

設問 5.

実写背景画像

アニメ風背景画像

A

B

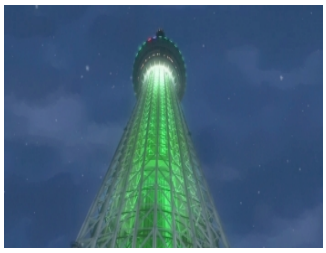
C

D

E

設問 6.

実写背景画像



アニメ風背景画像

A

B

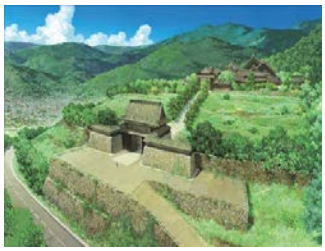
C

D

E

設問 7.

実写背景画像

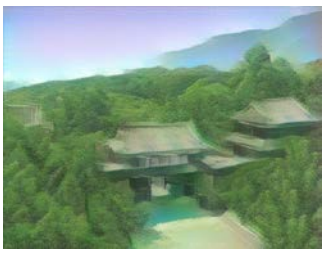


アニメ風背景画像

A

B

C



D

E

設問 8.

実写背景画像

アニメ風背景画像

A

B

C

D

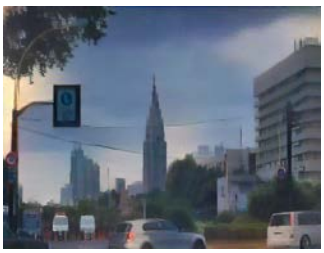
E

設問 9.

実写背景画像

アニメ風背景画像

A



B

C

D

E

設問 10.

実写背景画像

アニメ風背景画像

A

B

C

D

E

設問 11.

実写背景画像

アニメ風背景画像

A

B

C

D

E

設問 12.


実写背景画像


アニメ風背景画像


A


B


C

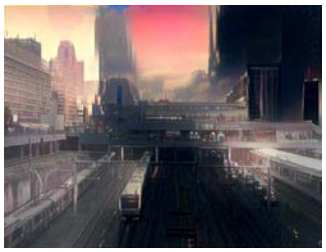

D

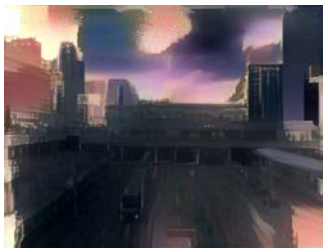

E

設問 13.

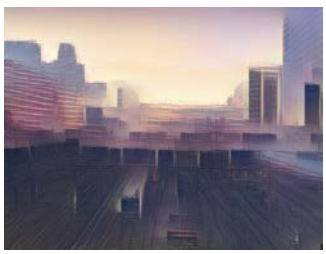

実写背景画像


アニメ風背景画像


A


B


C


D


E

設問 14.

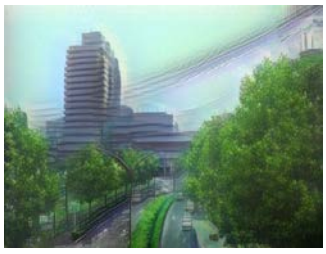

実写背景画像


アニメ風背景画像


A


B


C


D


E

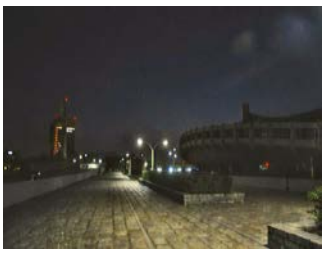
設問 15.


実写背景画像


アニメ風背景画像


A


B


C


D


E

設問 16.

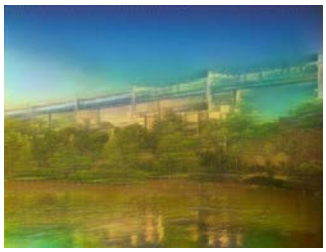

実写背景画像


アニメ風背景画像


A


B


C


D


E

設問 17.


実写背景画像


アニメ風背景画像


A


B


C

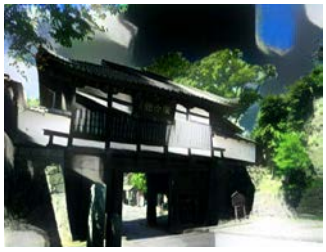

D


E

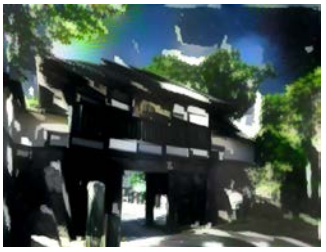
設問 18.

実写背景画像

アニメ風背景画像



A



B

C

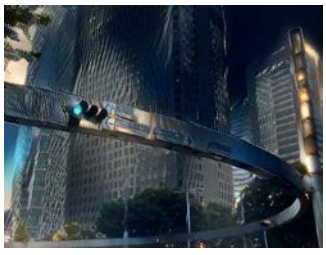
D

E

設問 19.

実写背景画像

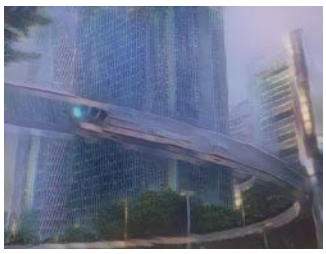
アニメ風背景画像



A

B

C



D

E

設問 20.

実写背景画像

アニメ風背景画像

A

B

C

D

E

表 0-1、表 0-2、表 0-3 と表 0-4 に設問毎の 5 つの検討手法の Rank 平均値を示す。これらの表では、一番左の列は手法の名前を表し、その右の列は各手法の Rank 平均値を表している。

表 0-1　設問 1 から 10 の Rank 平均値

手法名	1	2	3	4	5	6	7	8	9	10
A Gatys et al.	2.29	3.48	2.14	3.26	2.81	3.02	2.93	3.17	4.1	2.74
B Li et al.	2.5	3.1	2.43	2.74	2.5	3.45	2.76	2.02	2.81	1.43
C Johnson et al.	3.93	2.86	4.19	3.05	3.76	3.5	3.74	3.1	3.05	4.33
D Chen et al.	2.95	1.79	2.83	2.14	1.69	1.6	1.76	2.55	1.83	2.71
E Luan et al.	3.33	3.79	3.4	3.81	4.21	3.43	3.81	4.17	3.21	3.76

表 0-2　設問 11 から 20 の Rank 平均値

手法名	11	12	13	14	15	16	17	18	19	20
A Gatys et al.	3.4	3.62	3.43	2.5	2.76	3.1	3.62	2.62	2.93	3.57
B Li et al.	1.64	2.5	1.5	2.07	2.57	2.36	2.36	2.24	3.07	2.5
C Johnson et al.	3.43	3.33	4.1	3.52	3.55	3.52	2.19	2.02	3.38	2.93
D Chen et al.	2.31	1.79	1.95	2.55	2.26	1.48	3.86	3.52	1.17	2.48
E Luan et al.	4.21	3.76	4.02	4.36	3.86	4.55	2.98	4.6	4.45	3.52

表 0-3　設問 21 から 30 の Rank 平均値

手法名	21	22	23	24	25	26	27	28	29	30
A Gatys et al.	2.12	3.71	1.95	3.5	2.81	2.9	2.6	2.64	3.43	2.93
B Li et al.	2.1	2.4	1.76	2.19	2.05	3	3.07	1.6	2.52	1.62
C Johnson et al.	3.81	3.79	4	4.02	3.83	4.29	4.07	3.12	3.52	4.31
D Chen et al.	3.9	1.45	3.98	1.69	1.81	1.45	1.71	3.45	1.88	2.57
E Luan et al.	3.07	3.64	3.31	3.6	4.5	3.36	3.55	4.19	3.64	3.57

表 0-4　設問 31 から 40 の Rank 平均値

手法名	31	32	33	34	35	36	37	38	39	40
A Gatys et al.	2.57	3.31	3.02	2.64	3.1	2.76	3.29	2.9	3.1	4.1
B Li et al.	1.62	1.83	1.24	1.43	2.21	1.93	2.6	1.86	2.98	1.71
C Johnson et al.	3.45	4.48	4.64	3.4	4	4.1	3.86	2.12	3.81	3.31
D Chen et al.	3.98	2.1	2.1	3.16	1.69	2.14	2.19	3.52	1.5	2.26
E Luan et al.	3.38	3.29	4	4.36	4	4.07	3.07	4.6	3.62	3.62