

pix2pix を用いた線画生成における 特徴的な線の自動描画に関する研究

柳田一樹[†] 行天啓二[†] 大城英裕[†] 高見利也[†]

概要: pix2pix と呼ばれる技術により、線画の自動着色が可能となった。しかしその精度は、線画に描かれた対象物の特徴を際立たせる線である、特徴的な線の質に依存する。そこで本研究では、その特徴的な線の自動描写法を提案する。その手段として、輪郭のみの線画と、特徴的な線が描かれた線画を利用した pix2pix を用いる。具体的には、輪郭のみの線画と特徴的な線が描かれた線画との関係性を pix2pix に学習させ、学習後のモデルに輪郭のみの線画を入力することで特徴的な線を自動描画させる。その際、学習用の画像群にぼかし処理を加えることで、pix2pix を効果的に学習させることができ、特徴的な線の生成を可能にした。実験では、靴を対象として、ぼかし処理を含む4つの加工を施したペア画像を用意し、提案手法により生成した線画を定性的に評価した。また、その線画に対して、pix2pix による自動着色を適用した。その結果、複雑な線が少ない線画については、所望の自動描画に成功し、質の良い着色結果が得られた。一方、複雑な線が多い線画については、うまく自動描画されず着色の質も低い結果が得られた。

キーワード: ニューラルネットワーク, 画像生成, 線画生成, 自動着色, pix2pix

A Study of drawing characteristic lines automatically for line drawing generation using pix2pix

KAZUKI YANAGIDA[†] KEIJI GYOHTEN[†]
HIDEHIRO OHKI[†] TOSHIYA TAKAMI[†]

Abstract: A technology called pix2pix has made it possible to automatically color line arts. However, its accuracy depends on the quality of the characteristic lines, which are the lines that emphasize the characteristics of the object drawn in the line arts. In this study, we propose an automatic drawing method for the characteristic lines in the line arts. The proposed method uses pix2pix to learn the relationship between a contour-only line drawing and a line drawing with characteristic lines. By inputting a contour-only line drawing to the obtained model, characteristic lines can be drawn automatically. By adding a blurring process to the image groups for learning, pix2pix could train them effectively, and it became possible to generate characteristic lines. In the experiment, we prepared paired images of shoes and qualitatively evaluated the line art generated by the proposed method. In addition, automatic coloring by pix2pix was applied to the line art. As a result, we succeeded in the desired automatic drawing, and obtained high-quality coloring results for line arts with few complicated lines. On the other hand, line drawings with many complex lines were not automatically drawn well, resulting in low quality coloring.

Keywords: neural network, image synthesis, line drawing generation, automatic coloring, pix2pix

1. はじめに

pix2pix とは、GAN (Generative Adversarial Networks) の技術を利用して、何らかの処理適用前後のペア画像を学習させることにより、汎用的な画像変換を行う生成器を獲得する手法である[1]。pix2pix を用いることにより実現が可能となった画像変換の1つとして、線画の自動着色がある。イラストレータは、色付けに習熟していなくても、完成度の高い色付きイラストを作成することができるようになった。

しかし、この自動着色には2つの問題点が存在する。1つは、自動着色で得られる画像は、入力する線画の質によって、着色の質が異なる点である。線画に基本的な特徴しか

描かれていなければ、着色は淡泊になる傾向にある。線画が対象物の様々な特徴を捉えて描かれていれば、高レベルな自動着色が可能である。もう1つは、入力する線画はユーザが手書きで用意しなければならない点である。したがって、高度なイラスト画像を獲得したい場合には、様々な特徴を捉えた線画を手書きで用意する必要がある。しかし、イラスト作成の初心者にとって、このような線画を自作することは煩雑な作業である。特徴を捉えた線画の作成を支援する従来技術も、現状ではあまり見受けられない。

本研究では、高レベルな自動着色を可能にする、特徴を捉えて描かれた線を自動的に生成する技術の確立を目的とする。描画対象の線画の一部を入力するだけで、その対象を特徴づけるその他の線を、自動的に補完することが可能

[†]大分大学
Oita University

であるかについて検証する。そのための手段として、輪郭のみの線画と特徴的な線が描かれた線画のペア画像を学習する pix2pix を提案する。

以下、2.で線画の自動着色を可能とした pix2pix の説明を述べた後、3.にて従来の手法に 4 つの加工を適用した提案手法の詳細を述べる。その後、4.で提案手法を用いた実験の結果と考察を示し、今後の展望について触れていく。

2. 従来手法

本章では、本研究の従来手法である pix2pix について説明する。その際に、pix2pix の基盤となった GAN について先に述べる[2]。

2.1 GAN

GAN とは敵対的生成ネットワークの略称であり、教師なし学習で使用するアルゴリズムの一種である。GAN は、学習時に、学習用画像に関する特徴を学習してモデル化する。テスト時には、そのモデルに基づいて学習用画像に似ている疑似画像を生成することができる。

GAN は、生成器と識別器と呼ばれる 2 つのネットワークから構成される。これら 2 つのネットワークを互いに競合させることで、生成器がより本物の画像に似た画像を生成できるようになる。生成器と識別器の基本的な役割として、生成器は実在しない画像を生成し、識別器は画像が実在するものかしないものかを識別する。

生成器と識別器による学習の流れを詳しく説明する。図 1 にて、その学習の流れを簡易的に表した GAN のモデルを示す。まず、生成画像の特徴の種に相当するランダムノイズベクトルを獲得する。そのベクトルを生成器に入力し、生成器は与えられたベクトルで学習用画像に似た画像を生成できるよう学習する。これにより、本物画像と偽物画像の 2 つが存在することになる。これらの画像の一方を識別器に与え、与えられた画像が本物か偽物かを識別器に判断させる。識別後、識別器は識別結果を基に、生成器に再度新たな本物に近い画像を生成させるようフィードバックし、生成器に学習させる。識別器も同様に、識別結果からどのような判断に基づくことで、より正確な識別ができるかを学習する。この一連の過程を何度も繰り返すことによって、最終的には生成器が限りなく本物の画像に近い偽物画像を生み出すことができるようになる。

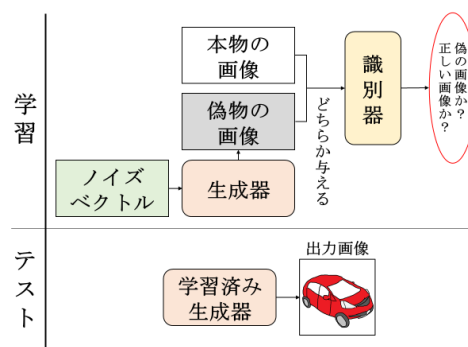


図 1 GAN における学習とテストの構造

Figure 1 Learning and testing of GAN.

2.2 pix2pix

pix2pix は、GAN の派生技術であり、基本的な構造および学習過程は通常の GAN と同じであるが、異なる点が 2 つ存在する。GAN との違いを明示した pix2pix の概要を図 2 に表す。

1 点目として、生成器が最初に偽物の画像を生成する際、ランダムノイズベクトルではなく、実在する学習用の変換前画像を用いて学習する。この処理に変更する理由として、GAN で最初に作成されるランダムノイズベクトルは、生成器によって生成される画像の種類を細かく分類できない。学習時に生成器が変換前画像から学習を開始することで、画像の種類判別を容易にする。また、GAN は何も手掛かりがない状態から学習を始めていたため、変換前画像が与えられることにより学習時間の短縮も可能となった。

2 点目は、識別器に与えられる画像が、単一画像ではなくペア画像が与えられる点である。学習内容について、複数の学習用画像から共通する特徴を導き出す完全推論問題の GAN とは異なり、pix2pix は画像から画像への対応関係を導き出す変換問題であるため、識別器は特徴を捉えやすくなる。識別内容が変わったことで学習内容の明確化にも繋がり、GAN よりも完成度の高い偽物画像を生成することができる。

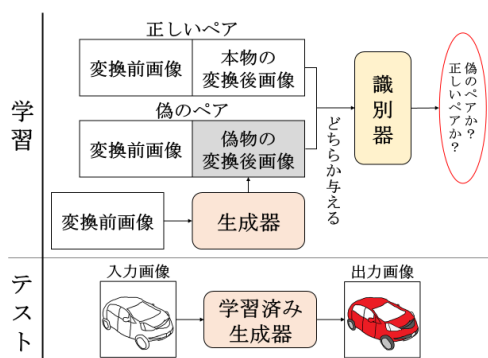


図2 pix2pixにおける学習とテストの構造

Figure 2 Learning and testing of pix2pix.

pix2pixによって、多くの画像変換が可能となった。[1]で紹介されている変換例が一般的であり、白黒画像のカラー画像化、昼の空模様を夜の空模様へ変換、ラベル画像のリアル画像化など、幅広い画像変換に応用することができる。その中でも、1.で述べた通り、筆者らは線画の自動着色に着眼した。

pix2pixによる線画の自動着色では、不足していると判断した線を自動的に補完する特長がある。図3にその様子を示す[3]。人の顔の自動着色を目的としたpix2pixへ、髪の毛等の特徴がないキャラクターの顔の線画を入力しても、髪の毛や鼻の線を補完した着色画像を出力できることを表している。当然、線の補完の可否は学習時に用いる学習用画像に依存するので、全ての特徴が補完される訳ではない。さらに、pix2pixの線の補完によって描かれる部分の特徴は、線が描かれた画像を入力した場合のpix2pixで描かれる特徴と比較して、補完される線によって得られる物体の境界線は曖昧になるため、極力線を描いた方が望ましい。



生成器A：人の顔の自動着色を目的としたpix2pixの学習済み生成器

図3 pix2pixによる自動着色の特長である、線の補完 ([3]より一部引用)

Figure 3 Line completion, the advantage of automatic coloring using pix2pix.

3. 提案手法

提案手法は、2つのpix2pixから構成されている。最初に、輪郭線のみの線画を入力し、特徴的な線の概略を出力するpix2pixを適用する。次に、その出力結果を、特徴的な線が描かれた最終的な線画に変換するpix2pixを適用する。以

下に、これらのpix2pixをどのように導入したか、また、オリジナルのpix2pixに対してどのような工夫を加えたかについて説明する。

3.1 概要

pix2pixの実行手順は、下記の3段階に分けることができる。

- ① 学習用画像の用意
- ② 学習
- ③ テスト

輪郭線のみの線画から特徴的な線の概略を生成するpix2pixについては、「①学習用画像の用意」段階にて、「輪郭のみの線画生成」「ぼかし処理」「射影変換」「外接矩形抽出」の4つの処理を適用した。これらについて、3.2から3.5で説明する。また、特徴的な線の概略から最終的な線画を生成するpix2pixについては、3.6で説明する。

3.2 輪郭のみの線画作成

変換前画像として輪郭のみの線画、変換後画像として特徴的な線が描かれた線画を用意する必要がある。図4にその用意する変換前画像と変換後画像の一例を示す。特徴的な線が描かれた線画は、カラー画像を対象とした一般的なエッジ検出手法を適用することにより獲得することを想定している。また、輪郭のみの線画は、特徴的な線が描かれた線画の輪郭を抽出して作成する。

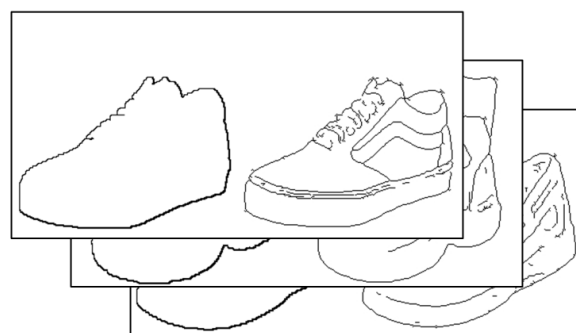


図4 提案手法の学習用データセット

Figure 4 Examples of the training dataset for the proposed method.

3.3 ぼかし処理

3.2にて用意した、輪郭のみの線画と特徴的な線が描かれた線画をそのまま用いてpix2pixに学習させたところ、得られた結果に特徴的な線は描かれなかった。原因として、

線画には空白が多く情報量が極端に少ないこと、特徴的な線の複雑性から学習時に線間の関係性を見いだしにくいことによる学習の難化が推測される。線画はほぼ白画素で構成されており、黒画素も細い線上に依存する。このような画素分布には画素値の空間的な勾配が存在しないため、pix2pix 学習時に勾配法を適用するには不向きである。効率的に pix2pix の学習を進めるためには、黒画素の周辺に、その黒画素が存在することを示す情報を追加する必要があると思われる。

この問題を解決するための手段として、本研究では、ぼかし処理の一種である Gaussian Blur を適用した[4]。この手法を用いてぼかし処理を学習用画像に施したところ、特徴的な線の概略が生成された。この結果から、学習用画像にぼかし処理を適用するものとした。

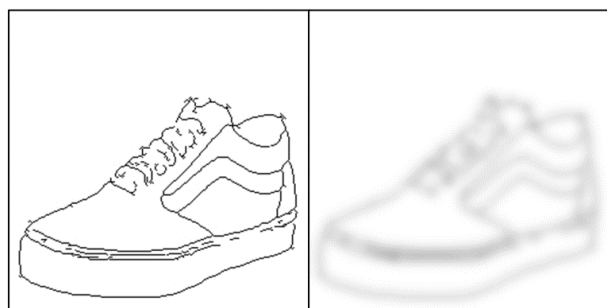


図5 Gaussian Blur によるぼかし処理
Figure 5 Blurring using Gaussian blur.

3.4 射影変換

事前に与えられた学習用画像について、対象の撮影方向が限られている場合がある。また、所望の特徴的な線を柔軟に生成するには、学習用画像の数が少ない恐れもある。そこで、多角度の視点から描かれた線画に対応させるため、また、十分な量の画像を学習させるために、3.3 のぼかし処理を加えた学習用画像にランダムな射影変換を施す。ここで用いる射影変換とは、2次元画像における左上、右上、左下、右下の四隅を画像の中心から外側に引き延ばすことで、ねじれの加わった画像を生み出す2次元射影空間での射影変換のことを表す。

変換過程について具体的に説明する。図6にその過程を示す。まず、入力画像を1/2サイズに縮小し、図6の緑枠内の領域を確保する。これらの領域内に、変換後画像の四隅の座標をランダムに決定し、射影変換により引き延ばす。例えば、図6の縮小画像の右上の点であれば、右上方向に伸びる赤色の矢印がその処理を表している。以上の過程により、基の画像とは異なる形状の線画を作成することができる。図7の出力画像1から出力画像4に、獲得できる様々な形状の線画の一例を表している。

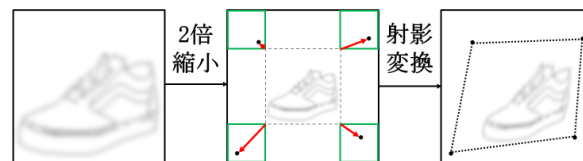


図6 射影変換
Figure 6 Perspective transform.

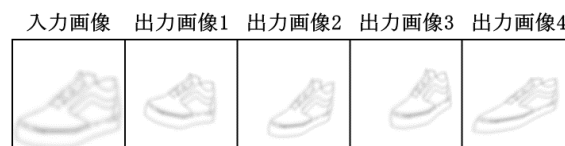


図7 射影変換後画像
Figure 7 Images after applying the perspective transform.

3.5 外接矩形抽出

3.4の射影変換を施した線画は、元の線画と比較してサイズが小さくなる。3.4の縮小された画像で学習を行い、テスト画像を基のサイズの画像として実験したところ、所望の結果が全く得られなかった。この原因として、画像サイズが同じでなければ、学習内容を適用することができないと考えられる。そこで、図8に示す手順で、射影変換にて縮小した線画の外接矩形を獲得し、画像サイズを正規化する。

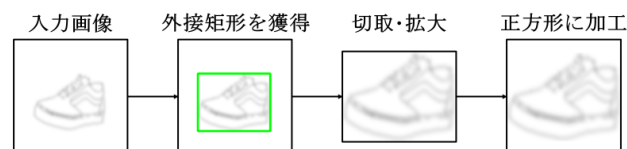


図8 外接矩形獲得による画像正規化
Figure 8 Image normalization using minimum bounding rectangle.

3.6 ぼかし画像の線画変換

3.2から3.5にかけて紹介した加工を適用して特徴的な線が描かれた線画を生成することができる。しかし、ぼかし処理を適用しているため、獲得する線画もぼかし処理が加わったままである。そこでぼかした画像を元に戻す手順として、ぼかし画像を線画へ変換する pix2pix を導入する。

学習用画像として、線画に同じぼかし処理を施したぼかし画像を用意すればよい。学習する線画に3.3と全く同じ Gaussian Blur を適用する。ぼかした画像を変換前画像、線画を変換後画像とし、pix2pix を用いて学習させることにより、ぼかし画像を線画に変換することができるモデルを獲得することができる。このぼかし復元処理によって生成し

た線画を図 9 に示す。入力したぼかし画像を入力画像，出力された線画を出力画像と表記している。



図 9 ぼかし画像を線画へ変換する pix2pix

Figure 9 Results of pix2pix that converts blurred images to line drawings.

4. 実験

本章では，3.で説明した提案手法を検証する実験の概要について説明し，実験結果と考察を述べる。本実験では，pix2pix を用いた自動着色において公開されている学習用データセット[a]のうち，靴画像のデータセットを用いた。ここでは，輪郭のみの線画から特徴的な線が描かれた線画を生成する実験、及び，生成された線画に対して[1]で提案されている線画の自動着色を適用し，最終的にどのような画像が得られるかに関する実験を行った。

両実験はどちらも同じ計算機の構成で実行している。ディープラーニングフレームワークは TensorFlow，言語は Python，OS は Windows 10，CPU は Core i9-9900k，GPU は GeForce RTX 2080 Ti を用いた。

4.1 線画生成

まず初めに，輪郭のみの線画から，特徴的な線の概略を表す画像を生成する実験を行った。3.で紹介した，輪郭のみの線画生成，ぼかし処理，射影変換，外接矩形の4つの加工を，[a]の学習用画像，テスト画像それぞれに適用する。実際に使用したデータセットの様子を図 10 に示す。学習用画像の左画像は図 2 の学習時の変換前画像であり，右画像は図 2 の学習時の本物の変換後画像である。テスト画像の左画像は図 2 のテスト時の入力画像であり，右画像は入力画像の正解画像である。

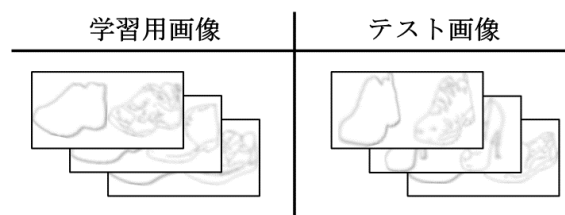


図 10 線画生成におけるデータセット

Figure 10 Datasets for the experiment on generating line drawings.

学習用画像の枚数は 1,000 枚である。内訳として，100 種類の靴画像に 10 パターンのランダムな射影変換を適用した，100 種類の靴画像 1,000 枚である。具体的には，靴の種類としてヒール，革靴，スニーカー，サンダルの画像を用いた。画像の枚数は，ヒール 10 枚，革靴 40 枚，スニーカー 40 枚，サンダル 10 枚である。したがって，本実験で取り扱う学習用靴画像は，大きく 4 種類に分けられる合計 100 枚の靴画像である。

学習回数は 100 回である。先に述べた学習用画像 1,000 枚を用いて，学習回数の最適な値を決めるため，25 回，50 回，100 回，1,000 回，10,000 回の学習回数で実験を行い，調査した。結果として，学習回数が 25 回と 50 回のモデルでは主観的に不自然な線が描かれた出力画像が多かった。学習回数が 100 回，1,000 回，10,000 回のモデルで得られた画像は，それぞれ不可なく線画が得られるものの，同じ靴画像の入出力結果に線画の違いは見られなかった。以上の結果から，学習回数は 100 回が適切であると考え，設定した。

4.1.1 実験結果

4.1 の実験結果を図 11 に表す。テスト画像の入力画像である輪郭のみの線画に対し，得られた線画を出力画像として表している。入力画像の基となった本来の線画を正解画像として出力画像の下に添付している。テスト画像は 500 種類の靴画像に 10 パターンの射影変換を適用した 5,000 枚が対象であり，その一部を選別している。

a) <https://people.eecs.berkeley.edu/~tinghuiz/projects/pix2pix/datasets/edges2shoes.tar.gz>



図 11 線画生成の実験結果

Figure 11 Experimental results of generating line drawings.

4.1.2 考察

出力結果が線画であり、明確な比較対象がない画像であるため、数値による定量的な評価が難しい。そのため、「見た目が良い」か「見た目が悪い」かの定性的な評価を行い、考察を述べる。

複雑な形状が少なく、特徴的な線の数が比較的少ないヒールと革靴では、見た目が良い出力結果が得られた。一方で、形状が複雑なスニーカーとサンダルは見た目が悪い靴画像が幾らか出力された。特にスニーカーは、特徴的な線も複雑であるため、多くの出力画像に見た目が悪いと評価した。

しかし、2.2 にて述べた通り、従来手法の自動着色時に必要な線が一部補完される。したがって、線画の見た目が悪い靴も着色時に補完され、見た目の良い着色画像が獲得できる可能性がある。そこで、図 11 に示した出力結果ではなく、図 11 に自動着色を施した画像で最終的な見た目の良し悪しを判断するべきと考える。

4.2 着色

4.1.2 の考察に基づき、4.1.1 において生成された全ての線画に対して、3.6 のぼかし画像を線画へ変換する pix2pix を適用した後に、[1]で提案されている線画の自動着色を適用した。着色実験での学習用画像は、[a]の学習用画像に射影変換と外接矩形の 2 つの加工を適用した画像である。図 12 に着色テストにおける学習用画像とテスト画像を示す。図 10 と同じく、学習用画像の左画像は変換前画像であり、右画像は変換後画像である。テスト画像では、提案手法により生成した画像を入力画像として用いているため、図 10 とは異なり提案手法により正解画像は存在しない。



図 12 着色におけるデータセット

Figure 12 Datasets for the experiment on coloring line drawings.

着色実験では、5,000 枚の靴画像に 10 パターンのランダムな射影変換を適用した 50,000 枚の学習用画像で学習する。4.1 同様、靴の種類として、ヒール、革靴、スニーカー、サンダルの画像を用いた。

4.2.1 実験結果

図 13 に 4.2 の実験結果を示す。4.2 で述べたように、正解画像は存在しない。テスト画像は 4.1.1 の出力結果を使用しているため、5,000 枚であり、4.1.1 と同様にその一部を抜粋している。



図 13 着色の実験結果

Figure 13 Experimental results of coloring line drawings.

4.2.2 考察

4.1.2 で述べた通り、テスト画像の着色時に線が補完された。線画生成時に既に見た目が良いと評価したヒールと革靴は、問題なく着色された。形状が複雑なため一部見た目が悪いと評価した。サンダルもうまく補完され、着色画像は見た目が良いと判断することができる。一方、複数の形状が存在し、特徴的な線が複雑なスニーカーに関しては、見た目が悪いと評価した。着色の補完を適用しても、線が複雑なスニーカーを、リアルな靴に再現することはできなかった。

柄模様で構成されている靴の着色も失敗した。ヒールからサンダルまで全ての靴において、派手な模様の靴への着色は失敗している。以上の結果から、線がシンプルな形状の靴に対してはうまく着色できたが、線が複雑な形状の靴

にはうまく着色できないことがわかった。改善点として、まずは構造が複雑な画像に対する画像拡張が必要であると考ええる。

5. まとめ

本研究では、イラストレータが色付きイラストを作成する際に、輪郭のみの線画を描くだけで自動的に色付きイラストを生成する手法を提案した。提案手法は、従来手法の pix2pix に、輪郭のみの線画生成、ぼかし処理、射影変換、外接矩形の抽出など、4 つの処理を適用し、輪郭のみの線画から、特徴的な線の概略を表す画像を生成する。その後、特徴的な線の概略から最終的な線画を生成する pix2pix を適用し、特徴的な線が描かれた線画を生成する。この線画に対して、従来手法である線画を着色する pix2pix を適用すると、色付きイラストを生成することができる。

実際に、輪郭のみの線画から特徴的な線が描かれた線画を生成する実験と、獲得した線画を用いて従来手法の自動着色を検証する実験を行った。その結果、シンプルなデザインのヒール、革靴、サンダルについては、見た目が良い着色画像が得られた。一方、構造が複雑なスニーカーや柄模様のある靴では、見た目が悪い結果となった。

この結果を受け、今後、構造が複雑な靴の学習用画像を拡張していく。また、様々な評価方法を確立することで、得られた出力画像の評価を明確に査定していく必要がある。

謝辞

本研究では、JSPS 科研費 JP19K12045 の助成を受けたものです。

参考文献

- [1] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, Alexei A. Efros, "Image-to-Image Translation with Conditional Adversarial Networks"; Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 1125-1134
- [2] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, Yoshua Bengio, "Generative Adversarial Networks"; Advances in Information Processing Systems 27(NIPS), 2014
- [3] Rob Prince, "This AI will turn your doodled self-portraits into horrifying nightmares", <https://www.businessinsider.com/pix2pix-ai-doodle-self-portraits-photo-realistic-pictures-2017-6>
- [4] Moo K. Chung, "University of Wisconsin-Madison – STAT 692 Medical Image Analysis – 3. The Gaussian kernel", <http://pages.stat.wisc.edu/~mchung/teaching/MIA/reading/diffusion.gaussian.kernel.pdf>