

ニューラル画像対画像変換による毛筆体文字の生成

島崎早智[†] 多田信洋[†] 藤岡寛之[†] 日高章理^{††}[†]福岡工業大学情報工学部 ^{††}東京電機大学理工学部

1 はじめに

ヒトの書字運動をヒントにした“動的フォント法”とよばれるフォント生成法が提案されている [1]。この方法では、仮想的な筆記具と筆記面を考え、B スプラインを用いた滑らかな 3 次元軌道に沿って筆記具を運動させた結果として文字フォントを生成する。特に、運筆時の太さが重要となる毛筆体文字の設計・再設計において有用であることが示されている [2]。しかし、筆記具が円錐形状という単純モデルで定義されており、実際に毛筆を用いた書字時にみられるかすれやにじみといった墨線の表現までできていない。このような自然な墨線の表現まで可能な毛筆体文字フォントの生成をめざし、ごく最近、Sawangphol ら [3] により畳み込みネットワークを用いたペン字から毛筆体文字への生成法が提案された。しかし、ネットワーク学習用の文字画像データを手動で生成しており、データ数が小数であったこともあるが、複雑な運筆をもつ文字に対してはデコードした画像がややぼやけるといった課題が残っている。

そこで本研究では、以上のような課題を解決し、太さが一定の細線文字からかすれ等をもつより自然な毛書体文字を生成するためにニューラル画像対画像変換法である Cycle GAN [4] を導入した毛筆体文字を生成する方法を提案する。その際の学習用のデータとして手書き毛筆フォントである衡山毛筆フォント [5] の 2273 文字分の画像データと、それらを Zhang-Suen 法により細線化した画像データの集合データを用いる。実験において生成例を示すことで、本手法の有用性を示す。

2 画像対画像変換による毛筆体文字の生成

ここでは、太さが一定の細線文字からかすれ等をもつより自然な毛書体文字を生成する問題を考える。そのような生成問題に対して、以下のように画像対画像変換法である Cycle GAN [4] を導入する。

今、一定の太さをもつ文字画像データの集合を X 、毛筆体文字の画像データの集合を Y とする。このとき、Cycle GAN により、集合 X と集合 Y の集合間の写像の学習を行い、その結果として得られる集合 X から Y への写像を毛筆体生成法として利用する。

集合 X から Y への変換を行う画像生成写像を $G: X \rightarrow Y$ 、その逆方向の生成写像を $F: Y \rightarrow X$ とする。加えて、 X および Y がそれぞれ G および F により生成された偽のデータか否かを判別する鑑別器を D_Y と D_X として定義する。このとき、集合 X から Y を利用して、生成器 G, F と鑑別器 D_X, D_Y を以下の評価関数を利用して学習する。

$$\begin{aligned} L(G, F, D_X, D_Y) = & L_{GAN}(G, D_Y, X, Y) \\ & + L_{GAN}(F, D_X, Y, X) \\ & + \lambda L_{cyc}(G, F) \end{aligned} \quad (1)$$

ただし、 $\lambda (> 0)$ はハイパーパラメータである。 $L_{GAN}(G, D_Y, X, Y), L_{GAN}(F, D_X, Y, X)$ は Adversarial Loss、 $L_{cyc}(G, F)$ は Cycle Consistency Loss と呼ばれる損失であり、以下のように与えられる。

$$\begin{aligned} L_{GAN}(G, D_Y, X, Y) = & \mathbb{E}_{y \sim P_{data}(y)} [\log D_Y(y)] \\ & + \mathbb{E}_{x \sim P_{data}(x)} [\log(1 - D_Y(G(x)))] \end{aligned} \quad (2)$$

$$\begin{aligned} L_{GAN}(F, D_X, Y, X) = & \mathbb{E}_{x \sim P_{data}(x)} [\log D_X(x)] \\ & + \mathbb{E}_{y \sim P_{data}(y)} [\log(1 - D_X(F(y)))] \end{aligned} \quad (3)$$

$$\begin{aligned} L_{cyc}(G, F) = & \mathbb{E}_{x \sim P_{data}(x)} [\|F(G(x)) - x\|_1] \\ & + \mathbb{E}_{y \sim P_{data}(y)} [\|G(F(y)) - y\|_1] \end{aligned} \quad (4)$$

特に、(4) 式の Cycle Consistency Loss は、いわゆる再構築誤差であり、学習がうまくいった場合には、「太さが一定の細線文字画像データの集合 X と毛筆体文字の画像データの集合 Y に共通する構造を保ったまま、集合 X に属する文字画像をもう一方の集合 Y に属する文字画像へと変換する」写像が生成器 G, F として得られることになる。その際の生成器 G を毛筆体文字を生成するために利用する。

3 生成実験

2 節の Cycle GAN を使った毛筆体文字の生成法の有用性を実験により検討する。以下では、3.1 節で Cycle GAN の実装の概略を述べる。3.2 節で毛筆体文字の生成実験の結果を示す。

Generation of Hairy-Brush Style Characters Using Image-to-Image Translation

[†] Sachi SHIMASAKI (s15m1034@bene.fit.ac.jp)

[†] Nobuhiro TADA (s16a1041@bene.fit.ac.jp)

[†] Hiroyuki FUJIOKA (fujioka@fit.ac.jp)

^{††} Akinori Hidaka (hidaka.akinori@mail.dendai.ac.jp)

Faculty of Information Engineering, Fukuoka Institute of Technology ([†])

Faculty of Science and Engineering, Tokyo Denki University (^{††})

	Best5				Worst5			
	Input	Output	Ground Truth	R	Input	Output	Ground Truth	R
1	い	い	い	0.986232	鶴	鶴	鶴	0.906485
2	っ	っ	っ	0.983661	画	画	画	0.911678
3	づ	づ	づ	0.978722	諭	諭	諭	0.913418
4	べ	べ	べ	0.978576	籠	籠	籠	0.913708
5	ウ	ウ	ウ	0.978274	番	番	番	0.91753

図 1: 毛筆体文字の生成実験の結果



(a) 入力



(b) 生成結果

図 2: ペンタブレットで筆記した文字からの生成例

3.1 実装

2 節の Cycle GAN の生成器 G, F および鑑別器 D_X, D_Y のネットワーク構造に関しては, Zhu ら [4] のものと同一のものを使用する。(1) 式の λ と学習係数に関しても [4] に合わせて $\lambda = 10$ および 2×10^{-4} とする。また, Cycle GAN の学習には手書き毛筆フォントである衡山毛筆フォント [5] の約 14000 文字のうちランダムに選んだ 2073 文字分の画像データと, それらを Zhang-Suen 法 [6] により細線化した 2073 文字分の画像データの集合データをそれぞれ画像データの集合 Y と X とする。画像サイズは集合 X, Y ともに 256×256 [pix] である。以上の実装は, GPU(NVIDIA GeForce GTX980) を装備した PC (CPU: Intel Core-i7-6700K, RAM: 32GB) 上で Python および Tensorflow[7] を用いて行う。なお, 学習におけるバッチ数は 1, エポック数は 150 とした。

3.2 生成結果

3.1 節の実装および学習により得られた生成器 G を用いて毛筆体文字の生成実験を行う。テスト画像として, 学習には使用していない衡山毛筆フォント [5] のランダムに選んだ 200 文字分の画像を Ground Truth に用い, さらにそれらの文字画像を 3.1 節と同様の方法で細線化した文字画像を生成器 G への入力として用いる。その際に生成器 G により生成される毛筆体文字の画像と Ground Truth との正規化相互相関 R を求めることで本手法の有用性を確認する。図 2 に実験結果をまとめる。図右側と左側は, それぞれ正規化相互相関 R が高いと低いものから 5 つずつ表している。最も正規化相互相関 R が低いものでも 0.90 程度である。また, 図 3 に, ペンタブレットで筆記して取得した文

字画像から毛筆体文字を生成した結果を示す。以上から, 本手法は一定の太さをもつ文字から自然な毛書体文字を十分に生成することができるがわかる。

参考文献

- [1] K.Takayama and H.Kano et al, Dynamic Font: A New Representation Technology, *FUJITSU Sci. and Tech. J.*, Vol.32, No2, pp.192-202, 1996.
- [2] H. Fujioka, H. Kano, H. Nakata and H. Shinoda, Constructing and Reconstructing Characters, Words and Sentences by Synthesizing Writing Motions, *IEEE Trans. Systems, Man and Cybernetics, Part A*, Vol.36, No.4, pp.661-670, 2006.
- [3] J. Sawangphol, R. Soontornvorn and H. Fujioka, Design of Hairy Brush Characters Using Convolutional Encoder-Decoder Network, 第 71 回電気・情報関係学会九州支部連合大会講演論文集, 2 ページ, 大分, 2018 年 9 月 27 日-28 日.
- [4] J-Y. Zhu, T. Park, P. Isola, A. A. Efros, Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks, *arXiv.org*, Computer Vision and Pattern Recognition, arXiv:1703.10593v6, Nov, 2018.
- [5] 青柳 衡山, 衡山毛筆フォント, <https://opentype.jp/kouzanmouhitufont.htm>.
- [6] R. C. Gonzalez, R. E. Woods and S. L. Eddins, *Digital Image Processing Using MATLAB*, Pearson Education Inc., 2004.
- [7] T. Hope, Y. S. Resheff and I. Lieder, *Learning TensorFlow: A Guide to Building Deep Learning Systems*, O'Reilly Media, 2017.