Vol.2021-NL-248 No.32
Vol.2021-CVIM-226 No.32
2021/5/21

深層手書き漢字生成

和田 有輝也[1,a)]　延原 章平[1,b)]　西野 恒[1,c)]

概要：少量の教師なしデータで学習する手書き漢字生成の既存手法では文字全体の構造的特徴の再現が十分になされていない．そこで，訓練済み手書き漢字認識モデルを用いて訓練データの一部に自動でアノテーションを行うことで，比較的少ない数の教師なしデータを用いながらも，高精度に構造的特徴を再現する手書き漢字生成手法を提案する．教師なしデータで学習できるように CycleGAN をベースとしたネットワークを設計する．さらに，自動アノテーションによって得られた教師ありデータを活用すべく，生成画像と対応する画像を直接比較する損失や，変換前後の対となる画像のペアに対する敵対性損失を導入する．また，生成画像が漢字としての構造を維持するように，目的関数に訓練済み文字認識器を用いた損失を追加する．実験によって，提案手法が既存手法と比較して，特に構造的特徴や部首等の特定の構成要素についての書き癖の再現の点で高い精度の生成を実現することが確認された．

1. はじめに

同一フォントの活字や同一の筆者によって書かれた手書き文字はそれぞれ共通したスタイルを有している．共通のスタイルをもつ文字画像集合からそのスタイルを獲得し，同一のスタイルをもつ新たな文字画像を生成する手法についてはその幅広い応用のために様々な研究がなされている．応用例としてはフォント作成支援や画像中の文字編集，文字認識等の文字画像データを必要とする技術のためのデータ拡張が挙げられる．

漢字は算用数字やアルファベットに比べて，文字の種類が膨大であり，かつ画数が多く，形状も複雑である．そのため，例えば，漢字のフォント作成にかかるコストは非常に高く，漢字生成による支援の効果は大きい．漢字生成の中でも，特に手書きの漢字を対象とした手書き漢字生成がある．手書き文字は活字と比較して，筆者の書き癖のためにしばしばはねやはらいが省略されたり，文字全体が大きく歪んでいたりするため，スタイルの獲得は難しい．

漢字生成は文字の見本となる既存のフォント（ソースフォント）の漢字画像から生成したいスタイル（ターゲットフォント）の漢字画像への変換として定式化されてきた．しかしながら，手書き漢字生成モデルの学習のために筆者から大量の手書き漢字を収集したり，その一つ一つにアノテーションするのは大きな労力を要する．一方で，ある筆者が手書きで作成した文書やメモ等にはその筆者の手書き漢字のサンプルが既に存在している．こうしたサンプルはその漢字が何であるか等の付加的な情報をもたない教師なしデータである．手書き漢字生成はそうした限られた数の教師なしデータから変換を学習できることが望ましい．しかしながら，比較的少ない数の教師なしデータで手書き漢字生成を行う既存の手法 [1] では，字画の質感等の局所的特徴は十分に再現できるものの，文字全体の構造的な特徴の再現が不十分である．

そこで，本研究では比較的少ない数の教師なしデータによる手書き漢字生成の精度を，特に構造的な特徴の再現の点において改善することを目的とする．そのために，変換を学習する前に訓練済み文字認識器を用いて，教師なし訓練データの一部に自動でその漢字が何であるかを示すラベルを付与し，そのラベルを活用することで精度の向上を図る手法を提案する．

2. 関連研究

ここでは，我々が提案する手書き漢字生成手法の関連研究として，深層学習を用いた画像のスタイル変換と手書き漢字生成についての代表的なものを紹介する．

2.1 深層学習を用いた画像のスタイル変換

深層学習を用いた，一般の画像変換問題に対する汎用的なフレームワークとして pix2pix [2] が広く知られている．pix2pix は Generative Adversarial Network (GAN) [3] のフレームワークを用いたモデルであり，変換前後の対となる画像のペアから画像間の対応関係を学習する．

[1] 京都大学大学院情報学研究科
[a)] ywada@vision.ist.i.kyoto-u.ac.jp
[b)] nob@i.kyoto-u.ac.jp
[c)] kon@i.kyoto-u.ac.jp

ⓒ 2021 Information Processing Society of Japan

一方で，ペアになっていない 2 つの画像データセットからそれらのドメイン間の変換を学習するフレームワークとして CycleGAN [4] がある．CycleGAN では画像のペアを用いずに変換を学習するために，一方のドメインから他方のドメインへ画像を変換する生成器に加えて画像を逆方向に変換する生成器を導入する．また，ドメイン間の一対一の対応を学習するために，順方向の変換と逆方向の変換を経た画像が元の画像と一致するように促すサイクル一貫性損失を導入する．

ペアになっていない画像のデータセットの他に少量の画像のペアが用意できる場合に有効なものとして，CycleGAN を半教師あり学習に適応させたいくつかの手法が提案されている．Mondal らは，画像のペアに対しては生成器の出力と対応する画像とを直接比較することにより，ペアをなさない画像に対してはサイクル一貫性損失と敵対性損失により生成器を学習する手法を提案した [5]．また，Nguyen らは半教師あり学習のためのモデルとして Semi-Supervised Adversarial CycleGAN (SSA-CGAN) を提案した [6]．SSA-CGAN は入力された 2 枚の画像が画像のペアらしいかどうかを識別するモジュールを CycleGAN に加えたモデルである．この識別器は本物の画像のペアと少なくとも一方が生成画像であるペアによって訓練される．本研究では教師なしデータの一部に付与したラベルを活用するために，半教師あり CycleGAN の手法を取り入れる．

2.2 手書き漢字生成

EasyFont [7] は漢字を構成する字画についての事前情報を活用することで，少数の手書き漢字のサンプルから，ソースフォントからターゲットフォントへの変換を学習する手法である．しかしながら，あるターゲットフォントの漢字を生成するためには対応するソースフォントの漢字にその字画についての情報を与える必要がある．これには漢字についての専門知識が必要である上に，大きな手間がかかる．

深層学習を用いた漢字生成の手法としては zi2zi [8] がある．zi2zi は pix2pix [2] をベースとした手法である．漢字についての専門的な事前情報は不要であり，ソースフォントとターゲットフォントの漢字画像のペアのみから end-to-end な変換を学習する．一方で，zi2zi は訓練データとして 1 つのスタイルあたり 1000 字以上という大量の教師ありデータを必要とする．大量の手書き漢字を収集し，その一つ一つにラベルを付与するのは大きな労力を要する．そのため，大量の教師ありデータを必要とする zi2zi は手書き漢字生成には向いていない．

教師なし学習によるアプローチとしては CycleGAN を用いた手法 [1], [9] がある．特に手書き漢字に着目した手法 [1] では 400 字程度の比較的少ない数の教師なし訓練データからでも生成が可能であることを実験により示している．しかしながら，この手法により生成された漢字画像は，字画の質感等の局所的特徴は十分に再現できている一方で文字全体の構造的な特徴の再現が不十分であり，生成精度の観点から改善の余地がある．本研究では必要とするデータの条件を変更しないまま，特に構造的な特徴の再現について生成精度を改善する．

3. 深層手書き漢字生成モデル

本研究では手書き漢字生成をソースフォント画像 $x \in X$ からターゲットフォント画像 $y \in Y$ への変換として定式化する．ただし，X はソースフォントドメイン，Y はターゲットフォントドメインである．ソースフォントは文字の見本として用意した既存の活字漢字のフォントであり，ターゲットフォントは生成したいスタイルの手書き漢字であるとする．また，生成器 $G : X \to Y$ が手書き漢字生成モデルであるとする．

ここで，本研究の提案手法は次の 2 つの段階に分割することができる．

教師なしデータに対する自動ラベル付与

教師なしターゲットフォントデータセットの一部に自動でラベルを付与し，半教師ありデータセットとする．ここで漢字画像に対するラベルとはその画像の漢字が何であるかを表す情報であり，例えば Unicode によって表現される．

手書き漢字生成モデルの学習

ソースフォントデータセットと半教師ありターゲットフォントデータセットを訓練データとして，手書き漢字生成モデル G を訓練する．ここで，ソースフォント画像 x は全てラベルを有している．同じラベルを有している画像 x_p, y_p は画像のペア $\{x_p, y_p\}$ として用いることができる．X, Y の部分集合で，ペアとして用いることができる画像の集合をそれぞれ X_p, Y_p，ペアをなさない画像の集合をそれぞれ X_u, Y_u とする．

3.1 教師なしデータに対する自動ラベル付与

ターゲットフォント画像 y を訓練済み手書き漢字認識器に入力し，その認識結果として得られた漢字に対応するラベルを y に付与する．ラベル付与により得られた半教師ありデータセットをモデルの学習のための訓練データセットとして用いる．ラベル付与によって増大した情報を活用することで生成精度の向上が期待できる．

ここで，教師なしデータセットの全ての画像にラベルを付与することで教師ありデータセットにすることも可能である．しかし，一般に認識器の認識結果は必ずしも正確ではない．ラベル付与に用いる訓練済み手書き漢字認識器の認識結果に誤りがあった場合，その画像には誤ったラベルが付与される．誤ったラベルが付与されたデータはモデルの学習に悪影響を及ぼす．そこで，誤った認識結果を採用しないために，認識器の予測の確信度の高い一部の画像にのみラベルを付ける．

ⓒ 2021 Information Processing Society of Japan

Vol.2021-NL-248 No.32
Vol.2021-CVIM-226 No.32
2021/5/21

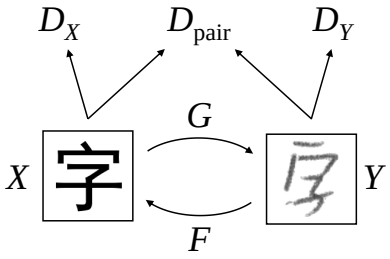

図 1: 提案手法におけるネットワークの構成

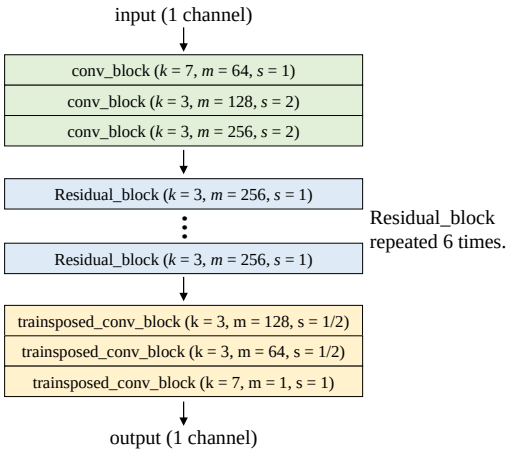

図 2: 生成器 G, F の実装．conv_block は畳み込み層，インスタンス正規化，ReLU 層から成るブロックである．Residual_block は ResNet [10] を構成するブロックである．transposed_conv_block は転置畳み込み層，インスタンス正規化，ReLU 層から成るブロックである．なお，k はカーネルサイズ，m はフィルタ数，s はストライドを表す．

3.2 ネットワークの構成

本手法で用いるネットワークは CycleGAN [4] をベースとしており，生成器 G, F，識別器 D_X, D_Y, D_{pair} によって構成されている．ネットワークの全体図を**図 1** に示す．以下，各構成要素の設計について述べる．

生成器 G はソースフォント画像 x をターゲットフォント画像 $G(x)$ に変換する．また，生成器 F はターゲットフォント画像 y をソースフォント画像 $F(y)$ に変換する．生成器 G, F はいずれも**図 2** に示す Encoder-Decoder モデルのネットワークである．なお，エンコーダとデコーダの間の変換ネットワークとして 6 ブロックの ResNet [10] (ResNet-6) を用いた．

識別器 D_X,D_Y は入力された画像がドメイン X,Y の真のデータであるか，生成器 F,G によって生成されたデータであるかをそれぞれ分類する．D_X,D_Y はパッチサイズが 70×70 の PatchGAN [2] である．PatchGAN は入力画像全体に対して真偽を判定する識別器と比較してパラメータが少ないという利点がある．

識別器 D_{pair} は SSA-CGAN [6] で導入された半教師あり学習のためのモジュールである．D_{pair} は入力された 2 枚の画像が真の画像のペア $\{x_p, y_p\}$ であるか，少なくとも一方が生成器 G または F によって生成されたペア $\{x_p, G(x_p)\}$，$\{F(y_p), y_p\}$，$\{F(y_p), G(x_p)\}$ であるかを分類する．**図 3** に示すように，D_{pair} は 3 つのモジュール $D_{\text{head}/X}, D_{\text{head}/Y}, D_{\text{foot}}$ によって構成されている．$D_{\text{head}/X}, D_{\text{head}/Y}$ は X, Y の画像をそれぞれ特徴マップに変換する．D_{foot} は $D_{\text{head}/X}, D_{\text{head}/Y}$ から得られる 2 つの特徴マップを結合したテンソルを入力として，D_{pair} としての識別結果を出力する．ここで，ある画像のペアに対して，その一方が真のデータであるかどうかという情報とそのペアが真の画像のペアであるかどうかという情報は独立ではない．そのため，D_X, D_Y と D_{pair} の層を部分的に共有することでそれぞれの学習の効率を向上させることができると考えられる．そこで，$D_{\text{head}/X}, D_{\text{head}/Y}$ をそれぞれ D_X, D_Y の一部とすることで，D_{pair} は D_X, D_Y と層

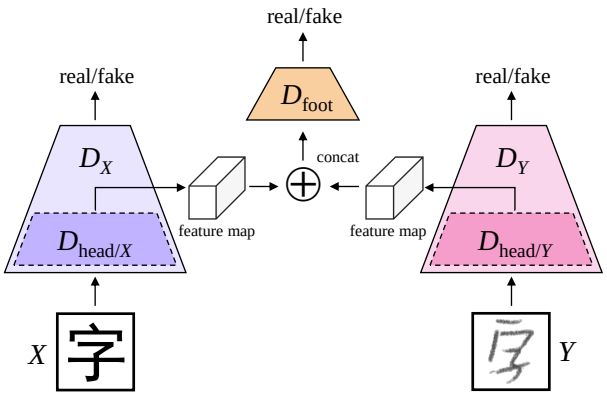

図 3: 識別器 D_{pair} の設計．D_{pair} は $D_{\text{head}/X}$,$D_{\text{head}/Y}$, D_{foot} の 3 つのモジュールによって構成されている．

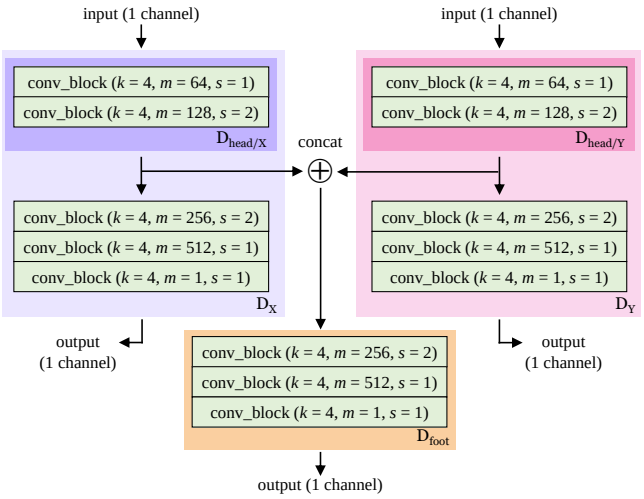

図 4: 識別器 $D_X, D_Y, D_{\text{pair}}$ の実装．ただし，各ブロックにおいて ReLU 層の代わりに $\alpha = 0.2$ の Leaky ReLU 層を用いる．また，$D_{\text{head}/X}, D_{\text{head}/Y}$ の各層には 50%のドロップアウトを適用する．

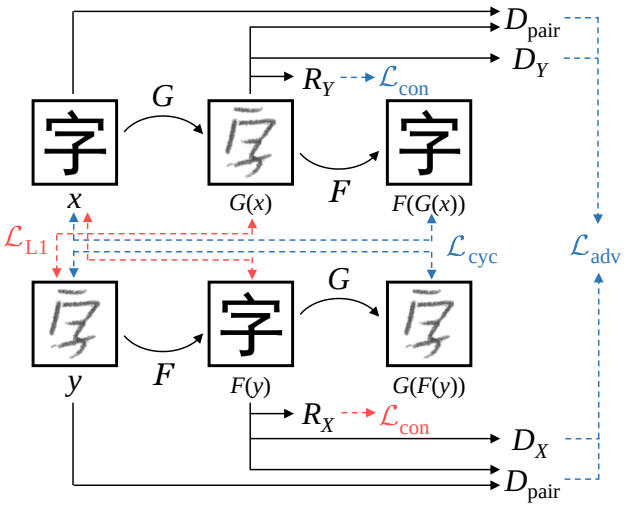

図 5: 生成器 G, F と損失項の関係．青で示した損失項は入力がペアをなすかどうかによらずに与えられ，赤で示した損失項は入力がペアをなすときのみ与えられる．

を部分的に共有している．

3.3 目的関数

生成器 G, F を学習するための目的関数は，

$$\begin{aligned}\mathcal{L}_{\text{G}}(G,F) = \mathcal{L}_{\text{adv}}(G,F) + \lambda_{\text{cyc/u}}\mathcal{L}_{\text{cyc/u}}(G,F) \\ +\lambda_{\text{cyc/p}}\mathcal{L}_{\text{cyc/p}}(G,F) + \lambda_{\text{L1}}\mathcal{L}_{\text{L1}}(G,F) \\ +\lambda_{\text{con}}\mathcal{L}_{\text{con}}(G,F)\end{aligned} \tag{1}$$

である．ここで $\mathcal{L}_{\text{adv}}$ は生成器についての敵対性損失，$\mathcal{L}_{\text{cyc/u}}$ 及び $\mathcal{L}_{\text{cyc/p}}$ はサイクル一貫性損失，$\mathcal{L}_{\text{L1}}$ はラベルを用いた L1 損失，$\mathcal{L}_{\text{con}}$ は文字認識器を用いた損失である．生成器 G, F と損失項の関係を**図 5** に示す．また，識別器 $D_X, D_Y, D_{\text{pair}}$ を学習するための目的関数は，

$$\begin{aligned}&\mathcal{L}_{\text{D}}(D_X, D_Y, D_{\text{pair}}) = \\ &\quad \mathcal{L}_{D_X}(D_X) + \mathcal{L}_{D_Y}(D_Y) + \lambda_{D_{\text{pair}}}\mathcal{L}_{D_{\text{pair}}}(D_{\text{pair}})\end{aligned} \tag{2}$$

である．ここで $\mathcal{L}_{D_X}$, $\mathcal{L}_{D_Y}$, $\mathcal{L}_{D_{\text{pair}}}$ はそれぞれ識別器 D_X, D_Y, D_{pair} に対する敵対性損失である．学習では $\mathcal{L}_{\text{G}}(G,F)$ が G, F について，$\mathcal{L}_{\text{D}}(D_X, D_Y, D_{\text{pair}})$ が $D_X, D_Y, D_{\text{pair}}$ についてそれぞれ最小となるよう最適化する．以下，式 (1), (2) における各損失項について述べる．

3.3.1 敵対性損失

敵対性損失は生成される画像の分布と真のデータの分布を一致させることを目的とした損失である．なお，敵対性損失には LSGAN [11] を採用する．LSGAN は Goodfellow らが提案した敵対性損失 [3] に比べて学習が安定することで知られている．

識別器 D_X, D_{pair} に対する敵対性損失 $\mathcal{L}_{D_X}$, $\mathcal{L}_{D_{\text{pair}}}$ は，

$$\begin{aligned}\mathcal{L}_{D_X}(D_X) =& \frac{1}{2}\mathbb{E}_{x\sim X}\left[\left(D_X(x)-1\right)^2\right] \\ &+ \frac{1}{2}\mathbb{E}_{y\sim Y}\left[\left(D_X(F(y))\right)^2\right]\end{aligned} \tag{3}$$

$$\begin{aligned}\mathcal{L}_{D_{\text{pair}}}(D_{\text{pair}}) =& \frac{1}{2}\mathbb{E}_{x,y\sim X_p,Y_p}\left[\left(D_{\text{pair}}(x,y)-1\right)^2\right] \\ &+\frac{1}{2}\mathcal{L}_{\text{fake}}(D_{\text{pair}})\end{aligned} \tag{4}$$

である．識別器 D_Y に対する敵対性損失 $\mathcal{L}_{D_Y}$ は $\mathcal{L}_{D_X}$ と同様に定義する．ここで，$\mathcal{L}_{\text{fake}}$ は，

$$\begin{aligned}\mathcal{L}_{\text{fake}}(D_{\text{pair}}) = \frac{1}{3}\mathbb{E}_{x,y\sim X_p,Y_p}\Big[\Big\{&\left(D_{\text{pair}}(x,G(x))\right)^2 \\ &+ \left(D_{\text{pair}}(F(y),y)\right)^2 + \left(D_{\text{pair}}(F(y),G(x))\right)^2\Big\}\Big]\end{aligned} \tag{5}$$

であり，識別器 D_{pair} がデータの組 $\{x_p, G(x_p)\}$, $\{F(y_p), y_p\}$, $\{F(y_p), G(x_p)\}$ を生成されたペアであると識別するように促す．真のペア 1 つに対して生成されたペアが 3 通りあるので，式 (4) の第 1 項と第 2 項で均衡を取るために式 (5) では 3 つの項に対する平均を取る．

生成器 G, F に対する敵対性損失 $\mathcal{L}_{\text{adv}}$ は，

$$\begin{aligned}\mathcal{L}_{\text{adv}}(G,F) = \mathcal{L}_{\text{adv}/G}(G) + \mathcal{L}_{\text{adv}/F}(F) \\ + \lambda_{\text{pair/u}}\mathcal{L}_{\text{pair/u}}(G,F) + \lambda_{\text{pair/p}}\mathcal{L}_{\text{pair/p}}(G,F)\end{aligned} \tag{6}$$

である．ここで，$\mathcal{L}_{\text{adv}/G}$ は生成器 G に対する敵対性損失であり，

$$\mathcal{L}_{\text{adv}/G}(G) = \mathbb{E}_{x\sim X}\left[\left(D_Y(G(x))-1\right)^2\right] \tag{7}$$

である．生成器 F に対する敵対性損失 $\mathcal{L}_{\text{adv}/F}$ は $\mathcal{L}_{\text{adv}/G}$ と同様に定義する．また，$\mathcal{L}_{\text{pair/u}}$, $\mathcal{L}_{\text{pair/p}}$ はそれぞれペアをなさない画像，ペアをなす画像が入力であるときの，G と F に対する識別器 D_{pair} との敵対性損失であり，

$$\begin{aligned}\mathcal{L}_{\text{pair/u}}(G,F) =& \mathbb{E}_{x\sim X_u}\left[\left(D_{\text{pair}}(x,G(x))-1\right)^2\right] \\ &+\mathbb{E}_{y\sim Y_u}\left[\left(D_{\text{pair}}(F(y),y)-1\right)^2\right]\end{aligned} \tag{8}$$

で定義する．損失 $\mathcal{L}_{\text{pair/p}}$ は X_u, Y_u をそれぞれ X_p, Y_p として $\mathcal{L}_{\text{pair/u}}$ と同様に定義する．

ここで，D_{pair} はペアをなす画像のみから学習するので，D_{pair} にとってペアをなさない画像は常に未知のデータである．そのため，D_{pair} の識別の精度はペアをなす画像に対してより高くなる．より高い精度の識別による学習の影響を強めるために重み係数 $\lambda_{\text{pair/u}}, \lambda_{\text{pair/p}}$ について，$\lambda_{\text{pair/u}} \leq \lambda_{\text{pair/p}}$ とすることが望ましい．

3.3.2 サイクル一貫性損失

サイクル一貫性損失は CycleGAN [4] で導入された，生成器 G, F が互いに順変換と逆変換の関係になるように促す損失である．ペアをなさない画像についてのサイクル一貫性損失 $\mathcal{L}_{\text{cyc/u}}$ は，

$$\begin{aligned}\mathcal{L}_{\text{cyc/u}}(G,F) =& \mathbb{E}_{x\sim X_u}\left[||F(G(x))-x||_1\right] \\ &+\mathbb{E}_{y\sim Y_u}\left[||G(F(y))-y||_1\right]\end{aligned} \tag{9}$$

である．ペアをなす画像についてのサイクル一貫性損失 $\mathcal{L}_{\mathrm{cyc/p}}$ は X_u, Y_u をそれぞれ X_p, Y_p として $\mathcal{L}_{\mathrm{cyc/u}}$ と同様に定義する．他の損失との均衡のためにそれぞれに対する重み係数 $\lambda_{\mathrm{cyc/u}}, \lambda_{\mathrm{cyc/p}}$ は個別に調整する．

3.3.3 ラベルを用いた L1 損失

図 5 に示すように，同じラベルを有し，ペアをなす画像 $\{x_p, y_p\} \in \{X_p, Y_p\}$ については生成画像 $G(x_p), F(y_p)$ とそれぞれ対応する画像である y_p, x_p を直接比較することができる．生成器 G, F が $\{x_p, y_p\}$ からより直接的に変換前後の対応関係を学習するための損失として，

$$\mathcal{L}_{\mathrm{L1}}(G,F) = \mathbb{E}_{x,y\sim X_p,Y_p}\left[||G(x)-y||_1 + ||F(y)-x||_1\right] \tag{10}$$

で表される L1 損失 $\mathcal{L}_{\mathrm{L1}}$ を導入する．

3.3.4 文字認識器を用いた損失

生成器によって変換された画像が変換前の画像と同じ漢字として読める状態を保つように，その意味的構造を保持するべく，文字認識器を用いた損失 $\mathcal{L}_{\mathrm{con}}$ を導入する．

損失 $\mathcal{L}_{\mathrm{con}}$ を計算するために訓練済み文字認識器 R_X, R_Y を用いる．R_X は活字漢字に，R_Y は手書き漢字にそれぞれ適応した文字認識器である．R_X, R_Y のパラメータは学習によって更新しない．文字認識器 R_X は活字漢字画像 x を入力とし，x に対する認識の予測分布として c_X 次元ベクトル $R_X(x)$ を出力する c_X 値分類器であるとする．同様に R_Y は手書き漢字画像 y を入力として c_Y 次元ベクトル $R_Y(y)$ を出力する c_Y 値分類器であるとする．ここで，

$$\begin{aligned}\mathcal{L}_{\mathrm{con}}(G,F) = \mathbb{E}_{x\sim X}\left[H\Big(l(x), R_Y\big(G(x)\big)\Big)\right] \\ +\mathbb{E}_{y\sim Y_p}\left[H\Big(l(y), R_X\big(F(y)\big)\Big)\right]\end{aligned} \tag{11}$$

と定義する．ただし，$H(p,q)$ は離散確率変数 p, q に対する交差エントロピー誤差関数であり，

$$H(p,q) = -\sum_i p(i)\log(q(i)) \tag{12}$$

と表される．また，$l(x)$ は漢字画像 x のもつラベルに対応する要素について one-hot 表現したベクトルである．損失 $\mathcal{L}_{\mathrm{con}}$ の目的は変換の前後で同じ文字として読める状態を保つことのみである．そこで，変換後の画像に対する予測分布 $R_Y(G(x)), R_X(F(y))$ において，変換前の画像 x, y のもつラベルに対応する要素の値が大きくなるようにのみ促すべく，one-hot 表現 $l(x), l(y)$ を用いる．式 (11) の各項は生成器の出力が，入力画像に付されたラベルの表す漢字と同じ漢字として読めるように促している．文字認識器を用いた損失を与えるためには変換前の画像がラベルを有している必要がある．そのため，式 (11) の第 1 項は全ての $x \in X$ について適用でき，第 2 項はラベルの付与された画像 $y_p \in Y_p$ にのみ適用できる．

(a) SIMHEI (b) HW252

図 6: SIMHEI と HW252 の漢字画像の例.

3.4 モデルの訓練

一般的なデータ拡張として左右反転，上下反転，拡大，縮小等があるが，漢字画像に対しては真の分布に存在し得ないデータを増やすことになる可能性が高い．そのため，提案手法におけるモデルの訓練ではデータ拡張を行わない．

学習のための最適化アルゴリズムには Adam を用いる．Adam のハイパーパラメータは CycleGAN ベースの手法で広く用いられている $\beta_1 = 0.5$，$\beta_2 = 0.999$，$\epsilon = 10^{-8}$ を用いる．学習は 200 エポック行う．学習率は，初めの 100 エポックでは 2×10^{-4} で一定にし，その後は 1 エポック毎に 2×10^{-6} ずつ減少させる．各エポックでは訓練データセットの全てのサンプルについて 1 度ずつ訓練を行う．その際，訓練するサンプルの順序はエポック毎にランダムに変化させる．また，訓練では 1 バッチ毎に $D_X, D_Y, D_{\mathrm{pair}}$ を固定して $\mathcal{L}_{\mathrm{G}}(G,F)$ を計算し，G, F を更新した後，G, F を固定して $\mathcal{L}_{\mathrm{D}}(D_X, D_Y, D_{\mathrm{pair}})$ を計算し，$D_X, D_Y, D_{\mathrm{pair}}$ を更新する．なお，バッチサイズは 1 とする．

4. 評価実験

我々は提案手法の生成精度を評価するための実験を行った．ここでは，実験に用いたデータセット，訓練済み文字認識器，定量的評価指標について述べた後，実験の結果やそれに対する考察を述べる．なお，全ての実験について，目的関数における各損失項の係数は $\lambda_{\mathrm{pair/u}} = 0.5$，$\lambda_{\mathrm{pair/p}} = 1$，$\lambda_{\mathrm{cyc/u}} = 5, \lambda_{\mathrm{cyc/p}} = 1, \lambda_{\mathrm{L1}} = 10, \lambda_{\mathrm{con}} = 0.5, \lambda_{D_{\mathrm{pair}}} = 0.25$ に設定した．また，モデルの実装には Tensorflow を用いた．

4.1 データセット

我々はターゲットフォントとして CASIA-HWDB1.1 [12] の HW252 (1252-f.gnt) を用いた．CASIA-HWDB1.1 は手書き中国語漢字のオフライン単一文字画像が収録されたデータセットである．HW252 は HWDB1.1 に含まれるファイルの一つであり，単一の書き手が書いた，簡体字の符号化文字集合である GB2312-80 のレベル 1 に指定されている漢字 3755 字を含む文字の画像 3926 枚が含まれている．なお，画像はいずれも 128×128 のグレースケール画像である．また，ソースフォントとしては，簡体字フォントとして広く用いられている SIMHEI を用いた．ソースフォント画像はターゲットフォントに合わせて 128×128 のグレースケール画像とした．**図 6** は SIMHEI 及び HW252 の漢字画像の例である．

GB2312-80 のレベル 1 に指定されている漢字 3755 字の

ソースフォント画像及びターゲットフォント画像を本実験で用いるデータセットとした．ランダムに選択した，データセットの 10% にあたる 376 字の画像を訓練データとし，残りの 3379 字の画像をテストデータとした．

なお，手書き漢字認識器によってターゲットフォントの訓練データの一部に自動でラベルを付与する際には，予めラベル付与データの割合 r を定めた．そして，訓練データのうち認識器による認識の確信度の高かった rN 枚の画像についてラベルを付与した．ただし，N はターゲットフォントの訓練データの画像の総数である．

4.2 訓練済み文字認識器

我々は訓練済み手書き漢字認識モデルとして Melnyk-Net [13] を，訓練済み活字漢字認識モデルとして chinese_ocr [14] をそれぞれ用いた．

Melnyk-Net は手書き漢字に適応した漢字認識モデルであり，その訓練済みモデルは GB2312-80 のレベル 1 に指定されている漢字 3755 字に対応している．また，学習データセットには CASIA-HWDB1.0-1.1[12] が用いられている．

chinese_ocr は活字漢字に適応した漢字認識モデルである．しかし，それは単一文字画像を入力としてその予測分布を表す単一のベクトルを出力するものではなく，高さ 32px の文字列画像を入力として幅に応じた個数の予測ベクトルの系列を出力するモデルである．各ベクトルは簡体字，アルファベット等を含む 5990 種の文字についての予測分布である．予測された文字の系列は規則に従ってデコードされることで最終的な予測結果となる．本実験ではこの仕様のために式 (11) における第 2 項を

$$\mathbb{E}_{x,y\sim X_p,Y_p}\left[\frac{1}{M}\sum_{i=1}^{M} H\Big(l\big(R_X(x)_i\big), R_X\big(F(y)\big)_i\Big)\right] \quad (13)$$

として代用した．ここで，M は予測ベクトルの個数，$R_X(x)_i$ は chinese_ocr にソースフォント画像 x を入力して得られる i 番目のベクトル，$l(R_X(x)_i)$ は $R_X(x)_i$ において最大となる要素について one-hot 表現したベクトルである．また，chinese_ocr の訓練済みモデルの学習データセットには中国語コーパスを用いてランダムに生成された 280×32 の文字列画像 364 万枚が用いられている．

なお，いずれの認識器に漢字画像を入力する場合も適切にリサイズ等の前処理を行った．特に chinese_ocr に入力する画像は縦横比を変化させずに 32×32 にリサイズした．

4.3 定量的評価指標

生成結果に対する定量的評価指標として Structural Similarity Index Measure (SSIM) [15] と Style discrepancy を用いる．SSIM は 2 つの画像の類似度を測る指標である．$[0,1]$ の値を取り，値が大きいほど 2 つの画像が類似していることを示す．また，局所的なテクスチャのパターンの違いによる影響を抑制するため，画像にガウシアンブラーをかけてから SSIM を計算した．

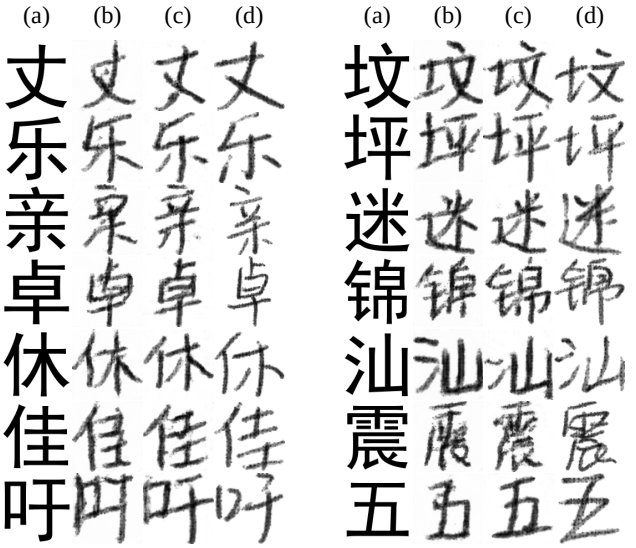

図 7: 既存手法と提案手法の生成結果の例．各行には (a):SIMHEI，(b):既存手法 [1] による生成，(c):提案手法による生成，(d):HW252 の同じ漢字が並んでいる．

Style discrepancy は，生成された文字に対する定量的評価指標として Chang らが提案したものであり，それぞれが共通したスタイルをもつ 2 つの画像集合について，それらのスタイルの相違を測る [1]．なお，値が小さいほど画像集合間のスタイルが類似していることを表す．スタイルを表現する特徴量として，ある畳み込みニューラルネットワークの l 番目の層の出力となる特徴マップのグラム行列を用いる．また，ある画像集合のもつ共通したスタイルを表現する特徴量を，集合の各要素に対するグラム行列の平均とする．さらに，ある 2 つの画像集合間の Style discrepancy をそれぞれの平均グラム行列の二乗平均平方根とする．グラム行列の計算に用いる畳み込みニューラルネットワークの層として，訓練済み手書き漢字認識モデル Melnyk-Net [13] の activation_7 を選択した．ここで，基準値として次の 2 つの Style discrepancy を計算した．一つ目は HW252 の画像集合をランダムに二等分して作った 2 つの画像集合間の Style discrepancy であり，これは 0.32 であった．2 つの集合のもつスタイルはいずれも HW252 であり，一致している．したがって，0.32 は本実験における Style discrepancy の最小値の基準となる．二つ目は SIMHEI の画像集合と HW252 の画像集合の Style discrepancy であり，これは 29.70 であった．SIMHEI は活字漢字であり，HW252 は手書き漢字であるので，この値は本実験における Style discrepancy の最大値の基準となる．

4.4 提案手法による手書き漢字生成の精度

提案手法の生成精度の評価のために，提案手法と Chang らが提案した CycleGAN を用いた既存手法 [1] による生成

表 1: 既存手法と提案手法の定量的評価による比較

	平均 SSIM (↑)	Style discrepancy (↓)
既存手法 [1]	0.352	10.33
提案手法	**0.377**	**7.83**

結果の比較を行った．提案手法におけるラベル付与データの割合は $r = 0.25$ とした．なお，比較のために，既存手法における敵対性損失及び生成器のエンコーダ，デコーダ間の変換ネットワークはそれぞれ提案手法に合わせて LSGAN [11]，ResNet-6 とした．また，既存手法におけるサイクル一貫性損失の重みは $\lambda_{\text{cyc}} = 10$ とした．

図 7 に提案手法と既存手法による生成結果の例を示した．提案手法の方が既存手法と比較して字全体の歪みや傾き，字の細部の形状の両方についてターゲットフォントの画像に近い生成を実現できている．例えば，「卓」について，提案手法による生成にはターゲットフォントのもつ，全体の形状として垂直方向に細長く，また，特に 6，7 画目のように横棒が右肩上がりになる傾向があるという特徴がより明確に現れている．さらに，字画がそれぞれはっきりしているために可読性が高い．他の例としてターゲットフォントの「休」は旁である「木」について，3 画目が 1，2 画目と接していない，左右非対称的であるという特徴を有している．既存手法による生成ではこれらの特徴が確認できない一方で，提案手法による生成ではこれらの特徴が確認できる．また，HW252 の特徴として土偏の 2,3 画目が連続して書かれることで「レ」のようになる傾向が見られる．既存手法による土偏をもつ漢字の生成ではその傾向が現れていない一方で，提案手法による生成ではその傾向が現れている．以上のように，提案手法は既存手法と比較して，特に文字全体についての構造的な特徴や特定の構成要素における書き癖の再現について，定性的に精度の高い生成を実現できていることが確認できた．

また，**表 1** に示すように，提案手法と既存手法のそれぞれの生成結果について，テストデータ全てに対する生成画像とターゲットフォント画像の SSIM の平均値（平均 SSIM）と，ターゲットフォントとの Style discrepancy を求めた．既存手法と比較して，平均 SSIM は提案手法の方が高く，Style discrepancy は提案手法の方が低い．このことから定量的な生成精度の改善が確認できた．

一方で，提案手法でも生成が上手くできなかった例が存在する．「汕」は変換の過程において点が消失した例であり，「震」は字画が密集しているために縺れてしまい，漢字として崩れた例である．文字認識器を用いた損失は変換の前後で同じ文字として読める状態を保つことには寄与する一方で，点の消失や字画の縺れが生じても形状としてユニークである限り，その是正には寄与しない．そこで，改善策としては入力された文字画像がその文字として書き損じのない正確なものであるかを判別するモジュールの導入

図 8: アブレーションスタディにおける生成結果の例．各行には (a):SIMHEI，(b):提案手法から (A) を取り除いた場合の生成，(c):提案手法から (B) を取り除いた場合の生成，(d):提案手法から (C) を取り除いた場合の生成，(e):提案手法による生成，(f):HW252 の同じ漢字が並んでいる．

が挙げられる．また，「五」のようにソースフォントとターゲットフォントにおける形状が大きく異なる漢字については再現が困難であった．このような漢字の再現を実現するためには訓練データを増やすことで，より多様なターゲットフォントの書き癖を学習する必要があるだろう．

4.5 アブレーションスタディ

提案手法で導入した損失及びモジュールが生成精度の改善に寄与していることを確かめるべく，(A) 文字認識器を用いた損失 $\mathcal{L}_{\text{con}}$，(B) 識別器 D_{pair} 及びそれに関する損失，(C) ラベルを用いた L1 損失 $\mathcal{L}_{\text{L1}}$ の 3 つについてのアブレーションスタディを実行し，その生成結果を比較した．**図 8** に (A),(B),(C) をそれぞれ取り除いた場合と提案手法の 4 つについての生成結果の例を示した．なお，ハイパーパラメータは全て同一とし，$r = 0.25$ とした．

(A) を取り除いた場合の生成は，提案手法による生成と比較して漢字として崩れており，可読性が低いものが多い．例えば，「虹」は漢字の崩れが顕著な例である．このことから文字認識器を用いた損失は変換の前後でその意味的構造を保持することを促し，同じ文字として読める状態を保つことに寄与していると考えられる．(B) を取り除いた場合の生成は，提案手法による生成と比較して文字全体の構造的な特徴の再現が上手くなされていないものが多い．例えば，ターゲットフォントの「余」は縦に細長い形状である．また，ターゲットフォントの「巷」は横棒が右肩上がりである，「巳」の矩形の部分が正方形に近いといった特

表 2: アブレーションスタディにおける定量的評価による比較

	平均 SSIM (↑)	Style discrepancy (↓)
(A) を取り除いた場合	0.362	9.18
(B) を取り除いた場合	0.371	**7.50**
(C) を取り除いた場合	0.370	8.33
提案手法	**0.377**	7.83

徴を有している．提案手法による生成の方が (B) を取り除いた場合の生成よりこれらの特徴が顕著に再現されている．このことから識別器 D_{pair} 及びそれに関する損失は，特に文字全体の構造的な特徴の再現に寄与していると考えられる．(C) を取り除いた場合の生成結果は特に細部についての精度が提案手法による生成結果を下回っている．例えば，ターゲットフォントの「珠」は 2 画目と 4 画目が合わさって「レ」のような 1 つの字画になっている．また，ターゲットフォントの「迂」は 1 画目と 3 画目は接していない．(C) を取り除いた場合の生成ではこのような細部の特徴が再現されていないが，提案手法による生成では再現されている．このことからラベルを用いた L1 損失は特に文字の細部の特徴の再現に寄与していると考えられる．以上により，提案手法で導入した損失やモジュールが定性的な生成精度の改善に寄与していると確かめられた．

また，**表 2** に示すように，4 つの場合の生成結果について，それぞれ平均 SSIM とターゲットフォントとの Style discrepancy を求めた．提案手法は平均 SSIM では最も高い値を取り，Style discrepancy では (B) を取り除いた場合に次いで 2 番目に低い値であった．両方の指標において (B) を取り除いた場合と提案手法は僅差であるが，これは 2 つの指標が定性的なスタイルの再現の精度の変化を敏感に反映できていないためであると考えられる．

5. 結論

本研究では，教師なし訓練データの一部に自動で付与したラベルを活用することで，比較的少ない数の教師なしデータを用いながらも，質の高い生成を得ることができる手書き漢字生成の手法を提案した．また，評価実験によって，提案手法を定性的，定量的に評価した．特に文字全体についての構造的な特徴や特定の構成要素における書き癖の再現について，提案手法が定性的に高い精度の生成を実現することを確認した．さらに，アブレーションスタディによって，提案手法で導入した損失やモジュールがそれぞれ生成精度の改善に寄与していることを確認した．

今後解決すべき課題としては，字画の欠損や縺れのある生成が見られることが挙げられる．漢字の書き損じの程度を識別するモジュールによる損失を目的関数に追加することによって，この課題に対する改善が可能であると考える．また，漢字を日常的に用いている人は定量化の難しい細かなスタイルの変化にも気付くことができる．本研究では定量的評価指標として SSIM，Style discrepancy を用いたが，新たな評価指標としてクラウドソーシング等を利用した不特定多数の漢字使用者による評価の導入も検討したい．

謝辞

本研究の一部は (公財) 日本漢字能力検定協会からの研究助成により実施された．

参考文献

[1] Chang, B., Zhang, Q., Pan, S. and Meng, L.: Generating Handwritten Chinese Characters Using CycleGAN, *WACV*, pp. 199–207 (2018).
[2] Isola, P., Zhu, J.-Y., Zhou, T. and Efros, A. A.: Image-to-Image Translation with Conditional Adversarial Networks, *CVPR*, pp. 5967–5976 (2017).
[3] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. and Bengio, Y.: Generative Adversarial Nets, *NeurIPS*, Vol. 27, pp. 2672–2680 (2014).
[4] Zhu, J.-Y., Park, T., Isola, P. and Efros, A. A.: Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks, *ICCV* (2017).
[5] Mondal, A. K., Agarwal, A., Dolz, J. and Desrosiers, C.: Revisiting CycleGAN for semi-supervised segmentation, *arXiv:1908.11569* (2019).
[6] Nguyen, H., Luo, S. and Ramos, F.: Semi-supervised Learning Approach to Generate Neuroimaging Modalities with Adversarial Training, *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pp. 409–421 (2020).
[7] Lian, Z., Zhao, B., Chen, X. and Xiao, J.: EasyFont: a style learning-based system to easily build your large-scale handwriting fonts, *ACM TOG*, Vol. 38, No. 1, pp. 1–18 (2018).
[8] Tian, Y.: zi2zi: Master chinese calligraphy with conditional adversarial networks, `https://github.com/kaonashi-tyc/zi2zi`, last accessed on 2021-2-11 (2017).
[9] Zhou, P., Zhao, Z., Zhang, K., Li, C. and Wang, C.: An end-to-end model for chinese calligraphy generation, *Multimedia Tools and Applications*, Vol. 80, No. 5, pp. 6737–6754 (2021).
[10] He, K., Zhang, X., Ren, S. and Sun, J.: Deep residual learning for image recognition, *CVPR*, pp. 770–778 (2016).
[11] Mao, X., Li, Q., Xie, H., Lau, R. Y., Wang, Z. and Paul Smolley, S.: Least squares generative adversarial networks, *ICCV*, pp. 2794–2802 (2017).
[12] Liu, C.-L., Yin, F., Wang, D.-H. and Wang, Q.-F.: CASIA online and offline Chinese handwriting databases, *International Conference on Document Analysis and Recognition (ICDAR)*, IEEE, pp. 37–41 (2011).
[13] Melnyk, P., You, Z. and Li, K.: A high-performance CNN method for offline handwritten Chinese character recognition and visualization, *Soft Computing*, Vol. 24, pp. 7977–7987 (2020).
[14] Yang, C.: chinese_ocr, `https://github.com/YCG09/chinese\_ocr`, last accessed on 2021-2-11 (2018).
[15] Wang, Z., Bovik, A. C., Sheikh, H. R. and Simoncelli, E. P.: Image quality assessment: from error visibility to structural similarity, *IEEE TIP*, Vol. 13, No. 4, pp. 600–612 (2004).

ⓒ 2021 Information Processing Society of Japan