スタイル転移によるフォント画像変換

成沢　淳史[1,†1,a)]　柳井　啓司[1,b)]

概要：本研究では文字のフォントの自動生成, 自動変換のタスクに取り組んでいる. 従来のフォント生成のタスクでは文字をいくつかのストロークから成り立つものとしモデル化を行い作成する手法が取られてきた. それに対して, 本研究では深層学習により, 画像中のフォント画像ないしパターン画像からストロークに相当する特徴を自動で抽出し, 変換元のフォントから任意のデザインパターンへの変換に挑戦している. この仕組みにより手書き文字のような個人ごとのオリジナルフォントの作成が用意にできるようになる. 実験ではケチャップ文字を始めとしたユニークな質感パターン画像セットを作成し, 深層学習のクロスドメイン学習による手法と Neural Style Transfer の手法とを組み合わせ, 生成結果の可読性を改善した.

1. はじめに

文字にまつわる分野において深層学習のおかげで様々な研究タスクを考えられるようになっている. 情景文字認識のほか, 古文書の解析や画像中のテキストを使ったイメージキャプションができるようになった. 近年, コンピュータビジョン分野において画像生成タスクが盛り上がりを見せており, フォントの形状変換や新しいフォントの生成を実現するための応用が始まりつつある. フォント生成は中国語や日本語のように文字種が数千を超えるような言語でのフォント作成におけるコスト削減の点で有益な研究である. すでにフォントからフォントへの変換は文字の分野で重要なタスクとして様々な研究者が深層学習の手法を用いて試みている. 本研究ではフォントからフォントへの変換タスクをさらに発展させ, 身の回りに存在するパターンや共通のデザインが統一的に使われた画像からユニークな特徴をフォント画像に対して転写するタスクに挑戦する. こうした流れは，従来の文字を認識することが最終目標であった文字認識研究の枠を超えた新しい動きであり，文字にまつわる様々な研究ということで文字工学と呼称されている.

2. 実験目的

本論文では画像を扱う深層学習に基づいた文字画像の形状変換ないし生成によるフォント生成の最適な手法の探求を目的として行う.

日本語や中国語は数千におよぶ文字の種類があり, フォントの製作にはコストと時間が必要となる. したがって, 画像生成技術により数種類のデザインパターンから日本語フォントを自動生成することが期待されている. しかしながら, 深層学習を文字画像に利用する場合には生成結果の可読性の向上や少数サンプルからの学習といった課題が残っている.

深層学習による画像生成タスクの困難な点としては, 深層学習固有の問題として多様かつサンプル数の多いデータセットが必要となる点が挙げられる. また, 現状ではフォント間での変換には同じ対応する文字から変換パターンを学習するためデータセットの用意が難しく, フォント間以外への応用が困難である. そこで, 本研究では以下の取り組みを行った.

(1) Neural Style Transfer の導入による生成結果の可読性の改善
(2) 質感パターンセットを作成し, フォントに対する質感の転写
(3) 入力画像の工夫による少数サンプルからの学習

3. 関連研究

3.1 従来手法によるフォント画像生成

フォントにはベクトルフォントとビットマップフォントがある. 一般的なベクトルフォント生成では文字を直線やカーブ, ハネなどのストロークの集合または部首のようなまとまったコンポーネントの組み合わせから成ると考え, 文字をストロークにまで分解し, 対応する変換先のストロークを割り当てる手法 [1], [2] やストロークやコンポーネントの組み合わせから漢字を作り出す手法 [3], [4] が提案されている.

[1] 電気通信大学 総合情報学科
[†1] 現在，同大学院 情報理工学研究科 情報学専攻所属
[a)] narusawa-a@mm.inf.uec.ac.jp
[b)] yanai@cs.uec.ac.jp

表 1　質感画像セット概要

画像セット名	サンプル数
ケチャップ文字	445
砂文字	483
紐文字	796

そのため, 現状では予めフォントのベクトル情報を利用してスケルトンを作成することでストロークへの分割が容易になり, 漢字を構成する上で必要なストロークを特定, 抽出しフォントの自動生成を行う. 漢字を構成するために必要なストローク成分は研究されているが, 漢字や記号毎にストロークの対応関係を記述するのは大変な労力となる. そこで深層学習の手法を用いることでフォント画像から自動でストロークを抽出し, 変換対応をネットワークで学習させることで自動化を行うことができるようになる.

また, これまでの研究ではフォントの生成に絞った研究が多く見られたが, 近年はフォントの自動生成よりもタイポグラフィーを含めて飾り文字のようなアーティスティックな文字画像の生成 [5] に注目が集まる中で文字画像に対するテクスチャのような質感の付与のほか, 文字以外のパターン画像に対する手法の適用が期待されている. 本研究では深層学習の手法によりテクスチャのような質感の付与や文字画像からストローク間での対応関係を変換ネットワークで学習させることに挑戦する.

3.2　深層学習によるフォント生成

深層学習を用いた文字画像生成タスクの多くは中国圏の研究成果が多く著名な成果に Rewrite[*1], Zi2Zi[*2]プロジェクトが存在する. Rewrite は Neural Style Transfer [6] をフォント画像の生成に適した変更を加えたプロジェクトである. Neural Style Transfer は二枚の画像を合成する手法であり, Rewrite の他にもフォントでの応用結果 [7] が報告されている.

さらに敵対学習の手法を利用した Zi2Zi がある. これは Pix2Pix [8] をベースに拡張を施したプロジェクトであり, 画像から画像への変換を目的とした仕組みである. 画像を特徴表現するためのエンコーダ, 特徴表現から画像に復元するデコーダの変換ネットワークが使われており変換ネットワークの学習では変換元と変換先で同じ対応関係のペアを用意し学習を行う.

本研究ではこれらの研究をさらに発展させ, ペア画像無しでのクロスドメイン学習, フォント間以外の質感パターン画像セットからの学習に応用した.

4.　手法概要

図 1　質感パターン画像セット (左からケチャップ文字, 砂文字, 紐文字)

本研究ではフォント画像からケチャップ文字, 砂文字, 紐文字の 3 種類の質感パターン画像セット (図 1) への変換実験を行う. 本研究では Fast Style Transfer (図 2)　[9], [10] を参考に敵対学習のひとつであるクロスドメイン学習による手法 [11], [12], [13], [14] を組み合わせた, ネットワーク (図 3) を考案した. クロスドメイン学習は変換ネットワーク部分 G, F (図 4) と Discriminator 部分 D_x, D_y (図 5) から成り, 順変換と逆変換を行うため, それぞれ 2 つずつ独立してネットワークが準備される.

変換ネットワークは入力画像の形状を保つため, 逆変換を行った画像と Element-Wise での比較による Cycle Loss (L_{cycle}) に加え, Adversarial Loss ($L_{adversarial}$) が Disctiminator によりフィードバックされ, 敵対学習を進めることで入力の形状に対してターゲットのデザイン, スタイルを転写することができる.

本研究ではクロスドメイン学習の生成結果における可読性の改善のため, Style Transfer の Content Loss ($L_{content}$) に注目し, また、新しいデザインの獲得に繋がる結果が期待される Style Loss (L_{style}) を導入している. Style Loss は式 1 の通り, スタイル画像 (x_s) と 生成画像 (x_g) のあるレイヤー (l) での各チャンネルのアクティベーション ($F_l(x)$) から求まるグラムマトリクス (式 2) の差で定義される.

$$L_{style} = \sum ||G_{l,i,j}(x_s) - G_{l,i,j}(x_g)||^2 \qquad (1)$$

$$G_l(x) = F_l(x)F_l^T(x) \qquad (2)$$

Style Transfer を導入するに辺り, 特徴抽出には VGG 16 のプレトレインモデルから行い, 図 6 に示す通り, スタイルレイヤーは $Relu\_1\_1, Relu\_2\_1, Relu\_3\_1, Relu\_4\_1, Relu\_5\_1$, コンテンツレイヤーは $Conv\_4\_1$ から特徴を抽出する. 提案手法ではこれら 4 つのロスを統合しており, 変換ネットワーク G, F は式 3 のロスから学習される. それぞれの重

[*1] https://github.com/kaonashi-tyc/Rewrite
[*2] https://kaonashi-tyc.github.io/2017/04/06/zi2zi.html

表 2　ロスの重みの設定

Style Weight	3.00e5
Content Weight	1.50e0
Adversarial Weight	4.50e8
Cycle Weight	3.50e12

み $\alpha, \beta, \gamma, \delta$ は実験的に決めており, 最終的な重みの設定は表 2 の通りとなっている.

$$L_{total} = \alpha L_{style} + \beta L_{content} + \gamma L_{adversarial} + \delta L_{cycle} \quad (3)$$

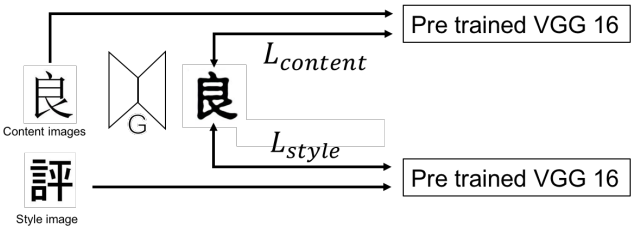

図 2　Fast Style Transfer 概要

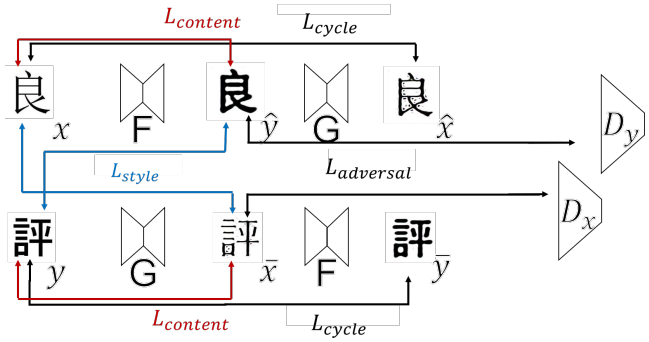

図 3　CycleGAN with Neural Style 概要 Adversarial Loss, Cycle Loss に加え, Style Loss, Content Loss 導入する.

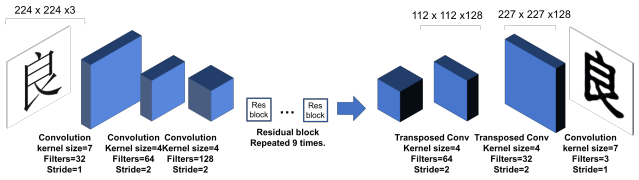

図 4　変換ネットワークの詳細

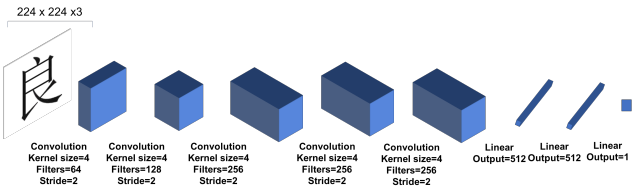

図 5　Discriminator ネットワークの詳細

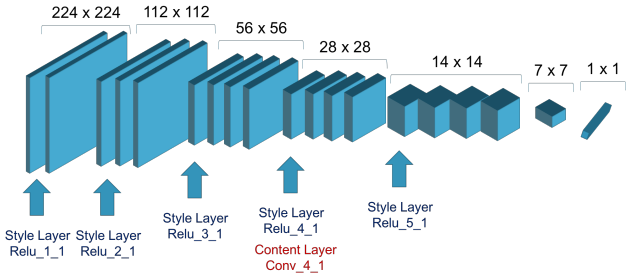

図 6　スタイルレイヤーとコンテンツレイヤーの選択

5. 実験

5.1 画像と実験環境

ケチャップ文字画像セットは文字の画像が多く, 砂文字には英字と簡単なひらがなに加えて絵の一部から構築している. また, 紐文字に至っては文字が含まれておらず、紐で作られたアート画像からランダムで切り出しを行い手作業で整形を行った. さらに入力画像に関して, 画像中から十分なスタイル情報を得るために単一画像中に 16 文字を画像セットから選び, 配置を行い 500 枚として学習した. この入力の工夫はスタイル情報の獲得を行う目的だけでなく, 図 7 の例のようにノイズが減り綺麗な文字の画像を生成することができるようになる.

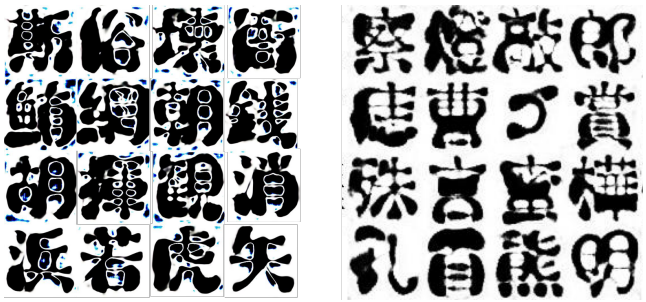

図 7　単一文字画像と複数文字画像の比較 (単一文字画像 (左) 複数文字画像 (右))

5.2 単一文字画像での変換例

まず, はじめにクロスドメイン学習の性能を確認するために単一画像中に一文字を配置して生成テストを行った. 質感パターン画像セットでは図 8 のように成功例もみられるが, 図 9 のように可読性が失われた変換失敗例もみられる.

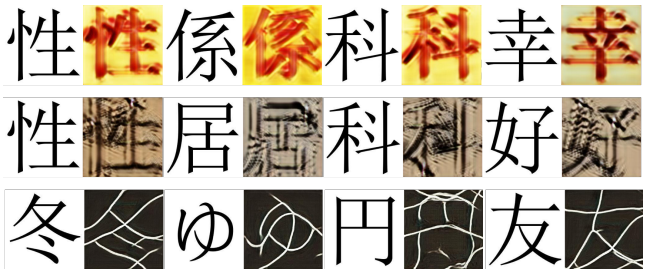

図 8　単一文字画像でのクロスドメイン学習による画像生成成功例 (上からケチャップ文字, 砂文字, 紐文字への変換例)

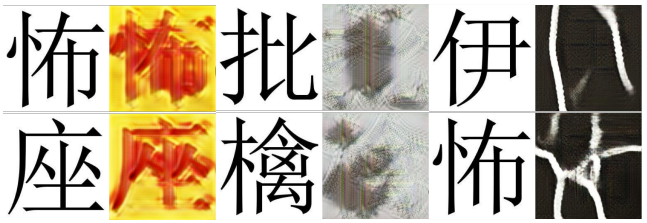

図 9　単一文字画像でのクロスドメイン学習の失敗例

5.3　生成例

Neural Style, Cycle GAN, 提案手法での生成例を図 10 に掲載する．フォントからそれぞれの質感画像セットへの変換が目的となるため, Cycle GAN では順方向への変換結果のみ掲載する．また, 表 2 とは異なり, Content Loss の重みが大きく設定されている．Cycle GAN と提案手法において質感パターン画像への変換では文字の形状がはっきりしなかった生成画像がコンテンツロスの導入により改善される結果が得られた．特に紐文字の Neural Style (Style Loss + Content Loss) での生成はスタイルに重みをおいた場合に可読性を保持できないが, 提案手法では可能となっている．また, Style Loss と Content Loss はトレードオフの関係にあるが, Adversarial Loss, Cycle Loss が可読性や質感の転写を補う結果も得られた．

図 10　生成結果 (Neural Style, Cycle GAN, 提案手法)

5.4　各ロスの重みの調整方法

4 つのロスの重みを最適にするために, 検証を行った．まずは, Content Loss と Style Loss の組み合わせから調整をはじめ, Content Loss を固定した状態で Style Loss の重みを変化させた．図 11 は SimSun フォントから Gothic 体への変換結果である．本実験を行う前にフォント間の変換にて Style の重みを 3.0e5 に決定した．Style Loss はこの値で固定し, 図 12 に示すよう Adversarial Loss, Cycle Loss を調整した．また, 図 12 の 2 行目以降は Adversarial Loss を除外し, Style Loss, Content Loss, Cycle Loss の調整を行った結果である．Content Loss は 1.5e0 にてスタイル特徴の影響が出始め, Cycle Loss の重みを上げ, 3.5e5 とすることで更に自然なスタイルの獲得に繋がった．Style Loss + Cycle Loss + Content Loss の組み合わせにさらに Adversal Loss を加えることで図 12 の最終行のようにさらに自然なデザインに近くようになる．

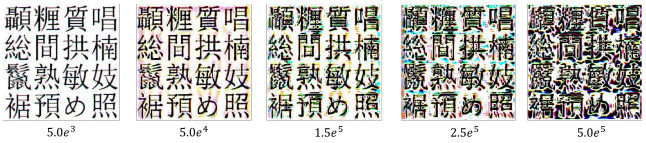

図 11　Style Weight による生成結果の変化

図 12　様々な重みを変えた際の変化

5.5 主観評価と客観評価

Style Loss と Content Loss の導入の効果を確認するため, 様々なロスの組み合わせを検証した. 図 13 では Content Loss の有無により, 可読性が保持されていることが確認できるほか, Style + Cycle + Content の組み合わせに Adversarial Loss を加えると生成画像のテクスチャが自然になることが視覚的にわかる.

また, 対象のパターン画像のスタイルをネットワークが獲得できているかを Style Loss の平均 (スタイル平均) により比較を行った. それぞれ画像セットから 1 枚のスタイル画像を選び, 生成した検証画像セットの Style Loss を全て計り, 表 3, 表 4, 表 5 にまとめる. Neural Style を使った場合のスタイル平均が最も小さい. 本研究が目指すところは, この値と Cycle GAN をベースラインとして中間値をターゲットとしている. ケチャップ文字では Style Loss + Cycle Loss + Content Loss の組み合わせで達成しており, Adversarial Loss を加えた場合には Cycle GAN より微減の結果となった. 全ての組み合わせにおいて視覚的には良い結果が得られている.

一方, 砂文字では Neural Style と Cycle GAN でのスタイル平均に大差はなく, Style Loss を組み合わせたパターンは図 13 にみられるように, 背景が自然なテクスチャで生成されない一方で, Adversarial Loss を追加することで背景が自然になり, 表 4 ではスタイル平均の減少の結果が得られた. 結果の表 4 より, Style がうまく獲得できない場合に Adversarial Loss や Cycle Loss を導入することで Style の獲得に繋がる傾向がみられる.

最後に紐文字では, Content Loss の有無により可読性が大きく異なる結果が得られた. また, 砂文字と同様に Style + Content + Cycle における Adversarial Loss の有無は背景のテクスチャに影響を与える結果が観察できるほか, この実験では表 5 にみられるように Style Loss の有無がスタイル平均に大きく影響する結果となった.

表 3 スタイルロス平均の比較 (ケチャップ文字)

	平均 (e6)	分散 (e10)
Style + Content (Neural Style)	5.17	64.8
Adversarial + Cycle (Cycle GAN)	6.03	26.7
Style + Cycle + Content	5.66	5.65
Adversarial + Style + Cycle	5.98	5.00
Adversarial + Style + Cycle + Content	6.00	5.28

表 4 スタイルロス平均の比較 (砂文字)

	平均 (e6)	分散 (e10)
Style + Content (Neural Style)	2.77	5.87
Adversarial + Cycle (Cycle GAN)	3.11	14.2
Style + Cycle + Content	3.36	2.80
Adversarial + Style + Cycle	2.69	4.17
Adversarial + Style + Cycle + Content	2.71	4.59

表 5 スタイルロス平均の比較 (紐文字)

	平均 (e7)	分散 (e11)
Style + Content (Neural Style)	2.03	31.3
Adversarial + Cycle (Cycle GAN)	2.74	82.9
Style + Cycle + Content	2.17	14.3
Adversarial + Style + Cycle	2.06	3.74
Adversarial + Style + Cycle + Content	1.99	10.8

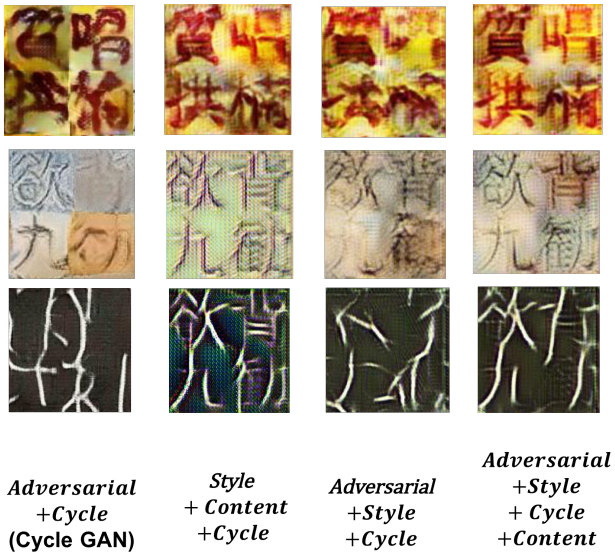

図 13 ロスの組み合わせの比較

6. 考察

本提案手法では Adversarial Loss と Style Loss, Cycle Loss と Content Loss が競合する形になっている. しかしながら, 3 つの画像セットを試したところ, それぞれでうまく機能する組み合わせとそうでないケースがあった. 共通して Content Loss は入力形状の保持に働くが, Content + Style ではスタイル特徴をうまく合成できないケースがあった. この場合には, Cycle Loss を加えることで改善するケースがあったが, 考えられる要因として直線的なデザインの多い人工のフォント画像と輪郭線に歪みのあるパターン画像とで Gram matrix を通して共起関係を得られ

たかの差であると考えられる. Cycle Loss では 2 つの変換ネットワークを通すため, 直線部分に歪みが生じることで共起関係が得やすいことが予想される. Content Loss の重みを固定した状態で Cycle Loss を変化させるとスタイルの特徴が大きく混ざった視覚的に意味のある結果 (図 12) も実験より得られている.

また, Style Loss と Adversal Loss は性質が異なるものであると予想のもとから導入を行い実験を行った. 実際には Adversal Loss が Style loss を改善するケースは砂文字画像セットでのみ結果が得られた. ケチャップ文字, 紐文字における提案手法ではスタイル平均が Neural Style 以上, Cycle GAN 以下となっている. Adversal Loss の重みを大きくすると背景のテクスチャの結果が改善する結果が砂文字より得られており, 同等の効果は Content Loss により得られると思われる. Discriminator はテクスチャ, 形状などのディテールに対する制約なので, 自然な画像生成に繋がると考えられ, Content Loss は前景と背景に対する制約と考えられるが, 現状 Adversal Loss の方が自然な画像生成に繋がる印象を得ている.

様々なネットワーク構造で実験を行ったところ, Adversal Loss + Content Loss + Cycle Loss ないし, これに Adversarial Loss を加えた場合の組み合わせが最も良く, 特に Cycle GAN に Content Loss を導入した場合には可読性が高い可能性で改善すると考えている.

7. 今後の課題

生成結果の評価を行うためには質感 (デザイン) と可読性の二つの指標で測る必要がある. 実験では質感の転写がなされているのかを Style Loss を使って比較を行った. 今後, この Style Loss による問題点があるのかを議論したいと考えている. また, 文字の可読性については今回測ることができなかった. 有力な手段として万能な文字分類器を作成し, 文字の分類率により比較を行うのが最も良いと思われる.

今回の手法はパッチレベルでの変換対応がうまくいかない場合への対処として裏付けがなされていない. 文字は弾性を持っているため, 形状が大きく変わった場合でも可読性が変わらない場合がある. スタイルに適した形状まで予め入力画像を摂動させることができれば, 変換対応の学習に繋がり, ストロークの消失に対するアプローチになる可能性がある.

謝辞: 本研究に対しご助言，ご議論をして下さった九州大学内田誠一先生に感謝いたします．また本研究は JSPS 科研費 17H06100「機械可読時代における文字科学の創成と応用展開」の助成を受けたものです.

参考文献

[1] A. Zong and Y. Zhu. Strokebank: Automating personalized chinese handwriting generation. In *Advancement of Artificial Intelligence*, pp. 3024–3030, 2014.

[2] T. Miyazaki, T. Tsuchiya, Y. Sugaya, S. Omachi, M. Iwamura, S. Uchida, and K. Kise. Automatic generation of typographic font from a small font subset. In *arXiv preprint arXiv:1701.05703*, 2017.

[3] J. Lin, C. Hong, R. Chang, Y. Wang, S. Lin, and J. Ho. Complete font generation of chinese characters in personal handwriting style. In *International Performance Computing and Communications Conference (IPCCC)*. IEEE, 2015.

[4] X. Songhua, L. Francis C.M., C. Kwok-Wai, and P. Yunhe. Automatic generation of artistic chinese calligraphy. In *IEEE Intelligent Systems*, 2005.

[5] Y. Shuai, L. Jiaying, L. Zhouhui, and G. Zongming. Awesome typography: Statistics-based text effects transfer. In *Proc. of IEEE Computer Vision and Pattern Recognition*, 2016.

[6] L. A. Gatys, A. S. Ecker, and M. Bethge. Image style transfer using convolutional neural networks. In *Proc. of IEEE Computer Vision and Pattern Recognition*, 2016.

[7] A. Gantugs, B. K. Iwana, A. Narusawa, K. Yanai, and S. Uchida. Neural font style transfer. In *International Conference on Document Analysis and Recognition*, 2017.

[8] I. Phillip, Z. Jun-Yan, Z. Tinghui, and E. Alexei. Image-to-image translation with conditional adversarial networks. In *Proc. of IEEE Computer Vision and Pattern Recognition*, 2017.

[9] J. Johnson, A. Alahi, and L. Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *Proc.of European Conference on Computer Vision*, 2016.

[10] D. Ulyanov, V. Lebedev, A. Vedaldi, and V. Lempitsky. Texture networks: Feed-forward synthesis of textures and stylized images. In *arXiv:1603.03417v1*, 2016.

[11] Z. Jun-Yan, P. Taesung, I. Phillip, and E. Alexei. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proc.of IEEE International Conference on Computer Vision*, 2017.

[12] Z. Yi, H. Zhang, P. Tan, and M. Gong. Dualgan: Unsupervised dual learning for image-to-image translation. In *Proc.of IEEE International Conference on Computer Vision*, pp. 2849–2857, 2017.

[13] T. Kim, M. Cha, M. Kim, J. K. Lee, and J. Kim. Learning to discover cross-domain relations with generative adversarial networks. In *International Conference on Machine Learning*, 2017.

[14] Y. Taigman, A. Polyak, and L. Wolf. Unsupervised cross-domain image generation. In *International Conference on Learning Representation*, 2017.