Vol.2018-MPS-119 No.10
2018/7/30

近代書籍のための自動フォント生成

竹本有紀[†1]　石川由羽[†1]　髙田雅美[†1]　城和貴[†1]

概要：国立国会図書館は，貴重な資料を Web 上で一般公開している．明治から昭和初期にかけて刊行された書籍である近代書籍も，画像データとして閲覧可能である．利便性向上のためには画像データをテキストデータに変換する必要があるが，近代書籍の文字画像は，一般的な OCR ソフトウェアで正確に文字認識を行うことが困難である．そこで，近代書籍の文字認識に特化した多フォント活字認識手法が提案されている．多フォント活字認識手法による正確な文字認識には，学習データとして膨大な近代書籍の文字画像が必要である．しかしながら，現在の方法では文字画像の収集に限界がある．本稿では，Deep Learning を用いて文字画像を自動生成することで，多フォント活字認識手法の学習データの補完を目指す方法を提案する．そのために，近代書籍と類似した特徴を持つ文字画像を自動生成するニューラルネットワークを構築する．

キーワード：フォント生成，Deep Learning，ニューラルネットワーク，近代書籍

1. はじめに

古い時代の書物は，当時の歴史や文化，思想などを知るための有用な手段の一つである．明治から昭和初期に刊行された書籍は近代書籍と呼ばれる．近代書籍の多くは絶版となっており，入手が困難で貴重な資料である．国立国会図書館[1]は，近代書籍の画像データを Web 上で一般公開している[2]．これにより紙の書籍のような，破損や紛失，盗難の心配なく，誰でも手軽に近代書籍を閲覧できる．しかしながら，画像データでは文書内容に対して用語の検索ができない．資料の利便性向上のためには，文書内容の早急なテキスト化が求められる．

画像データをテキストデータに変換する方法の一つに，光学文字認識（Optical Character Recognition）がある．ところが，近代書籍は市販のソフトウェアでは正確な認識が難しい．そこで，近代書籍の文字認識に特化した多フォント活字認識手法が提案されている[3][4][5]．この手法には，学習データとして，膨大な数の近代書籍の文字画像が必要である．現在は，近代書籍の画像データから文字画像を手動で切り出し，学習データを収集している．しかしながら，手動による収集には限界がある．よって，文字認識の精度向上のためには，手動による文字画像の切り出し以外の方法で，学習データを効率的に増やす方法が必要である．

本稿では，Deep Learning を用いて文字画像を生成し，多フォント活字認識手法で用いる学習データの補完を目指す．生成する文字画像は，近代書籍の特定の出版者・出版年代のフォントの特徴を持つ文字画像である．入手が容易な文字画像を入力すると，近代書籍の特定の出版者・出版年代のフォントの特徴を持つ文字画像を自動生成するニューラルネットワークの構築を目標とする．

2. Deep Learning を用いたフォントの自動生成手法

2.1 フォントの自動生成アルゴリズム

本稿では，Deep Learning によってフォント固有の特徴変換を目指す．Deep Learning を用いて特定の出版者・出版年代のフォントの文字画像を自動生成する手法の手順は以下の通りである．

1. データセットの読み込み
2. フォントの特徴の学習
   a. 文字画像の生成
   b. 誤差の算出
   c. 誤差の逆伝搬
   d. パラメータの更新
   e. 精度の確認
   f. b.に戻る
3. フォントの自動生成
   a. 生成の元となる文字画像の読み込み
   b. 文字画像の生成

手順 1 でデータセットの文字画像を読み込み，データセットを訓練データとテストデータに分ける．訓練データはニューラルネットワーク（これ以降、NN と呼ぶ）の学習にために用いられる．テストデータは NN の未知データに対する再現精度を確認するために用いられる．次に手順 2 で NN の学習を行う．手順 2-a では，手順 1 で読み込んだ訓練データの文字画像を用いてフォントを変換した文字画像を生成する．手順 2-b では，手順 2-a で自動生成した文字画像と，近代書籍の文字画像の誤差を算出する．手順 2-c では，手順 2-b で算出した誤差を NN に逆伝搬し，手順 2-d でパラメータの更新を行う．手順 2-e では，テストデータを用いて NN の誤差を確認する．手順 2-a から 2-e の工程を所定回数繰り返したら，学習を終了する．学習終了後，手順 3 でテストデータに対する精度が最も高かった NN を用いてフォントの自動生成を行う．手順 3-a でフォント生成の元となる文字画像を読み込み，手順 3-b でフォントを変換した文字画像を生成する．それぞれの工程の詳細を次の 2.2 節から 2.4 節で述べる．

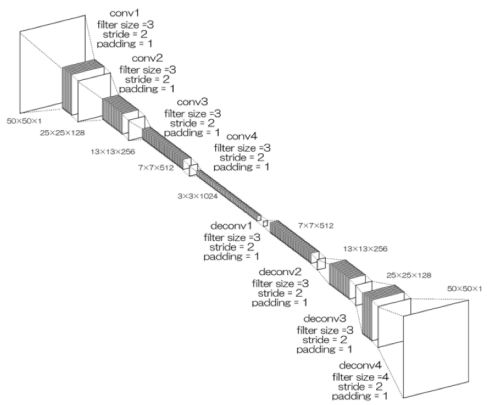

Fig. 1 フォントを自動生成する NN

2.2 データセットの読み込み

本稿で NN の学習に用いるデータセットは，2 枚 1 組の文字画像である．データセットを構成する文字画像は，フォント生成の元となる文字画像と近代書籍の特定の出版者・出版年代のフォントの文字画像である．フォント生成の元となる文字画像に用いるフォントは，3 つの条件を満たす必要がある．その条件とは，入手が容易であること，収録している文字の種類が豊富であること，異字体や旧字体も多く収録されていることである．本稿では，これらの条件を満たすフォントとしてゴシック体を用いる．

多フォント活字認識手法で扱う画像は bmp 形式であるため，用意する文字画像と NN が生成する文字画像も bmp 形式とする．読み込まれた文字画像を 2 値化し，画像サイズを縦横 50 ピクセルに統一する. データセットは訓練データとテストデータに分割する．訓練データを用いて NN を学習し，テストデータを用いて NN の精度を確認する．

2.3 フォントの特徴の学習

ゴシック体の文字画像と構築した NN を用いて，近代書籍の特定フォントの文字画像の自動生成を行う．NN は，入力された文字画像のフォントを学習したフォントに変換した文字画像を生成する．本稿で構築した NN を Fig. 1 に示す．NN の中間層は，それぞれ 4 層の畳み込み層と逆畳み込み層によって構成されている．畳み込み層と逆畳み込み層では，stride を 2 にすることでそれぞれプーリング層，アンプーリング層の役割を担っている．出力される文字画像の大きさを入力画像と統一するため, 全ての層の padding を 1，最後の逆畳み込み層のフィルタサイズを 4 に設定した．全ての畳み込み層と 3 つの逆畳み込み層では，フィルタサイズを 3 とする．活性化関数には，ReLU 関数を用いる. 4 つの畳み込み層には, dropout 関数を適用する. dropout する割合は，学習の様子を見て調整する必要がある．

学習データに用いた近代書籍のフォントにより近い文字画像を生成するため，自動生成された文字画像と近代書籍の文字画像の誤差を算出する．この誤差を NN の出力層から入力層へ逆伝搬し，誤差がより小さくなるようにパラメータを更新して学習していく．本稿で算出する誤差は，画素値の平均二乗誤差（Root Mean Squared Error）である．

逆伝搬された誤差から，学習の最適化手法を用いて新たなパラメータを求める．本稿では，Adam[6]を用いる．学習の最適化手法には様々なものがあるが，Adam はメモリ消費が少なく，学習の収束も早い手法である．

学習を繰り返すことで，様々なパラメータを持つ NN を得る．フォントの自動生成に最も適しているのは，未知データに対する再現精度が最も高いモデルである．学習終了後に最も適したモデルを選ぶことを目的として，学習のたびにテストデータに対する再現精度を確認する．そのために，テストデータから自動生成された文字画像と近代書籍の文字画像の誤差を算出する．求める誤差は 2 枚の文字画像の画素値の平均二乗誤差である．この値が最も小さくなるモデルが，フォントの自動生成に最も適している．

2.4 フォントの自動生成

学習の終了後, NN を用いてフォントの自動生成が可能となる．ここで用いるのは，テストデータに対する再現精度が最も高いパラメータを持つ NN のモデルである．この NN に任意の文字のゴシック体の文字画像を入力すると，学習に用いた近代書籍の特定フォントの特徴を持つ文字画像が自動生成できる．これにより，多フォント活字認識手法に必要な学習データを自動生成された文字画像によって補完できる．学習データの補完によって，多フォント活字認識手法の精度向上が期待される．

3. フォントの自動生成実験

3.1 実験方法

提案手法の有用性を検証するため，フォントの自動生成実験を行う．実験で学習するのは，明治中期に駸々堂で出版された書籍の印刷に用いられたフォントである．これ以降，このフォントを駸々堂明治中期フォントと呼ぶ．学習に用いるデータセットは，ゴシック体と駸々堂明治中期フォントの文字画像である. 1 つの文字に対して，それぞれ 1 枚，計 2 枚の文字画像を 1 組とするデータセットを 1297 組用意した．これらのデータセットのうち，ランダムに選んだ 1200 組を訓練データ，残りの 97 組をテストデータとする．学習回数を 5000 回，dropout の割合を 0.7 として NN を学習する．

学習終了後，自動生成された文字画像と駸々堂明治中期フォントの文字画像を比較して，学習データに利用可能であるか否かの判定を行う. 比較対象は, PDC 特徴[7]のユークリッド距離である．PDC 特徴は，外郭方向寄与度特徴とも呼ばれ，手書き文字認識に利用される特徴量である．比較に PDC 特徴を用いるのは, 多フォント活字認識手法では文字画像の特徴量として，PDC 特徴を用いるためである．多フォント活字認識における PDC 特徴は，1536 次元のベクトルである．近代書籍の文字画像は，印刷の度に滲みや

Table. 1　訓練データのフォント生成結果

ゴシック体の文字画像	駸々堂中期フォントの文字画像	自動生成された文字画像
遙		
光		
禮		

Table. 2　テストデータのフォント生成結果

ゴシック体の文字画像	駸々堂中期フォントの文字画像	自動生成された文字画像
紀		
公		
嬉		

擦れの影響が異なる．そのため，近代書籍の文字画像から抽出した PDC 特徴は, 同一文字であっても印刷された文字ごとに差異が生じる．よって，PDC 特徴のベクトル空間において，同一フォントで印刷された同一文字は，PDC 特徴のベクトルが完全には一致しないが，ある程度の誤差の範囲に集中して分布する．その誤差の範囲内に自動生成された文字画像から抽出した PDC 特徴が存在すれば, 自動生成された文字画像は近代書籍の文字画像と類似した PDC 特徴を持ち，学習データとして利用可能であると言える．

3.2 実験結果

実験において，テストデータの再現精度が最も高かったモデルのフォントの生成結果を示す．Table. 1 は，ゴシック体，駸々堂中期フォントの文字画像と，訓練データから自動生成された文字画像の例である．Table. 2 は，ゴシック体，駸々堂中期フォントの文字画像と，テストデータから自動生成された文字画像の例である．訓練データから自動生成された文字画像は，駸々堂中期フォントの文字画像をほぼ完全に再現できることが分かる．テストデータから自動生成された文字画像は, ノイズが含まれているものの, 変換の元となったゴシック体の文字画像の特徴が，駸々堂中期フォントの特徴に変換されていることが分かる．ピクセルごとの画素値を比較した結果，訓練データにおける画素値の平均一致率は 99.79%, テストデータにおける画素値の平均一致率は 73.69%であった．訓練データに比べてテストデータの場合の平均一致率が低い原因として，訓練データにない未知の特徴を正確に再現できなかった点が考えられる．しかしながら，近代書籍から切り出される文字画像は，インクの滲みや擦れ，撮影時のページのたわみなどの影響を受けている．そのため，同じ漢字であっても，切り出される文字画像ごとに，線の太さや形状の歪みの程度は異なる．よって，自動生成された文字画像が駸々堂中期フォントの特徴を再現できていたとしても，画素値の一致率が下がる可能性も考えられる．

Table. 3　PDC 特徴を比較する文字画像の一覧

自動生成された文字画像	駸々堂中期フォントの文字画像の例
	
	
	

自動生成された文字画像が学習データに利用可能であるか否かを判定するために，PDC 特徴の比較を行う．テストデータから自動生成された文字画像のうち，6 種類の文字画像を選び, 駸々堂明治中期フォントの文字画像と PDC 特徴を比較した．6 種類の文字は，近代書籍から切り出された文字画像の多い文字の中から，画数や文字の形が偏らないように選ぶ．選んだ 6 種類の文字について，自動生成された文字画像と，駸々堂中期フォントの文字画像の例を Table. 3 に示す．用意した駸々堂中期フォントの文字画像は，「出」が 51 枚，「大」が 68 枚，「者」が 33 枚，「時」が 36 枚，「見」が 39 枚，「手」が 32 枚である．PDC 特徴の比較方法について説明する．まず，自動生成された文字画像と駸々堂中期フォントの文字画像の PDC 特徴を求める. 次に, 求めた PDC 特徴から 2 つのユークリッド距離を算出する．1 つ目は，自動生成された文字画像と，複数枚の中から任意に選んだ 1 枚の駸々堂中期フォントの文字画像の PDC 特徴のユークリッド距離である．2 つ目は，複数枚の中から任意に選んだ 2 枚の駸々堂中期フォントの文字画像の PDC 特徴のユークリッド距離である．これ以降，1 つ目のユークリッド距離をユークリッド距離Ⅰ，2 つ目のユークリッド距離をユークリッド距離Ⅱと呼ぶ．ユークリッド距離Ⅰの分布範囲がユークリッド距離Ⅱの分布範囲内に収まっていれば，自動生成された文字画像は，駸々堂中期フォントの文字画像と類似した PDC 特徴を持つと言える．

6 種類の文字において，算出した 2 つのユークリッド距離のヒストグラムを Fig. 2 に示す．横軸がユークリッド距離，縦軸は度数を表す．ヒストグラムより，6 種類全ての文字でユークリッド距離Ⅰの分布範囲がユークリッド距離Ⅱの分布範囲内にあることが分かる．よって，自動生成された文字画像は，駸々堂中期フォントの文字画像と類似した PDC 特徴を持つと言える．このことから，Deep Learning によって，多フォント活字認識手法の学習データに利用可能な文字画像の自動生成が可能であることが分かった．

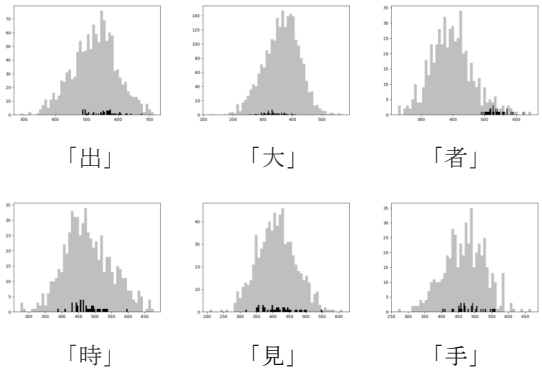

Fig. 2 ユークリッド距離のヒストグラム

Table. 4 フォント変換に失敗した文字画像の例

駸々堂中期フォントの文字画像	自動生成された文字画像
十	𠂇
寧	寔
云	亞

NN によって，全ての文字においてフォント変換が成功したわけではない．自動生成された文字画像の中には，フォントを正しく変換できなかった文字画像もある．フォント変換に失敗した文字画像の例を Table. 4 に示す．いずれの文字画像も，文字の形状に崩壊や欠損が見られる．この原因としては，訓練データから学習した特徴では，これらの文字画像を正しく再現することができなかった点が考えられる．この問題を解決するためには，様々な特徴を学習して再現できるように，NN の改善や学習データの増強を行う必要がある．

4. まとめ

本稿では，近代書籍の文字認識に特化した多フォント活字認識手法の学習データを，Deep Learning によって自動生成された文字画像で補完する手法を提案する．そのため，近代書籍の特定の出版者・出版年代のフォントの特徴を持つ文字画像を自動生成する NN の構築を目指す．

提案手法の有用性を検証するため，フォントの自動生成実験を行なった．学習する近代書籍のフォントは，明治時代中期の駸々堂で出版された書籍に用いられていたフォントである．これ以降，このフォントを駸々堂明治中期フォントと呼ぶ．入手が容易なフォントには，ゴシック体を用いる．実験のために，ゴシック体と駸々堂明治中期フォントの文字画像 2 枚 1 組のデータセットを 1297 組用意した．

フォントの自動生成実験の結果を示す．駸々堂明治中期フォントの文字画像と自動生成された文字画像のピクセルごとの画素値を比較すると，平均一致率は訓練データでは 99.79%，テストデータでは 73.69%であった．訓練データに比べ，テストデータにおける画素値の平均一致率が低かった原因として，学習データにない未知の特徴を再現できなかったことが挙げられる．しかしながら，近代書籍の文字画像は，同じ漢字であっても，線の太さや形状の歪みの程度に差異がある．そのため，駸々堂中期フォントの特徴を再現した文字画像を自動生成できていても，学習データの文字画像との一致率が低い可能性が考えられる．自動生成された文字画像が学習データに利用可能であるか否かを判定するために，PDC 特徴の比較を行う．多フォント活字認識手法では, 文字の特徴量として PDC 特徴を用いるためである．テストデータに対する精度の最も高かったモデルに自動生成された文字画像のうち，6 種類の文字で PDC 特徴を比較した．その結果，6 種類全ての文字において，自動生成された文字画像は駸々堂明治中期フォントの文字画像と類似した PDC 特徴を持つことが確認できた．よって，自動生成された文字画像は多フォント活字認識手法の学習データに利用可能であると言える．自動生成された文字画像によって，多フォント活字認識手法の学習データを補完できれば，多フォント活字認識手法によってより正確な文字認識が可能となる．それにより，近代書籍の画像データをより正確にテキストデータへ変換できるようになる．しかしながら，構築した NN ではフォントを変換した文字画像を正確に自動生成できない文字もある．より多くの文字に対してフォントを変換した文字画像を自動生成するためには，NN の改良や学習データの増強が必要となる．

謝辞　本研究は MEXT 科研費 JP17H01829 の助成を受けたものです。

参考文献

[1] 国立国会図書館. http://www.ndl.go.jp.　(参照 2018-1-8).
[2] 国立国会図書館デジタルコレクション．http://dl.ndl.go.jp. (参照 2018-1-8).
[3] Ishikawa, C., Ashida, N., Enomoto, Y., Takata, M., Kimesawa, T. and Joe, K.: Recognition of Multi-Fonts Character in Early-Modern Printed Books, Proceedings of International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA09), Vol. Ⅱ, pp. 728-734(2009).
[4] Fukuo M., Enomoto Y., Yoshii N., Takata M., Kimesawa T. and Joe K.　: EvaluaTion of the SVM based Multi-Fonts Kanji Character Recognition Method for Early-Modern Japanese Printed Books, Proceedings of The 2011 International Conference on Parallel and Distributed Processing Technologies and Applications (PDPTA2011), Vol. Ⅱ, pp. 727-732(2011).
[5] 粟津妙華，上坂和美，高田雅美，城和貴：近代書籍を対象とした多フォント活字認識手法，情報処理学会論文誌．数理モデル化と応用(TOM)，Vol. 9(2)，pp. 33-40(2016).
[6] Diederik P. Kingma, and Jimmy Lei Ba. Adam: A Method for Stochastic Optimization. ICLR, 2015.
[7] 萩田博紀，内藤誠一郎，増田功 : 外郭方向寄与度特長による手書き漢字の認識，　電子通信学会論文誌，Vol.J66-D，No.10，pp.1185-1192(1983).