7ZC-01

古典籍自動翻刻のための変体仮名切り出し手法

臺原　学†　　三輪　貴信‡　　澤田　秀之‡　　橋本　周司‡

早稲田大学先進理工学部†　　早稲田大学理工学術院‡

1. はじめに

古典籍翻刻の自動化において，文字切り出しは非常に重要である．文字切り出しの手法としては，文書画像にラベリング処理を行って得られる要素を分離，統合する手法が報告されている[1]．しかしながら，実用的な方法が確立されているとは言い難く，実際には専門家が手動で切り出しているのが現状である[2].

ここでは，文字の切り出し位置の候補を探索し，それぞれの候補が文字であるかを判定することで，再帰的に文字切り出しを行う新たな手法を検討したので，その結果を報告する．

2. 再帰的な文字切り出し手法

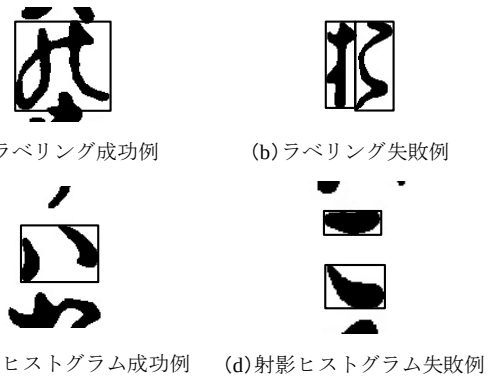
文書画像に前処理を施した後，ラベリング処理を用いて切り出し位置の候補を決定する．次に，候補を畳み込みニューラルネットワーク（CNN）による文字識別器にかけることで，正しく切り出せているかを判定する．その後，水平射影ヒストグラムを用いて切り出した候補に対して，ラベリング処理と同様の判定手続きを行う．そして，ラベリングと水平射影ヒストグラムを用いて切り出せた文字をテンプレートとして，文書画像と非線形テンプレートマッチングを行い，テンプレートと同じ文字の切り出しを行う．これらの処理を切り出せる文字がなくなるまで繰り返す． 各処理の詳細について以下に述べる．

2.1 前処理

はじめに前処理として行の切り出しを行う．古典籍の文書画像から白色画素に対する垂直射影ヒストグラムを作成し，ピークとピークの間を1行として扱う．次に，背景領域と文字領域の分離を行う．本研究では寺沢らの手法[3]と同様に，閾値以上の画素値の部分を背景として白にセットし，文字の部分はグレースケールのままにした.

2.2 連結部分のラベリング処理

ラベリング処理では，画像中で黒色画素が連結している領域，つまり，文字の連続するストロークに対して一つのラベルを付与する．それぞれのラベルを文字切り出し位置候補とする．この処理により，図 1(a)に示すようにかすれがなく続け字でない文字を切り出すことができる．しかしながら，図 1(b)に示すように，「お」「い」などの複数の独立した要素から成る文字は正しく切り出すことはできない．そこで CNN によって正しく切り出されたと判定された文字を背景と同じ色に塗りつぶし，次の処理の入力画像とする.



(a)ラベリング成功例　(b)ラベリング失敗例

(c)射影ヒストグラム成功例　(d)射影ヒストグラム失敗例

図 1　文字切り出し例

2.3 水平射影ヒストグラム

ラベリング処理の後，各行の黒色画素に対する水平射影ヒストグラムを作成する．値が 0，つまり，黒色画素が存在しない位置を切り出し位置の候補とする．この処理により，図 1(c)に示すように隣の字と重なりがなく，続け字になっていない箇所を切り出すことができる．しかしながら，図 1(d)に示すように，「こ」などの上下に分かれた要素から成る文字は，要素間が切り出し位置候補になるため正しく切り出すことはできない．CNN によって正しく切り出されたと判定された文字を背景と同じ色に塗りつぶし，次の処理の入力画像とする.

2.4 CNN による文字識別器

前述のようにラベリングおよび射影ヒストグラムで得た切り出し位置の候補が，正しいか否かを判別するために，CNN を用いた文字識別を行う．ここでは，「日本古典籍字形データセット」（国文研ほか所蔵／CODH 加工）[4]に含まれる変体仮名 228,334 字を学習データとして CNN

Segmentation Method for Reprinting of Anomalous Kana in Japanese Historical Documents

† M. Daihara, School of Advanced Science and Engineering, Waseda University

‡ T. Miwa, H. Sawada, S. Hashimoto, Faculty of Science and Engineering, Waseda University

を学習し，文字識別器を作成した．出力には変体仮名46字に対するsoftmax関数を用いた．各切り出し位置の候補画像をこの識別器に入力し，softmax 関数の値が閾値を超える場合を文字として認識する．事前実験の結果を踏まえ，閾値を0.9999に設定した．

2.5 非線形テンプレートマッチング

ラベリングおよび射影ヒストグラムによって切り出された文字をテンプレートとして，画面全域でテンプレートマッチングを行う．マッチングには寺沢らの手法[3]を用いた．この処理により，すでに切り出された文字と同じ文字を文書中から抽出することができる．このとき，同じ文字が文書中に存在しない場合には，違う文字や文字と文字の間でマッチングすることがある．この問題を解決するために，マッチングによる類似度上位3位までを前述の文字識別器にかけ，文字の位置に正しくマッチングしているかを判定する．ここでも CNN を用いて正しく認識できていると判定された領域のみを文字として切り出し，背景と同じ色に塗りつぶし，2.2 のラベリング処理に戻る．

以上の処理を新たな文字切り出しがなくなるまで繰り返す．

3. 文字切り出しの実験

図2は，例題として用いた人間文化研究機構国文学研究資料館所蔵「大鏡」の2頁である．これに対して提案手法を適用した結果を示す．図2に含まれる422字の変体仮名が切り出しの対象である．このうち正しく切り出せたのは111字であった．これは文書中に含まれる変体仮名全体の26%にあたる．一方，softmax 関数の値が閾値を超えなかったが，ラベリングまたは水平射影ヒストグラムにより正しく切り出されていると目視で判断できる文字は 29 字あった．このことから，softmax 関数の閾値を下げることにより認識できる文字数を増やすことができる可能性がある．ただし，閾値を下げることは誤認識の割合を増やすことにもつながる．

図3に切り出しの失敗例を示す．四角で囲った部分が切り出された文字である．図 3(a)に示すように隣の文字の一部を含んだ状態や図 3(b)に示すように文字の一部しか切り出されていない状態でも文字として認識されてしまう場合があった．これらの状態で認識してしまうと次の処理に移る際に，文字の一部が欠けていて認識できない，残された文字の一部がノイズとして残ってしまうといった問題が生じる．

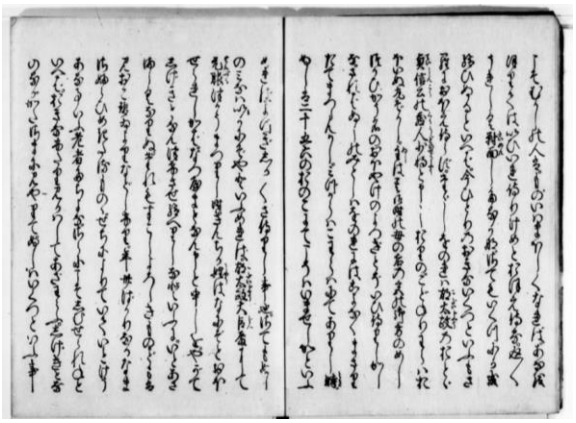
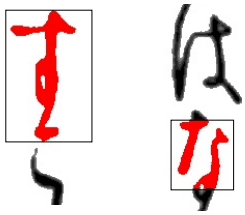


図2　実験に用いた画像.



(a)　(b)

図3　切り出しの失敗例.

切り出せなかった文字の多くは続け字になっている部分であった．提案手法では，続け字部分はテンプレートマッチングでマッチングしなければ切り出せず，切り出し位置を定めるのが難しい．[1]では，続け字部分の黒色画素に対する水平射影ヒストグラムを作成し，最小値の位置で続け字を分離していることから，続け字部分に対しても基準の最適化によってある程度は改善が可能であると考えられる．

4. まとめ

ラベリング処理と射影ヒストグラム，非線形テンプレートマッチングを用いて古典籍から変体仮名を切り出す手法を検討した．切り出し判定を行う CNN の認識性能の問題もあるが，続け字の多い古典籍においては，文字単位の扱いには限界があると考えられる．今後は，語単位，文節単位などのコンテキスト情報を取り入れて認識率向上を図る予定である．

参考文献

[1] 坪井, 他: 情報処理学会研究報告, 2005-CH-066, 53–60. (2005)
[2] 山本,大澤: 情報管理 58,11,819-827.(2016)
[3] 寺沢, 他: 電子情報通信学会論文誌 D, J89-D, 8, 1829–1839. (2006)
[4] 日本古典籍字形データセット, 人文学オープンデータ共同利用センター. http://codh.rois.ac.jp/char-shape/ (参照 2017.12.23)