N-020

画像処理とクラスターを用いたくずし字セグメンテーション
Kuzushiji Segmentation with Image Processing and Cluster Analysis

高 涵† 呂 氷† 王 志辰† 孟 林†
Gao Han Lyu Bing Wang Zhichen Meng Lin

1. はじめに

日本の古典籍は多くの情報を記録しており、これらの古典籍を解読することは歴史、政治、文化に非常に役立つ。しかし、沢山の古典籍はくずし字で書かれた。くずし字は今ほどんと使われていなく、極わずかな専門家しか解読できない書体である。長年を経て、虫食いや汚れなどノイズが存在している。特に、多くの古典籍は文字と文字を繋がっているくずし字を使用しているため、古典籍の解読が難しくなる。本論文では、日本の古典籍の縦書きという特徴に着目し、画像処理とクラスターを用いて、古典籍画像の自動文字抽出を目指す。人文学オープンデータ共同利用センターの古典籍を用いた評価実験により、提案手法の有効性を確認した。

2. 文字セグメンテーション手法

今画像処理を使用して文字を直接抽出する手法が多い。しかし、日本の古典籍を抽出する状況は複雑で、主に二つの問題がある。一つは長年を経て、虫食いや汚れなどノイズが存在している。もう一つ問題はくずし字が文字を簡略化し、迅速に記述できるため、繋がっている文字がたくさん存在している。本論文の目的は、古典籍の縦書きという特徴に着目し、くずし字をよく抽出することを実現する。提案手法は図 1 のように、文字列の中心線を探して、文字の位置を推定し、文字と仮名セグメンテーションを行う。そして、水平射影分布でにより、再セグメンテーションを行う。最後の出力画像は文字抽出画像である。

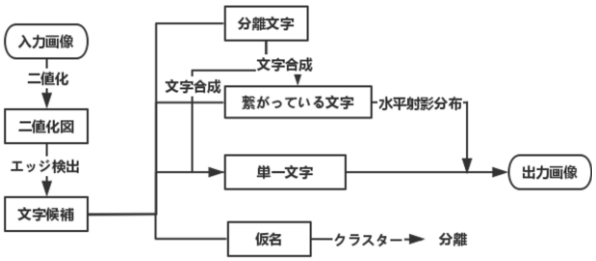


図 1　セグメンテーション手法

まず、入力データを二値化処理する。次に、エッジ検出により、文字列から画像中の文字候補のセグメンテーションを行う。文字候補はノイズデータ、くずし字データ、および仮名データがあるデータセットである。セグメンテーションされた文字候補から、クラスターで文字列の中心線を計算する。文字候補に左右で分けられた場合はあるので、文字の位置を推定し、文字合成を行う。クラスターでくずし字の抽出を行う。繋がっている文字に対して、水平射影分布により、再セグメンテーションを行う。

† 立命館大学 大学院理工学研究科, Graduate School of Science and Engineering, Ritsumeikan University

2.1 二値化

二値化処理は、画像中のピクセルについて、閾値未満のピクセルに対して 0 に変換し、閾値より大きいピクセルに対して 255 に変換する処理である。図 2 のように、画像を白と黒のみに変換する。二値化処理により、画像と背景の境界を明確化させることで、次の処理の速度を向上させるだけでなく、結果に影響を与える多くのノイズデータを削除できる。

本論文では、二値化の大津法を用いて実現する。大津法の特徴は、分離度が最も大きくなるときの閾値を探して画像二値化を行う。具体的に、入力画像のヒストグラムから、ピクセルの最大値、最小値、平均値を計算する。最小値から最大値の範囲内で選ぶ閾値 Tで二つのクラスに分けて、クラス 1 とクラス 2 の分離度が最大になる時の閾値 Tを二値化閾値とする。画像を二値化する結果が図 2 のように、図 2(a)は閾値が 200 に設定された出力画像である。図 2(b)は大津二値化の出力である。図 2 からわかるように、大津法を使用して、ノイズが大幅に低減される。

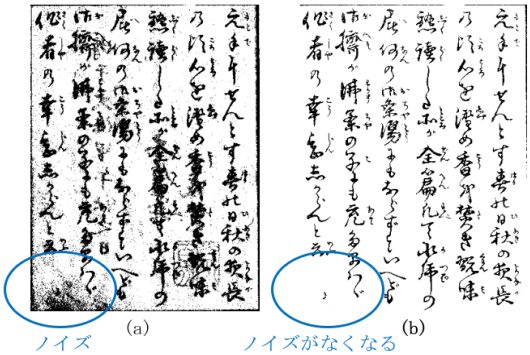


図 2　二値化画像

2.2　エッジ検出

エッジ検出は、コンピューター ビジョンの分野で非常に重要な画像特徴を抽出できる方法である。入力画像に対して、エッジ検出演算子により、画像にフィルタリングされて、勾配を取得する。勾配は画像のピクセルの変化を反映することができる。勾配が大きいはピクセルの変化速度が速いことを示すため、画像の特徴を抽出することができる。エッジ検出は、画像のピクセルの明るさが大幅に変化したピクセルの集合を見つけることである。これらの集合は通常、輪郭である。

本論文では、エッジ検出の Findcontours 関数を使用している。Findcontours 関数は二値化画像の輪郭を抽出できる関数である。図 3 は画像中の文字のエッジ検出結果である。

図 3 エッジ検出

2.3 クラスター

クラスター分析は、オブジェクトとそれらの関係を説明するデータ内の情報に基づいてデータ オブジェクトをグループ化することである。グループ内のオブジェクトが互いに類似している一方で、異なるグループ内のオブジェクトは異なる。

本論文ではクラスターの Kmeans 関数を使用して文字の分類を実現する。Kmeans アルゴリズムは図 4(b)を示すように、最初にK個（図 4 で$K = 2$として説明する）の初期重心を選択してピクセルを分類する。初期重心はランダムで選択して各重心から全てのピクセルまでの距離を計算して、図 4(c)のように、ピクセルをそれぞれに最も近い重心に割り当てる。各グループの平均ベクトルを計算して、図 4(d)のように、新しい重心を選ぶ。新しい重心から全てのピクセルまでの距離を計算してピクセルを割り当てる。上記の操作を繰り返して、図 4(f)ように、重心が変化しなければ、または最大反復回数に達するまで、出力する。このようにして、すべてのピクセルがグループ化される。

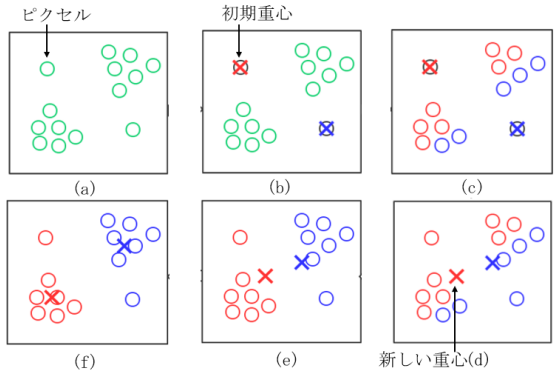


図 4　Kmeans アルゴリズム(K=2 とする)

Kmeans クラスターを用いて画像中の全てのエッジ検出する文字を分類する。K は画像の文字列の数であり、分類する結果が図 5 を示すように、各列に色をつけて分類する。

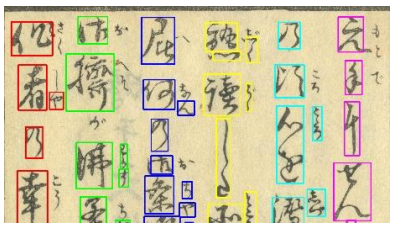


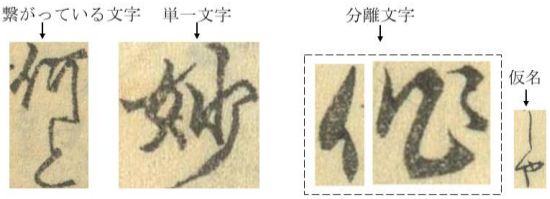
図 5　Kmeans で分類結果

2.4 文字処理

エッジ検出により得られたデータは単一文字、繋がっている文字、分離文字と仮名を構成されている。

2.4.1 文字合成

文字合成は、エッジ検出された分離文字に対して、文字の特徴の分析し、提案する方法で文字を合成を行う。図 6(a)のように、単一文字と繋がっている文字は有効なデータと定義される。図 6(b)のように、分離文字と仮名は無効なデータと定義される。無効なデータは使用する前に削除または処理する必要がある。この提案方法は、無効なデータを処理する方法である。



(a)有効データ　(b)無効データ

図 6　エッジ検出されたデータ

分離文字に対して、文字合成を行う。図 7 のように、データ A とデータ B のy軸に重なるΔYは、データのy座標間隔がどのくらいが重なるかどうかを判断する重要なパラメーターである。

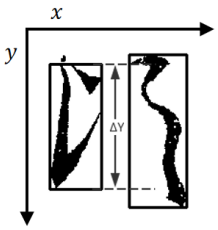


図 7 y軸に重なるΔY

図 8 を示すように、古典画像では、分離文字におけるデータ A、B の配列の組み合わせが六種類ある。

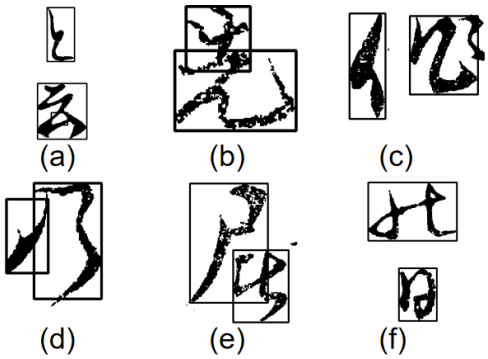


図 8　ΔY の六種類

図 8 から分かるように、図 8(a)と(f)のΔY が存在していなくて、文字合成処理を実行しない。図 8(b)、(c)、(d)と(e)はΔYがあって、文字合成を実行する。

文字合成は、主に各文字列の中心線を探すことで文字を合成する。図 9 のように、中心線は、文字列のすべてのボ

ックスの中心点のx座標と面積に重みを付けて計算された回帰線である。

図 9 のように、wはデータの幅であり、hはデータの長さである。Wは文字列のすべてのデータ面積の加重平均である。Sは、文字列すべてのデータの面積合計である。Xは、この列の中心線の横座標である。

$$W = \sum x_n \times w_n \times h_n$$
$$S = \sum w_n \times h_n$$
$$X = W/S$$

面積の重みを使用して計算された中心線は、古典籍の文字分布特徴をより適切に示すことができる。また、くずし字データの面積の重みが大きいため、中心線はくずし字に偏る。中心線に最も近いデータは、くずし字またはくずし字の一部と考えられる。

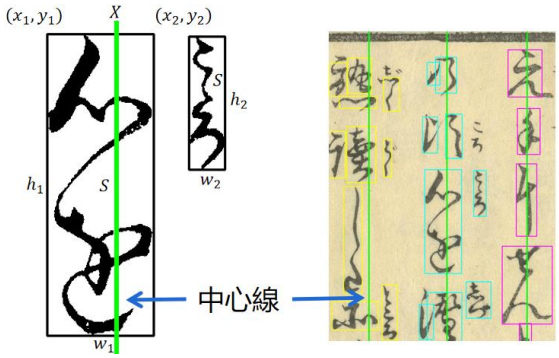


図 9 中心線

中心線に近いデータをボックス合成を行う。合成したボックスをもう一度合成処理を行う。図 10 のように、データ(a)(b)がくずし字データであり、データ(c)が仮名データである。文字列の中心線を計算すると（みどり線）、データ(a)(b)が中心線に近いことが分かった。そして、データ(a)(b)を文字合成して図(b)になる。そのあと、データ(d)(c)から中心線までの距離を計算して合成するかを判断する。データ(c)が中心線に遠いので、図(c)のように、データ(d)(c)が合成しない。

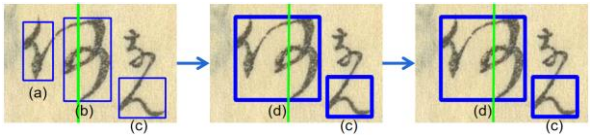


図 10　文字合成の例

2.4.2 仮名処理

古典籍の縦書きという特徴あるので、理想的な状況では、各列の間隔に最大で 1 つ文字のサイズがある。しかし、図 11(a)を示すように、文字の周りに仮名がついている古典籍もあるので、仮名が文字の抽出に影響し、捨ててしまう必要がある。文字列の特徴を分析すると、くずし字データは文字列の左側にあり、仮名データは文字列の右側にある。各文字列にクラスター処理を行って、図(a)のように、くずし字と仮名が違う色で表記されていて、黒ボックスが仮名である。この方法で、仮名を探して切り捨てて、結果は図(b)のように、仮名が捨ててしまう。

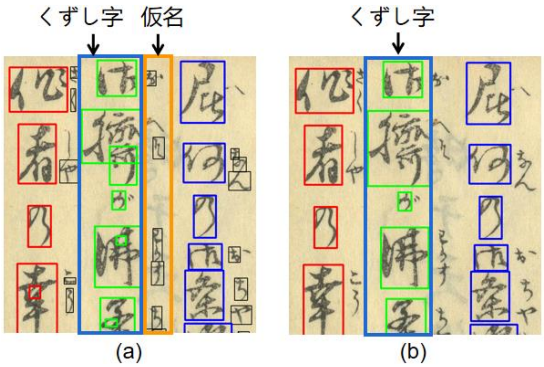


図 11　くずし字と仮名の分離

2.4.3 繋がっている文字の処理

繋がっている文字の文字分割処理を実行する。アスペクト比とは、画面や画像の縦と横の長さの比である。単一文字のアスペクト比が約 1:1 であるので、繋がっている文字のアスペクト比は普通に 1.8 より大きい。この特徴に基づいて、アスペクト比を計算して文字を分割する方法を提案した。図 12 のように、アスペクト比が 1.8 より大きい場合は繋がっている文字と考えられ、セグメンテーションする必要がある。

水平射影分布とは、画像を水平方向に投影したときに記録される黒ピクセルの分布のことである。図 12 のように、文字と文字の間に黒いピクセルが少ない特徴を着目し、繋がっている文字の境界エリアを探す（赤い枠）。境界エリアのピクセル数が少ないため、エリア内のピクセルの平均値は他のエリアよりも小さくなる。

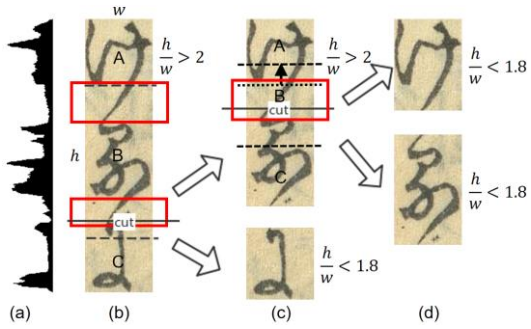


図 12　水平射影分布でセグメンテーション

具体的な手法は図 13 のように、まず、画像のアスペクト比を計算し、アスペクト比が 2 より大きい場合は文字分割する。図(a)の頂点と下端から 1x1 の大きさのエリアを選択して A エリアと C エリアとして、他のエリアが B エリアとする。各エリアの平均黒いピクセルを計算して、それぞれT_A、T_B、T_Cである。$T_B < T_A$と$T_B < T_C$の場合は B エリア内の平均黒いピクセルが最も小さい行$min(t_B)$をカットラインとしてカットする。$T_B > T_A$あるいは$T_B > T_C$の場合は B エリアを拡大することが必要である。新 D エリアの最も小さい行$min(t_D)$をカットラインとしてカットする。

カットされた画像を図 13 のように、もう一度カットする。カットされた画像のアスペクト比もまだ 2 より大きい場合は上記の操作を繰り返す。カットされた画像のアスペクト比が 2 未満で 1.8 より大きい場合に、$T_B < T_A$と$T_B < T_C$

の場合は B エリア内の平均黒いピクセルが最も小さい行$min(t_B)$をカットラインとしてカットする。新カットされた画像を出力する。$T_B > T_A$あるいは$T_B > T_C$の場合は出力する。カットされた画像のアスペクト比が 1.8 より小さい場合は出力する。

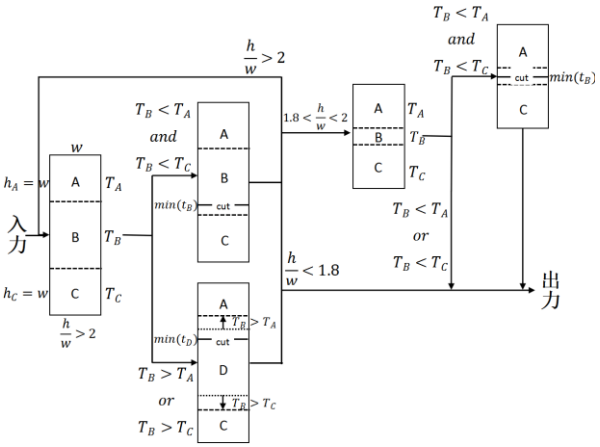


図 13　セグメンテーション手法

3. 実験

3.1 実験データ

本論文では、人文学オープンデータ共同利用センターが所蔵している「虚南留別志」(国文研書誌 ID：100241706)を用いて、評価実験により、提案手法の有効性を確認した。OS は Windows で、プログラミング言語は Python である。

3.2 実験結果

3.2.1 実験 1：画像中のノイズを削減する実験

入力画像を大津二値化して、ノイズ が大幅に削減され、次のデータ実験へのノイズの影響が軽減されることが明らかになる。9 枚の日本古典籍画像を用いてノイズ削減する実験を行う。大津法で画像のノイズの割合が 26.6%から 3.4%に減少する。

3.2.2 実験 2：提案手法で文字セグメンテーション実験

図 14 に示すように、文字候補に左右で分けられた場合はあるので、文字列の中心線を用いて、文字の位置を推定し、文字合成を行う。

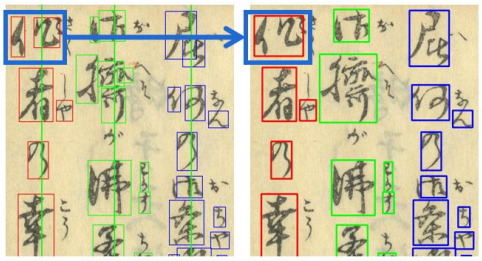


図 14　文字合成の結果

図 15 に示すように、文字合成により、データ数を大幅に削減する。また、取得したデータの 86%以上が有効データである。8 列のデータに対して、提案手法はもっと良い効果がある。

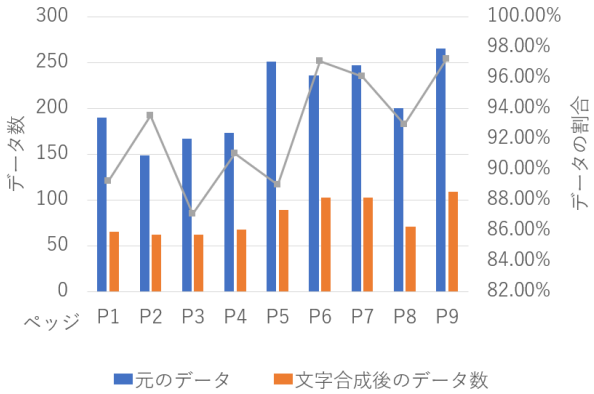


図 15　データ数の変化と有効データの割合

繋がっている文字のセグメンテーションの成功率の平均値は 75.9%である。図 16 のように、単一文字データ数の割合は大幅に向上する。提案手法の改善により、セグメンテーションの成功率の向上が今後の課題となる。

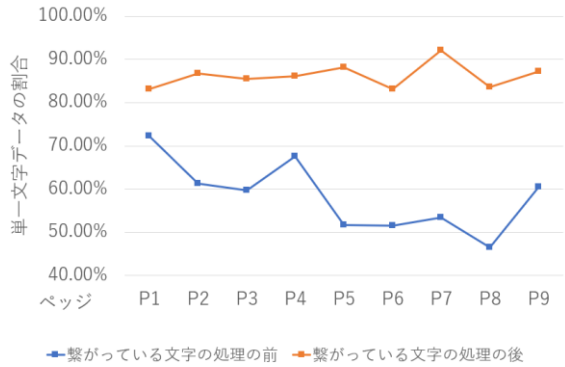


図 16　単一文字データ数の割合の変化

4. おわりに

本研究では、提案する文字合成と文字再セグメンテーションの手法でくずし字の抽出を実現する。文字合成処理の後、仮名がなくなる。93.1%の出力データは使える。75.9%の複数なくずし字があるデータを再セグメンテーションできる。提案手法の有効性を確認した。提案手法の改善により、セグメンテーションの成功率の向上が今後の課題となる。更なる実験の追加と、セグメンテーション後のくずし字の自動的な認識も今後の課題となる。

謝辞

本研究は人文学オープンデータ共同利用センターの助成を受けたものです。

参考文献

[1] 日本古典籍くずし字データセット, 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/book/100241706/(2021.6.15)
[2] 大津 展之, “判別および最小 2 乗規準に基づく自動しきい値選定法”, 電子通信学会論文誌, Vol. J-63D, No. 4 (1980)
[3] SatoshiSuzuki, “Topological structural analysis of digitized binary images by border following”, Computer Vision, Graphics, and Image Processing, Vol30, Issue 1(1985)
[4] Clustering, scikit-learn : https://scikit-learn.org/stable/modules/clustering.html#k-means(2021.6.15)