情報処理学会論文誌　数理モデル化と応用　Vol.8　No.1　72–79 (Mar. 2015)

活字データの分類を用いた進化計算による近代書籍からのルビ除去

粟津 妙華[1,a)]　高田 雅美[1,b)]　城 和貴[1,c)]

受付日 2014年5月28日，再受付日 2014年7月18日，
採録日 2014年9月5日

概要：国立国会図書館では，所蔵する明治から昭和前期の近代書籍を近代デジタルライブラリとしてWeb上でページごとの画像データとして公開しているが，文書内容での検索を行うことができない．そのため，自動でのテキスト化が望まれている．その際，問題となっているのがヒストグラム法では除去できないルビであり，我々はすでに近代書籍に特化したルビ除去手法を提案している．しかしながら，その提案した手法は書籍に付加された版者や時代などの外部情報を利用しなければならず，近代デジタルライブラリのすべての外部情報を利用することはきわめて困難である．そこで本論文では，対象とする書籍画像から直接得られるデータをもとに，進化計算によってルビ除去式を生成し，近代書籍から自動でルビを除去する手法を提案する．

キーワード：近代書籍，自動テキスト化，ルビ除去，遺伝的プログラミング，クラスタリング

Ruby Removal Filters by Genetic Programming Using the Classification of Printing Type Data for Early-modern Japanese Printed Books

Taeka Awazu[1,a)]　Masami Takata[1,b)]　Kazuki Joe[1,c)]

Received: May 28, 2014, Revised: July 18, 2014,
Accepted: September 5, 2014

Abstract: In the web site of National Diet Library, the digital library from the Meiji era is open to the public. Since the early-modern Japanese printed books are given as image data, namely, full-text search is not available, automatic conversion to the text is needed. There is a major obstacle to the text conversion because of ruby, which is found in early-modern printed books. Ruby cannot be removed by the existing and traditional histogram method. Therefore, we have proposed a ruby removal method for early-modern printed books. Since the proposed method is based on the external information added to the books, the feasibility is very low. In this paper, we propose a new method to remove the ruby automatically from early-modern Japanese printed books by generating ruby removal formula by Genetic Programming using the training data based on the book images.

Keywords: ruby remove, early-modern printed books, genetic programming, aspect ratio of the print

1. はじめに

国立国会図書館関西館では，明治期から昭和前期にかけての書籍約34万冊を公開している．これらの近代書籍は，哲学・自然科学・文学などの幅広い分野にわたり，また，現在は絶版になっている書籍も多く，学術的に貴重な資料である．そこで国立国会図書館では，図書館資料を文化財として永く後世に伝えるとともに広く利用に供にするという目的のもと，所蔵資料のデジタルアーカイブ化を行い，近代デジタルライブラリとして電子図書館サービスを提供している．近代デジタルライブラリのWebサイトでは，タイトル・著者名のほかに出版者や出版年など外部情報を設

[1] 奈良女子大学
Nara Women's University, Nara 630–8506, Japan
[a)] awazu-taeka0802@ics.nara-wu.ac.jp
[b)] takata@ics.nara-wu.ac.jp
[c)] joe@ics.nara-wu.ac.jp

定して近代書籍の検索を行うことが可能である．しかしながら，近代書籍の本文は画像として公開されているため，本文の全文検索を行うことができない．全文検索を行うには，画像データである現在の近代デジタルライブラリのテキスト化が必要となる．近代書籍は学術的に貴重なものを多く含むとはいえ，数十万冊に及ぶ書籍の人手によるテキスト化はコスト的に不可能である．

このような背景のもと，我々は国立国会図書館関西館に協力を仰ぎ，近代デジタルライブラリの自動テキスト化に関する研究 [1], [2] に着手している．近代書籍をテキスト化する際，画像データに既存 OCR を適用しても認識率が低く実用に耐えうるものではない．国立国会図書館では，平成 22 年度に近代書籍の全文テキスト化実証実験を行い，その報告書をサイトで公開している [3]．その中で既存 OCR を用いた認識実験が報告されているが，明治期・大正期の書籍における認識率はおよそ 88%であり，またルビのある書籍で 41.9%となっている．そこで，我々は手書き文字認識の手法を利用することで近代書籍から切り出された活字の認識が可能であることを報告している [1], [2]．実際，近代書籍では出版者ごとに用いる活版が異なることは当然予測されることであるが，同じ出版者であっても時代によって活版が異なることも報告されている [4]．近代書籍の活字認識に手書き文字認識の手法を利用するのはこのような背景があるためである．

近代書籍の自動テキスト化を行うためには，認識対象の活字も自動で切り出さなければならないが，一般にルビによる文字切り出しの失敗がその後の文字認識率を劣化させることが知られている [5]．特に近代書籍では，現在の書籍のように決まった規格はないため，既存のルビ除去技術を適用したのでは，肝心の文字認識率が大幅に低下してしまう．この問題を解決するため，我々は近代書籍に特化したルビ除去手法を開発している [6]．これは，遺伝的プログラミング [7] を用い，出版者・時代ごとの専用ルビ除去式を生成する手法である．しかしながら，近代書籍の出版者は個人出版のものも多く含まれ，数は 1 万を超える．そのため，出版者・時代ごとに手動で教師データを整理することは非常に困難である．そこで，本論文では，分類方法を見直し，活字画像から得られる活字のアスペクト比を用いて近代書籍のクラスタリング [8] を行う．そして，その分類をもとに遺伝的プログラミングを用いてルビ除去式を生成し，自動でルビを除去する手法を提案する．

本論文の構成は，以下のとおりである．

2 章において，遺伝的プログラミングによるルビ除去式生成の概略について述べ，著者らの先行研究 [6] で提案した分類によるルビ除去手法と問題点についてまとめる．3 章において活字データを用いた近代書籍のクラスタリングと，それをもとにしたルビ除去式の生成について提案する．4 章において，提案手法の有効性を調べる実験について述べる．活字データによる分類を用いた提案手法と，黒画素射影ヒストグラム法の結果を比較し，考察を行う．

2. ルビ除去のための出版者・時代ごとの分類とその問題点

近代書籍は活版印刷であり，現在のように統一された規格はない．現在の規格に沿ったルビは，ルビとルビの付いている文字（親文字）の間に一定の間隔が空いている．しかし，近代書籍によく見られるルビは，親文字とルビの近接度が高く，連結しているものも多い．連結している場合，親文字とルビのそれぞれの特徴値を求めることが困難であるため，サポートベクターマシンなどの機械学習では，ルビだけを分離することはできない．また，黒画素射影ヒストグラムを用いた場合も，近接度の高さが原因でルビの除去率は高くない．この問題を解決するために，我々はすでに近代書籍に特化したルビ除去手法を提案している [6]．ここでは提案したルビ除去手法を簡単に説明し，その問題を述べる．

2.1 遺伝的プログラミングを用いたルビ除去式の生成

この手法は，親文字とルビの境の近似式を求める手法である．以下に概要を述べる．**図 1** は著者らの先行研究 [6] で提案したフローチャートである．

(1) 教師データの原画像である各行からルビ付き文字列の座標位置と文字の幅の推定．
(2) 手順 (1) の値を与え，遺伝的プログラミングを用い除去式を生成．
　(a) 初期個体群の生成．
　(b) 手順 (1) で求めた位置情報と幅を終端要素として与え，適応度を計算．
　(c) 終了条件の確認．
　(d) ルーレット選択で，個体群の半数を交叉．
　(e) ランダム選択で選んだ個体を突然変異．
　(f) 適応度の計算．

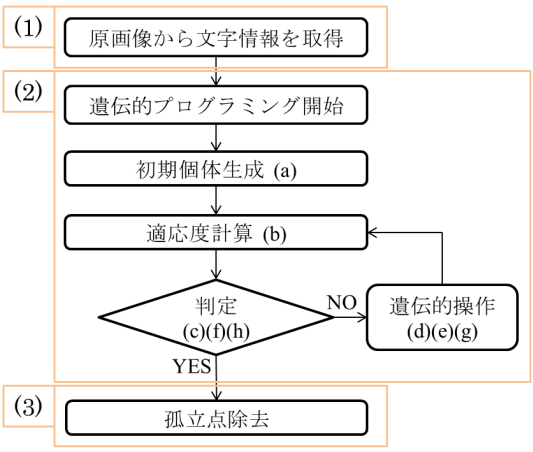


図 1　著者らの先行研究 [6] で提案したフロー
Fig. 1　Flow of Ref. [6] method.

(g) 適応度の低い個体を削除，新たに個体を生成．
(h) 手順 (2c) に戻る．
(3) 除去式で除去後，メディアンフィルタを適用し，残ったルビの一部に対し孤立点除去を行う．

この手法では，遺伝的プログラミングを用い，行における親文字とルビの境の近似式を自動生成する．初めに，教師データである各行から文字の位置情報や高さ・幅などを推定し，それらの値を遺伝的プログラミングの終端要素として与え，ルビ除去式を自動生成する．適応度は目標画像との輝度値の一致率である．除去式を適用後，残ったルビの一部を除去するため，孤立点除去を行う．

2.2　問題点

近代書籍は活版印刷であるため，活版の数だけフォントが存在し，それらは出版者や時代によって特徴が異なることは容易に想像できる．そこで，この手法では，出版者・時代ごとの専用ルビ除去式を生成することとした．教師データとして，出版者・時代ごとに分類した近代書籍の行を用い，遺伝的プログラミングによって出版者・時代ごとの専用ルビ除去式を生成した．

実験では，98%前後の除去率となった．たとえば「春陽堂・明治中期」で生成された式ではルビ除去率は 99.0%であった．また，判別分析法を用いた黒画素射影ヒストグラム法では，およそ 83%の除去率であり，提案した手法の有効性は示された．

しかしながら，近代書籍の出版者の数は膨大である．近代デジタルライブラリで公開されている書籍の出版者だけでも 1 万を超える．出版者・時代という書籍に付加された外部情報を用いて，手動で分類することは難しく，この分類方法を用いた近代デジタルライブラリすべての自動テキスト化はコスト的に不可能である．そのため，近代書籍の自動テキスト化には，活字画像から直接得られるデータをもとに自動でクラスタリングし，ルビ除去式を生成することが必要である．

3.　活字データによるクラスタリング

本論文では著者らの先行研究 [6] で提案した，遺伝的プログラミングを用いたルビ除去式の生成手法を用いる．この手法は，親文字とルビの境の近似式を求める手法である．文献 [6] では，教師データとして，出版者・時代という外部情報をもとに手動で分類したが，本論文では，画像から直接得られる活字のアスペクト比によって自動でクラスタリングし，それらを教師データとして遺伝的プログラミングを用いてルビの除去式を生成する．生成された除去式の有効性を検証するため，4 章で，実文書を用い評価実験を行う．

3.1　活字のアスペクト比

本論文で提案するクラスタリング方法は，活字のアスペクト比を用いるものである．著者らの先行研究 [6] では，フォントは出版者・時代ごとに特徴があると仮定したが，実際のフォントは，活版の数，つまり印刷所の数だけ存在していると考えられる．これは出版者・時代という括り以上に，手動での分類は不可能である．そのため，実際に印字された活字データから特徴値を求めることが最善であると考える．ルビ除去式の生成において，親文字とルビの間の距離や，字体ごとに異なるアスペクト比は重要な要素であり，各印刷所ごとに異なる活版を用いることで，活字のアスペクト比に違いが生じる．そこで，行ごとのルビを含む活字の平均アスペクト比を用いてクラスタリングし，遺伝的プログラミングによってルビ除去式を生成する．

活字のアスペクト比を求めるために，まず行の中の親文字の高さとルビを含む幅を算出する．初めに，行方向に垂直に黒画素射影ヒストグラムをとり，谷の部分で分離し，その高さの平均を求める．その際，求めた高さが，実際の高さと大きな差が出ることがある．インクのにじみなどにより親文字が上下で連結した文字は，その他の文字の高さよりも大幅に大きくなり，漢数字の「二」「三」のように小さく分離されてしまう文字や句読点は，その他の文字よりも非常に小さくなる．そのため，平均値を求める際には，上記の実際の高さの値と大きく異なる高さの値を省く必要がある．これにより，実際の高さと平均値の差異が小さくなることが期待される．省く値は，いったんすべての高さの値から平均を求め，その平均値から 20%以上離れたものとする．

次にルビを含む幅の平均を求める．高さの平均を求める際に省いた文字と，高さの 1.2 倍以下の幅となる文字は除き，親文字のルビを含む幅の平均を算出する．文字の縦横の理想比率は 1 : 1 であり，幅が高さの 1 倍以上であればルビがあると判断できるが，実際の文字の縦横比率は 1 : 1 ではない．高さの 1 倍以上の文字を親文字とすると，ルビの付いていない文字も複数含まれてしまう．そこで確実にルビのある文字だけを対象とするため，高さの 1.2 倍以上をルビ付きの文字であるとする．そして，ルビのついている親文字を対象として，その高さとルビを含む幅の比を求める．アスペクト比を f とすると，

$$f = \frac{\text{ルビを含む幅}}{\text{高さ}} \tag{1}$$

と表される．**図 2** は，アスペクト比に用いる高さとルビを含む幅を示している．

3.2　アスペクト比によるルビ除去式の生成

著者らの先行研究 [6] で用いた，近代デジタルライブラリで公開されている近代書籍 900 冊を対象にアスペクト比 f を求めたところ，およそ 1.4 から 1.8 の間となり，大半は 1.5 から 1.7 であり，1.4 以下と 1.8 以上は非常に少なく，それぞれ全体の 5%ほどである．

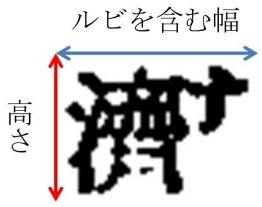


図 2　アスペクト比に用いる高さとルビを含む幅
Fig. 2　Width with ruby and height for aspect ratio.

表 1　各グループにおける目標画像の輝度値との一致率（%）
Table 1　The best agreement rates for each class.

f	目標画像との一致率
[-:1.4]	99.022
[1.35:1.45]	99.018
[1.4:1.5]	99.021
[1.45:1.55]	99.021
[1.5:1.6]	99.013
[1.55:1.65]	99.022
[1.6:1.7]	99.001
[1.65:1.75]	99.212
[1.7:1.8]	99.207
[1.75:1.85]	99.018
[1.8:-]	99.022

はじめに f の値によって暫定的にクラスタリングし，ルビの除去率を精査した後，もう一度詳細にクラスタリングを行う．まず，f の値が 1.4 以下，1.35–1.45，1.4–1.5，1.45–1.55，1.5–1.6，1.55–1.65，1.6–1.7，1.65–1.75，1.7–1.8，1.75–1.85，1.8 以上の 11 に分類する．教師データ数の確保のため，f 値は 1.0 刻みとする．1.4 以下と 1.35–1.45，1.7–1.8，1.75–1.85，1.8 以上のグループは，教師データが 100 行集まらなかったが，50 行以上あるので，それを教師データとする．それ以外のグループは教師データ 100 行である．実験はグループごとに 10 回行い，各グループでの除去式によるルビ除去後と目標画像の輝度値の最も良い一致率を求める．結果を**表 1** に示す．

すべてのグループで 99%以上の輝度値の一致率となっている．生成された式を精査したところ，1.4 以下と 1.35–1.45，1.4–1.5，1.45–1.55，1.5–1.6，1.55–1.65，1.6–1.7 の各グループで生成された式は，すべて同じ式となる．つまり 1.65 以下は同一の式でルビを除去できる．生成された式を以下に示す．

$$y = \text{高さの平均値} + 6 \tag{2}$$

f の値が 1.65–1.75，1.7–1.8，1.75–1.85，1.8 以上のグループで生成された式は，それぞれ異なっている．ここで目標画像との一致率から重複している範囲の詳細なクラスタリングを行う．重複区間で生成された 2 つの除去式のうち，ルビ除去率の良い式を採用する．2 つのルビ除去率がほぼ同じ場合は，重複区間をさらに細かく分割し，2 つの除去式を適用し，ルビ除去率を求める．**図 3** は，f の値と除去式によるルビ除去後の画像と目標画像との輝度値の一致率，重複区間を示したものである．

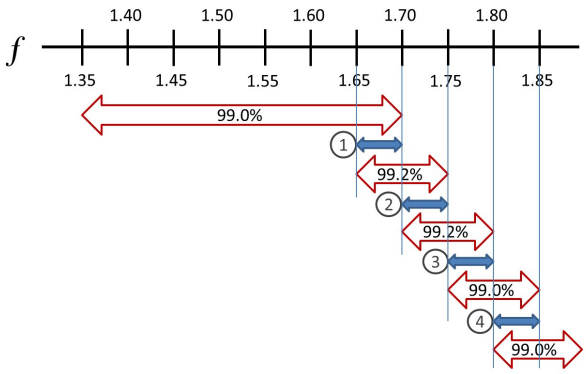


図 3　f 値と重複区間
Fig. 3　The value f and overlap section.

表 2　f 値とそれぞれの式における輝度値の一致率（%）
Table 2　The agreement rates for value f and each class.

f 値	[1.65:1.75] の式	[1.7:1.8] の式
[1.70:1.71]	99.193	99.016
[1.71:1.72]	99.094	99.142
[1.72:1.73]	99.154	99.185
[1.73:1.74]	99.188	99.191
[1.74:1.75]	99.179	99.199

図 3 の重複区間 ① 1.65–1.7 では，暫定クラスタ 1.6–1.7 の式での目標画像との一致率はおよそ 99.0%，1.65–1.75 の式でおよそ 99.2%である．そこで f 値 1.65–1.7 の教師データを用い，両方の式を適用すると 1.6–1.7 の式の一致率 98.82%，1.65–1.75 の式で 99.21%である．そのため，この区間では 1.65–1.75 の除去式が適切であると考える．

次に，図 3 の重複区間 ② 1.7–1.75 では，暫定クラスタ 1.65–1.75 と 1.7–1.8 で生成された式のどちらもおよそ 99.2%となっている．そこで f 値を 0.01 刻みで 15 行ずつ用意し，1.65–1.75 と 1.7–1.8 の除去式を適用する．f 値を 0.01 刻みにした場合，使用した教師データでは 15 行ずつしか用意できないため，15 行で実験を行う．結果を**表 2** に示す．

1.7–1.71 では，1.65–1.75 の式を適用した方が目標画像との一致率は高くなる．また 1.71–1.72，1.72–1.73，1.73–1.74，1.74–1.75 では，わずかだが 1.7–1.8 の式を適用した方が一致率は高い．この結果から，1.7–1.75 の重複区間は，1.71 で適用区間を分けるのが適切であると考える．

図 3 の重複区間 ③ 1.75–1.8 では，暫定クラスタ 1.7–1.8 の式での目標画像との一致率はおよそ 99.2%，1.75–1.85 の式でおよそ 99.0%である．そこで f 値 1.75–1.8 の教師データを両方の式に適用すると 1.7–1.8 の式で一致率は 99.183%，1.75–1.85 の式で 99.047%である．そのため，この区間では 1.7–1.8 の除去式が適切である．例として，1.7–1.8 で生

成された式は以下のとおりである．

$$
\begin{aligned}
y = {} & (8 - ((\text{高さの平均値}/8) - (\text{高さの平均値} - \\
& (5/(\cos((2 \times \pi \times x/(((8 \times 5)/2)) - \pi/2)) - \\
& (\text{高さの平均値} - ((4/(6 \times \cos((2 \times \pi \times x/ \\
& ((\cos((2 \times \pi \times x/(((8 \times 5)/2)) - \pi/2))/2)) \\
& -\pi/2))))/\text{高さの平均値})))))))
\end{aligned}
\tag{3}
$$

図 3 の重複区間 ④ 1.8–1.85 では，暫定クラスタ 1.75–1.85 と 1.8 以上で生成された式のどちらもおよそ 99.0%となっている．f 値 1.8 以上は数が非常に少なく，教師データとした行で全体の 5.5%しかない．0.01 刻みで 15 行を集めることは困難であるため，1.8 以上の教師データすべてを使用し一致率を検証する．結果，1.75–1.85 の式での一致率は 99.041%，1.8 以上の式で 99.022%である．これより，1.8 以上では 1.75–1.85 の式を適用する方が良い結果となることが分かる．

次に，重複なく 1 段階のクラスタリングを行った場合を考える．f 値を 1.35 から 1.0 刻みに重複なく分割すると，1.75 から 1.80 の区間では，およそ 99.0%の 1.75–1.85 の除去式を適用することとなり，2 段階でクラスタリングを行った場合よりも，ルビ除去率の低い除去式を用いることになる．また，f 値を 1.40 から 1.0 刻みに重複なく分割すると，1.65 から 1.70 の区間で，2 段階クラスタリングによる除去式よりも低い除去率の 1.60–1.70 の除去式を用いることになるため，重複なく 1 段階でクラスタリングを行うよりも，2 段階でクラスタリングを行う方が有効であることが分かる．

この結果より，f 値が
・1.65 以下なら，式 (2)
・1.65–1.71 なら，[1.65:1.75] の除去式
・1.71–1.80 なら，[1.7:1.8] の除去式
・1.8 以上なら，[1.75:1.85] の除去式
で，ルビを除去でき，出版者・時代という人手に頼った分類よりも効率的にルビを除去できることが分かる．図 4 は除去式 (2) を適用し，ルビを除去したものである．この活字データによるクラスタリングは，読み込んだ行の画像から直接得られる特徴値を用いることから，出版者・時代という外部からの付加情報を用い，手動で分類する方法に比べ，効率良くルビを除去することができる．

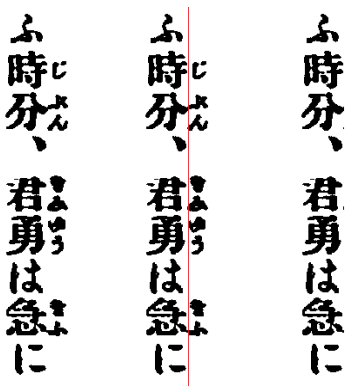


図 4　f 値が 1.65 以下の式を適用した行の一部
Fig. 4　The original image, the line by formula (2) and the ruby character removal result.

4.　有効性の検証

提案の分類方法を用いて生成されたルビ除去式の有効性を調べるため，除去式を用いてルビ除去の実験を行う．

4.1　実験条件

本研究は，近代書籍全般を対象としたものであるが，近代書籍の数は膨大であり，それらすべてを実験対象とすることは困難である．そこで，近代デジタルライブラリで公開されている約 34 万冊を対象として実験を行う．まず，これらの書籍の中でルビの付いている書籍の数を調べる．

近代デジタルライブラリでは，書籍を分野ごとに分けている．たとえば，哲学（8,153 冊），歴史（32,826 冊），社会科学（105,861 冊），言語（11,125 冊），文学（36,698 冊）など全部で 10 に分類されている．それぞれの分野ごとに 1%の書籍をランダムに選び，ルビの有無を調べる．結果，ルビのある割合が 5%以下の分野が 4 つ，およそ 10%前後の分野が 2 つである．そのほか総記と言語で約 20%，哲学でおよそ 50%，文学でおよそ 80%の書籍にルビが存在する．この結果から，近代デジタルライブラリの書籍の中でルビのある書籍の総数は 64,304 冊と推定される．

このルビのある書籍の中から信頼度 95%，誤差 5%で標本数を求めると，およそ 1,537 冊となる．1,537 冊を分野ごとの割合に分けると，哲学で 148 冊，文学で 166 冊などとなる．分野ごとにルビのある書籍を所定数ランダムに選び出し，その書籍から 1 行切り出しルビ除去の実験を行う．1 冊の書籍では，当然同じフォントを使用しており，ルビの特徴も同じであるため，今回の実験ではルビが複数ついている行を 1 行切り出す．実験では，行は分野に関係なく f の値だけで分類し実験を行い，さらに判別分析法を用いた黒画素射影ヒストグラムによる手法との比較も行う．除去の成否は目視である．通常，有効性の判断は認識率によって判断されるべきであるが，近代書籍の文字認識においては，既存 OCR を用いた場合，認識率が低く実用性はない．3 章で使用した行を対象に，e.Typist [9] と読取革命 [10] を用い認識させた結果，ルビを含む行の認識率は e.Typist で 48.9%，読取革命で 33.2%，ルビ除去後の認識率は e.Typist で 83.2%，読取革命で 81.4%である．実用性のない既存 OCR を用いて，有効性を検証することはできない．また，手書き文字認識手法による近代書籍のテキスト化は，現在研究中であり，ルビ除去の検証実験に用いることは不適切であると考える．そのため，ルビ除去の成否は目視とする．

4.2　結果

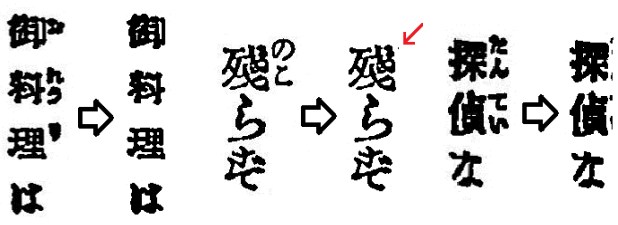
f の値が 1.65 以下の行は 853 行，1.65–1.71 は 245 行，1.71–1.8 は 366 行，1.8 以上は 73 行である．ルビ除去の結果を**表 3** に示す．

すべての f 値グループで提案手法の除去率は 90%以上の除去率となり，1.65 以下と 1.65–1.71 では，95%を超えており，非常に良好な結果である．それに対し，f 値が 1.8 以上のグループでは，提案手法とヒストグラム法を比較した場合，除去できた行数は 1 本しか異ならないという結果となっている．f 値が大きいということは，ルビを含めた幅が高さに比べ，非常に大きいことを意味する．活版印刷の場合，ルビは通常親文字の活字の縦横半分の大きさの活字であることが多いが，活字の文字の周りの余白部分の大きさが活版によって異なり，これが f 値の違いとなる．余白が大きければ，3 章で求めた活字の高さは小さく，またルビを含めた幅余白分だけ大きくなり，結果 f 値は大きくなる．つまり，f 値が大きいものは，親文字・ルビともに余白が大きく，親文字とルビの近接度が低いものが多いと考えられる．そのため，判別分析法を用いた黒画素射影ヒストグラムによる除去法とほとんど差のない結果となっている．通常，親文字とルビの近接度が低い場合，ヒストグラム法で良好な結果が得られると考えられるが，ルビが少ないと判別分析法では正しい閾値を求められないことがある．これは，ルビの少なさから大きな山の部分ができず，ヒストグラムの谷の部分が明瞭に出ないため，正しい閾値を求めることができないからである．そのため約 90%という除去率となっている．

ルビの除去に成功した例と失敗した例を**図 5** に示す．図 5 (1)，(2) は成功例である．(2) では，右上にわずかにルビの一部である黒画素が残っているが，非常に小さいものであり，ルビの除去に成功したといえる．図 5 (3) は失敗した例である．ルビの一部であったと容易に推測される大きな黒画素の塊が残っている．そのため，これは失敗となる．

表 3　f 値による手法とヒストグラムによる手法のルビ除去率（%）
Table 3　Removal success rate of the histogram and the proposal method.

f 値	提案手法	ヒストグラム
[-:1.65]	96.2	74.3
[1.65:1.71]	95.5	82.4
[1.71:1.8]	93.9	89.1
[1.8:-]	91.8	90.4



(1) 成功例　(2) 成功例　(3) 失敗例

図 5　ルビ除去の成功例と失敗例
Fig. 5　The example of success and failure of ruby removal.

そこで，f 値 1.8 以上の場合のルビ除去法として，外接矩形による手法を用いる．著者らの先行研究 [6] で，ルビ除去に適用可能な文字切り出しにおける既存手法として，外接矩形を用いる手法を紹介した．近代書籍のルビは親文字と近接度が高く，連結しているものが多いため，親文字とルビが 1 つの矩形となり，f 値 1.8 以下の行ではルビの除去率は，ほぼ 0%である．しかしながら，f 値 1.8 以上の場合，ほとんどの親文字とルビが離れている．そのため，外接矩形を用いる手法で，親文字とルビで別々の矩形となり，ルビを分離できると考える．まず縦・横・斜めの 8 方向に連結した黒画素部分にラベリング処理を行い，外接矩形を求める．次に矩形範囲が重複している矩形を統合し，複数に分割されることが多い文字を 1 つの矩形とする．こうすることで，親文字とルビの矩形を求めることができる．次に平均矩形面積を出し，その平均値より大きいものを主たる行の文字であるとする．主たる行のそれぞれの文字の矩形の中心を求め，それを平均し，主の行の縦の中心線とする．ここで，文字の縦横比率を 1 : 1 と仮定し，中心線は文字の幅の半分の位置にあると推定しているので，ルビの矩形の中心は中心線よりも右に平均の高さの 1/2 以上離れていると考えられる．そのため，中心線から左へ 3 章で求めた平均の高さの 1/2 以上離れた位置に矩形の中心があるものをルビであると判断し，除去する．結果は，73 本中 71 本の除去に成功し，除去率はおよそ 97.3%である．失敗したものは，句読点が一部除去されてしまったことによるものである．

この結果から，f 値が 1.8 以下であれば，除去率は平均で 95.2%となり，提案手法は有効な手法であるといえる．また f 値が 1.8 以上では，外接矩形を用いることでルビの除去率は 97.3%となり，ルビの分離が可能である．

出版者・時代という分類による除去式の生成では，98%前後の除去率であったが，これは書籍に付加された外部情報をもとに手動で分類しなければならず，実現不可能である．それに対し，提案手法は活字画像から直接データを求め，それをもとにクラスタリングするため自動で行うことができ，提案手法は近代書籍のルビ除去に有効であるといえる．

5.　まとめ

本論文では，活字データの分類を用いた進化計算による近代書籍からのルビ除去手法を提案した．本手法を用いることにより，現在の書籍を対象としたルビ除去手法には適さない近代書籍において，ルビを自動で除去することができ，近代書籍の自動テキスト化が進むことが期待される．

提案手法では，画像データからルビのついている活字の高さとルビを含む幅を求める．そして，それらを用いたア

スペクト比をもとに分類し，遺伝的プログラミングを用いてルビ除去式を生成する．結果，教師データから求まった4つの除去式によって，目標画像との一致率は99%を超えている．これらの式を使い，近代デジタルライブラリで公開されている書籍画像から自動でルビを除去する実験を行ったところ，ルビを含むアスペクト比で1.8以下であれば，提案手法により平均95%を超えるルビ除去率となった．また，1.8以上の場合は外接矩形を用いることでルビの分離が可能であり，ルビの除去率はおよそ97%である．本手法は，著者らの先行研究 [6] の手法とは異なり，読み込んだ活字データを使い自動で分類するため，近代書籍のためのルビ除去手法として，非常に有効である．

今後は，レイアウト解析が重要であると考える．近代書籍を既存のレイアウト解析手法によって解析した場合，精度は非常に低く実用性に乏しい．近代デジタルライブラリで公開されている自然科学や技術などの分野では，文章と図や表などが複雑に配置されているページも多く，近代書籍の自動テキスト化のためには，文章や図・表をそれぞれ正確に切り出さなければならない．この問題を解決するために，近代書籍に特化したレイアウト解析技術が必要であると考える．

参考文献

[1] Ishikawa, C., Ashida, N., Enomoto, Y., Takata, M., Kimesawa, T. and Joe, K.: Recognition of Multi-Fonts Character in Early-Modern Printed Books, *Proc. 2009 International Conference on Parallel and Distributed Processing Technologies and Applications* (*PDPTA 2009*), Vol.II, pp.728–734 (2009).

[2] 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, *Proc. 2011 International Conference on Parallel and Distributed Processing Technologies and Applications* (*PDPTA 2011*), Vol.II, pp.727–732 (2011).

[3] 国立国会図書館全文テキスト化実証実験報告書 (online), 入手先 ⟨http://www.ndl.go.jp/jp/aboutus/digitization_fulltextreport.html⟩（参照 2014-07-18）.

[4] 福尾真実，高田雅美，城　和貴：同一出版者の近代書籍に対する漢字認識評価，情報処理学会研究報告，Vol.2012-MPS-90, No.26 (2012).

[5] 曹　宇，佐藤匡正：文字寸法の違いに着目したOCR認字率の改善法，電子情報通信学会技術研究報告SS，ソフトウェアサイエンス，Vol.100, No.678, pp.17–22 (2001).

[6] 粟津妙華，高田雅美，城　和貴：遺伝的プログラミングを用いた近代書籍からのルビ除去，情報処理学会論文誌数理モデル化と応用，Vol.6, No.2, pp.53–62 (2013).

[7] 伊庭斎志：遺伝的プログラミング入門，東京大学出版会 (2001).

[8] 齋藤堯幸，宿久　洋：関連性データの解析法—多次元尺度構成法とクラスター分析法，共立出版 (2006)

[9] e.Typist v.15.0 (online), available from ⟨http://mediadrive.jp/products/et/⟩ (accessed 2014-07-18).

[10] 読取革命 Ver.15 (online), 入手先 ⟨http://panasonic.co.jp/pstc/products/yomikaku/⟩（参照 2014-07-18）.

付　　録

A.1　f 値による各区間で生成された除去式

1.40 以下

$y = (\text{高さの平均値} + 2) + 4$

1.35–1.45

$y = 4 + \text{高さの平均値} - (6/3 + 4)$

1.4–1.5

$y = (2 + 2 \times 2) + \text{高さの平均値}$

1.45–1.55

$y = \text{高さの平均値} + 5 + 1$

1.5–1.6

$y = 5 \times 2 - 4 + \text{高さの平均値}$

1.55–1.65

$y = 10/2 - 3 + (\text{高さの平均値} + (6/3 + 2))$

1.6–1.7

$y = 3 + 1 + \text{高さの平均値} + 2$

1.65–1.75

$$y = (abs((\text{高さの平均値} + abs(\sin((2 \times \pi \times x/ (((((abs(\text{高さの平均値})/(\sin((2 \times \pi \times x/ ((((\text{高さの平均値} \times \text{高さの平均値}) \times ((x/(abs(\text{高さの平均値}) - (x/9))) + 3)) /2)) - \pi)) + 2)) \times x)/2)) - \pi))))) + 2)$$

1.7–1.8

$$y = (8 - ((\text{高さの平均値}/8) - (\text{高さの平均値} - (5/(\cos((2 \times \pi \times x/(((8 \times 5)/2)) - \pi/2)) - (\text{高さの平均値} - ((4/(6 \times \cos((2 \times \pi \times x/ ((\cos((2 \times \pi \times x/(((8 \times 5)/2)) - \pi/2))/2)) -\pi/2))))/\text{高さの平均値})))))))$$

1.75–1.85

$$y = (1 + (\text{高さの平均値} + (((\sin((2 \times \pi \times x /(((\text{高さの平均値} + (1 + (6/2)))/2)) - \pi)) +((\sin((2 \times \pi \times x/(\text{高さの平均値} + ((3/2))) -\pi)) + \sin((2 \times \pi \times x/(\text{高さの平均値} + (((7/2)/2))) - \pi)))/3))/3) + (3/2))))$$

1.80 以上

$$
\begin{aligned}
y = ((&\sin((2\times\pi\times x/((\text{高さの平均値}+\sin((2\\
&\times\pi\times x/((\text{高さの平均値}+(\sin((2\times\pi\times x/\\
&((\text{高さの平均値}+(\sin((2\times\pi\times x/\\
&((\text{高さの平均値}+(x/8))/2)-\pi))+5))/2)\\
&-\pi))+3))/2)-\pi))))/2)-\pi))+(((\sin(\\
&(2\times\pi\times x/((\text{高さの平均値}+\text{高さの平均値})\\
&/2)-\pi))+1)/3)+\text{高さの平均値}))\\
&-\sin((2\times\pi\times x/((\text{高さの平均値}+(x\\
&\times abs((x/8))))/2)-\pi)))
\end{aligned}
$$

粟津 妙華　（学生会員）

2012 年奈良女子大学理学部情報科学科卒業．2013 年同大学大学院人間文化研究科情報科学科修士課程修了．修士（理学）を同大学より取得．2013 年同大学院人間文化研究科複合現象科学専攻博士後期課程進学，現在に至る．パターン認識，機械学習に関する研究に従事．

高田 雅美　（正会員）

2004 年奈良女子大学大学院人間文化研究科複合領域科学専攻修了．博士（理学）を同大学より取得．2004 年独立行政法人 JST 戦略的創造研究推進事業において，京都大学大学院情報学研究科にて委嘱研究員．2006 年奈良女子大学大学院人間文化研究科助手．2007 年奈良女子大学大学院人間文化研究科助教．2013 年奈良女子大学理学部講師．2014 年奈良女子大学研究院生活環境科学系生活情報通信科学領域講師．数値計算ライブラリの開発，分散メモリ環境を対象とする並列プログラムの開発に関する研究に従事．

城 和貴　（正会員）

大阪大学理学部数学科卒業．日本 DEC，ATR 視聴覚研究所（日本 DEC より出向），（株）クボタ・コンピュータ事業推進室で勤務の後，1993 年奈良先端科学技術大学院大学情報科学研究科博士前期課程入学，1996 年同研究科後期課程修了，同年同研究科助手．1997 年和歌山大学システム工学部講師，1998 年同助教授．1999 年奈良女子大学理学部情報科学科教授．2014 年奈良女子大学研究院生活環境科学系生活情報通信科学領域教授，現在に至る．博士（工博）．情報処理学会論文誌数理モデル化と応用編集委員長．