Vol.2020-CH-123 No.1
2020/6/6

貼り込み形式の資料に対するフォント画像を用いたテキスト検索手法の検討 - 東京大学総合図書館所蔵『君拾帖』を対象として

高橋　大成[1,a)]　中村　覚[1,b)]

概要：近年，深層学習技術の発展とともに様々な言語の文字認識精度が向上しており，画像内の文字列に対するテキスト検索を可能とするサービスが数多く展開されている．このような検索機能は，画像資料が多く含まれるデジタルアーカイブにおいて，資料検索の効率化という点で特に重要である．この検索タスクのうち，本研究では様々な種類の資料が貼り込まれている貼り込み形式の資料を検索対象とする．この形式の資料には，印刷された資料や崩れた毛筆手書きの資料，絵やラベルが含まれ，さまざまな字体が混在しているために文字認識が難しい．本研究では，文字認識を行わずに様々な字体に対応するフォント画像と資料画像の類似度を考えることで資料画像内の文字列に対するテキスト検索を行う手法を提案する．そして，貼り込み形式の資料に対するテキスト検索において，フォント画像を用いることの有用性を検討する．

1. イントロダクション

図書資料や博物資料をデジタル化して公開するデジタルアーカイブの構築や提供が盛んに進められている．しかし，これらの多くは画像データであるため，メタデータが十分に付与されている場合もあるが，その内容に対する検索の困難さは依然として課題である．

このような資料の分類や解読や検索に対する取組みとして，John Resig が開発した浮世絵の画像検索サービス Ukiyo-e Seach[1] や，資料中から自動的に切り出された画像や図版に基づく検索を可能とする国立国会図書館の次世代デジタルライブラリー [2]，市民参加型の翻刻プロジェクトであるみんなで翻刻 [3] などがある．画像データ内のキーワード検索においては寺沢ら [4] によってワードスポッティングと呼ばれる手法が研究されている．

さらに，画像内の文字列に対する検索ニーズに対して，OCR に関する研究が進められている．奈良文化財研究所と東京大学史料編纂所の共同開発した木簡，くずし字解読システム MOJIZO[5]，江戸時代のくずし字を機械学習を用いて高精度で認識できる KuroNet くずし字認識サービス [6] 等がある．また，中国の古代文字に対する認識を行う既存研究として李ら [7] のフォント画像を利用するものもある．李らの研究では中国の古代文字 10 種について，資料中の文字と対応するフォント画像との特徴ベクトルの類似度を元に文字認識可能かを検討しているが，書体が 1 種類であることと資料から文字を 1 文字単位で切り出せることが前提となっている．この検索タスクのうち，本研究では様々な種類の資料が貼り込まれている貼り込み形式の資料を検索対象とする．この様々な書体で書かれており，文字を切り出す事が難しい貼り込み形式の資料に対してフォント画像を利用した類似度計算を行う．

2. 提案手法

2.1 貼り込み形式資料の特徴

本研究で対象としている『君拾帖』は貼り込み形式の資料であり，様々な印刷物や手書きの資料など同一の本に含まれている．『君拾帖』に，現在一般的に使われている OCR ツールを適用した例を図 1，2 に示す．『君拾帖』の画像は 文字認識が容易になるように前処理として二値化を行なっている．

矩形が部分が文字または文字列として認識している部分を示す．比較的読むことが容易な印刷部分の精度も十分でなく，毛筆で書かれている部分に関しては文字として検出することすらできていないことがわかる．また，文字切り出しにおいて正しく切り出すことが出来ずに異なる文字として認識してしまう可能性がある．本研究でもキーワード検索において OCR の利用を検討したが，統一的でない字

[1] 東京大学
The University of Tokyo
[a)] takahashi@eidos.i.ic.u-tokyo.ac.jp
[b)] nakamura.satoru@mail.u-tokyo.ac.jp

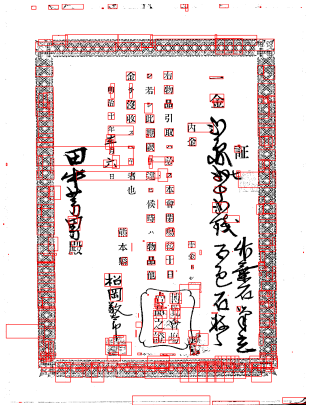

図 1　Tesseract[8] による認識

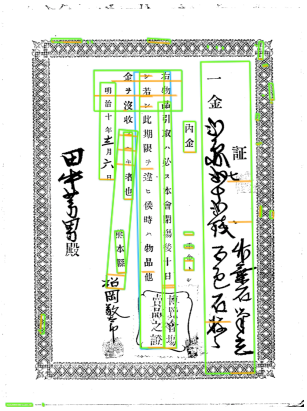

図 2　Google Cloud Vision API による認識

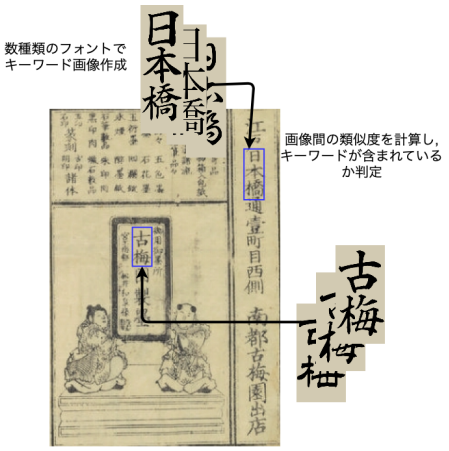

図 3　提案手法の概略

図 4　草書体の「田中」

図 5　行書体の「田中」

図 6　草書体の「東京」

図 7　図 4 の細線化

図 8　図 5 の細線化

図 9　図 6 の細線化

体が含まれている資料に対して汎用的な方法でテキスト認識をすることは難しく，それぞれの字体に特化した文字認識が必要であると考えた．そこで本研究ではテキスト認識を行わないキーワード検索手法を検討する．

2.2　提案手法

文字認識を用いず，様々な字体が含まれるデジタルアーカイブに対するキーワード検索手法として検索対象キーワードのフォント画像と対象資料の一部の画像との特徴量から計算される類似度による検索手法を提案する．一つの資料内に様々なサイズの文字列が含まれること，様々な画像サイズの資料に対応すること等を検討し，シンプルなテンプレートマッチングではない手法を提案する．提案手法の概略を図 3 に表す．

3.　実験及び評価

3.1　実験資料

『君拾帖』第 15 帖の切り出し済み資料を本実験の対象資料とした．またフォントとしては，衡山毛筆フォントの楷書，行書，草書と Ordano 明朝 GSRR を用いた．

3.2　前処理

本実験の画像の前処理として画像の二値化と文字の細線化を行なった．二値化とは入力画像を黒のピクセルと白のピクセルのみの画像に変換するという手法である．本研究では OpenCV を用いて大津の二値化 [9] のアルゴリズムで二値化を行なった．また，細線化には Zhang-Suen の細線化 [10] のアルゴリズムを用いた．例として二値化処理を行ったフォント画像を図 4，5，6 に示す．また，二値化処理と細線化処理をどちらも行なったフォント画像を図 7，8，9 に示す．

3.3　実験 1（特徴量選択の実験）

提案手法における類似度計算に用いる特徴ベクトルの選定に向け，最初に複数のフォント画像どうし類似度を 5 種類の特徴ベクトル用いて算出した．この実験では同じキーワードで異なる書体のフォントの画像間の類似度と，異なるキーワードで同じフォントの画像間での類似度を計算した．これにより，書体が異なるが同じ文字である画像間の類似度が高くなるような特徴量を選定する．この時，フォント画像には二値化処理または二値化処理と文字の細線化の 2 種類の前処理を用いる．

このフォント画像同士の類似度を計算するに当たって，類似度として次の 4 種類を用いて実験を行なった．ディープラーニングを用いない代表的かつ OpenCV によって実装が可能な BRISK，ORB，AKAZE と比較的シンプルな

表 1　画像を二値化した場合の類似度

	草書体「田中」 草書体「東京」 二値化処理	草書体「田中」 行書体「田中」 二値化処理
BRISK	585.824979	609.656295
ORB	48.339535	50.562791
AKAZE	98.981579	103.076316
VGG16	0.34004748	0.632908
ResNet50	0.1294196	0.4848013

表 2　画像を細線化した場合の類似度

	草書体「田中」 草書体「東京」 細線化処理	草書体「田中」 行書体「田中」 細線化処理
BRISK	526.855005	534.561362
ORB	42.533019	42.433962
AKAZE	100.526316	99.368421
VGG16	0.461798	0.682324
ResNet50	0.717608	0.703838

アーキテクチャを持ち，分類タスクでの精度が高いニューラルネットワークモデルの VGG16 と ResNet50 を用いた．

1 つ目は BRISK(Binary Robust Invariant Scalable Keypoints) の特徴点マッチングにおける全ての特徴点間の距離平均を類似度の指標とした．

2 つ目は ORB(Oriented FAST and Rotated BRIEF) の特徴点マッチングにおける全ての特徴点間の距離平均を類似度の指標とした．

3 つ目は AKAZE(Accelarated KAZE) の特徴点マッチングにおける全ての特徴点間の距離平均を類似度の指標とした．

4 つ目は現・産業技術総合研究所の手書きの教育漢字と平仮名が 979 クラスに分かれているデータセット「ETL-8b Character Database」[11] を用いて学習させた VGG16[12] の出力層に対する入力を特徴ベクトルとしてコサイン類似度を指標とした．

5 つ目は「ETL-8b Character Database」で学習させた ResNet50[13] の出力層直前のドロップアウト層への入力を特徴ベクトルとしてコサイン類似度を指標とした．

VGG と ResNet の学習においては入力画像サイズを 72 × 72 とし，どちらも 75 エポック学習させた．

3.4　特徴量の評価

5 つの特徴量でフォント画像間の類似度を表 1，2 に示す．BRISK，ORB，AKAZE は特徴点間の距離平均の値を算出しているため，値が低いほど類似度が高いことを意味する．また，VGG16 と ResNet50 に関しては特徴ベクトルのコサイン類似度を求めているため，値が高いほど類似度が高いことを意味する．

BRISK，ORB，AKAZE 特徴量においては，特徴量がコーナーなどで取られることがあるので，草書体のフォント同士の類似度の値が高くなっていると考えられる．VGG16 と ResNet の中間層出力はほとんどの場合で同じ文字どうしの方が類似度が高くなっていた．BRISK，ORB，AKAZE 特徴量においてのメリットは回転やスケールサイズに関してロバストなことであるが，本実験では同じサイズで回転していないフォント画像を使用するため，その利点を十分に生かすことはできない．また，提案手法を適用する『君拾帖』においても回転している資料が少ないため，以降では VGG16 と ResNet50 を採用する．

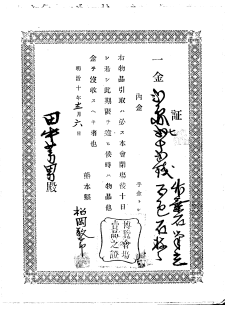

図 10　二値化した『君拾帖』第 15 帖 1 ページ

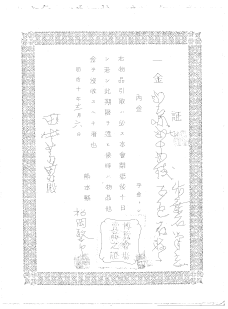

図 11　細線化した『君拾帖』第 15 帖 1 ページ

3.5　実験 2（前処理の差異）

VGG16 と ResNet50 から抽出した特徴ベクトルを用いて，図 10，11 に示した前処理済みの『君拾帖』第 15 帖 1 ページ目に貼り込まれている資料に手法を適用し，キーワードのフォント画像との類似度が最大となる矩形を抽出した．また『君拾帖』の資料のほとんどが縦書き資料であるため，本実験では縦書き文字に限定して実験を行なった．

対象となるキーワードは「明治」と「田中」とした．『君拾帖』には縦書きでも文字と文字の間にスペースが存在している資料が多く存在している．そのため，キーワードの文字間に空白をいれた「明　治」,「田　中」をフォント画像にしたものも用意した．

対象資料である『君拾帖』において様々な形態の貼り込み資料が存在しており，正確に文字列を切り出すことができない．そのため，本実験ではフォント画像と似ている部分を画像中から探す方法に関して，資料画像上で窓画像を作って走査していき，窓画像との類似度を計算し最近傍を

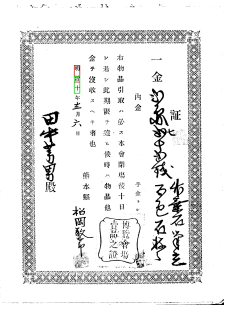

図 12　二値化画像で VGG16 を用いた類似度検索

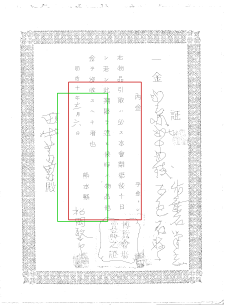

図 13　細線化画像で VGG16 を用いた類似度検索

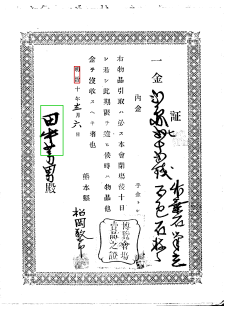

図 14　二値化画像で ResNet50 を用いた類似度検索

探索するという単純な方法とした．窓画像資料画像の縦と横の画素数のうち小さい方の 40 分の 1 の画素数ごとで窓画像の縦と横の長さを変更していく．また本実験の縦書き 2 文字の「明治」と「田中」をキーワードに検索しているので, 空白を入れた縦書き 3 文字「明　治」と「田　中」を考慮し，窓サイズの縦横比が縦長で 9:5 から 16:5 の間になる場合のみの類似度を計算した．

二値化したフォント画像は二値化した資料画像の各部分に対して類似度計算を行い，細線化したフォント画像に対しては細線化した資料画像の各部分に対して類似度計算を行なった．「明治」をキーワードとした場合に全種類のフォントの内で最も類似度の高かった部分を赤い矩形で，「田中」に関して最も類似度の高かった部分を緑の矩形で示したものを図 12，13，14，15 に示す．

これらの結果から，細線化処理を前処理として行なった場合は類似している部分を見つけられていないということがわかる．フォント画像の方が黒の領域が多く，エッジによる概形などの特徴を抽出できるためなどと考えられる．

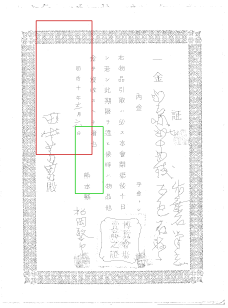

図 15　細線化画像での ResNet50 を用いた類似度検索

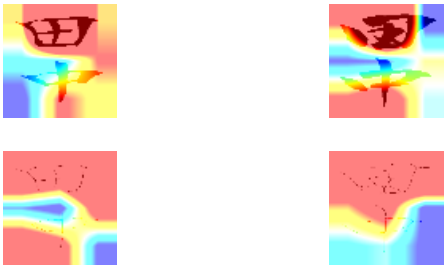

図 16　二値化と細線化を行なった「田中」の中間層可視化

図 16 に二値化と細線化をそれぞれ行なった 2 書体の「田中」を入力とした ResNet50 の中間層を Grad-CAM[14] により可視化したものを示す．

ヒートマップが赤いほどその部分が最終出力に影響することを表している．二値化処理を行なっている画像に関しては，主に「田」の字自体を注目していることがわかる．それに比べて細線化処理を行なっている画像に関しては，「田」の文字の周りの部分も注目している．これはどこに注目して分類を判断するかが正確に決定出来ていないことを示していると考えられる．したがって，「田」の文字を例にすると細線化処理を行なった場合は文字の周りに少しでも黒いピクセルが存在してしまうと抽出される特徴ベクトルが大きく変わり得る．それによって，細線化処理を行なった画像を入力とした時に同じ文字の場所を類似度が高いと判定出来ないと考えられる．

その結果，細線化処理を行うよりも二値化したフォント画像で形状を維持させたまま類似度を計算する方が良い結果を得られた．よって，提案手法を実際の複数資料に適用する場合は前処理として画像の二値化処理のみ行うこととした．

3.6　複数資料への適用と評価

本提案手法の評価として，先行研究 [15] で手動アノテーションが行われていた『君拾帖』15 帖の一部から文字が掠れていて明らかに読むことが出来ないのものをのぞき，40 枚の資料画像を抽出し二値化処理を行なった後，VGG16 と ResNet50 の中間層出力を特徴ベクトルと手法を適用した．検索を行うキーワードとして，時代としてよく現れる「明治」，署名や多数の資料の中に現れる「田中」，地名としてよく現れる「東京」を選択した．そして類似度の高い

表 3　VGG16 を用いた「明治」検索

上位 (件数)	適合率	再現率
3	0.667	0.133
6	0.5	0.2
9	0.667	0.4
12	0.667	0.533
15	0.667	0.667

表 4　VGG16 を用いた「田中」検索

上位 (件数)	適合率	再現率
3	1.0	0.273
6	0.667	0.364
9	0.667	0.545
12	0.583	0.636
15	0.533	0.727

表 5　VGG16 を用いた「東京」検索

上位 (件数)	適合率	再現率
3	0.333	0.071
6	0.167	0.071
9	0.222	0.143
12	0.333	0.286
15	0.267	0.286

表 6　ResNet50 を用いた「明治」検索

上位 (件数)	適合率	再現率
3	1.0	0.2
6	0.667	0.267
9	0.556	0.333
12	0.5	0.4
15	0.467	0.467

表 7　ResNet50 を用いた「田中」検索

上位 (件数)	適合率	再現率
3	1.0	0.273
6	0.833	0.455
9	0.667	0.545
12	0.667	0.727
15	0.533	0.727

部分があった順番に資料を出力させ，実際に含まれているかを判定し評価を行なった．資料 40 点中「明治」を含む資料が 15 点，「田中」を含む資料が 11 点，「東京」を含む資料が 14 点であった．これを各キーワードに対する正解データとして，検索の上位 3 件，上位 6 件，上位 9 件，上位 12 件，上位 15 件において適合率と再現率を計算した．それを表 3，4，5，6，7，8 に示す．

「東京」に関しての適合率と再現率が著しく低い．これは「東京」を含んでいる資料の多くが縦書きで 20 行以上で書かれた資料であるからだと考えられる．本実験で行なった類似度を計算するための窓画像の走査方法では正確に「東京」という文字を含めた窓画像を生成することが出来なかったことが原因として考えられる．

表 8　ResNet50 を用いた「東京」検索

上位 (件数)	適合率	再現率
3	0.333	0.071
6	0.333	0.143
9	0.556	0.357
12	0.5	0.429
15	0.4	0.429

また，「明治」という文字の検索適合率が高かった要因としては「明治」という文字は公的な文書が貼り込み資料となったものに多く書かれており，このような文書では活版印刷がなされていることが多い．したがって，毛筆によって書かれた「明治」が少なく，活版印刷の「明治」が多かったため，他のキーワードと比較的にフォントとの類似度が高くなったと考えられる．

また，本実験の走査方法では長い時間を要してしまうため，『君拾帖』内の様々な形式の資料を正確に行切り出しが可能な手法や，特徴ベクトルとの最近傍を計算するのではなく，近似最近傍探索を行うことによって実用的な高速化や精度の向上が可能であると考えられる．

4. まとめ

デジタルアーカイブで公開されている画像データに対するキーワード検索に関する問題を指摘し，文字のフォント画像を用いた類似度によるキーワード検出手法を提案し，文字フォントが資料内でのキーワード検索に有効であることを示した．そして，本手法を『君拾帖』の貼り込み資料に適用したところ 60%程度の適合率で検索できることがわかった．『君拾帖』には明治と大正時代の様々な字体の資料が貼り込まれているため，この資料に対して有効なキーワード検索が可能な手法であるということは，明治や大正時代に様々な字体で書かれている資料に対する本手法の有用性も示唆される．

今後の課題としては，キーワード検索手法において，窓画像の走査に時間を要する点が挙げられる．古文書に対する行切り出し手法はすでに研究がされている [16] が『君拾帖』のような資料に対しては適用が難しい．今後，『君拾帖』のような資料に対しても適用可能な汎用的な行切り出しが可能になれば，さらに計算時間の短縮や，画素密度の推定など他の特徴量を考えることができるため，検索精度の向上が期待できる．

また，フォントに関連しても近代書籍用フォント生成に関する研究 [17] がされており，歴史的な時代にあったフォントが生成できるようになれば本キーワード検索手法の精度や有効性の向上を期待できる．

参考文献

[1] Resig, J.: Japanese Woodblock Print Search - Ukiyo-e Search, https://ukiyo-e.org/. (Accessed on

09/16/2019).
[2] Aoike, T., Satomi, W., Kawashima, T. and Diet, N.: 資料画像中の挿絵領域の自動抽出及び画像検索システムの実装，pp. 97–102 (2018).
[3] 国立歴史民族博物館. 東京大学地震研究所．京都大学古地震研究所．：みんなで翻刻- MINNA DE HONKOKU，https://honkoku.org/. (Accessed on 05/12/2020).
[4] 寺沢 憲吾. 長崎健. 川嶋稔夫.：古文書画像を対象としたワードスポッティング，画像の認識・理解シンポジウム (MIRU 2005)，Vol. 9（オンライン），DOI: 10.1007/s10032-006-0027-8 (2005).
[5] 奈良文化財研究所：木簡・くずし字解読システム:解析:，https://mojizo.nabunken.go.jp/. (Accessed on 09/16/2019).
[6] ROIS-DS 人文学オープンデータ共同利用センター：KuroNet くずし字認識サービス，http://codh.rois.ac.jp/kuronet/. (Accessed on 02/01/2020).
[7] Biligsaikhan, B.: 古代文字検索のためのフォントからの字形特徴量の抽出および活用可能性の検討 (2019).
[8] Google: tesseract-ocr · GitHub，https://github.com/tesseract-ocr/. (Accessed on 02/07/2020).
[9] 展之大津：判別および最小 2 乗規準に基づく自動しきい値選定法，*The Transactions of the Institute of Electronics and Communication Engineers of Japan*, Vol. 63, No. 4, pp. p349–356 (1980).
[10] Zhang, T. and Suen, C.: A Fast Parallel Algorithn for Thinning Digital Patterns, *Commun. ACM*, Vol. 27, No. 3, pp. 236–239 (online), DOI: 10.1145/357994.358023 (1984).
[11] 電子技術総合研究所：Japanese Technical Committee for Optical Character Recognition, ETL 文字データベース (1973-1984).
[12] Simonyan, K. and Zisserman, A.: Very deep convolutional networks for large-scale image recognition, *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, pp. 1–14 (2015).
[13] He, K., Zhang, X., Ren, S. and Sun, J.: Deep residual learning for image recognition, *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Vol. 2016-Decem, pp. 770–778 (online), DOI: 10.1109/CVPR.2016.90 (2016).
[14] Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D. and Batra, D.: Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization, *International Journal of Computer Vision*, pp. 1–23 (online), DOI: 10.1007/s11263-019-01228-7 (2019).
[15] 中村覚．：IIIF とオープンデータを用いた『君拾帖』内容検索システムの開発，デジタルアーカイブ学会誌，Vol. 3, No. 2, pp. 155–158 (2019).
[16] UMEDA, M. and HASHIMOTO, T.: Character Segmentation and Recognition of Ancient Documents Based on Thinning of Background Region, *IPSJ journal*, Vol. 45, No. 4, pp. 1188–1197 (2004).
[17] Takemoto, Y., Kousaka, K., Ishikawa, Y. U., Takata, M. and Joe, K.: Automatic Font Generator for Early-Modern Printed Books, Vol. 2017, No. 15, pp. 1–6 (2017).