第 19 回画像センシングシンポジウム（SSII2013）インタラクティブ&ショートオーラルセッション リファレンスポイントを用いた情景内文字認識の高速化

松田崇宏† 小林拓也‡ 岩村雅一‡ 黄瀬浩一‡

†大阪府立大学工学部 ‡大阪府立大学大学院工学研究科

E-mail: {matsuda,kobayashi}@m.cs.osakafu-u.ac.jp,{masa,kise}@cs.osakafu-u.ac.jp

Abstract

カメラで撮影した情景画像中の文字を認識し，その中から重要な情報を得られれば非常に有用であると考えられる．それを実現させる手法の一つとして Iwamura らの手法がある．しかし，これをスマートフォン上でリアルタイム動作させることは難しい．本稿では，Iwamura らの手法において，リファレンスポイントを用いて特徴点数を削減し，処理速度を向上させることを考える．提案手法を用いて行った実験では，認識率を落とすことなく，特徴点数を約 30 分の 1 にすることができ，認識処理に必要な処理時間を約 67%減らし，全体の処理時間を約 5%削減できた．

1 はじめに

近年，スマートフォンの普及に伴い，多くのスマートフォン用アプリケーションが開発されている．その一つとして,「写して翻訳」という情景内文字認識アプリケーションがある [1]．このアプリケーションでは，撮影された画像内にある単語や文字を認識することができ，更にその翻訳を自動で行なってくれる．これによりユーザーはわざわざ文字を入力せずに，その単語や文字の意味を知ることができる．このようなアプリケーションでは，認識精度や認識に要する時間はとても重要な要素となる．そこで，いくつかの条件を満たす必要がある．まず，ユーザーは文字を様々な角度から撮影することが考えられるため，射影変換された画像にも頑健でなければならない．次に，文字は直線に配置されていないことも考えられるため，レイアウト変化に対しても頑健でなければならない．また，文字は無地ではなく，色や模様などがついた背景上にあることも考えられる．このような複雑背景上にある文字を認識することは非常に難しく，これは多くの場合，認識結果を低下させる原因となる．「写して翻訳」では，このような角度がついて撮影された画像や，複雑背景上にある文字の認識は難しい．そしてこのようなアプリケーションは，ユーザーを長時間待たせないようにリアルタイムで動作しなければならない．

そこで，それらの条件を満たした情景内文字認識手法の一つとして，Iwamura らの手法がある [2]．Iwamura らの手法では局所特徴というものを用いて文字認識を行なっている．この局所特徴が複雑背景上の文字の認識を可能にしている．この手法は大きく分けて 3 つのステップに分けることができる．まず，文字画像から局所特徴点を検出し，局所特徴量を抽出する．次に，抽出された特徴点とデータベースとのマッチングを行う．最後に，マッチングされた特徴点の配置を用いて文字の認識とその領域検出を行う．この手法で使用される特徴記述子は射影変換に頑健であり，また複雑背景上にある文字の認識にも利用できる．それゆえ，彼らの手法は情景内文字認識において高い認識率を得ることができている．しかし，その処理時間は大きな問題となっている．Iwamura らの手法は，高性能計算機では約 1〜2fps で動作するものの，未だにスマートフォン上での動作は難しい．そのため，その処理時間を削減する必要がある．

本稿では，Iwamura らの手法の処理時間の削減を図る．特徴点のマッチング後，マッチングされた特徴点は多くの冗長な特徴点を含んでいる．それらは，処理時間の増大の原因となっていると考えられる．そこで，それらを取り除くために，リファレンスポイント (RP) [3] を用いる．RP を用いることにより，特徴点がどの程度正しく検出されたのか，その信頼度を知ることができる．それゆえ，信頼度の低い特徴点を捨てることで，文字の認識とその領域検出の処理時間を削減することができると考えられる．

2 局所特徴を用いた文字認識

本節では，Iwamura らの手法について説明する．図 1 の従来手法は Iwamura らの手法の概略図である．前章で既に述べたが，Iwamura らの手法における認識処理は主に 3 つのステップに分けることができる．以下

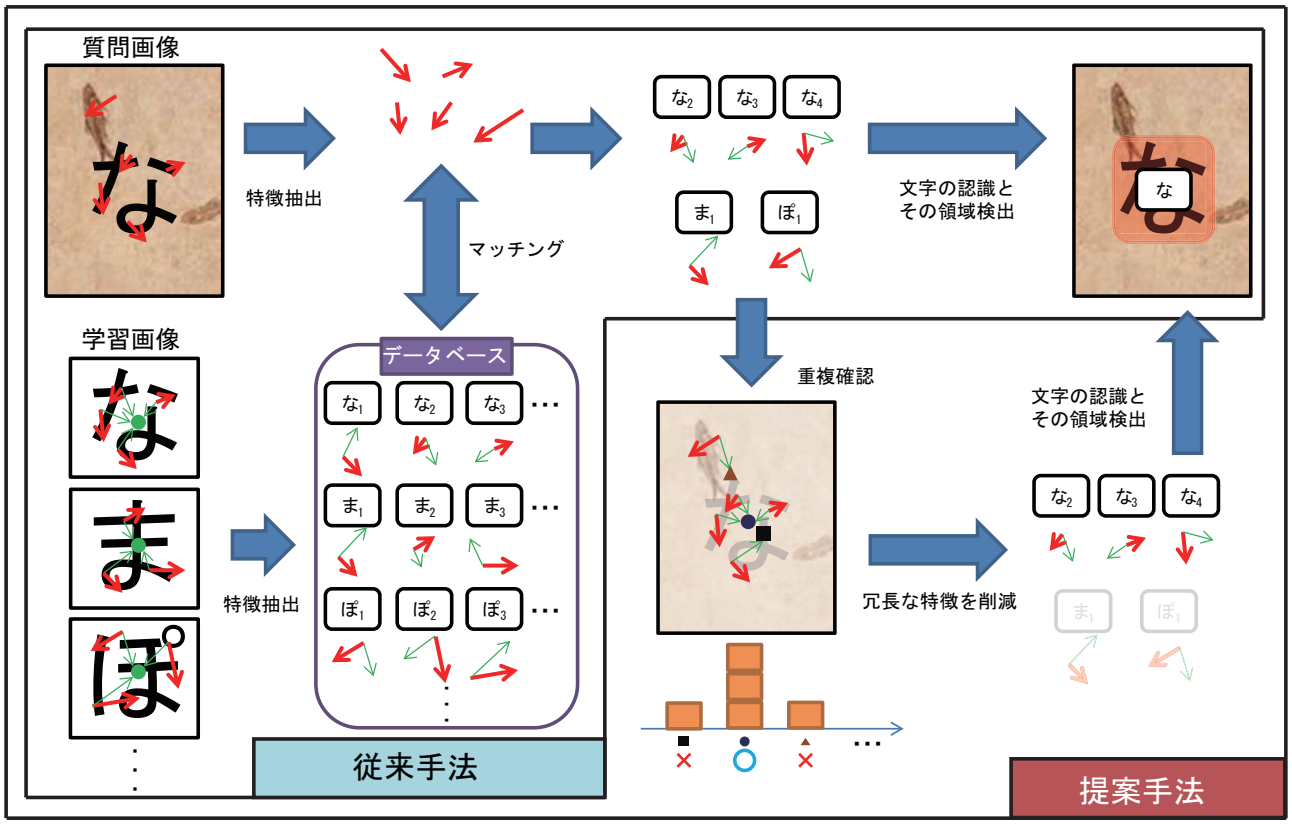


図 1　従来手法である Iwamura らの手法 [2] と提案手法の概略図.

に各ステップについて詳しく述べる.

2.1　特徴抽出

特徴抽出ステップでは，質問画像から ASIFT [4] を用いて特徴点の検出を行い，その点から特徴量を求める．特徴点の検出は，それぞれの画素に対して近傍の画素と画素値の比較を行う．もし画素値が極値であれば，その画素は特徴点とみなされる．特徴点の検出後，各特徴点に対して特徴ベクトルの記述を行う．特徴ベクトルは，特徴点の周辺の画素値から求められる 128 次元のベクトルとなる.

2.2　特徴点のマッチング

抽出されたそれぞれの特徴ベクトルに対して，データベースにある特徴ベクトルとのマッチングを行う．この時，特徴ベクトル同士のユークリッド距離を計算し，その距離が一番小さいものをマッチングの結果とする．距離の計算には，近似再近傍探索である佐藤らの手法 [5] を使用する．この処理の後，抽出されたそれぞれの特徴点に，マッチングした特徴点が持つ文字ラベルを与える．文字ラベルは，データベースの特徴点がどの参照画像，すなわちどの学習画像の文字から抽出されたかを表す．そのマッチングした特徴点の位置と文字ラベルは，次の文字の認識とその領域検出ステップで使用される.

2.3　文字の認識とその領域検出

図 2 に処理の概略を示す．まず，ある特徴点に対して，その周辺に一定の領域を定める．ここで，その領域内にその特徴点と同じ文字ラベルを持つ特徴点が複数個存在している場合，その領域には文字が存在する可能性が高い．しかし，その領域はかなり曖昧であり，文字領域であるとは言えないので，もっと厳密に領域を求める必要がある．次に，領域内にある同じ文字ラベルを持つ任意の 3 点を用いて，射影変換行列を求める．この変換行列を用いて参照画像の文字領域を質問画像に射影することで，正確な文字領域を得ることができる．また，それらの特徴点に対して RANSAC [6] を適用することにより，検出精度を上げることができる．更に，RANSAC を用いることで，検出された文字の信頼度も求めることができ，信頼度が低い文字を取り除くことで，誤検出を大幅に削減することができる．このようにして，文字の認識とその領域検出を同時に行うことができる.

3　提案手法

質問画像から抽出しマッチングを行った特徴点には，冗長な特徴点を多く含む．それらの冗長な特徴点は，質問画像の背景や類似している文字部分などから抽出されたものである．RANSAC を使うことによりこれらの

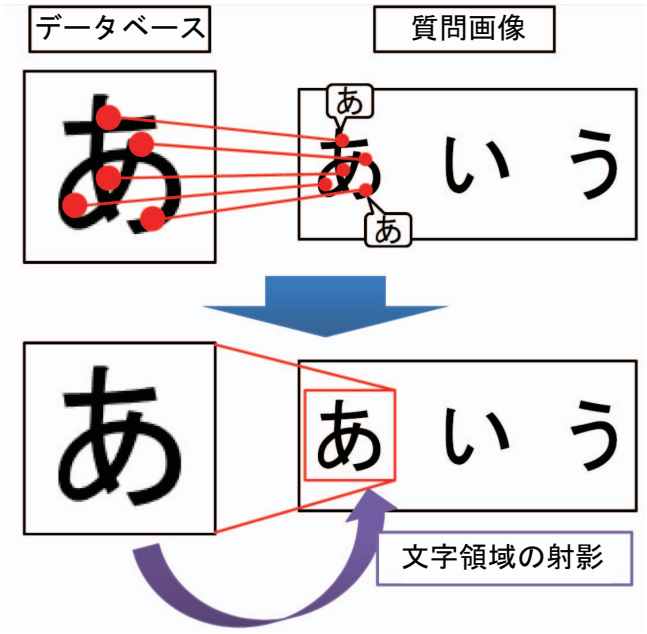
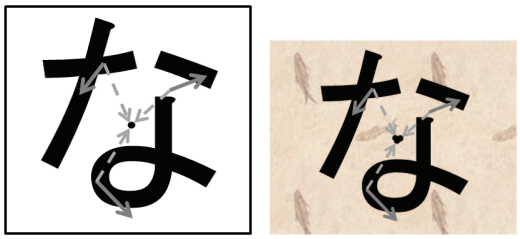


図 2　文字の認識とその領域検出ステップの概略図.



(a) 学習画像　　(b) 質問画像

図 3　RP の概略図．実線が局所特徴，破線が RP の位置を指す.

特徴点による誤検出は回避することはできるが，処理時間の増大は避けることができない．そこで，もし冗長な特徴点を削減することが出来れば，処理時間を削減できると考えられる．本節では，そのような冗長な特徴点をリファレンスポイント (以下 RP) [3] を用いて削減する手法を提案する．

図 1 に Iwamura らの手法を RP を適用した提案手法を示す．また，RP の部分のみを取り出したものを図 3 に示す．学習時には，図 3(a) のように，特徴ベクトルだけではなく，それぞれの特徴点からその文字の中心までの方向と距離も求める．ここで，その特徴点からその方向と距離に位置する点，つまり文字の中心を RP と定義する．もし特徴点が正しく抽出・マッチングされれば，図 3(b) のように，RP は一箇所に密集するはずである．逆に，背景や誤検出によって得られた特徴点の RP は一箇所に密集しない．そこで，RP が t 個以上重

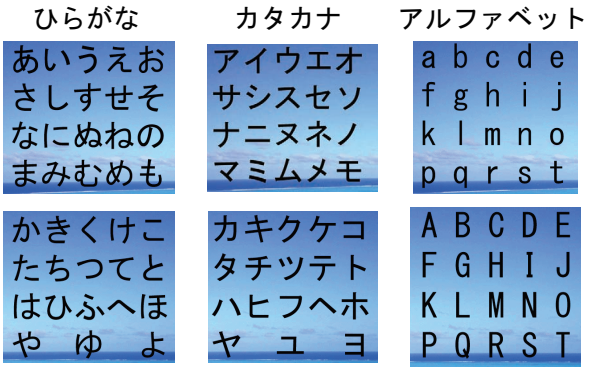


図 4　質問画像.

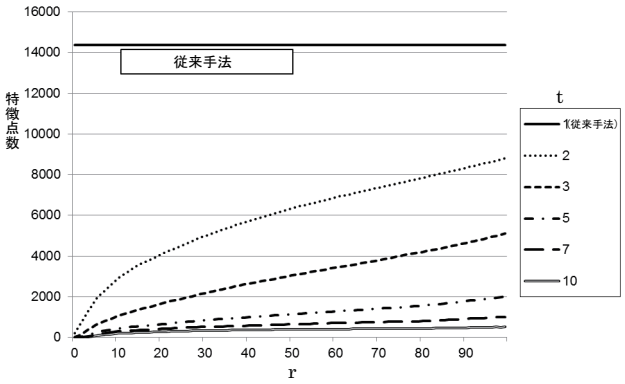


図 5　特徴点数.

なった場合，それらの特徴点を全て残す．逆に，RP が t 個以上重ならなかった場合，その特徴点は捨てる．この処理を全ての特徴点に対して行うことで，最終的に認識に寄与しない特徴点だけを削減することができる．しかしながら多くの場合は，RP はその文字の中心からは幾分ずれてしまい，完全に重なることはほとんどない．これは，画像が変われば得られる特徴点の位置が幾分ずれてしまうためである．そこで，RP 同士の距離が r 以内であれば，それらは重なっているとみなす.

4　実験

提案手法の有効性を検証するために，提案手法における閾値 t と r を変化させて比較実験を行った．本節では，特徴点数と処理時間，認識率の 3 つの結果を示す.

4.1　実験条件

学習画像には，MS ゴシックのひらがな 71 文字，カタカナ 72 文字，漢字 1945 文字，英字 52 文字，数字 10 文字の計 2150 文字を使用した．質問画像には，A4 サイズの紙に MS ゴシックのひらがな，カタカナ，アルファベットを 20 文字ずつ印刷し，それらをカメラで撮影し直した計 60 枚用意した．カメラの解像度は，640 × 480 であった．使用した計算機の CPU は core i5 2.3GHz,

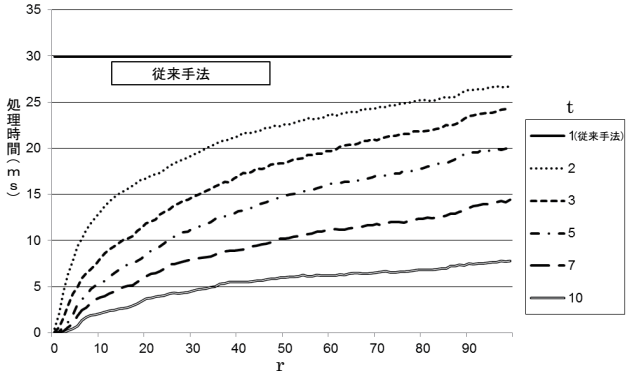


図 6　文字の認識とその領域検出の処理時間.

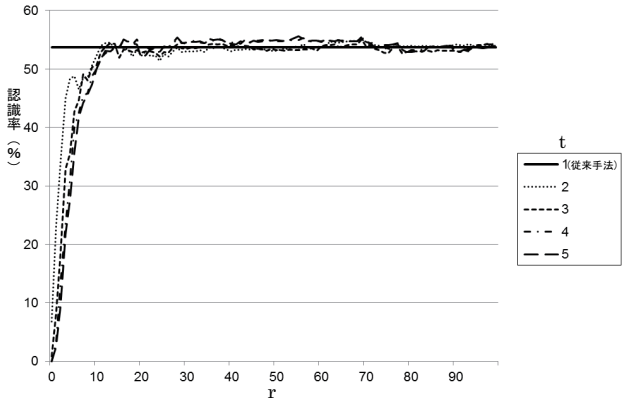


図 7　t が 2 から 5 の時の認識率.

メモリは 6GB であった．また，提案手法においては，t を 2 から 10，r を 0 から 99 の間で変化させた.

4.2　結果

図 5 に特徴点数を示す．図より，提案手法を用いることにより多くの特徴点が削減できていることがわかる．しかし，削減された特徴点の中には冗長な特徴点だけでなく，有効な特徴点も含まれている可能性がある．そのため，この結果だけからはどの閾値の時が最適か，また冗長な特徴点だけを削減できたかどうかは言い切れない.

図 6 に文字の認識とその領域推定ステップにおける処理時間を示す．図より，RP を用いることで全体的に処理時間が削減できていることが明らかである．また，t が大きくなればなるほど，また r が小さくなればなるほど，処理時間を大幅に削減できたこともわかる.

図 7 と図 8 に認識率を示す．グラフが滑らかでないのは，RANSAC がランダムアルゴリズムであり，認識結果が毎回微妙に変わるためである．図 7 より，t が 2 から 5 で r が 15 以上の時，認識率は全て等しいことがわかる．また，図 8 より，t が 5 から 10 まで変化する時，t が大きくなればなるほど認識率が低下していることがわかる.

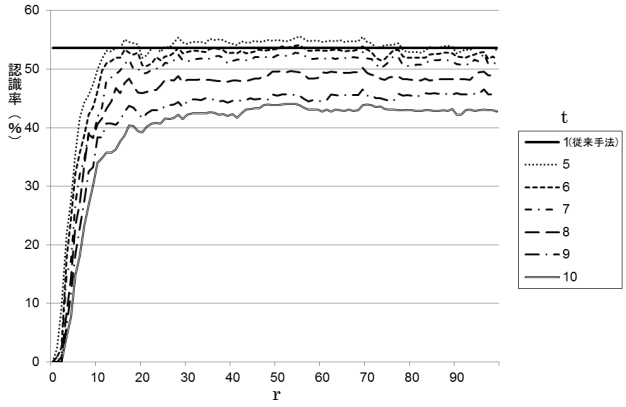


図 8　t が 5 から 10 の時の認識率.

表 1　最良の閾値 (t=5，r=16) のときの結果.

	従来手法	提案手法
認識率	55%	55%
特徴点数	14370	557
文字の検出とその領域検出での処理時間 (RP の処理時間を含む)	30ms	10ms
認識処理全体の処理時間	414ms	391ms

これらの実験結果から，認識率を維持しつつ処理時間を最大限に削減できる閾値は，t=5 かつ r=16 であることがわかる．表 4 は最適な閾値の時の提案手法と従来手法の結果の比較を示す．表より，特徴点数は約 30 分の 1 に削減できたことがわかる．また，特徴点数が削減できたことにより，処理時間も約 3 分の 1 に削減できたことがわかる．しかし，手法全体の処理時間は 5 %程度しか削減できておらず，大きく削減できたとは言い難い.

5　まとめ

本稿では，情景ない文字認識手法である Iwamura らの手法において，リファレンスポイントを用いて冗長な特徴点を削減することで処理時間を削減する手法を提案した．その結果，認識率を落とすことなく，特徴点数は約 30 分の 1 に，また処理時間は 5 %削減することができた．今後の課題としては，閾値 t と r を動的に変化させてみることや，画像や文字のサイズが異なる様々な画像に対して実験してみることが考えられる．また，Iwamura らの手法の全体の処理時間を大きく削減できたとは言えないので，特徴抽出やマッチングステップの処理を高速化する必要がある.

謝辞

本研究の一部は JST CREST の補助を受けた．ここに記して感謝する．

参考文献

[1] http://www.nttdocomo.co.jp/service/information/utsushite_honyaku/.

[2] M. Iwamura, T. Kobayashi and K. Kise: “Recognition of multiple characters in a scene image using arrangement of local features”, Proc. of 11th International Conference on Document Analysis and Recognition (ICDAR 2011), pp. 1409–1413 (2011).

[3] K. Martin and K. Koichi: “Using a reference point for local configuration of SIFT-like features for object recognition with serious background clutter”, IPSJ Transactions on Computer Vision and Applications（CVA）, **3**, pp. 110–121 (2011).

[4] J. Morel and G.Yu: “ASIFT: A new framework for fully affine invariant image comparison.”, SIAM Jour. on Imaging Sciences, **2**, (2009).

[5] 佐藤, 岩村, 黄瀬：“空間インデクシングに基づく距離推定を用いた高速かつ省メモリな近似近似最近傍探索”, 信学技報 PRMU2012-142, **112**, 441, pp. 73–78 (2013).

[6] M. A. Fischler and R. C. Bolles: “Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography”, Commun. ACM, **24**, 6, pp. 381–395 (1981).