

特徴点探索と投票による孤立文字認識

岩村 雅一[†] 根岸 和也^{††} 大町真一郎^{††} 阿曾 弘具^{††}

[†] 大阪府立大学大学院工学研究科 〒599-8531 大阪府堺市学園町 1-1

^{††} 東北大学大学院工学研究科 〒980-8579 仙台市青葉区荒巻字青葉 6-6-05

E-mail: [†]masa@cs.osakafu-u.ac.jp, ^{††}{negishi,machi,aso}@aso.ecei.tohoku.ac.jp

あらまし 通常、情景中の文字情報を利用するためには文字認識が必要である。しかし、その前段階である文字切り出しの一般的な手法は文字列を成さない「孤立文字」、レイアウトが複雑な数式中の文字、隣接した2文字が接触した接触文字等を切り出すことができない。従って、これらの文字を認識し、情報を得ることができない。本稿ではこのような対象からも文字情報が得られるように、特徴の検索と投票に基づく、文字の切り出しと認識を同時に行う手法を提案する。本手法で用いる投票は一般化 Hough 変換と類似の方法である。提案手法は正しく切り出された文字画像に対する個別文字認識手法を切り出されていない文字画像にも適用できるように一般化した手法と捉えることができる。提案手法を実際の画像に適用したところ、辞書として用いるフォントと認識対象のフォントが異なる場合でも、孤立文字である時計の文字盤や、数式中の接触文字を良好に切り出すことができた。

キーワード 孤立文字, 接触文字, 一般化 Hough 変換, 文字認識, 文字切り出し

Isolated Character Recognition by Searching Feature Points and Voting

Masakazu IWAMURA[†], Kazuya NEGISHI^{††}, Shinichiro OMACHI^{††}, and Hirotomo ASO^{††}

[†] Graduate School of Engineering, Osaka Prefecture University

1-1 Gakuencho, Sakai-shi, Osaka, 599-8531 Japan

^{††} Graduate School of Engineering, Tohoku University

6-6-05 Aoba, Aramaki, Aoba-ku, Sendai-shi, 980-8579 Japan

E-mail: [†]masa@cs.osakafu-u.ac.jp, ^{††}{negishi,machi,aso}@aso.ecei.tohoku.ac.jp

Abstract To utilize a character information in scene images, character recognition and segmentation are required. Character segmentation is essential preliminary of character recognition. However, most character segmentation methods cannot extract *isolated characters* which do not constitute a character string, characters in a complex equation, touching characters where two characters are connected and so on. Therefore, they cannot be recognized. In this paper, to utilize information of such characters, we propose a novel recognition method based on extracting feature points and voting. The voting algorithm of the proposed method is similar to the generalized Hough transform. The proposed method generalizes the conventional character recognition for character images segmented correctly to one for *unsegmented* character images. In experiments, the proposed method extracted isolated characters in the face of a clock and touching characters well even if the fonts of a reference image and a target one were different.

Key words isolated character, touching character, generalized Hough transform, character recognition, character segmentation

1. ま え が き

情景中の文字は様々な情報を持ち、非常に利用価値が高いが、文字情報を得るために必要な文字認識技術にはまだ課題が多い[1]。情景画像中の文字を認識する従来の一般的なアプローチ

は、文字画像を情景画像中から切り出し、切り出された文字画像に対して個別文字認識の手法を適用する。正しく切り出された活字文字に対する個別文字認識の性能は実用上十分高いと考えられるが、切り出しに失敗した文字画像に対しては無力であるため、文字認識性能は文字切り出しの精度に大きく依存する。

ところが、文字認識の前提となる文字切り出しには制約が多い。

例えば、文献 [2] では画像中の局所的な複雑度を利用して一度文字列を抽出した後に文字を抽出する。しかし、このように一度文字列を抽出する手法では文字列を成さない「孤立文字」や、複雑なレイアウトの数式中の文字を切り出すことができない。また、文字切り出しの際に連結成分を用いる手法が多いが、連結成分に着目した手法では、隣接した 2 文字が接触した接触文字を切り出すことができない [3]。

このような制約は、文字の切り出し位置を決定した後に初めて文字の種類を決定（認識）するために起こると考えられる。そもそも文字の位置と種類は、位置を決定するためには字種依存の特徴が必要で、種類を決定するためには位置を特定しなければならないという表裏一体の関係にある。本来同時に行うべき 2 つの処理を別々に行うことが切り出しの失敗を招き、個別文字認識の性能を十分に引き出せない原因であると考えられる。

この状況を改善するため、文字である可能性が高い領域（仮説）を多数切り出し、詳細な判別を認識処理に委ねる多重仮説切り出し法が提案されている [4], [5] が、このような手法は文字行に対して適用する手法であり、画像全体に適用すれば計算量が膨大になる。従って、仮定と検証の過程を経ずに、一度の処理で終わる認識手法が求められる。

そこで我々はアクティブ探索法 [6] を利用して文字の切り出しと認識を同時に行う手法 [7], [8] を提案した。これらの手法は、従来は困難であった孤立文字や接触文字に対しても良好な認識結果を得ている。しかし、認識対象の文字画像と辞書にあたる文字画像のフォントが異なる場合には必ずしも高い認識精度が得られない。

そこで本論文では、文献 [8] の手法をフォントが異なる場合でも認識出来るように改良し、特徴の探索と、位置ずれを許容する投票に基づく手法を提案する。具体的にはまず文字の特徴を探索し、特徴点を発見する。そして、特徴点を基に一般化 Hough 変換 [9] ~ [11] に類する投票を行い、文字の切り出し位置を決定する。特徴には図形の角と曲線を用いる。これらは比較的単純な特徴であるが、文字を構成する基本的な要素であると考えられ、単純な図形であるが故に異なるフォントに対する頑健性を有すると考えられる。

従来の個別文字認識は認識性能、計算速度、使用する計算資源等の面からテンプレートマッチングを用いたものが主流であるが、Hough 変換はテンプレートマッチングと基本的に等価な処理であると考えられている [11]。そのため、今後提案手法を発展させることにより、個別文字認識と同等の処理を切り出されていない文字を含む画像に適用することが可能になると考えられる。

2. 特徴を探索するための準備

2.1 入力画像と参照画像

本手法では、情景画像等、切り出したい文字を含む画像を入力画像、辞書にあたる個々の文字の画像を参照画像と呼ぶ。参照画像としては、対象文字である英数字 62 字種の画像をあらかじめ用意しておく。

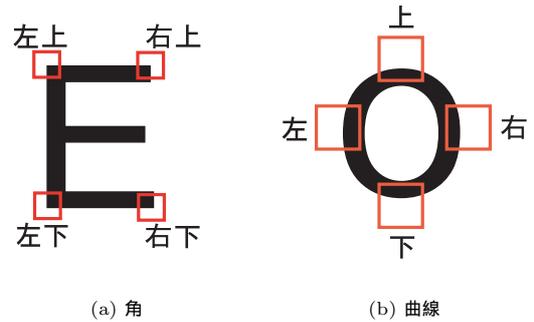


図 1 特 徴

2.2 エッジの方向を利用したヒストグラム

本手法では特徴の検出に用いる特徴量として、エッジの方向を利用したヒストグラムを用いる。あらかじめ画像中の各画素においてエッジの方向と強度を Sobel Filter を用いて算出しておく。その際、雑音の影響を排除するために、エッジの強度が一定値以下の画素に「エッジを持たない画素」というラベルを付けておく。

ある領域のヒストグラムは、エッジの方向を幅 θ ラジアン（階級）に分割し、同じ階級に割り当てられた画素数をその階級の度数となるように作成する。 θ は特徴によって異なり、2.3 で述べる。ただし、「エッジを持たない画素」とラベル付けされた画素は、ヒストグラムを作成する際に利用せず、画素数には数えない。さらに度数の総和が 1 になるように正規化し、正規化ヒストグラムとしたものを特徴量とする。この特徴量を、以下エッジヒストグラムと呼ぶ。特徴量の次元数は $\frac{2\pi}{\theta}$ である。

2.3 特 徴

文字認識で使われる特徴としては、端点・分岐点・交点・屈折点等が検討されている [12]。本論文では文字の判別に有効であり、探索にも適していると考えられる角と曲線を用いる。角は左上、左下、右上、右下の 4 種類（図 1(a)）、曲線は上、左、右、下の 4 種類（図 1(b)）の計 8 種類の特徴を用いる。これらの特徴のエッジヒストグラムは、図 1 の画像からあらかじめ作成しておく。

エッジの方向の刻み幅 θ について考えると、フォントの異なる文字の特徴点も柔軟に検出するためには可能な限り θ を大きくしたほうがよい。一方、複雑な構造を持つ特徴を検出するためには θ を小さくする必要がある。そのため、特徴の複雑さに応じて θ を変え、角は $\theta = \pi/2$ 、曲線は $\theta = \pi/8$ としてエッジヒストグラムを構築する。

具体的には、角の場合、角度の範囲を $[-\frac{\pi}{4}, \frac{7}{4}\pi)$ とし、

$$\left[-\frac{\pi}{4} + \frac{\pi}{2}n, \frac{\pi}{4} + \frac{\pi}{2}n\right), \quad 0 \leq n \leq 3 \quad (1)$$

の各区間に含まれる角度を持つ画素数からエッジヒストグラムを作成する。区間を上記のように広く設定することで図 2 に示すように斜体字に対しても立体と同様に特徴抽出が可能である。図 2 は左上の角の特徴を検出した結果で、図中の赤色の領域が特徴の検出箇所である。

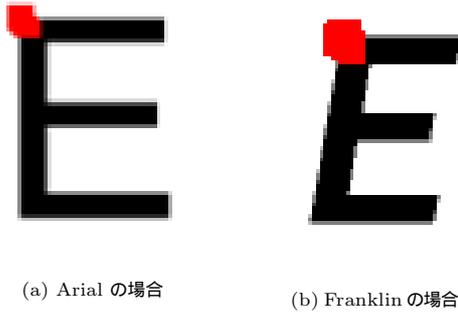


図 2 斜体字に対する角の検出の比較

曲線の場合、角度の範囲を $[-\frac{\pi}{16}, \frac{31}{16}\pi)$ とし、

$$\left[-\frac{\pi}{16} + \frac{\pi}{8}n, \frac{\pi}{16} + \frac{\pi}{8}n\right), \quad 0 \leq n \leq 15 \quad (2)$$

の各区間に含まれる角度を持つ画素数からエッジヒストグラムを作成する。

2.4 類似値

特徴であるかどうかを検証する領域を探索領域と呼ぶ。特徴のエッジヒストグラムと、探索領域のエッジヒストグラムがどれだけ似ているかを示す尺度として、Swain ら [13] が提案した類似値を用いる。H を入力画像または参照画像中の探索領域のエッジヒストグラム、M をある特徴のエッジヒストグラム、Q をエッジヒストグラムの次元数 (角は 4、曲線は 16) とすると、H と M の類似値 S_{HM} は以下の計算によって求められる (H_i と M_i はそれぞれ H、M の階級である)。

$$S_{HM} = \sum_{i=1}^Q \min(H_i, M_i) \quad (3)$$

2 つのエッジヒストグラムは共に和が 1 になるように正規化されているので、類似値の最大値は 1、最小値は 0 である。閾値 T_1 を $0 \leq T_1 \leq 1$ の範囲で定め、類似値が T_1 以上の箇所を特徴 M の検出箇所 (特徴点) とする。

3. 提案する文字切り出しおよび認識手法

提案手法で文字の切り出しと認識を行う処理は一般化 Hough 変換の処理に似ている。そこで、まず一般化 Hough 変換について述べた後、提案手法について述べる。

3.1 一般化 Hough 変換

一般化 Hough 変換は Hough 変換を一般化した手法で、特定のシルエットを持つが解析的に簡単には表せない図形を検出するために用いられる [9] ~ [11]。以下、図形の回転と拡大・縮小を考えない場合の一般化 Hough 変換について述べる。

3.1.1 図形の記述

一般化 Hough 変換では最初に検出対象の図形を記述する。この処理は表 1 のような R テーブルを作成することに等しい。あらかじめ図 3 のように検出したい図形中に基準点を設け、図形のエッジ上の各点 (以後、特徴点と呼ぶ) に番号を付けておく。そして、基準点から j 番目の特徴点までのベクトル (図中の r_j

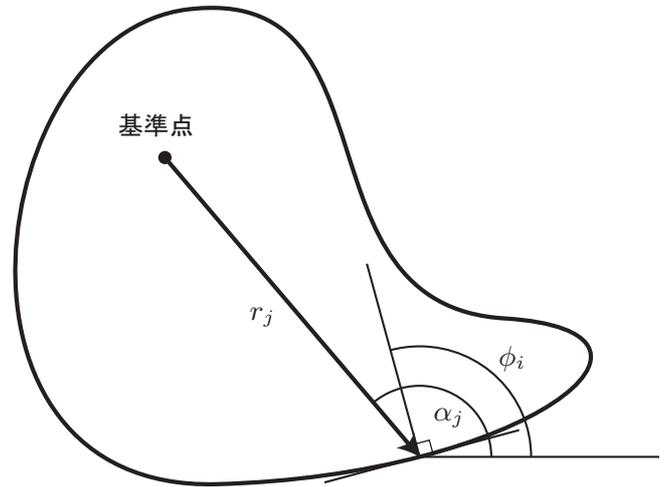


図 3 一般化 Hough 変換

表 1 一般化 Hough 変換の R テーブル

エッジの角度	基準点から見た特徴点の位置
ϕ_1	$(r_{11}, \alpha_{11}), (r_{12}, \alpha_{12}), \dots$
ϕ_2	$(r_{21}, \alpha_{21}), (r_{22}, \alpha_{22})$
\vdots	\vdots

と α_j の組) を求め、そのときのエッジの角度 (図中の ϕ_i) を R テーブルに記録する。その際、エッジの角度をインデックスにする。

3.1.2 図形の検出

図形の検出は、対象図形を含む画像の全てのエッジに対して以下の処理を行う。まず、各エッジの角度を調べる。説明の都合上、ここでは座標 (X, Y) のエッジの角度が ϕ だったとする。次にエッジ毎に R テーブルを見て、エッジの角度 ϕ に対応する基準点の位置を計算し、基準点の投票度数を 1 増やす。今、 $\phi = \phi_2$ だとすると、基準点の位置は $(X - r_{21} \cos \alpha_{21}, Y - r_{21} \sin \alpha_{21})$ と $(X - r_{22} \cos \alpha_{22}, Y - r_{22} \sin \alpha_{22})$ になるので、これら 2 点の投票度数を 1 ずつ増やす。全ての投票が終了した後、投票度数が多い点を基準点とする図形が検出される。

3.2 提案手法

提案手法と一般化 Hough 変換の違いはおおよそ以下の通りである。

(1) 一般化 Hough 変換ではエッジ上の各点の角度に応じて投票するのに対し、本手法では検出された特徴の種類に応じて投票を行う。

(2) 一般化 Hough 変換は基準点にのみ投票するが、本手法は斜体字にした場合等、違うフォントでも安定して抽出できるように、基準点の近傍にも投票する。

(3) 一般化 Hough 変換では投票度数の最大値は不定であるが、本手法では参照画像で検出された特徴点の総数 (後述する $F_S^{(k)}$) が最大値になる。

3.2.1 図形の記述：参照画像からの特徴点の検出

参照画像を記述するため、参照画像から全ての特徴点を検出し、各特徴点の種類と位置を記録する。参照画像の左上の画素を基準点とし、検出した特徴点の位置情報を基準点から特徴点

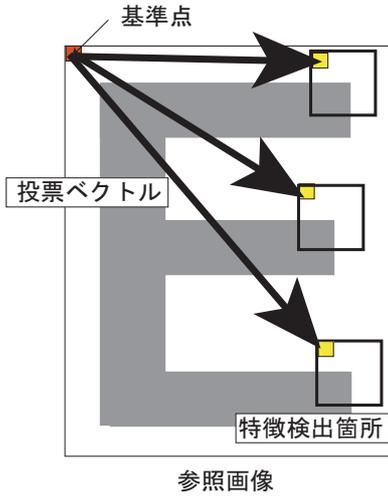


図 4 参照画像からの特徴点検出

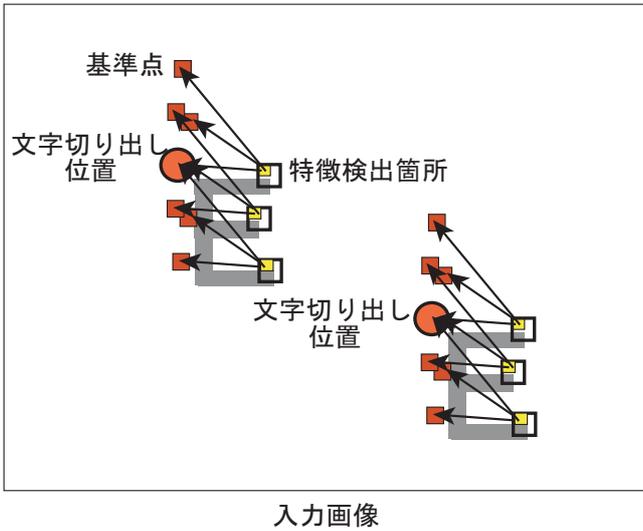


図 5 入力画像からの特徴点検出と投票

へのベクトルで表す (図 4 参照) . このベクトルを投票ベクトルと呼ぶ . これは一般化 Hough 変換での r_j と α_j の組に相当する .

投票ベクトルは以下のように定める . まず , 基準点を原点とし , そこから右に x 軸 , 下に y 軸を取る . 字種 k の参照画像から検出された i 番目の特徴点の個数を $N_i^{(k)}$ とおく ($1 \leq i \leq 8$) . 特徴 i が検出された箇所に 1 から $N_i^{(k)}$ までの番号をつけ , 特徴 i が j 番目に検出された位置を投票ベクトル $f_{ij}^{(k)}$ とする ($1 \leq j \leq N_i^{(k)}$) . こうして得られた特徴 i と投票ベクトル $f_{ij}^{(k)}$ の組が , 字種 k の R テーブルにあたる .

参照画像から検出された特徴点の総数を $F_S^{(k)}$ とおくと ,

$$F_S^{(k)} = \sum_{i=1}^8 N_i^{(k)} \quad (4)$$

と表すことができる . この値は 3.2.2 で用いる .

3.2.2 図形の検出 : 文字切り出し

一般化 Hough 変換での図形の検出に相当する文字切り出しについて述べる . まず , 入力画像内の各点 (x, y) において , 特徴 i が検出されたことによる字種 k の投票度数を

$v_{ij}^{(k)}(x, y)$, $0 \leq v_{ij}^{(k)}(x, y) \leq 1$ とおく . ここで j は参照画像中の特徴点 (投票ベクトル) の通し番号である . あらかじめ全ての (x, y) と k について $v_{ij}^{(k)}(x, y) = 0$ と初期化しておく .

8 種類の特徴について , 1 種類毎に入力画像から全ての特徴点を検出する . そして , 検出された各特徴点について , R テーブルを用いて文字の基準点と考えられる点に投票を行う (図 5 参照) . 特徴 i が l 番目に検出された位置を x_{il} とすると , 字種 k を仮定したときの基準点 $P_{ijl}^{(k)}$ は投票ベクトル $f_{ij}^{(k)}$ を用いて次式で得られる .

$$P_{ijl}^{(k)} = x_{il} - f_{ij}^{(k)} , \quad 1 \leq j \leq N_i^{(k)} , \text{ for all } k \quad (5)$$

本手法では異なるフォントでも安定して抽出できるように , 基準点の近傍にも投票する . 基準点付近での投票方法は , $P_{ijl}^{(k)}$ を中心とする半径 r 画素の円内に投票度数 1 を与える . すなわち ,

$$v_{ij}^{(k)}(x, y) = 1 , \quad \text{for } (x, y) \text{ s.t. } \|(x, y) - P_{ijl}^{(k)}\| \leq r \quad (6)$$

である . これは既に投票が行われた箇所に再度投票を行っても , 探索した特徴が同じで , かつ同じ投票ベクトルを用いる限り投票度数は 1 のままであることを意味する .

全ての投票が終了した後 , 座標 (x, y) が字種 k の基準点である可能性の高さを表す $V^{(k)}(x, y)$ を次式のように求める .

$$V^{(k)}(x, y) = \sum_i \sum_j v_{ij}^{(k)}(x, y) \quad (7)$$

$V^{(k)}(x, y)$ の最大値は $F_S^{(k)}$ であるので , 文字を切り出す際の得票割合の閾値 T_2 を $0 \leq T_2 \leq 1$ で定義すると ,

$$V^{(k)}(x, y) \geq F_S^{(k)} T_2 \quad (8)$$

を満たす箇所 (x, y) が字種 k の文字切り出し箇所となる . 閾値 T_2 の値を大きくすることは , 参照画像との類似度がより大きい箇所のみを切り出すことを意味する .

4. 実験

4.1 特徴点検出

提案手法が文字画像のどの部分を特徴点として検出するのかを調べた . 特徴点検出に用いる探索領域の大きさは任意に変更することができるが , 今回の実験では 5×5 ピクセルの領域を用いた . 探索領域を参照画像の左上から順に走査し , 検索する特徴との類似値 S_{HM} をそれぞれの箇所で計算し , 類似値が閾値 T_1 以上になった箇所を特徴点の検出箇所とした .

フォント Arial の「0」,「E」,「M」,「S」に対して 8 種類の特徴を検出した結果を図 6 に示す . 図 6(a) は「0」, 図 6(b) は「E」, 図 6(c) は「M」, 図 6(d) は「S」の特徴点を検出した結果である . 類似値が閾値 $T_1 = 0.75$ を越えた箇所を特徴の検出箇所とし , 探索領域と同じ大きさの矩形で囲って示している . オレンジ色の矩形がそれぞれ角の左上 , 角の右上 , 曲線の上 , 曲線の左であり , 緑色の矩形がそれぞれ角の左下 , 角の右下 , 曲線の下 , 曲線の右である . 特徴点の検出位置が塗りつぶ

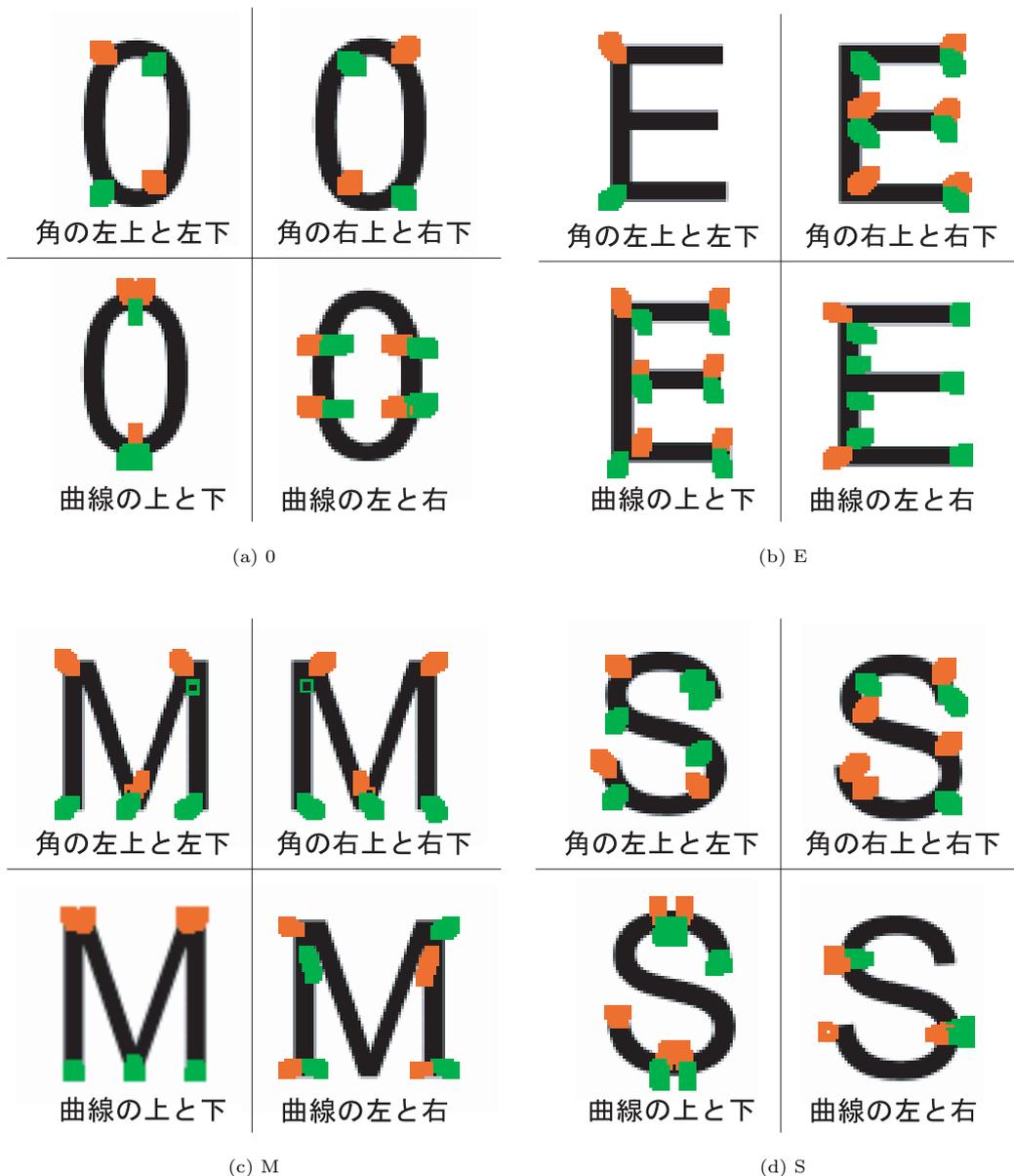


図 6 Arial の特徴点検出箇所

されているのは、角や曲線の付近において、類似値が閾値 T_1 を越えた箇所を全て検出しているからである。また、角の特徴を検出した時、角以外の箇所も多数検出されているが、参照画像と入力画像で同様の点が検出されているため、文字の特定には使用できる。

「0」の曲線の左右の特徴 (図 6(a) の右下) や「S」の曲線の上下の特徴 (図 6(d) の左下) が、曲線の中心からずれたところで 2 箇所ずつ検出されているのは、曲線の中心部分が直線になっていることが一因と考えられる。

4.2 認識実験

提案手法の有効性を確認するため、以下に示す 3 種類の認識実験を行った。

(1) 参照画像と入力画像のフォントが異なる場合の基礎的な認識実験

(2) 孤立文字に対する認識実験

フォント	文字画像の大きさ (ピクセル)
Arial	50 × 54
Arial Black	58 × 54
Franklin	45 × 51
丸ゴシック	60 × 61

表 2 参照画像の大きさ

(3) 接触文字に対する認識実験

いずれの実験でも、参照画像には Arial を用い、入力画像には Arial 以外のフォントを用いた。用いた各フォントの「A」の大きさは表 2 の通りである。低解像度の画像に対しても良好な検出結果を得るために、特徴の検出を行う前にあらかじめ平滑化フィルタを参照画像、入力画像の両方に掛けておいた。

本手法では、エッジヒストグラムを構築する際にエッジを持たない画素を用いない。そのため、探索領域中の大部分がエッ

フォント	Correct	Match	Miss
Arial Black	18 (29%)	43 (69%)	1 (2%)
Franklin	15 (24%)	44 (71%)	3 (5%)
Maru Gothic	22 (35%)	23 (37%)	17 (27%)

表 3 フォントの異なる文字に対する認識結果

ジを持たない画素の場合にはわずかな画素から信頼性の低いエッジヒストグラムが計算され、誤って特徴を検出してしまう場合がある。そこで、探索領域にエッジを持たない画素が7割以上含まれている場合は、その箇所での特徴検出は行わないことにする。

特徴検索での類似値は 4.1 同様、閾値 T_1 以上になった箇所を特徴点の検出箇所とした。投票の際は、得票割合が閾値 T_2 以上の領域を文字の切り出し箇所とした。基準点付近に投票するときの円の半径 r は全ての実験で $r = 8$ とした。

実験の評価方法について述べる。実験結果を Correct, Match, Miss の 3 種類に分類した。Correct は探索している文字と同じ文字のみを切り出した場合で、文字の切り出しと認識が同時に行われたことに相当する。図 7(a) はその例で、参照画像に「A」を用いた時に入力画像の「A」のみが切り出されている。4.2.2 で行う実験のように入力画像中に探索する文字が複数ある場合は、それらの文字を全て切り出し、それ以外の文字を一切切り出さなかった場合のみ Correct とする。

Match は探索する文字を切り出したが、他の箇所も切り出している場合である。図 7(b) はその例で、参照画像に「C」を用いた時に「C」を切り出してはいるが、それ以外に「6」、「G」、「O」、「Q」も切り出した。この場合、切り出した 5 文字に対して既存の個別文字認識手法を適用することで「C」は正しく認識でき、それ以外は「C」ではないと判断されると考えられる。従ってこれは文字の切り出しと認識の大分類に相当し、切り出しの失敗には当たらない。

Miss は探索している文字を切り出すことが出来なかった場合である。図 7(c) はその例で、参照画像に「K」を用いたが「K」は切り出せなかった。

4.2.1 フォントの異なる文字に対する認識実験

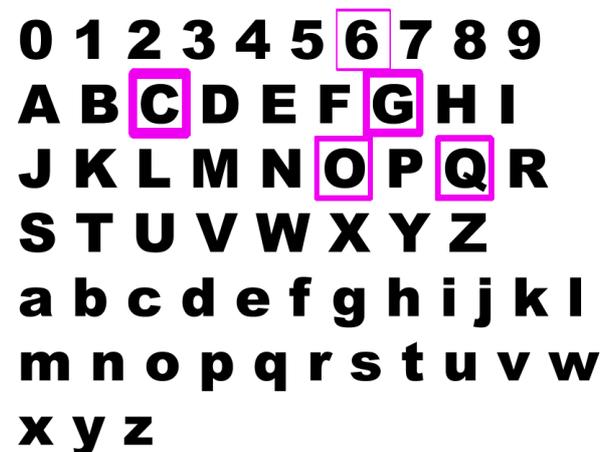
フォント Arial の文字を参照画像とし、太文字である Arial Black, 斜体字である Franklin, 丸みを帯びている丸ゴシック体を認識対象として認識実験を行う。なお、参照画像と認識対象の文字のフォントが同じ時の認識率は 100%であった。

入力画像には図 7 のように各フォントの英数字 62 文字が並んでいる画像を用いた。特徴を検出する時の類似値の閾値を $T_1 = 0.75$ とし、文字として切り出す時の得票割合の閾値を $T_2 = 0.85$ とした。

認識結果を表 3 にまとめた。実験の結果を見ると、認識まで正確に行えている Correct の割合は 1/3 程度であるが、参照画像と同じ文字を切り出すことが出来た割合 (Correct と Match の和) で考えると、Arial Black と Franklin に関しては 95% 以上であり、フォントが異なるにも関わらず、高い精度で切り出しに成功しているといえる。丸ゴシック体に関しては 70% 以上の文字を切り出すことが出来た。これにより、提案手法はフォ



(a) Correct: 参照画像に「A」を使用



(b) Match: 参照画像に「C」を使用



(c) Miss: 参照画像に「K」を使用

図 7 評価基準

ントが異なる場合でも一定の切り出し性能を持つといえる。

次に切り出しが出来なかった原因を考える。切り出しが出来なかったのは Arial Black が「K」、Franklin が「5」、「7」、「K」

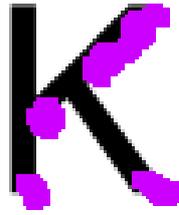


図 8 切り出しの失敗例:角の右下の特徴を検出

	Correct	Match	Miss
個数	9	1	0
割合	90%	10%	0%

表 4 時計の文字盤 10 字種に対する認識結果

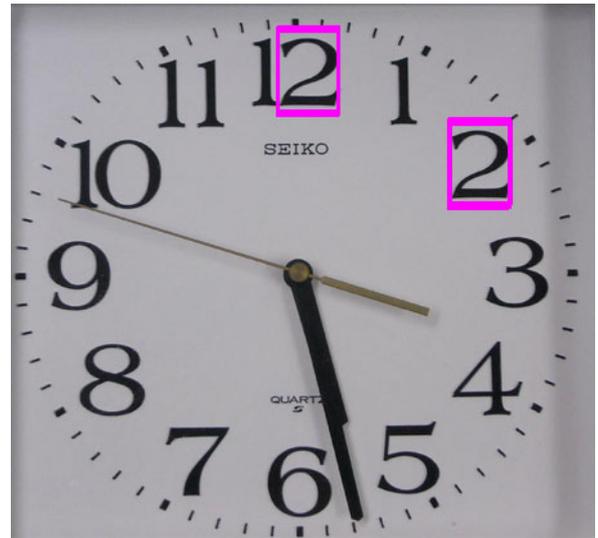
であった．両方のフォントで「K」が切り出されていない．そこで参照画像である Arial の「K」に対して角の右下の特徴を検索してみると，望ましくない箇所でも特徴が多数検出された（図 8 参照）．本手法で文字の切り出し位置を決定する際，参照画像から検出された特徴点の総数 $F_S^{(k)}$ に対する得られた投票度数の割合を判断材料にしている．図 8 のように，斜め直線を多量に検出してしまう場合は $F_S^{(k)}$ が異常に大きくなってしまいうために投票度数が相対的に低下し，文字の切り出し精度の低下を招く．

また，丸ゴシック体においては，切り出し出来なかった文字が約 27% 存在した．精度が低くなってしまった理由の一つに文字の大きさの差異が考えられる．表 2 が示す通り，フォントによって大きさが異なり，丸ゴシック体は Arial との大きさの差が最も大きい．投票の際には基準点から半径 r の領域に投票するため，フォントの大きさが多少違っていても正しく切り出せるが，その許容範囲を超えてしまったために閾値以上の投票度数が得られなかったものと思われる．このことを示すように，閾値 T_2 を 0.85 から 0.75 に変更することで約 85% の文字が正しく切り出せるようになった．従って，この問題は文字の拡大・縮小に対応することで解決できる可能性がある．

4.2.2 孤立文字に対する認識実験

孤立文字を含む時計の文字盤の画像に対して認識実験を行った．画像はグレースケールに変換してから用いた．画像の大きさは 454×420 ピクセルである．参照画像は Arial の数字 10 字種を 4.2.1 と同じ大きさで用いた．特徴を検出する時の閾値を $T_1 = 0.75$ とし，得票割合の閾値を $T_2 = 0.89$ とした．認識結果を表 4 にまとめた．また，参照画像に「2」を用いた時と「1」を用いた時の認識結果をそれぞれ図 9(a) と図 9(b) に示す．実験の結果は「1」が Match，それ以外の 9 字種が Correct であり，どの字種も Miss にはならなかった．

「1」と「2」に関しては文字が画像中に複数個存在するが，「2」は対象となる文字を全て切り出し，他の文字を一切切り出さなかったため，評価を Correct とした．一方「1」は対象となる文字を全て切り出したが「4」の一部も切り出したので，評価を Match とした．このように参照画像として「1」を用いた



(a) Correct: 参照画像に「2」を使用



(b) Match: 参照画像に「1」を使用

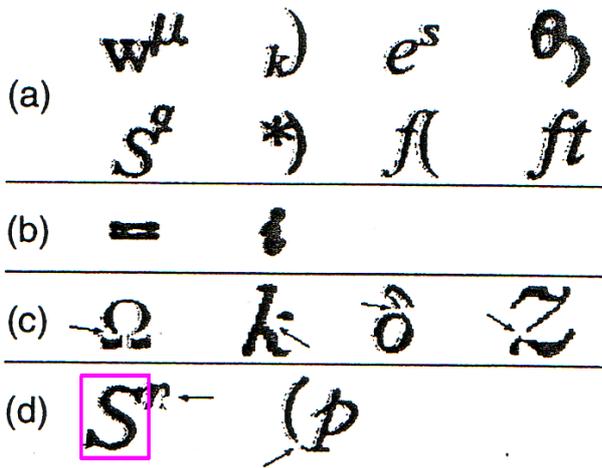
図 9 孤立文字に対する認識結果

場合は検索対象以外の文字も切り出してしまったが，文字の切り出しと認識の大分類という観点に立てば，切り出し精度としては十分である．

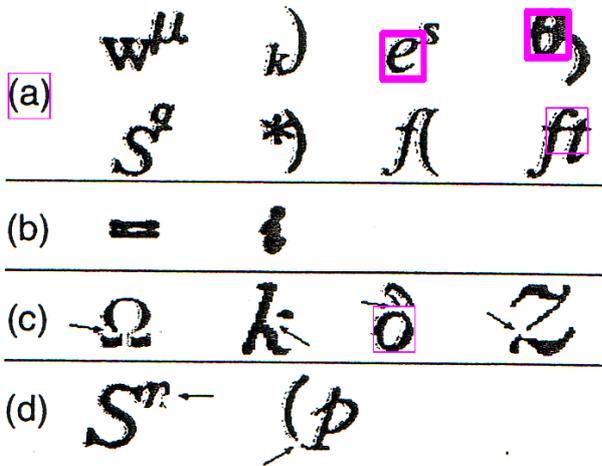
この結果は，本手法が孤立文字を含む実画像に対しても非常に有効であることを示している．特筆に値するのは，文字盤の数字の字形が参照画像である Arial とは大きく異なっていることである．それにも関わらず正しく切り出せていることは本手法の頑健性を示している．

4.2.3 接触文字を含む画像に対する認識実験

文献 [3] から接触文字を含む画像を借用して認識実験を行った．画像の大きさは 749×558 ピクセルである．実験で使った参照画像は Arial の「S」と「e」である．特徴を検出する時の閾値は $T_1 = 0.75$ を用い，文字を切り出す時の得票割合の閾



(a) Correct: 参照画像に「S」を使用



(b) Match: 参照画像に「e」を使用

図 10 接触文字に対する認識結果

値 T_2 は「S」の場合は $T_2 = 0.7$ 、「e」の場合は $T_2 = 0.8$ を用いた。本手法は文字の拡大・縮小にも対応可能であるが、現在は未実装であるため、今回の実験では参照画像の大きさを認識対象に合わせることで対応した。参照画像に「S」を用いたときの結果を図 10(a) に、参照画像に「e」を用いたときの結果を図 10(b) に示す。参照画像に「S」を用いた場合は Correct で、参照画像に「e」を用いた場合は Match であった。実験の結果より、従来法では切り出しが困難であった接触文字に対しても本手法は精度の高い切り出しを行うことができる。

図 10 には大きさが異なる「S」が 3 文字含まれている。今回の実験では、最も大きい「S」に参照画像の大きさを合わせたために、他の「S」は切り出せなかった。そのため、4.2.1 でも述べたが、今後は文字の拡大・縮小にも対応する必要があると考えられる。

5. まとめ

文字切り出しは文字認識の前段階と位置付けられているが、一般的な文字切り出し手法は文字列を成さない「孤立文字」、レイアウトが複雑な数式中の文字、隣接した 2 文字が接触した接触文字等を切り出すことができない。切り出しに失敗した文字は認識できないので、これらの対象は認識できない。

本論文では特徴の検索と、位置ずれを許容する投票に基づく、文字切り出しと認識を同時に行う手法を提案した。提案手法を実際の画像に適用したところ、辞書として用いるフォントと認識対象のフォントが異なる場合でも、孤立文字である時計の文字盤や、数式中の接触文字を良好に切り出すことができた。

提案手法は原理的に従来の個別文字認識と同等の処理が可能であると考えられるため、今後提案手法を発展させることにより、個別文字認識と同等の処理を切り出されていない文字を含む画像にも適用することが可能になると考えられる。

文字画像の回転や拡大・縮小への対応、閾値をフォントや画像状態に合わせて動的に変更する手法の開発は今後の課題である。

謝辞 本研究の一部は、科学研究費補助金 基盤研究 (C) 16500096、ならびに若手研究 (B) 17700205 の補助を受けている。

文 献

- [1] 黄瀬浩一, 大町真一郎, 内田誠一, 岩村雅一, カメラを用いた文字認識・文書画像解析の現状と課題, 信学技報 PRMU2004-246, Mar., 2005.
- [2] 松尾賢一, 上田勝彦, 梅田三雄, “適応しきい値法を用いた情景画像からの看板文字列領域抽出,” 信学論 (D-II), vol.J80-D-II, no.6, pp.1617-1626, 1997.
- [3] 内田誠一, 野村明弘, 鈴木昌和, 数学文書データベースの解析, 信学技報 PRMU2003-234, 2004.
- [4] H. Yamada and Y. Nakano, “Cursive handwritten word recognition using multiple segmentation determined by contour analysis,” IEICE Trans. Inf. & Syst., vol.E79-D, no.5, pp.464-470, May, 1996.
- [5] 丸山健一, 小林誠, 山田博文, 中野康明, “複数の文字認識を用いた連続手書き英単語認識,” 信学論 (D-II), vol.J82-D-II, no.9, pp.1435-1443, 1999.
- [6] 村瀬洋, V. Vinod, “局所色情報を用いた高速物体探索—アクティブ探索法—,” 信学論 (D-II), vol.J81-D-II, no.9, pp.2035-2042, 1998.
- [7] 根岸和也, 岩村雅一, 大町真一郎, 阿曾弘具, 切り出しを同時に行う孤立文字認識法, 電気関係学会東北支部連合大会講演論文集 2H-13, 2004.
- [8] 根岸和也, 岩村雅一, 大町真一郎, 阿曾弘具, 部分領域の探索による孤立文字認識, 第 3 回情報科学技術フォーラム講演論文集 I-018, 2004.
- [9] D. Ballard, “Generalizing the Hough transform to detect arbitrary shapes,” Pattern Recognition, vol.13, no.2, pp.111-122, 1981.
- [10] D. H. Ballard and C. M. Brown, コンピュータ・ビジョン, 日本コンピュータ協会, 東京, 1987. 福村晃夫 [ほか] 訳.
- [11] 松山隆司, 久野義徳, 井宮淳 (編), コンピュータビジョン: 技術評論と将来展望, 新技術コミュニケーションズ, 東京, 1998.
- [12] 森健一 (編), パターン認識, 電子情報通信学会, 1988.
- [13] M. Swain and D. Ballard, “Color indexing,” International Journal of Computer Vision, vol.7, pp.11-32, 1991.