

ピントのずれに頑健な文字認識手法の高速化

伊村 政志[†] 岩村 雅一[†] 日浦 慎作^{††} 黄瀬 浩一[†]

[†] 大阪府立大学 大学院工学研究科

^{††} 広島市立大学 大学院情報科学研究科

E-mail: fimura@m.cs.osakafu-u.ac.jp, {masa,kise}@cs.osakafu-u.ac.jp, ††hiura@hiroshima-cu.ac.jp

あらまし 文字認識では、ピントのずれによって認識率の低下が起きる。この問題の原因は、文字の特徴が潰れ、パターンが類似することにある。この問題は符号化開口を用いてピントがずれても文字の特徴が残るようにすれば解決可能と考えられる。しかし、ぼけ画像はレンズの焦点面からのずれにより変化するため、認識にはぼけの度合を推定する必要がある。1枚の画像からぼけの度合を推定するには、画像に対して何かしらの仮定を設ける必要がある。そこで本研究では、画像には予め学習されたテンプレートのいずれかの文字が写っていると仮定する。前報では、予めテンプレートのぼけたパターンを用意して、認識対象とテンプレートマッチングにより認識する手法を提案した。しかし、この手法ではマッチングに用いるパターン数がテンプレートとぼけの度合の候補数の掛け算になってしまうため、その数に比例して処理時間が膨大なものになってしまう。そこで本稿ではぼけの度合推定と文字認識の2つのプロセスに分割することで上記手法の高速化を図る。また、ぼけた文字の認識に適した開口形状について検討する。

キーワード 文字認識, 符号化開口, コンピュータショナルフォトグラフィ, テンプレートマッチング

1. はじめに

近年情景画像中の文字を認識する技術が発展しており、カメラ付き携帯電話やスマートフォンのアプリケーションとして、NTTドコモの「うつつて翻訳」^(注1)を始めとした様々なサービスが提供されている。しかし、文字認識のシステムはぼけによる撮影画像の劣化に脆弱である。通常、カメラに取り付けられている絞りは、ピントがずれると文字の特徴が潰れ、その形状が似通ってしまい認識精度を著しく低下させてしまう。

ピントのずれに対しては、オートフォーカスなど、撮影の段階でピントがずれないように撮影システムを構築するという方法がある。しかし、認識対象の前にガラスなどの遮蔽物が存在する場合、そちらにピントが合ってしまうため、認識対象に必ずピントが合うというわけではない。

そこで本研究では、画像に計算処理を加えることで鮮明な画像を得るコンピュータショナルフォトグラフィ [1] の技術の1つである符号化開口 [2] を用いることで解決を図る。これはピントのずれによるぼけが発生しても、計算処理により鮮明な画像を得る技術であり、様々な形状の絞りが設計されている。このように設計された絞りの形状によってはピントがずれても認識対象であるパターンの特徴が撮影画像に保持されるため、文字の認識に活用できると考えられる。

しかし、画像のぼけは焦点面から撮像面までの距離により変化する。そのため、異なる形状にぼけたパターンを認識するた

めには、それがどれだけぼけているかを推定する必要がある。ぼけの度合を推定する既存手法として、ステレオカメラなど特殊なデバイスを用いることで、一度に入手した複数枚の画像を用いる手法が存在する [3]。一方で、河本らは1枚の画像からぼけの度合を推定する手法として、対象をバーコード画像に限定し、画像は白と黒の2値で表現されていると仮定することで、ヒストグラムの分離度からぼけの度合を推定した [4]。このように、1枚の画像からぼけの度合を得るには、元画像に対して何らかの仮定を設ける必要がある。文字認識では、認識対象の背景色が未知であり、また2値で表現されてるとも限らないため、上記のような画像に色の仮定を用いることはできない。そこで本研究では、画像にはデータベースに存在するテンプレートのいずれかが写っていると仮定する。

我々は前報において、候補となるぼけの度合を複数用意して、ぼけた文字のデータベースを作成し、認識対象と総当りのマッチングを行うことで、ぼけたままの画像から文字を認識する手法を提案して、背景の存在する画像でも文字が認識可能であることを示した [5]。しかし、この手法で用意されるパターンの数はテンプレートの数とぼけの度合の候補数の積となるため、テンプレートの数に比例して処理時間も増加してしまう。そこで本稿では、従来手法のプロセスをぼけの度合の推定と文字認識に分割する手法を提案し、処理時間を短縮する。また、提案手法に用いるためのぼけの度合推定に適した開口形状について検討する。

本稿の流れを説明する。まず、2節で符号化開口について説明し、3節で本研究で用いる仮定と、前報 [5] の手法について説明する。4節で提案手法について説明した後、5節で実験とそ

(注1) : http://www.nttdocomo.co.jp/service/information/utsushite_onyaku/

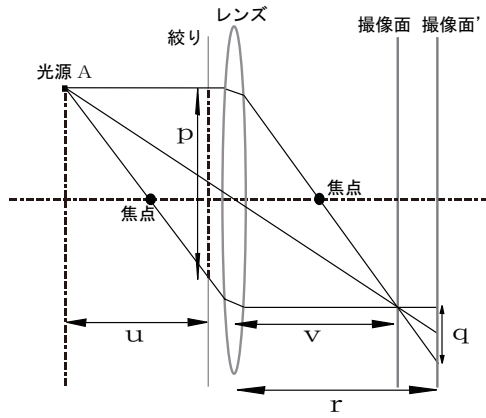


図1 レンズカメラのモデル

の結果について述べ、6節で提案手法に最適な開口形状について検討し、最後に7節でまとめと今後の課題について述べる。

2. 符号化開口

2.1 ぼけの発生原理とその除去

我々が普段使用する、一般的なレンズカメラをモデル化したものを図1に示す。焦点距離が f であるレンズでカメラから u 離れた光源 A を撮影する場合、ピントの合う距離 v は次式により求まる。

$$\frac{1}{f} = \frac{1}{u} + \frac{1}{v} \quad (1)$$

このとき撮像面レンズから撮像面までの距離 r が焦点面と同じ距離、つまり $r = v$ となるならば、ピントのあった画像を得ることができる。しかし、 $r \neq v$ の場合にはレンズを通った光源 A からの光は次式で表される直径 q の円として撮像面上に投影される。

$$q = \frac{p}{v} |r - v| \quad (2)$$

ここで、 p はカメラの絞りの大きさを表している。このとき、 q の大きさが撮像素子1つ当たりの大きさを超えると画像にぼけが生じる。その形状は絞りの形に依存しており、一般的なカメラの場合、絞りの形は円であるため、ぼけの形状も円となる。

ピントのずれによるぼけパターンは、レンズからの撮像面までの距離により変化する。この特徴を考えると、ぼけた画像は、ピントのあった画像がある関数に則って変化したものと言える。この関数をぼけ関数 (PSF) と呼ぶ。PSF を $d(x, y)$ とすると、ピントのずれによる PSF は開口形状と相似の関係を持ち、以下のように表すことができる。

$$d(x, y) = \frac{1}{M^2} a\left(\frac{x}{M}, \frac{y}{M}\right) \quad (3)$$

ここで、 $a(x, y)$ は開口形状を、 M は $a(x, y)$ に対する $d(x, y)$ の拡大倍率を指す。なお、 $\frac{1}{M^2}$ はレンズの透過光量を一定にするための正規化項である。

ぼけた画像を s 、元画像を t 、画像に含まれるノイズを z で表すことにより、ぼけた画像は

$$s(x, y) = d(x, y) * t(x, y) + z(x, y) \quad (4)$$



(a) MURA コード (b) Veeraraghavan

図2 符号化開口で用いられる開口形状

と畳み込み積分の式で表現することが可能である。式 (4) から、ぼけを除去した画像 \hat{t} はぼけた画像と PSF の逆畳み込み積分により求まることがわかる。畳み込み積分の式はフーリエ変換すると積の式に変化するので、式 (4) をフーリエ変換すると、

$$S(u, v) = D(u, v) \cdot T(u, v) + Z(u, v) \quad (5)$$

となり、式 (5) からぼけ除去画像のフーリエ変換 \hat{T} は、

$$\hat{T}(u, v) = \frac{S(u, v)}{D(u, v)} = T(u, v) + \frac{Z(u, v)}{D(u, v)} \quad (6)$$

として表すことができる。このように、 \hat{T} をフーリエ逆変換することにより、ぼけを除去した画像 \hat{t} を得ることができる。

ここで、もし開口形状が円である場合、 D は 0 または非常に小さい値を多く含んでしまうため、式 (6) 右辺においてノイズが強調されてしまう。その結果、 \hat{T} の値が不安定なものになるため、鮮明な画像を安定して得られるような開口形状を設計することが必要となった。

2.2 符号化開口

一般的に用いられるレンズカメラでは、ピントがずれた時にぼけによって対象の特徴が潰れてしまい、ぼけの除去を行っても鮮明な画像を得ることができない。逆にピンホールカメラを用いればピントのずれは発生しないが、光量が不足するため暗かった画像となってしまい、画像のノイズが強調されてしまう。そのため符号化開口は光量を確保しつつ、ピントがずれても特徴が残るように設計された。開口形状の例を図2に示す。黒が遮光部、白が透過部である。符号化開口には、X線やγ線を撮影するために考案された MURA コード [6] (図 2(a)) や、一般的なレンズカメラで鮮明な画像を得るために考案された Veeraraghavan の開口形状 [7] (図 2(b)) を始め、何を撮影するか等、その目的毎に様々な形状が存在している。しかし、符号化開口を用いてぼけの除去を行おうにも、焦点面からの距離によって式 (3) の M は変化するため、PSF d が未知であるという問題が残る。ぼけを除去する場合、 d が真の PSF と異なっていると画像をより改悪させることになってしまうため、画像からぼけの度合を推定する必要がある。前述のように、1枚の画像からぼけの度合を推定するには、画像に対して何かしらの仮定を設ける必要がある。本研究の目的はパターン認識であるため、扱う画像が2値であるとは限らず、その色も未知であるため、本研究では色情報に代わる別の仮定を設ける必要がある。

3. 従来手法

本節では前報 [5] の手法について説明する。

本研究で画像に何かしらの仮定を設けるにあたり、文字認識手法のプロセスに着目する。文字認識では認識対象となる文字が予めデータベース上に存在しているという前提のもと、データベース内の文字とのマッチングを行うことで認識している。そこで本研究では、データベースに存在するテンプレートのいずれかが、認識対象画像に写っていると仮定する。これにより、文字の認識のために画像のぼけを除去する必要はなくなり、逆にテンプレートをぼかすことでの認識が可能になる。

この仮定を用いた手法 [5] ではまず、ぼけの度合の候補を複数用意して、それに基づく PSF によって全てのテンプレートをぼかす。こうして全てのぼけの度合に対応したテンプレートの劣化パターンを作成して、認識対象に対してテンプレートを走査させる。そして、正規化相互相関 (ZNCC) によるマッチングを行い、相関係数のピークの値が最も高くなった文字を認識結果として返す。この方法を用いれば、ぼけ推定と同時に文字の認識が可能である。

次節で述べる提案手法の準備として、この手法を式で表す。相関によるマッチング処理を \circ で表現して、認識対象を x 、データベースに存在する i 番目のテンプレートを t_i 、 j 番目のぼけの度合に基づく PSF を d_j とおくことにより、作成されるぼけた文字 y_{ij} は $y_{ij} = t_i * d_j$ と畳み込み積分の式になる。したがってこの手法は、

$$[i^*, j^*] = \arg \max_{i,j} (x \circ y_{ij}) \quad (7)$$

と式表現できる。この手法ではテンプレートの数を n 、ぼけの度合の候補数を m とすると、処理時間は $O(mn)$ となる。そのため、テンプレートの数を増やした場合、それに比例して処理時間も大きく増加するという問題がある。

4. 提案手法

本節では前節で述べた従来手法の式を変形して、そこから導出される処理時間を削減した手法について説明する。

4.1 従来手法の高速化

本研究において認識対象 x が真のテンプレート t_x と真の PSF d_x の畳み込み積分 $x = t_x * d_x$ で表現できると仮定しているため、式 (7) は、

$$[i^*, j^*] = \arg \max_{i,j} ((t_x * d_x) \circ (t_i * d_j)) \quad (8)$$

と表すことができる。相関は周波数領域において複素共役の畳み込み積分により表現可能なので、 t_i 、 d_j 、 t_x 、 d_x のフーリエ変換をそれぞれ T_i 、 D_j 、 T_x 、 D_x とすると、 $(t_x * d_x) \circ (t_i * d_j)$ は周波数領域において、

$$(T_x \cdot D_x) \cdot \overline{(T_i \cdot D_j)} = (T_x \cdot \overline{T_i}) \cdot (F_x \cdot \overline{D_j}) \quad (9)$$

とできる。これに逆フーリエ変換を行うことで式 (8) は、

$$[i^*, j^*] = \arg \max_{i,j} ((t_x \circ t_i) * (d_x \circ d_j)) \quad (10)$$

と変形できる。式 (10) は文字同士の相関と、PSF 同士の相関を独立に行える可能性を示している。すなわち、最初に

$$j^* = \arg \max_j (d_x \circ d_j) \quad (11)$$

によりぼけの度合を求めて、次に

$$i^* = \arg \max_i (x \circ (t_i * d_{j^*})) \quad (12)$$

で字種を決定するという方法が採れれば、従来手法で $O(mn)$ の計算量を $O(m+n)$ まで削減できる。

しかし、ぼけた文字 $x = t_x * d_x$ は本来 t_x と d_x に分離不可能であるため、PSF 同士の相関 (式 (11)) を単独で行うことはできない。そのため、何らかのパターン \bar{t} を設定して、これを基に作成した $\bar{x} = \bar{t} * d_x$ と x の相関を

$$j^* = \arg \max_j (x \circ (\bar{t} * d_j)) \quad (13)$$

で求めることで式 (11) を実現することを考える。式 (11) と式 (13) が等価になるのは、 $t_x \circ \bar{t}$ により得られる画像が、点光源のように中心の画素のみ 1 で、他が 0 となるときである。したがって、 $\bar{t} = t_x$ とすることで、前述の処理の実現を試みる^(注2)。しかし、 t_x がどの文字であるかはそもそも未知であるため、 $\bar{t} = t_x$ を実現するには全てのテンプレートを \bar{t} とおいて相関をとってみるしかなく、処理時間が $O(nm+n)$ となる^(注3)。ため、従来手法よりも時間がかかってしまう。その一方で、 \bar{t} の数を n から減らしていくと、処理時間は削減されるが、除かれた字種のぼけの度合が正確に求まる保証はなく、誤認識を引き起こす要因となりかねない。

そこで、 \bar{t} の数を精度を落とさずに効果的に削減するために、類似の字種を統合したテンプレートの合成画像を用いる。統合の際には、同一クラスの文字ごとの特徴を強調するために、テンプレートの黒画素数の逆数で重み付けを行う^(注4)。どの字種を統合するかについては、本稿では文字同士の相関係数に基づくデンドログラムを作成して合成画像数を削減することで、認識率を落とさずに処理時間を削減することを試みる。すなわち、テンプレートの全字種間の相関を求め、1 から相関係数を引くことで距離を求める。そして郡平均法を用いたクラスタリングを階層的に行い、デンドログラムを作成する。作成したデンドログラムを図 3 に示す。そして、デンドログラムの階層ごとに、クラス内で文字を合成する。最適な \bar{t} の数 \bar{n} を予備実験で決定して用いるとすれば、処理時間は $O(\bar{n}m+n)$ となる。

4.2 提案手法の処理手順

処理の流れを説明する。まずぼけの度合いの候補を複数用意して、ぼけの度合い毎に PSF を作成する。作成した PSF を用

(注2)：厳密には、前報 [5] の図 8 で示したように、中心以外の画素値も 0 以外の値になってしまうため、これは近似になる。

(注3)：式 (11) の処理に $O(mn)$ 、式 (12) の処理に $O(n)$ の計算時間がかかるため。

(注4)：単純に平均画像を用いた場合、大きさが異なる字種がクラス内に存在すると、 \bar{t} の形状は黒画素が多い字種に近くなり、黒画素が少ない文字の相関が小さくなる。その結果、クラス内の全ての字種で正しくぼけの度合を求められなくなるという懸念がある。

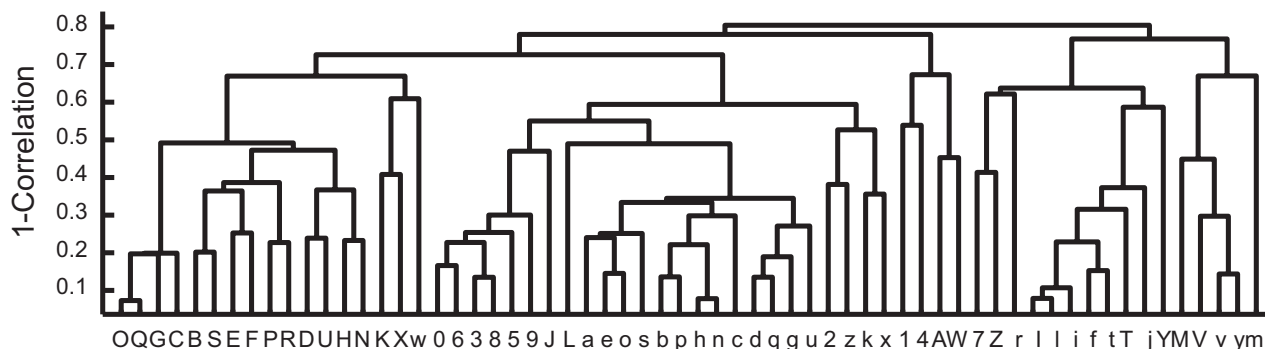


図3 テンプレートのデンドログラム

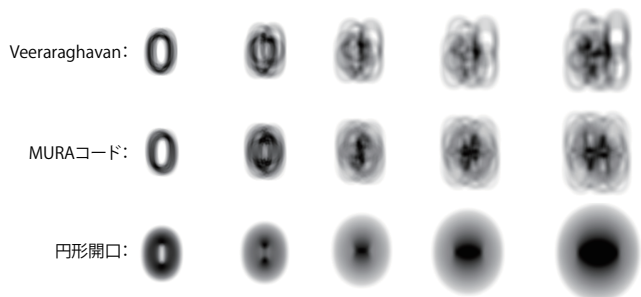


図4 クエリに用いた画像例。ぼけの度合は左から2, 4, 6, 8, 10である。

いてテンプレートの合成画像 \bar{t} を畳み込み積分によりぼかして、認識対象の画像に走査する。そして式 (13) より d_x との相関係数が最大となるぼけの度合 j^* を返す。次に、推定したぼけの度合に基づく $PSF_{d_{j^*}}$ で、 \bar{t} に対応する文字をぼかす。そして改めて認識対象の画像に走査して、式 (12) より x との相関係数が最大となる字種を認識結果として返す。

5. 実験

本節では、シミュレーションによる実験で提案手法の有用性を検証するとともにその考察を行う。

5.1 実験条件

実験で用いたテンプレートは MS ゴシックの数字及び英字大文字、英字小文字の計 62 字種である。開口形状は、Veeraraghavan の開口形状と MURA コード、円形開口の 3 つを用いた。ぼけ推定に用いるクエリとして、図 4 に示すように、ピントの合っている 0 段階目から大きくぼけた 20 段階目まで 21 段階にぼかした画像をシミュレーションにより作成した。各段階の PSF のサイズは 1 段階目から 7×7 [pixels], 14×14 [pixels], 21×21 [pixels], ..., 140×140 [pixels] となっている。これはぼけの度合が 1 上がるたび、焦点面から撮像面までの距離が等間隔に離れていることを意味する。また、用意したぼけの度合の候補にはクエリに用いたぼけの度合も含まれているため、従来手法の認識率は 100% である。なお、MS ゴシックの大文字の I と小文字の l は形状が酷似しているため、互いに誤認識を起こした場合でも認識できたものとする。

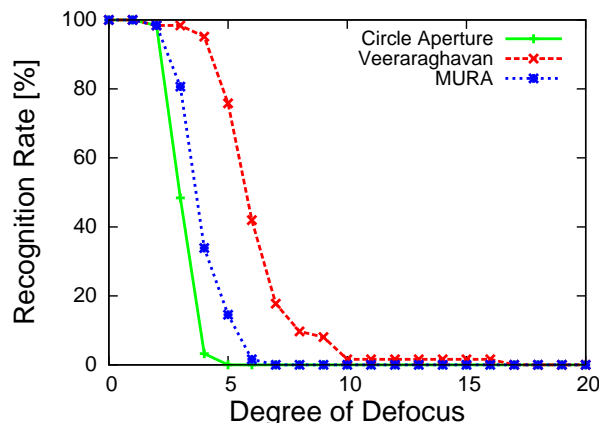


図5 ぼけ推定を行わない場合の認識率

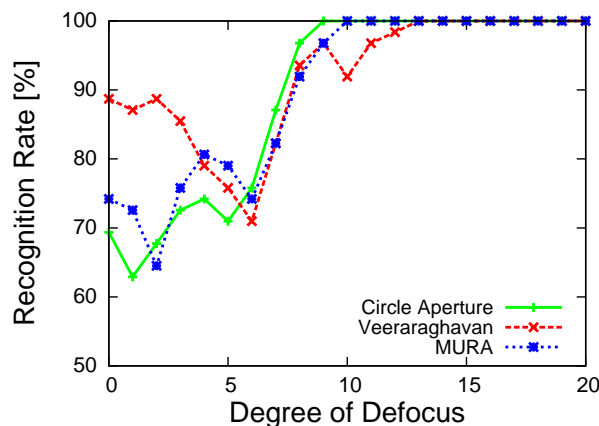


図6 合成画像を1枚にした時の認識率

5.2 ぼけ推定を行わない場合の認識率

提案手法での実験の前に、本節では各クエリに対してぼけの度合いを推定せず、画像にテンプレートを走査させるだけでどこまで認識が可能かを調べる。実験結果を図 5 に示す。どの開口形状もぼけの度合いが 2 の時までは認識できている。また円形開口に比べて、MURA コードと Veeraraghavan の開口形状はより大きなぼけに対しても高い認識率を示した。しかし、開口を符号化してもぼけ推定を行わなければ、認識性能に限界があることが確認できる。

5.3 提案手法の認識率

本節では、提案手法を用いた場合に各開口形状がどれだけの

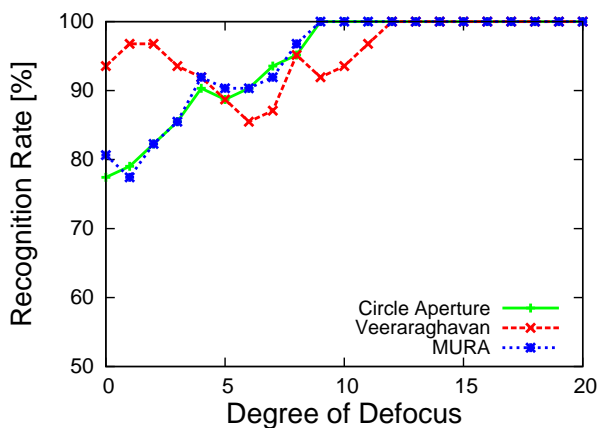


図 7 合成画像を 5 枚にした時の認識率

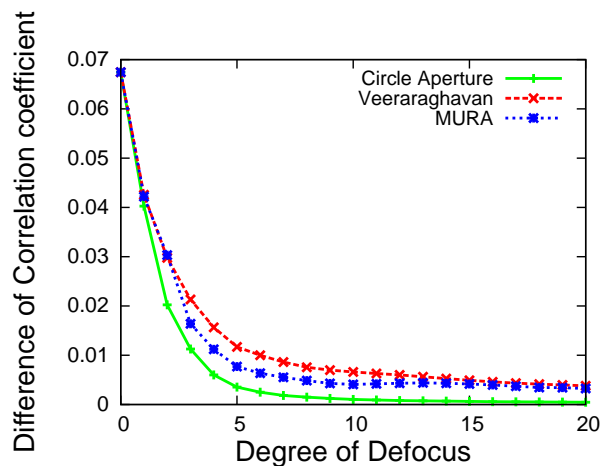


図 10 1 位と 2 位の字種の相関係数の差

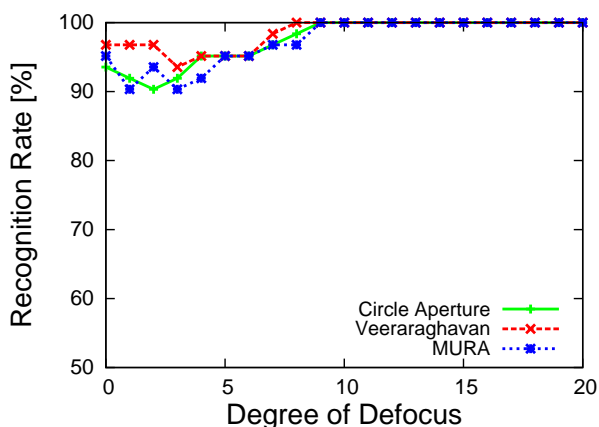


図 8 合成画像を 10 枚にした時の認識率

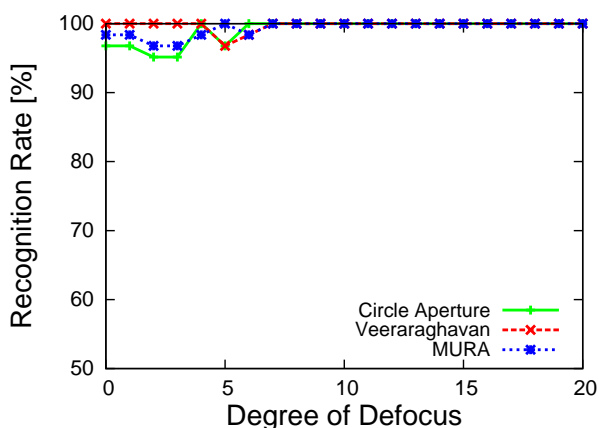


図 9 合成画像を 15 枚にした時の認識率

認識率を示すかを調べた。本実験で用いるクエリ、絞りの形状は前節と同じである。合成画像 \bar{n} の枚数が 1 枚、5 枚、10 枚、15 枚の時の結果をそれぞれ図 6, 7, 8, 9 に示す。どの開口形状も、ぼけの度合が大きくなるにつれて認識率が向上し、ぼけの度合が 10 を超えたあたりから認識率は 100% となった。ぼけの度合が 10 より小さい時の認識率は合成画像の枚数に依存していると言える。これは符号化開口だけではなく、円形開口でも同一の傾向が見られた。

合成画像の枚数が少なくても、ぼけの度合が 10 を超えると

認識率が 100% になった原因を調べるために、字種間の相関係数の 1 位と 2 位の差をぼけの度合ごとにプロットしたものを図 10 に示す^(注5)。ぼけが大きくなるにつれ、どの開口形状も字種間の類似度の差が狭まっているのが分かる。これはぼけが大きいために、クエリの文字が合成画像のぼけたパターンとも類似しているということであり、その結果、合成画像が少なくてもぼけの度合を正しく推定できたと考えられる。ぼけの度合を正しく推定できれば、文字の認識も容易であるため、同様の理由で円形開口でも、符号化開口と同程度の認識率を示すことができたと考えられる。しかしそれは言い換えると、ぼけの度合が大きくなると 2 位に誤認識し易くなるということであり、本稿で行ったノイズのない条件での実験で正しく認識できたとはいえ、実環境では誤認識が起こる可能性が高くなる。この傾向は円形開口で顕著に見られるため、開口の符号化なしに実環境で提案手法を実現することは困難な可能性がある。

開口形状ごとの比較を行うと、Veeraraghavan の開口形状が MURA コード以上の認識率を示す事が多かった。これは、ぼけの度合推定に成功した訳ではなく、ぼけの度合推定に失敗したとしても、認識に成功したという例が多かったためである。Veeraraghavan の開口形状は、仮にぼけの度合推定に失敗したとしても、図 10 に示すように、他の開口形状よりも文字が区別しやすいため、先の例が発生したものと考えられる。

次に、Veeraraghavan の開口形状における、ぼけの度合 10 までの平均認識率とクエリ 1 枚あたりの平均処理時間の関係を図 11 に示す。ぼけの度合を 10 までに絞ったのは、10 以降は概ね認識できることが分かったためである。平均認識率は合成画像の枚数が 20 枚の時に 100% になった。処理時間を見ると、合成画像の枚数と線形の関係にあることが分かる。処理時間と認識率の関係から、提案手法に用いる合成画像数 \bar{n} は 10 枚～20 枚が最適であるといえる。

6. 最適な開口形状の検討

本節では提案手法に適した開口形状について検討する。

まず、ぼけの度合い推定の重要性について検討する。式 (11)

(注5) : この結果を求める際には、大文字の I と小文字の l を同一の字種とした。

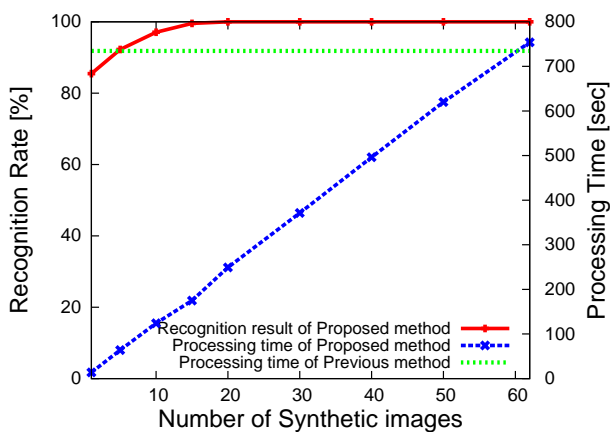


図 11 Veeraraghavan の開口形状におけるぼけの度合 10 までの平均認識率と処理時間

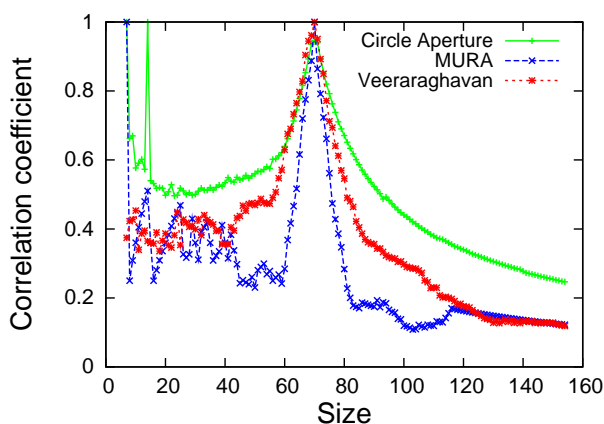


図 12 サイズが異なる開口形状同士の相関係数の推移

においてぼけの度合いを正しく推定することができれば、式 (12) において文字を正しく認識し易くなると考えられる。ぼけの PSF はぼけの度合いに基づいて開口形状が拡大されるものであるため、もし PSF のサイズが一致した時にだけ相関係数が高くなり、それ以外は低くなるような開口形状があれば、高精度なぼけの度合推定が実現できる。式 (11) と式 (13) が等価であると仮定した上で、大きさの異なる PSF を走査した時に相関係数のピークがどのように推移するかを図 12 に示す。これは、真の PSF のサイズを 70×70 [pixels] に設定して、同一の開口形状を 7×7 [pixels] から 1 ピクセルごとに拡大して走査した結果である。この結果から、MURA コードが他の開口形状に比べて、PSF の大きさが異なる時の相関係数が低くなっているため、ぼけの度合推定に適していると言える。しかし前節で述べた通り、ぼけの度合推定に最も適した開口形状と思われる MURA コードよりも、ぼけの度合推定の性能で劣る Veeraraghavan の開口形状の方が高い認識率を達成することが多かった。したがって、ぼけの度合推定の性能以外の性質も開口形状に求められると考えられる。

では、提案手法に適した開口形状が満たすべき要件とは何か。図 4 で示したように、Veeraraghavan の開口形状によってぼけた文字は MURA コードと比べて、文字の特徴が多く残っている。これが文字の認識に影響を与えているとみなすならば、満

たすべき要件は字種間の類似度の差ではないかと考えられる。改めて図 10 を見ると、Veeraraghavan の開口形状は MURA コードよりも類似度の差が大きかったため、認識率も高くなった可能性がある。

以上より、提案手法に適した開口形状は、ぼけの度合推定の精度もさることながら、字種間の類似度の差が大きいものであると考えられる。そのため、Veeraraghavan の開口形状よりも類似度の差が大きい開口形状を設計することで、更なる認識率向上を図れると考えられる。

7. まとめと課題

前報 [5] で提案したぼけた画像の文字を認識する手法には、処理時間の問題が存在した。この問題に対して、テンプレートの合成画像を用いたぼけの度合推定と、文字認識という 2 つのプロセスに処理を分割する手法を提案し、処理時間の削減を図った。実験の結果、提案手法は処理時間を大幅に削減しつつ高い認識率を維持した。

今後の課題として、提案手法に適した開口形状の設計や、文字のスケール変化に頑健にするための手法改良などが挙げられる。

謝辞 本研究は科研費補助金挑戦的萌芽研究 (23650090) の補助を受けた。ここに記して感謝する。

文献

- [1] 日浦慎作. コンピュータショナルフォトグラフィ理解のための光学系入門. 研究報告コンピュータビジョンとイメージメディア (CVIM), Vol. 2010-CVIM-171, No. 13, pp. 1-12.
- [2] 長原一. 符号化撮像. 研究報告コンピュータビジョンとイメージメディア (CVIM), Vol. 2010-CVIM-171, No. 14, pp. 1-9, 2010.
- [3] 武田祐一, 日浦慎作, 佐藤宏介, 符号化開口を用いた Depth from Defocus とステレオ法の融合. 画像の認識・理解シンポジウム (MIRU2012) 論文集, OS1-03, pp. 1-8, Aug. 2012
- [4] 河本悠, 日浦慎作, 浅田尚紀. 符号化開口を用いたバーコード画像のぼけ除去. 研究報告コンピュータビジョンとイメージメディア (CVIM), Vol. 2011-CVIM-177, No. 14, pp. 1-8, 2011.
- [5] 伊村政志, 岩村雅一, 日浦慎作, 黄瀬浩一. ピントのずれに頑健な文字認識手法の提案. 画像の認識・理解シンポジウム (MIRU2012) 論文集, IS1-72, pp. 1-8, Aug. 2012.
- [6] S. R. Gottesman and E. E. Fenimore. New family of binary arrays for coded aperture imaging. Applied optics, pp. 4344-4352, 1989.
- [7] A. Veeraraghavan, R. Rasker, A. Agrawal, A. Mohan, and J. Tumblin. Dappled photography: mask enhances cameras for heterodyned light fields and coded aperture refocusing. ACM Transactions on Graphics, Vol. 26, No. 3, p. 69, 2007.