社団法人　電子情報通信学会
THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

信学技報
TECHNICAL REPORT OF IEICE.

歪んだ文字の認識と自動ラベル付け
～大規模データベースの構築を目指して～

塚田　真規†　　岩村　雅一†　　黄瀬　浩一†

† 大阪府立大学大学院工学研究科　〒 599-8531 堺市中区学園町 1-1
E-mail: tsukada@m.cs.osakafu-u.ac.jp, {masa,kise}@cs.osakafu-u.ac.jp

あらまし　認識精度を向上させるために，膨大なデータ数を持ったデータセットは常に必要とされている．しかし，データ数が膨大なデータセットを作成するためには大量の画像にラベル付けする必要があり，コストや手間が非常にかかるといった問題が生じてしまう．本稿ではこの問題を解決し，膨大なデータセットを得るために，Self-corrective recognition algorithm に基づき自動でラベル付けを行い，学習する方法を提案する．この方法によって初めは認識できなかった歪んだ文字も認識することが可能となる．実験ではラベルなしデータを入力することで，どのような歪み文字が認識できたかを示し，またその精度についても示す．

キーワード　文字認識，self-corrective recognition algorithm，semi-supervised learning，大規模データセット，アフィン歪み

1. ま え が き

近年，スマートフォンなど小型端末のカメラ精度が向上し，普及が進んでいる．それに伴い，カメラを用いた様々な文字認識アプリケーションやサービスが求められている．このようなカメラの撮影画像から文字認識を行うアプリケーションの中には既に提供されているものもある．Google Goggles(注1)はスマートフォン用のアプリケーションであり，カメラの撮影画像から文字を認識し，その結果の翻訳などを行うことができる．Evernote(注2)は情景画像中の文字に対して検索のためのインデックスを作成し，キーワード検索によって目的の画像を検索するアプリケーションである．タンゴチュウ(注3)はユーザが撮影した画像から，写っている文字を認識するサービスである．このような情景中の文字認識を活用したアプリケーション，サービスは様々な応用が考えられる．

しかし現在提供されているアプリケーションは大きな問題を抱えている．それは人間が認識可能な文字と比べてコンピュータが認識可能な文字が非常に限られていることである．例えば情景中の文字はフォントや形状など多種多様であり，認識を困難なものにしている [1]．またカメラで撮影された文字画像の認識は射影変換，照明等の変化，オクルージョン，低解像度，ピンぼけなど様々な要因によって文字認識ができなくなってしまう．

このような多種多様な文字，変動を受けた文字を認識する有効な方法として事例ベースの方法が考えられる [2]．この方法ではあらかじめ大量のテンプレート画像をデータベースに蓄積しておく．そして認識対象となる文字画像と最も似ているテンプレート画像を探し出し，認識結果として出力する．事例ベース認識ではあらかじめ蓄積しておくテンプレート画像の数が大きいほど，認識精度が良くなることは明らかである．しかし，データベースを作成するために，どのようにして大量のテンプレート画像を集めるかが問題となる．

カメラベース文字認識の分野では，大規模データベースが不足している．現在利用できる最も大規模なデータベースは NEOCR である [3]．NEOCR は三ヶ月かけて手動でラベル付けされた 5,238 個の単語によって構成されている．より大きなデータセットとして Google Street View [4] から抽出された 100 万枚の数字データデットも近々公開されるだろうという発表があった．このデータセットの数字画像は Google Street View の番地画像から自動で抽出，認識され，最後に人間の手によって検証される．このようにデータセットを作成するためには人が介入する必要があり，当然ながら人が作業を行うにはコストや手間がかかってしまう．コストをより少なくするために，Amazon Mechnical Turk(注4) などを使う方法 [5] もあるが，さらに大規模なデータセットを作成するときには，検証などにコストがかかるため，根本的な解決策とはならない．よって，手動でラベル付けをするのではなく，人の介入を必要としない，自動でラベル付けを行う方法が必要となっている．

自動的に大規模なデータベースを作成するために Self-corrective recognition algorithm を用いる手法が考えられる [6, 7]．Self-corrective recognition algorithm では少数のラベル付きデータと多数のラベルなしデータを用いて学習する．

(注1)：http://www.google.com/mobile/goggles/
(注2)：http://www.evernote.com/about/intl/jp/
(注3)：http://tangochu.jp/
(注4)：https://www.mturk.com/mturk/

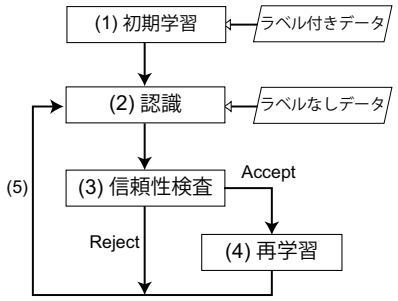
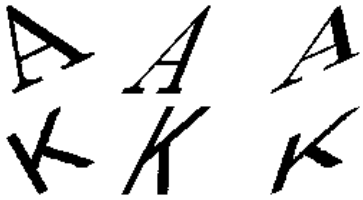


図 1: Self-corrective recognition algorithm の概要



(a) 回転 ($\theta = \frac{\pi}{4}$)　(b) シアー ($\phi = \frac{\pi}{6}$)　(c) 回転，シアー ($\theta = \frac{\pi}{4}, \phi = \frac{\pi}{6}$)

図 2: Century の 'A' と Arial の 'K' のアフィン変換の例

つまりラベル付きデータ数が少なくても識別精度を向上させることができ，自動的にラベル付けを行うシステムを実現できる．このアルゴリズムに基づいて学習を繰り返すことで，歪みなどで劣化した文字画像を認識することが可能となり，最終的に多種多様な文字を認識できる識別器とデータセットを得られる．この方法ではラベル付きデータとラベルなしデータは特徴空間上での分布が同一であることが仮定されている．しかし歪みなどで劣化した文字はそれぞれ異なった分布を持っているので，異なった仮定を設ける必要がある．

本稿では大規模データベースの構築と多種多様な文字を認識できる識別器の作成を目的として，ラベルなしデータに対して自動でラベル付けを行い，再学習に用いることによってラベル付きデータと類似していない歪んだ文字が認識可能となることを実験で示す．アルゴリズムとしては [6] と同様であるが，データに関しては異なった前提条件を設定する．前提条件として学習データとテストデータは特徴空間上の分布にずれが存在するとする．テストデータを認識するためにはラベルなしデータを学習し，このずれを補間する必要がある．実験ではこの考えが実現可能かを検証する．

2. 関 連 研 究

Self-corrective recognition algorithm は Semi-supervised learning の Self-training と見なすことができる [8]．

手書き文字認識では，Self-training を用いて認識精度を向上させる方法が提案されている [9–11]．しかしこの方法は他の Semi-supervised learning と同様にラベル付きデータとラベルなしデータが同一の分布を持つ仮定を用いている．

Graph-based semi-supervised learning はラベルなしデータを用いてデータ間を補間する方法 [8, 12] であり，距離の近いデータ間は同じクラスであるという仮定に基づいてラベルなしデータのクラスを推定している．提案手法はこの手法と同じ考えに基づいている．本稿ではラベル付きデータが各クラス一つであり，他は全てラベルなしデータを用いる．

3. Self-Corrective Recognition Algorithm

Self-correvite recognition algorithm の概要を図 1 に示す．処理手順は以下の通りである．(1) ラベル付きデータを用いて初期学習を行い，識別器を作成する，(2) 識別器でラベルなしデータを認識し，クラスを推測する，(3) 推測したクラスラベルの信頼性を調べる，(4) (3) の結果，クラスラベルが信頼できるとき，その結果を用いて識別器の再学習を行う，(5) (2)〜(4) を繰り返す．本稿での処理手順は [6] と同様であるが，データに関してはラベル付きデータとラベルなしデータとの間に大きな分布のずれがあるとする．

初期学習後の識別器は既学習のラベル付きデータに類似したデータしか認識できない．そのため類似していないデータのラベルは誤って推定される可能性が高く，類似していないデータを再学習に用いると精度が悪化する可能性がある．これを避けるためには，類似していないデータをなるべく再学習に使用しないことが重要である．そこで本稿ではラベルなしデータの入力を順序づけることを考える．すなわち，ラベルなしデータとラベル付きデータとの相関係数を求め，相関が高い順に入力していく．相関が高い文字は類似した形状を持つため認識可能であり，ラベル付けされた文字を学習することで更に歪んだ文字も認識可能となるはずである．実験 1 では信頼性検査を行わず，ラベルなしデータの入力順を制御することで歪み文字が認識可能かを確かめる．実験 2 ではデータをランダムに入力し，信頼性検査を用いたときに歪み文字を認識可能か調べる．

4. 歪み文字画像

4.1 アフィン変換

アフィン変換は二次元画像の幾何変換の一つである．回転，拡大縮小，シアーの変換を表すアフィン変換行列は以下のように表せる．

$$\boldsymbol{T} = \begin{pmatrix} \beta & 0 \\ 0 & \beta \end{pmatrix} \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} 1 & \tan\phi \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \alpha & 0 \\ 0 & \frac{1}{\alpha} \end{pmatrix} \tag{1}$$

ここで β，θ，ϕ，α はそれぞれスケール，回転，シアー，独立変倍のパラメータである．行列 $\boldsymbol{T}$ は座標 $\boldsymbol{x}$ を座標 $\boldsymbol{y}$ に $\boldsymbol{y} = \boldsymbol{T}\boldsymbol{x}$ によって射影する．本稿では，β，α を 1 に設定する．これはアフィン変換後の画像を正規化するためである．θ は $[-70, -69.75, -69.5, \ldots, +70]$ の範囲をとり，ϕ は $[-50, -49.75, -49.5, \ldots, +50]$ の範囲をとる．これらを組み合わせることで，それぞれの文字に対して 224,961(561×401) 種類のアフィン変換画像が得られる．アフィン変換を施した例を図 2 に示す．

4.2 相関グループ

実験でアフィン変換を施した画像と変換前のラベル付きデー

i. $[0.9, 1.0)$ ii. $[0.8, 0.9)$ iii. $[0.7, 0.8)$ iv. $[0.6, 0.7)$ v. $[0.5, 0.6)$ vi. $[0.4, 0.5)$ vii. $[0.3, 0.4)$ viii. $[0.2, 0.3)$ ix. $[0.1, 0.2)$ x. $(0.0, 0.1)$

図 3: 文字画像は各相関グループに含まれている歪み文字の例．上段は Century の ‘A’．下段は Arial の ‘K’．

タの相関係数を用いる．本稿では相関係数として Zero-mean Normalized Correlation Coefficient(ZNCC) を用いる．ZNCC は $[-1, 1]$ の値をとるが，本稿では相関係数が $[0, 1]$ の値をとる文字画像のみを用い，0 未満の相関係数を持つ文字画像は用いない．

相関係数を求めた後，その値に基づいて文字画像を 10 個の相関グループに分類する．図 3 はその分類例である．相関係数が $[0.9, 1.0)$ を持つ文字画像はグループ i に，$[0.8, 0.9)$ の範囲の相関係数を持つ文字画像はグループ ii に分類される．以下同様にして全ての文字画像を 10 個のグループに分類する．相関係数が大きくなると，その相関係数を持つ文字画像数は減少する．よってグループ i に含まれる文字画像数は最小となり，グループ x に含まれる文字画像数が最大となる．

5. 実　　験

5.1 実験条件

本稿では事例ベース認識を実現するため，近似最近傍探索に基づく最近傍識別器を用いた．近似最近傍探索法としては [13] を用いた．ハッシュベースの探索法を用いた理由はラベルなしデータを追加学習する際，木構造を用いる探索法と比べて高速に処理できるからである．

実験ではフォントとして Century と Arial を用い，認識対象はアルファベットの大文字とした．アフィン変換は 120pt の文字画像に対して行う．アフィン変換の後，変換画像を 64×64 画素に正規化し，二値化する．二値画像から画素値を各次元の値とした 4,096 次元の二値ベクトルを抽出する．4,096 次元の二値ベクトルでは近似最近傍探索において速度を十分に引き出せないため，主成分分析によって 4,096 次元の二値ベクトルを 40 次元の実数ベクトルに変換する．実験ではこの 40 次元の実数ベクトルを用いた．

5.2 実　験　1

実験 1 では，ラベルなしデータの入力順を制御したときの有効性について調べる．ラベルなしデータはフォント毎に，数の異なるデータセットを二種類ずつ用意し実験を行い，精度を向上させるために必要な条件も調べた．この実験では全てのラベルなしデータを学習し，信頼性検査は行わない．有効性を確かめるため，入力データの入力順を二種類用意し，比較する．一つは入力順を制御する方法であり，この方法を Controlled sequence と呼ぶ．これは相関係数の大きいグループ i から相関係数の小さいグループ x の順にラベルなしデータを入力する方法である．なおグループ内では文字画像はあらかじめシャッフルされている．もう一つの方法はラベルなしデータの入力順をランダムに設定する方法である．これを Random sequence と呼ぶ．識別器の性能を評価するためのテストデータとしてグループ i を除く各グループから文字画像を 100 枚ずつ選び，グループ毎に認識率を求めた．この実験で用いるラベルなしデータ数は Century のときは (a)71,500，(b)2,231,504 の二通り，Arial のときは (c)71,433，(d)2,179,850 の二通りの計四通り行った．

実験結果を図 4 に示す．Controlled sequence と Random sequence を比較すると，Controlled sequence の方が精度が高いことがわかった．Controlled sequence において，グループ ii を除いた全てのグループで認識率が向上している．Random sequence においてはグループ ix，x を除いたグループで認識率が低下してしまった．どのように再学習が行われているかより詳しく調べるために，正学習率 C，誤学習率 M，リジェクト率 R を以下のように定義する．

$$C = \frac{N_c}{N_c + N_m + N_r} \tag{2}$$

$$M = \frac{N_m}{N_c + N_m + N_r} \tag{3}$$

$$R = \frac{N_r}{N_c + N_m + N_r} \tag{4}$$

N_c は正しくラベル付けされて再学習に用いた文字数，N_m は誤った識別結果によりラベル付けされて再学習した文字数，N_r は再学習が行われなかった文字数を表している．Controlled sequence，Random sequence の C, M, R をそれぞれ表 1，2 に示す．この実験では信頼性検査を行わないため $R = 0\%$ となる．Controlled sequence の方が Random sequence よりも精度が良いため，ラベルなしデータの入力順を制御することは精度を向上させるために有効であることがわかった．また表 1，2 より Controlled sequence の方が C が大きく，M が小さいことから再学習を正しく行えていることがわかる．次に Controlled sequence の C と M を調べ，精度を向上させるための条件について考える．まず図 4(a) と図 4(b) を比較すると，図 4(b) の方が良い結果であり，精度を向上させるためにはより多くの文字を学習すれば良いことがわかった．初期学習後，特徴空間上のデータは非常に疎であるが，多くのデータを学習することによって空間上のデータが密になる．よってデータ間に存在する隙間を補間することができ，これはより多くのラベルなしデータを学習することによって識別器の精度を高めることを示している．図 4(c) と図 4(d) を比較すると，図 4(c) が良い結果であった．学習数の多い図 4(d) が学習数の少ない図 4(c) よりも精度が悪かった原因として相関グループの個数が不足していた可能性がある．相関グループの個数が不足していると，ラベル

表 1: 実験 1 での Controlled sequence の C, M, R

	正学習率 C[%]	誤学習率 M[%]	リジェクト率 R[%]
(a)	77.95	22.05	-
(b)	72.38	28.62	-
(c)	76.92	23.08	-
(d)	69.44	30.56	-

表 2: 実験 1 での Random sequence の C, M, R

	正学習率 C[%]	誤学習率 M[%]	リジェクト率 R[%]
(a)	34.44	65.56	-
(b)	13.29	86.71	-
(c)	33.85	66.15	-
(d)	14.16	85.84	-

なしデータを順序立てて学習できなくなり，歪み文字が正しく認識できなかったと考えられる．よって相関グループの個数を増やし，歪み文字の分類をより細かくしたときの実験も今後行う必要がある．

図 5 は再学習によって歪み文字が初期学習用データからどのような文字を経て認識されているかを示している．上段は Controlled sequnece において歪み文字の認識に成功した例であり，下段は Random sequence において歪み文字の認識に失敗した例である．Random sequence を用いたとき精度が下がってしまったが，これは誤ってラベル付けされたデータ数が多いためである．このような再学習の失敗を防ぐために，ラベルなしデータの入力順を制御する必要がある．

5.3 実　験　2

実験 2 では，信頼性検査にて再学習用データを選択したときの有効性について検証する．実験 1 の結果から，正しく再学習を行うためにラベルなしデータの入力順を制御することが有効であることがわかった．しかしこの方法は現実的には不可能である．なぜなら相関係数を求めるためにはラベルなしデータの真のクラスがわかっている必要があるからである．よって実験 2 では，実際の状況でもラベルなしデータを用いることによって歪み文字が認識できるか実験を行い，有効性を確かめる．つまり，入力順を Random sequence とし，誤って再学習をしないために信頼性検査を導入する．

信頼性検査では再学習に用いるラベルなしデータを選択する．その判定を行うために以下の式を定義する．

$$\frac{d_2}{d_1} > \text{Threshold} \tag{5}$$

d_1, d_2 はそれぞれラベルなしデータと最も近い近傍点との距離と 2 番目に近い近傍点との距離を表している．この式を満たすときのみ，識別器はラベルなしデータを学習する．

この節では閾値を固定した場合と，動的に変化させた場合の二種類の実験を行う．本節ではフォントが Century についてのみ実験を行った．まず閾値を固定した場合の実験を行った．設定する閾値は 1.5, 2.0, 2.5, 3.0 の四種類である．ラベルなしデータ数は 2,231,504 であり，これらはランダムに選ばれた．テストデータは実験 1 と同じデータを用いる．

表 3: 実験 2 で閾値を固定したときの C, M, R

閾値	正学習率 C[%]	誤学習率 M[%]	リジェクト率 R[%]
1.5	14.67	31.23	54.10
2.0	6.48	6.12	87.40
2.5	1.58	0.83	97.59
3.0	0.45	0.14	99.41

実験結果を図 6 に示す．またそれぞれの閾値における C, M, R の値を表 3 に示す．閾値が小さいほど，リジェクトされる文字数が減るため R の値が小さくなった．それに伴い，誤ってラベル付けされた文字による学習数も増えるため M の値が大きくなった．そのため閾値が小さいとき，例えば 1.5 のときは実験 1 の Random sequence に類似したものとなった．一方，閾値を大きくしたとき，例えば 3.0 のときでは再学習に用いるラベルなしデータ数が少ないため，初期状態からの変化が小さかった．閾値がこれらの中間であったとき，例えば 2.5 であったとき，認識率は向上した．この傾向はグループ viii，ix，x で顕著であった．これらの結果より C を大きく，かつ R を小さく保つことが精度を向上させるために重要であることがわかる．

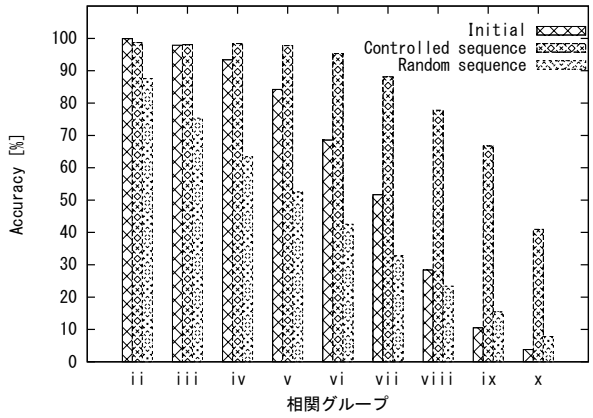
閾値を動的に変化させる実験では，閾値を減少させながら再学習を行う．初め閾値を高く設定しておくことで，M を低く保ったまま再学習を行うことができる．その後再学習に用いる文字数が少くなったとしても，閾値を減少させることによって，再び多くの文字を学習できるようになる．具体的には以下の不等式

$$\frac{N_{c-1}}{N_c} < 0.90 \tag{6}$$

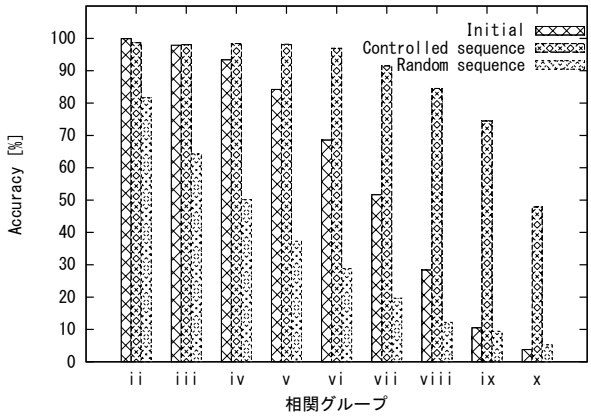
を満たしたとき，閾値を 0.3 ずつ減少させる．ここで N_c は c 回目のサイクルまでに再学習に用いた文字の総数である．図 7 は閾値の初期値を 2.8 に設定したときの結果である．最終的に閾値は 1 にまで減少した．これは入力した全てのラベルなしデータを学習したことを示している．そのとき $C = 64.50\%$, $M = 35.50\%$，$R = 0.00\%$ であった．閾値を動的に変化させたときの精度は，閾値を一定値に固定したときに比べて良くなった．そのため閾値を動的に変化させることが有効であるとわかった．閾値を動的に変化させたときは閾値を 1.5 に設定したときよりも M と R の値を小さくすることができた．

6. ま　と　め

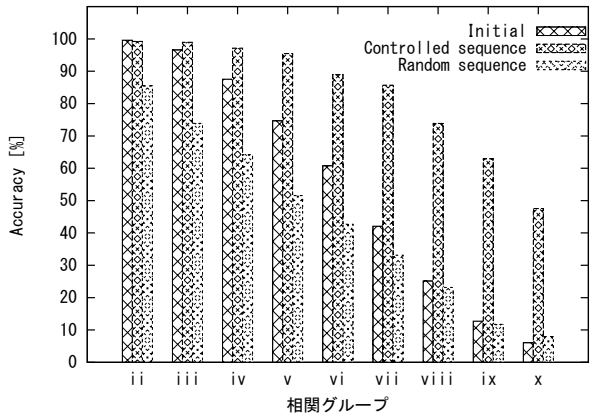
本稿では，大規模データベースの構築と多種多様な文字を認識できる識別器の作成を目的として，Self-corrective recognition algorithm に基づいて最初は認識できなかった文字でも自動的に認識可能とする方法を提案した．この方法を用いることで情景中の多種多様な文字やカメラで撮影することで生じる様々な変化を受けた文字であっても認識可能なシステムを実現することが可能となる．実験 1 では，ラベルなしデータの入力順を制御することによって初期学習に用いたラベル付きデータに類似していない歪んだ文字であっても認識ができることを示した．実験 2 では実際の状況に対応するために信頼性検査を用いた．その際，閾値を固定する方法，動的に変化させる方法を実験し



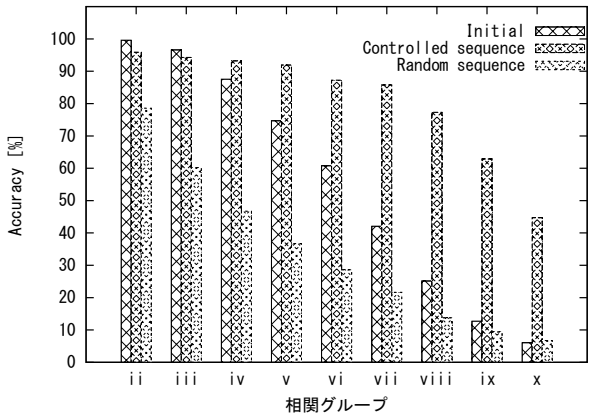
(a) フォントが Century，ラベルなしデータが 71,500 のとき.



(b) フォントが Century，ラベルなしデータが 2,231,504 のとき.



(c) フォントが Arial，ラベルなしデータが 71,443 のとき.



(d) フォントが Arial，ラベルなしデータが 2,179,850 のとき.

図 4: 相関グループと認識率の関係

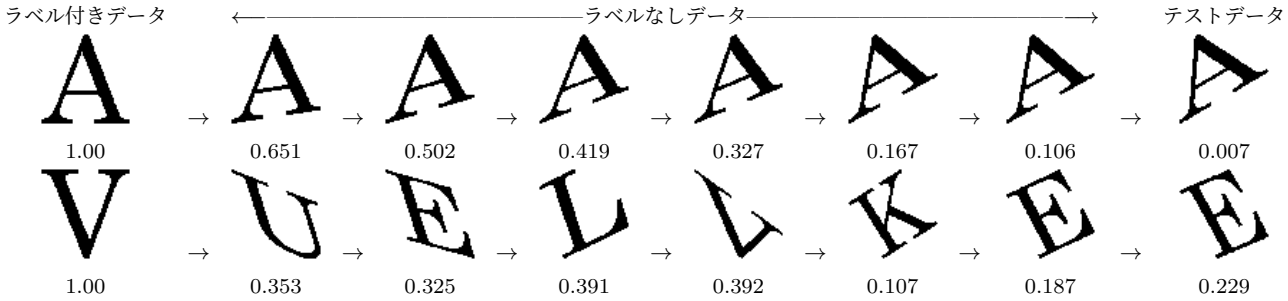


図 5: 再学習を行うことで，歪み文字がどのように認識されているかを示している．左端の文字画像がラベル付きデータであり，右端の文字画像はテストデータである．他の文字画像はラベルなしデータである．左端から 2 番目の文字画像は近似最近傍点として左端の画像を持ち，その結果を用いて再学習を行った．左端から 3 番目の文字画像は近似最近傍点として左端から 2 番目の文字画像を持ち，その結果を用いて再学習が行われた．このような再学習繰り返された後，右端の文字が認識された．上段は Controlled sequence を用いたときの結果 (真のクラス ‘A’) で，歪み文字を正しく認識している．下段は Random sequence を用いたときの結果 (真のクラス ‘V’) で，誤った再学習により誤認識を起こしている．各文字画像の下の数字は左端のラベル付きデータとの相関係数を表している．

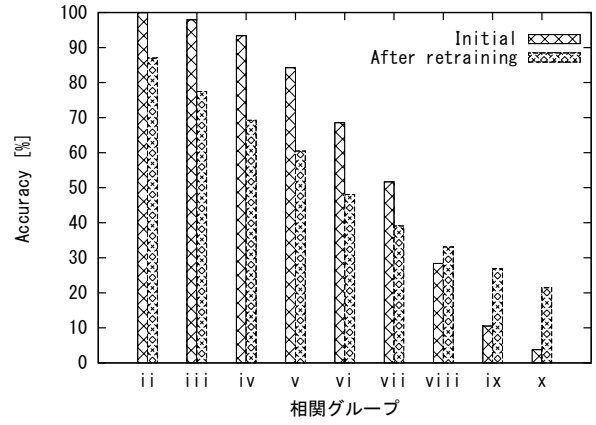
た．閾値を動的に変化させることで，閾値を固定したときよりも誤ってラベル付けされたデータによる学習を少なくし，より多くのデータを学習することができた．よって閾値を動的に変化させる方法が良いとわかった．

今後の課題として，変形によって分布が重なる文字に対応することである．例えば ‘b’ と ‘q’ は回転によって，‘L’ と ‘V’ はアフィン変換によって非常に類似した形状となる．このような文字は識別器によって区別が難しいため，正しく分離する方法を見つける必要がある．またラベルなしデータをより細かく分類し，相関グループの個数を増やして実験を行うことも今後の課題である．

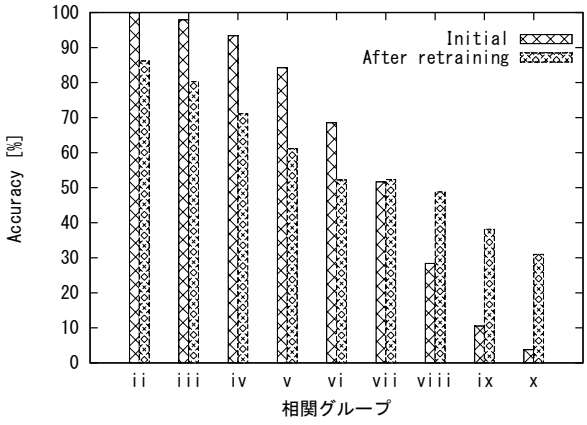
謝辞　本研究の一部は JST CREST の補助を受けた．ここに記して感謝する．

文　献

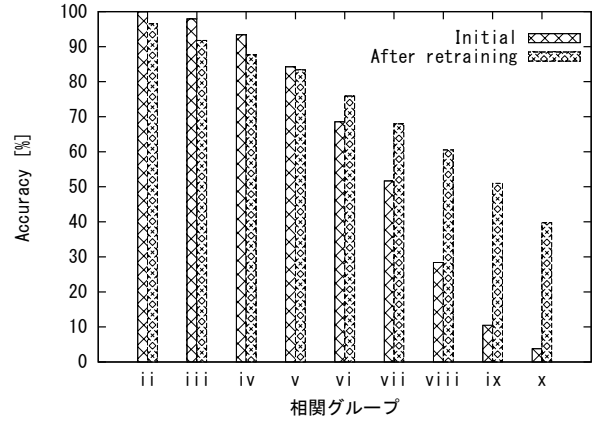
[1] T.E. de Campos, B.R. Babu, and M. Varma, “Character recognition in natural images,” Proc. International Conference on Computer Vision Theory and Applications, Lisbon,



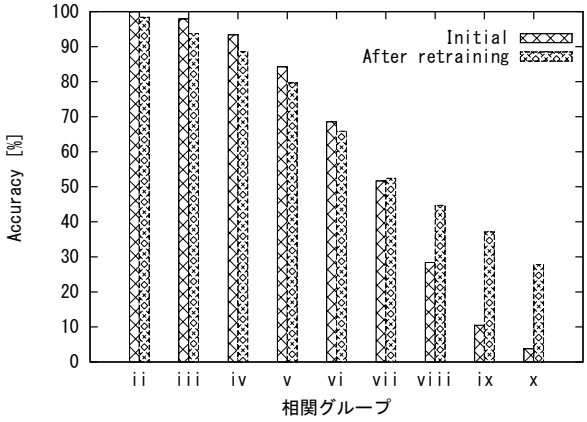
(a) Threshold = 1.5.



(b) Threshold = 2.0.



(c) Threshold = 2.5.



(d) Threshold = 3.0.

図 6: 信頼性検査を用いた時の相関グループと認識率

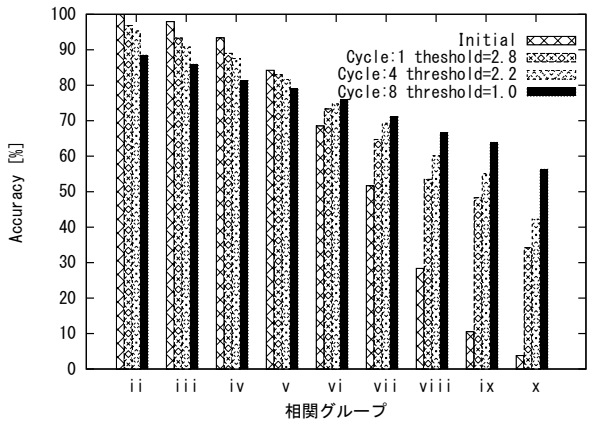


図 7: 閾値を動的に変化させたときの相関グループと認識率

Portugal, Feb. 2009.

[2] M. Iwamura, T. Tsuji, and K. Kise, “Memory-based recognition of camera-captured characters,” Proc. Document Analysis Systems (DAS2010), pp.89–96, June 2010.

[3] A.D. Robert Nagy and K. Meyer-Wegener, “NEOCR: A configurable dataset for natural image text recognition,” Proc. Camera-Based Document Analysis and Recognition (CBDAR2011), pp.53–58, Sept. 2011.

[4] A. Bissacco, “Reading text in google goggles and streetview images,” Proc. Camera-Based Document Analysis and Recognition 2011 (CBDAR2011), p.11, Sept. 2011.

[5] K. Wang and S. Belongie, “Word spotting in the wild,” Proc. European Conference on Computer Vision (ECCV2010): Part I, pp.591–604, Sept. 2010.

[6] G. Nagy and J. G. L. Shelton, “Self-corrective character recognition system,” IEEE Trans. Information Theory, vol.IT-12, pp.215–222, April 1966.

[7] G. Nagy and H.S. Baird, “A self-correcting 100-font classifier,” Proc., IS&T/SPIE Symp. on Electronic Imaging: Science & Technology, Feb. 1994.

[8] X. Zhu and A.B. Goldberg, Introduction to semi-supervised learning, Morgan and Claypool Publishers, Sept. 2009.

[9] V. Frinken and H. Bunke, “Self-training strategies for handwriting word recognition,” Proc. 9th Industrial Conference on Advances in Data Mining (ICDM ’09), pp.291–300, July 2009.

[10] V. Frinken and H. Bunke, “Evaluating retraining rules for semi-supervised learning in neural network based cursive word recognition,” Proc. International Conference on Document Analysis and Recognition 2009 (ICDAR2009), pp.31–35, July 2009.

[11] V. Frinken, A. Fischer, and H. Bunke, “Improving handwritten keyword spotting with self-training,” Proceedings of the 2011 ACM Symposium on Applied Computing, pp.840–845, March 2011.

[12] O. Chapelle, B. Schölkopf, and A. Zien eds., Semi-supervised learning, Cambridge, MA, MIT Press, 2006.

[13] 佐藤智一，岩村雅一，黄瀬浩一，“概算距離の精度向上による近似最近傍探索の高速化,” 信学技報 PRMU2011-67，Sept. 2011.