九州大学大学院
システム情報科学紀要
第9巻　第2号　平成16年9月

Research Reports on Information Science and
Electrical Engineering of Kyushu University
Vol.9, No.2, September 2004

粗密DPによる画像の弾性マッチングの高速化

宮崎洋光* ・ 内田誠一**・ 迫江博昭**

Fast Elastic Image Matching Algorithm Based on Coarse-to-Fine DP

Hiromitsu MIYAZAKI, Seiichi UCHIDA and Hiroaki SAKOE

(Received June 11, 2004)

Abstract: In image pattern recognition, elastic matching based on dynamic programming (DP) has been used as an effective technique to obtain a deformation-invariant distance between image patterns. A practical problem of elastic matching is its huge computation time. In this report, a fast elastic matching technique based on coarse-to-fine DP (CFDP) is proposed. In CFDP, a heuristic search strategy is employed to reduce computation time while keeping the global optimality of matching. The effect of the proposed technique on reducing computation time was indicated by experimental results.

Keywords: Image pattern recognition, Coarse-to-Fine DP, Elastic matching

1. はじめに

弾性マッチング法とは，2つの画像パターンの最適な非線形整合を目的とした画素対応関係の最適化問題として定式化される．この画像対応を組み合わせ的に扱う場合，動的計画法(DP)により最適解が求まる場合が多い．変形を補償した後の2画像間のマッチング距離が得られるので，この距離を用いることにより変形に対して頑強な識別系を実現できる．

DPに基づく弾性マッチングは，様々な制約条件が扱えるなどの利点がある一方，問題点として計算量の多さが挙げられる．この計算量低減のための一手法として，ビームサーチ法[1),2)]の利用が考えられる．ビームサーチ法とは計算過程において枝刈りを行い，準最適解を求める手法である．しかし，最適解からの逸脱の程度によっては認識率の低下を招く場合がある．

本研究では，DPに基づく弾性マッチングの高速化手法として，粗密DP(Coarse-to-fine DP)[3)]の利用を検討する．粗密DPは，A*アルゴリズム[4)]のようにヒューリスティックを利用して効率化を図りながら厳密な最適解を求める手法である．具体的には，最適解を与えると思われる個所を段階的に細かく探索することで，厳密な最適解を得ることができる．また，さらなる高速化を実現するために，分割方式による比較，低解像度画像の併用も検討する．以下，本論文では，粗密DPと区別するために，従来のDPを単純DP（plain DP）と呼ぶ．

実験で明らかになるように，本手法による計算量効果は平均して50%程度であり，前述したビームサーチのような1桁以上の高速化は図れていない．しかしながら，本手法はあくまで単純DPで求まる最適解と等価な解を保証した高速化手法である点でビームサーチ法とは目的が異なっている．さらに，本手法についてもビームサーチ法の組み込みを考えることもでき，この場合，単純DPにビームサーチを組み込んだものよりも効率のよいアルゴリズムが期待される．

本論文では，弾性マッチング法として比較的単純な手法[5),6)]を用いる．ただし，本手法の考え方自体は，単純DPに基づく他の弾性マッチング手法にも利用できる．

平成16年6月11日受付

* 知能システム学専攻修士課程
** 知能システム学部門

2. 単純DPと粗密DP

本節では，本手法の説明の準備として，単純DPおよび粗密DPの一般的な原理を述べる．

2.1 単純DPの原理

Fig. 1は, 各段にN個の状態が格子状に並んだトレリスと呼ばれる有向グラフである．ここで，第i段の状態集合を$S_i = \{(x \mid i) \mid x = 1, \ldots, X\}$，その状態の局所距離を$d(x \mid i)$と定義する．この第1段から第$N$段までの最適経路

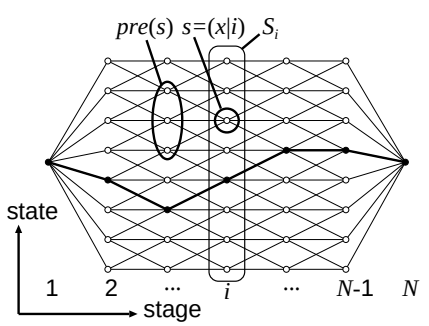

Fig. 1 Stage, states, and optimal path.

問題を考える．すなわち，$\sum_{i=1}^{N} d(x \mid i)$の最適化問題を考える．

Fig. 1のように便宜上1つの状態を$s \in S_i$で表すとすると，単純DPによりこの最適経路問題を解く場合，第1段から順次，各状態sの最小累積コスト$g(s)$を求めていく．最小累積コストは，各状態の局所距離とその状態に先行可能な前段の状態集合$pre(s)$ の最小な最小累積コストとの和で与えられる．

$$g(s) = d(s) + \min_{s' \in pre(s)} g(s') \tag{1}$$

ただし，第1段については$g(s) = d(s)$とする．このとき，第N段の最小累積コスト$g(s)$が最適解となり，最適解を与える経路が最適経路となる．

2.2 粗密DPの原理

粗密DPにより，**Fig. 1**の最適経路を求める場合，各段について状態$s \in S_i$を幾つかずつにまとめた複合状態$S_{i,l} = \{(x \mid i) \mid x = x_l^{min}, \ldots, x_l^{max}\}$を考える$(l = 1, \ldots, L)$．各段において，全ての状態はいずれかの複合状態に含まれ，重複することはない．すなわち，

$$\begin{cases} \cup_{l=1}^{L} S_{i,l} = S_i \\ S_{i,l} \cap S_{i,l'} = \emptyset \qquad (\text{for } l \neq l') \end{cases} \tag{2}$$

さらに準備として，複合状態$S_{i,l}$の局所距離には，そこに含まれる状態$s \in S_{i,l}$の局所距離$d(s)$の下限値$\tilde{d}(S_{i,l})$を与えることとする．

$$\tilde{d}(S_{i,l}) \begin{cases} = d(s) & (|S_{i,l}| = 1) \\ \leq \min\limits_{s \in S_{i,l}} d(s) & (\text{otherwise}) \end{cases} \tag{3}$$

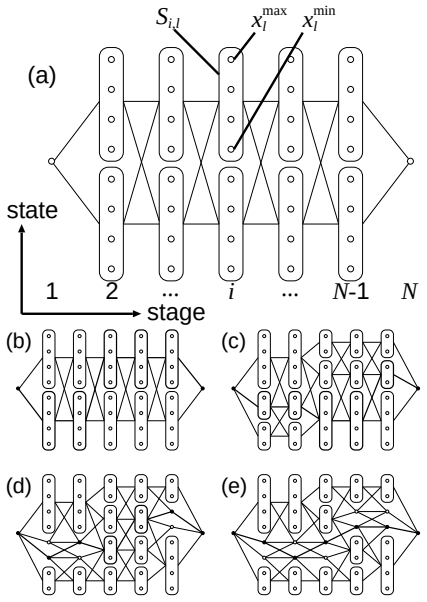

Fig. 2　Optimization process of coarse-to-fine DP.

このように，$|S_{i,l}| = 1$の場合は厳密な局所距離を与える．

この複合状態についてトレリスを構築し(**Fig. 2**(a))，その上で単純DPを用い，最適経路を得たとする(**Fig. 2**(b))．粗密DPでは，この最適経路上の複合状態を分割し，新たな複合状態を生成する．生成された複合状態には，新たに局所距離下限値$\tilde{d}(S_{i,l})$を与える．ただし，それ以上分割ができない複合状態$(|S_{i,l}| = 1)$については分割を行わない．そうして再び単純DPを用いる(**Fig. 2**(c))．この単純DPと分割の反復の結果，最適経路上のすべての複合状態が分割不能であった場合，その経路は最適経路となり，最適解が求まる(証明は文献3))．

粗密DPにおいて，計算量低減のためには，(i)局所距離下限値の精度，および(ii)その計算量，(iii)反復回数の低減が検討の鍵となる．特に反復回数については，最良の場合と最悪の場合で大幅な差が生じる．ここで，最良の場合とは各反復で同じ複合状態だけが分割されることであり，最悪の場合は1回の反復で1複合状態だけが分割されることである．したがって，こうした最良の場合に持っていくような工夫が必要である．

3. 粗密DPに基づく弾性マッチング

3.1 画像の弾性マッチング

3.1.1 定式化

弾性マッチングには様々な手法があるが，本報告では，画像Aの各列の両端点を制御点とし，それらの最適な写像先を画像B上に探索する手法[5]を用いる．この手法では**Fig. 3**に示すように，サイズ$N \times N$の画像$A = \{a_{i,j} \mid i, j = 1, \ldots, N\}$の第$i$行が，画像$B = \{b_{x,y} \mid x, y = 1, \ldots, N\}$の画素$(x_i^L, 1)$と$(x_i^R, N)$を両端とする線分に写像される．この際，同列上の画素

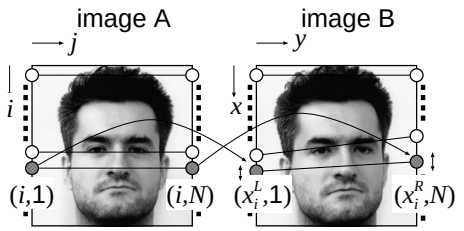

Fig. 3　Elastic image matching.

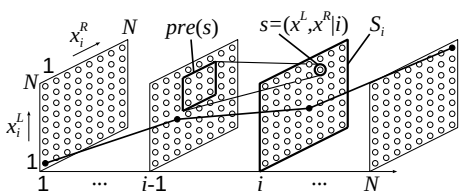

Fig. 4　Calculation space of plain DP for elastic image matching.

(i,j) $(j=1,\ldots,N)$の対応先は線形補間により定められるものとする．すなわち，

$$(i,j) \mapsto \left((x_i^R - x_i^L)j/N + x_i^L,\ j\right) \tag{4}$$

このとき，弾性マッチング法の最適化問題は次式のように定義される．

$$[\text{目的関数}] \sum_{i=1}^{N}\sum_{j=1}^{N} \left| a_{i,j} - b_{(x_i^R - x_i^L)j/N + x_i^L,\ j} \right| \rightarrow \text{minimize} \tag{5}$$

$$[\text{制御変数}] \{x_i^L, x_i^R \mid i = 1,\ldots,N\} \tag{6}$$

$$[\text{制約条件}] \begin{cases} 0 \le x_i^p - x_{i-1}^p \le 2 & \text{•• 単調連続性} \\ i - w \le x_i^p \le i + w & \text{•• 範囲制限} \\ x_1^p = 1,\ \ x_N^p = N & \text{•• 境界条件} \end{cases} \tag{7}$$

ここで，pはLまたはRを表し，wは正定数である．

以上の弾性マッチング法は，2画像間の画素対応を部分的な線形近似の下で最適化する手法であり，制御変数の数を非常に少なく抑えた最も単純な弾性マッチング法であるといえる．なお以下に述べる粗密DPの組み込みは，より一般的な弾性マッチング法にも適用可能である．

3.1.2 単純DPアルゴリズム

Fig. 4は，弾性マッチング法における，単純DPでの計算空間を示す．2.1節でトレリスとして示した状態空間は2次元的であったが，弾性マッチング法では，各i段について，それぞれ独立したx_i^L, x_i^Rで表される状態$s = (x^L, x^R \mid i)$が存在するので3次元的な表現となる．その中で最適経路を求める．つまり，$D(A,B) = \sum_{i=1}^{N} d(s)$の最小化問題を考える．$D(A,B)$は，弾性マッチング法により変形を吸収した2画像間の距離である．ここで，$d(s)$は状態sの局所距離であり，以下の式で与えられる．

$$d(s) = d(x^L, x^R \mid i) = \sum_{j=1}^{N} \left| a_{i,j} - b_{(x_i^R - x_i^L)j/N + x_i^L,\ j} \right|$$

また，$g(s)$は，第1列に始まり第i列が$(x_i^L, 1), (x_i^R, N)$間の線分に写像されるまでの最小累積コストを格納するワークエリアであり，DP漸化式は以下の式となる．

$$g(s) = d(s) + \min_{s' \in pre(s)} g(s')$$

$pre(s)$は制約条件(7)により規定される．

3.1.3 計算量

Fig. 4からわかるように，DP漸化式はN(段数)$\times N^2$(各段の状態数)回計算される．またDP漸化式一回あたりの計算量は，局所距離$d(s)$の計算に$O(N)$，最小値選択に$|pre(s)| \le 9 = O(1)$を要するので，まとめて$O(N)$である．以上より，アルゴリズム全体の計算量は$O(N^4)$となる．

3.2 粗密DPに基づく弾性マッチングの基本構成

3.2.1 概要

Fig. 5に粗密DPを用いた場合の弾性マッチング法の計算空間を示す．ここで，小さい白丸は元々の状態sを表す．囲まれた面の中の大きな丸は，その面に含まれる状態をまとめた，複合状態$S_{i,l} = \{(x^L, x^R \mid i) \mid x^L = x_l^{L,\min}, \ldots, x_l^{L,\max},\ \ x^R = x_l^{R,\min}, \ldots, x_l^{R,\max}\}$を表す．この計算空間内において，粗密DP内での単純DP計算と分割の反復を繰り返す．また，以下では$N = 2^n$(nは自然数)を仮定し，結果的に，粗密DP計算空間の複合状態は正方形領域をなす．複合状態の幅を$Q = |S_{i,l}|$と表す．

複合状態の分割は，x_i^L方向に2分割，x_i^R方向に2分割，計4分割し，新たな複合状態が4つ生成される．結果的に新たに生成された複合状態には，新たに局所距離

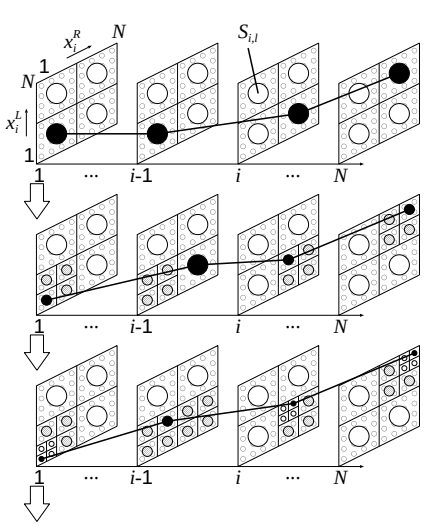

Fig. 5 Optimization process of coarse-to-fine DP for elastic image matching.

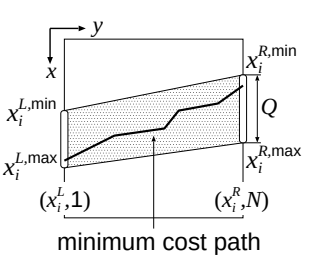

Fig. 6 Calculation of lower bound $\tilde{d}(S_{i,l})$ by searching for the minimum cost path in the region specified by $S_{i,l}$.

下限値を与える．なお，いずれの方向にも分割できない場合，その複合状態は元々の状態まで分割されているので，分割を行わない．なお，各i段の初期状態は，４つの複合状態とする．

3.2.2 局所距離下限値計算

複合状態$S_{i,l}$の局所距離下限値$\tilde{d}(S_{i,l})$を求める方法には，様々なものが考えられる．本手法では，複合状態の局所距離下限値を，線分から直線性の制約を外し，単調連続性な曲線分に関する距離の下限値とする(**Fig. 6**)．これは，(3)式を満たすので，解の最適性は保証される．計算量は，N(y方向の画素数)$\times Q$(探索の幅)$= O(NQ)$を要する．

3.2.3 計算量

反復一回あたりの下限値計算に要する計算量は，$4N$(新たに生成された複合状態数)$\times NQ$(下限値計算)$= 4N^2Q$となる．第k回反復におけるDP漸化式は，$4N$(初期の複合状態数)$+3Nt$(第t回反復までに増えた複合状態数)となる．したがって，最適経路が得られるまでの反復回数をTとすると，反復終了時までの計算量は，$\sum_{t=0}^{T}\Big((4N+3Nt)+4N^2Q\Big) = O(NT^2)+O(N^2QT)$となる．

3.3 単純 DP と粗密 DP の比較実験

弾性マッチング法における，粗密DPの利用の性能を把握するために，濃淡顔画像[7)]を対象とした実験を行った．計算時間は，画像Aを10パターン，標準画像Bを10パターンを用い，100通りのマッチング時間の平均を測定した．画像サイズは，16×16，32×32，64×64，の３通りとした．画像間のマッチング特徴には，各画素の輝度値，x,y方向のエッジ強度の計３特徴を用いた．実験には，Xeon$^{\mathrm{TM}}$ プロセッサ 3.20GHz，メモリ2GBのＰＣを用いた．なお以降の実験でも，同じＰＣを用いた．

実験結果を**Fig. 7**に示す．$N=32$の場合に限り低減されているが，全体をみると，Nが大きくなるに従い，計算量が増加する傾向がみられる．この計算量増加の原因は，計算終了までの反復回数(T)が多いためである．実験的に，反復回数は$T=O(N^2)$となることがわかった．よって，粗密DPにおける計算量は，$O(N^5)+O(N^4Q)$となる．これは単純DPにおける計算量$O(N^4)$ を大幅に超えており，優位性が得られなかったものと考えられる．

反復回数が多い原因は，効率の悪い分割があるためと考えられる．粗密DPでは，各反復において分割される複合状態は，最終的な最適経路を通る状態を含んでいることが望ましい．最良の場合(同じ複合状態だけが次々に細分化されていく場合)，反復回数は$\log Q$で済むので，計算量は，大幅に低減される．しかし実験を行った際の観察によると，効率の悪い分割（１回の反復で分割される複合状態が１つ，最終的な最適経路を通らない複合状態の分割）が多いことがわかっており，その結果，反復回数は増えているものと思われる．

さらに追求して，このような効率の悪い分割が起こった原因を考えると，ワープの単調連続性が考えられる．この制約により，分割前には先行可能であった複合状態が，分割後では先行不可能になってしまう問題が起こる．つまり，最適経路であった複合状態同士の繋がりがなくなり，集中した分割がなされなくなる．このため，次に選ばれる最適経路が前回の最適経路と異なる場合が多く，したがって計算終了までの反復回数が増加する．しかしながら，単調連続性はワープに位相保存性を与える重要な制約であり，排除すれば良いというものではない．

他の原因として画像の曖昧性が考えられる．この曖昧性が生じると，各複合状態同士の局所距離下限値の差が小さくなる．一方，ある複合状態が分割されると，分割により探索範囲が狭まるために，局所距離下限値が，分割前の複合状態よりも増加する．したがって，次の反復で同じ部分の複合状態が最適経路に選ばれず，違う部分が分割される可能性が高くなる．結果，近傍の複合状態が平均的に分割されていき，反復回数が増えてしまう．なお，この問題に対して，反復毎に近傍の複合状態を過剰に分割するという改良を試みた．付録Aに示すが，大幅な計算量低減の効果は得られなかった．

また，**Fig. 8**に$N=64$の場合のマッチング画像パターン別による単純DPとの計算量の比較をヒストグラムに示

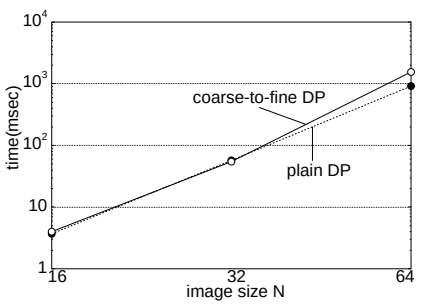

Fig. 7 Comparison between coarse-to-fine DP and plain DP on computation time.

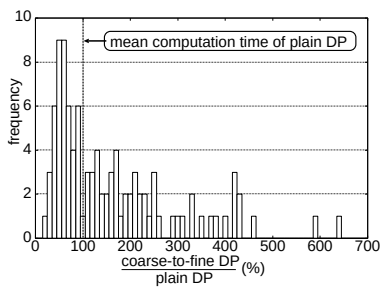

Fig. 8 Histogram of computation time reduction rate ($N=64$).

す．このように，マッチングする画像毎に計算量が変動している．これは，前述したようなマッチングに用いる2画像の曖昧性が原因だと考えられる．

4. 低解像度画像を併用した高速化

4.1 概要

粗密DPでの解法は，計算終了時までに多くの反復を要する．この原因は，3.3節で前述したように，画像の曖昧性により最終的な最適経路を通らない複合状態も分割してしまう点にある．本節では，低解像度画像[8]を用いることで，この画像の曖昧性により生じる，複合状態の平均的な分割に要する計算時間を低減する手法を検討する．

これまでの初期状態は，各段の状態をそれぞれ4つの複合状態にまとめたものとした．本節で与える改良手法では，すでに複合状態が粗く分割されている段階を初期状態とする．この粗い分割を行うために，低解像度画像を利用する．顔画像を含め多くの画像パターンでは，原画像と低解像度画像の類似性により，原画像同士のマッチングによる最適経路と低解像度画像での最適経路がほぼ重なり合うと考えられる．**Fig. 9**はその例証である．このため，低解像度画像について最適化した時の複合状態の様子が，原画像を粗密DPにより最適化する過程の複合状態の様子とほぼ重なり合う．その結果，通常の粗密DPでの，初期状態から途中の段階までの計算時間を低解像度画像の利用により低減できると考えられる．

この考えに基づくアルゴリズムでは，まずマッチングの対象となる原画像の$1/2^n$低解像度画像について粗密DPによる解法を行う．次に，計算終了時の粗密DP計算空間内の複合状態を，$1/2^{n-1}$低解像度画像による粗密DP計算空間に更新する．この際，第i段の更新先は第$2i$段と第$2i-1$段とする(**Fig. 10**)．更新後の複合状態の局所距離下限値については，再計算を行う．そして，それを初期状態として，$1/2^{n-1}$低解像度画像での粗密DPによる解法を行う．最終的に，原画像と同じ解像度での粗密DP計算空間において，最適化を行う．

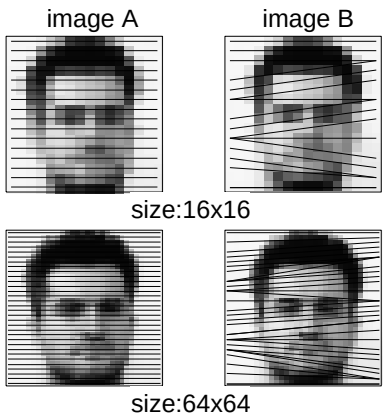

Fig. 9 Results of optimal matchings on different image resolutions.

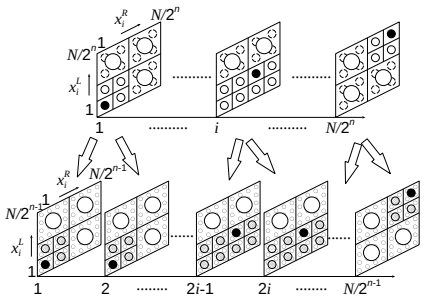

Fig. 10 Redefinition of super-states at increasing image resolutions.

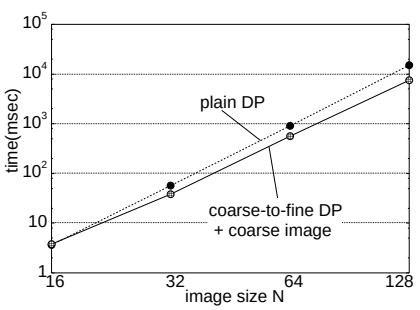

Fig. 11 The effect of the use of coarse image on computation time.

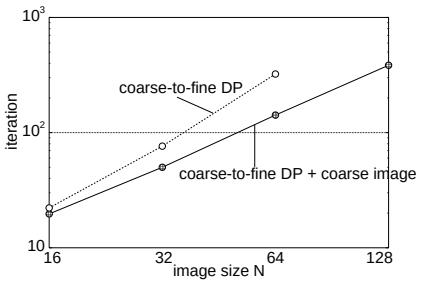

Fig. 12 The effect of the use of coarse image on the number of iterations.

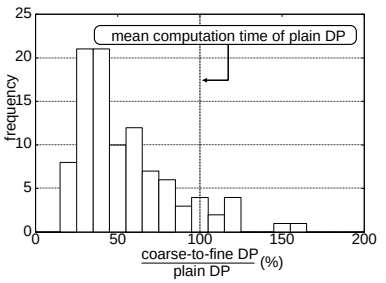

Fig. 13 Histogram of computation time reduction rate ($N = 64$).

4.2 低解像度画像の併用による効果の検証

3.3節と同じ顔画像を用いて実験を行った．実験結果を**Fig. 11**に示す．また，**Fig. 12**に反復回数，**Fig. 13**に$N = 64$の場合のマッチング画像パターン別による単純DPとの計算量低減率のヒストグラムを示す．

Fig. 11より単純DPでの計算量に比べ，大幅に低減されている．また，**Fig. 12**より反復回数も改良前の粗密DPに比べ，低減がみられる．したがって，4.1節に前述したような画像間の類似性による計算量の低減効果があったと考えられる．

Fig. 13より，改良前と比べ，画像による計算量の変動が小さくなっていることがわかる．特に，改良前では低減率200%を超えている場合が散見されるが，改良後にはなくなっている．

5. ま　と　め

単純DPに基づく弾性マッチングの高速化手法として，粗密DPの利用および改良手法について検討した．実験の結果，粗密DPのみを利用することでの計算量低減効果は確認することはできなかった．そこで，低解像度画像を併用したところ，結果的に単純DPに比べ，計算量低減効果を確認することができた．

将来的な課題としては，より変形自由度が高く計算量の多い弾性マッチング手法への適用を考えている．こうした手法は，状態数も多くなるため，粗密DPによる探索の効率化が有効であり，計算量低減効果が得られるものと考えている．また，本手法とビームサーチの併用も考えている．

参　考　文　献

1) 迫江博昭，藤井浩美，吉田和永，亘理誠夫，“フレーム同期化，ビームサーチ，ベクトル量子化の統合によるＤＰマッチングの高速化，” 電子情報通信学会論文誌，vol.J71-D，no.9，pp.1650-1659，Sep. 1988.
2) Seiichi Uchida and Hiroaki Sakoe，“An efficient two-dimensional warping algorithm，” IEICE Trans. Inf. & Syst.，vol. E82-D，no. 3，pp.693-700，Mar. 1999.
3) Christopher Raphael，“Coarse-to-fine dynamic programming，” IEEE Transactions on Pattern Analysis and Machine Intelligence，vol.23，no.12，pp. 1379-1390，Dec. 2001.
4) 白井良明，人工知能の理論，コロナ社. 1992.
5) 内田誠一，モハマッド．アサッド．ロニー，石田敏之，迫江博昭，“区分線形 2 次元ワープによる手書き文字の変形吸収の試み，” 信学技報，PRMU 99-228，pp.67-74，Feb. 2000.
6) 石田敏之，内田誠一，迫江博昭，“手書き文字認識における字形変動処理に関する一考察 –ダッチロールワープ–，” システム情報科学研究科報告，vol.5，no.1，pp.99-104，Mar. 2000.
7) http://www.iam.unibe.ch/~fkiwww/staff/achermann.html
8) Hava Lester and Simon R. Arridge，“A survey of hierarchical non-linear medical image registration，” Pattern Recognition，vol.32，pp.129-149，1999.

付　　録

A 過剰分割による効果の検討

3.3節での実験により，最終的な最適経路上の状態の近傍の複合状態も平均的に分割されることが，反復回数増加の原因として挙げられた．そこで本節では，反復毎に最適経路に選択された複合状態に加えその近傍も分割する方式を組み込み，その効果の検討を行う．過剰分割を行うことで，上記した平均的な分割を少ない反復回数で実現することが可能であり，総反復回数の減少とともに計算量低減効果が期待できると考えられる．

実験の結果を**Fig. 14**に示す．結果より，Nが大きくなるほど計算量低減効果が大きくなる傾向が見てとれる．しかし，4節で述べた改良手法ほどの低減効果は得られなかった．**Fig. 15**は各反復毎における総複合状態数を表した例証である．このように分割方式を変更したことで，反復回数を半減することができた．しかし過剰に分割を行うため，総複合状態数が増えてしまっている．このため，反復毎での最適経路選択に要する計算時間が増加してしまい，反復回数が減少されても全体としての計算量低減には至らなかった．

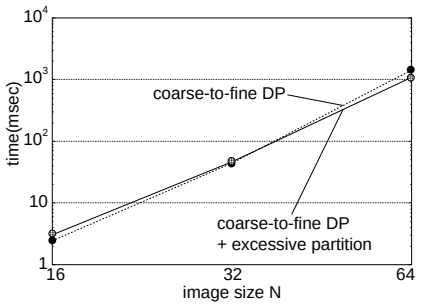

Fig. 14 The effect of excessive partitioning on computation time.

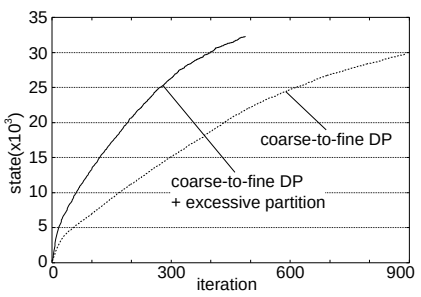

Fig. 15 Total number of super-states at each iteration ($N = 64$).