

1次元パターンの解析的DPマッチング

内田 誠一[†] 迫江 博昭[†]

[†]九州大学大学院システム情報科学研究所 〒812-8581 福岡市東区箱崎 6-10-1

E-mail: †{uchida,sakoe}@is.kyushu-u.ac.jp

あらまし パターン認識・画像処理において多用される弾性マッチング手法に動的計画法によるマッチング,いわゆるDPマッチングがある. DPマッチングは離散化された最適化問題の幅優先探索に基づく解法であり,従って探索の幅が非常に大きくなる問題に対しては適用困難であった. この問題を解決すべく本論文では解析的DPマッチングを提案する. 本手法では,マッチングの評価に用いられる局所的な誤差関数を2次関数近似表現することで,幅優先探索なしに厳密解を解析的に与えることができる. 本論文では1次元パターンに対するマッチングアルゴリズムを導出し,さらに実際の問題に適用し得ることをオンライン文字データを用いて実験的に検証する.

キーワード 動的計画法, DPマッチング, 解析的解法, オンライン文字認識

Analytical DP Matching for One-Dimensional Pattern

Seiichi UCHIDA[†] and Hiroaki SAKOE[†]

[†] Faculty of Information Science and Electrical Engineering, Kyushu University

Hakozaki 6-10-1, Higashi-ku, Fukuoka-shi, 812-8581 Japan

E-mail: †{uchida,sakoe}@is.kyushu-u.ac.jp

Abstract DP (dynamic programming) matching is one of the most fundamental techniques for various problems of pattern recognition and image processing. In this paper a novel DP matching algorithm, called analytical DP matching, is proposed. Conventional DP matching is organized as a breadth-first search algorithm. Thus, its computational complexity depends on the search width. In contrast, analytical DP matching has a different organization; specifically, it is an analytical solution method and can provide optimal matching with computational complexity which does not depend on the search width. The derivation of the algorithm and performance evaluation results are discussed in this paper.

Key words dynamic programming, DP matching, analytical solution, on-line character recognition

1. ま え が き

動的計画法に基づくパターンマッチング法 — DPマッチング — は,音声パターンの非線形対応付け(弾性マッチング)を目的として1970年代に開発された[1]~[3]. その後応用範囲は広がり,今ではパターン認識・画像処理における基本的道具の一つとなっている[4]. 特に,音声をはじめとする1次元時系列パターン(オンライン文字パターン,ジェスチャパターン)の弾性マッチング問題に対しては,ほとんどすべての場合においてDPマッチング(もしくはその確率的拡張であるHMM)が適用されていると言っても過言ではない.

このDPマッチングの特徴は,長さ I と J の2つの1次元パターンに対し,それらの大局的に最適な非線形対応付けを $O(IJ)$ の計算量で求める点にある. オンライン文字パターンを例にすると,パターン長は文字パターンを表現する筆点数に相当し,高々10~100程度である. 従って非常に高速に最適解が

求まる. この計算量の少なさが,DPマッチングが広く利用されてきたことの一つの理由でもある.

しかし,このことは I や J が非常に大きな問題にはDPマッチングを事実上適用できないことを意味する. 例えば,2次元パターンの弾性マッチング問題をDPで解く場合, J は画像のサイズ(画素数)に対して指数的に増加する[5],[6]. このためこれまで2次元パターンの弾性マッチング問題をDPで解く場合,マッチング自由度の部分的な制約や,ビームサーチなどの近似解法が援用されてきた.

本論文では,従来扱いにくかった問題にも適用し得る新しいDPマッチングアルゴリズム — 解析的DPマッチング — を提案する. 本手法では,原問題を一部連続近似して微分可能となるように工夫する. その結果,最適対応付けが微分に基づいて解析的に計算できるようになり,さらにその計算量は J に依存しない. 本論文では1次元パターンを対象とした解析的DPマッチングアルゴリズムを導出し,さらに実際の問題に適用し

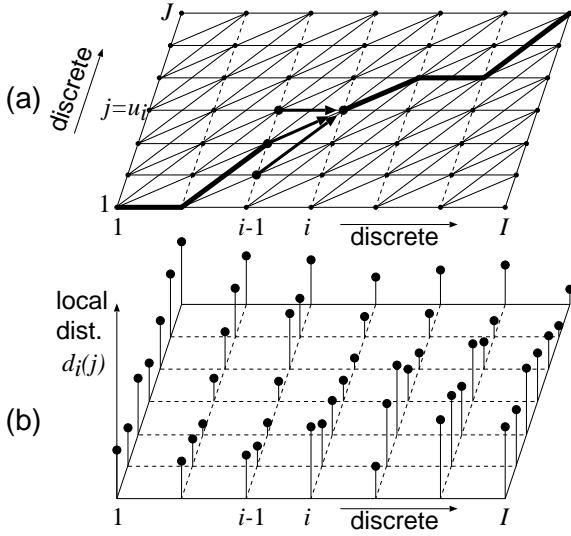


図1 従来の DP マッチング. (a) 最適経路探索空間としてのトレリス. (b) 局所距離.

得ることをオンライン文字データを用いて実験的に検証する.

2. 従来の DP マッチング

2つの1次元パターン $X = x_1, x_2, \dots, x_i, \dots, x_I$, $Y = y_1, y_2, \dots, y_j, \dots, y_J$ 間の弾性マッチングを考える. ここで x_i と y_j は特徴ベクトルである. 弾性マッチングは, X の第 i 要素 x_i と Y の第 j 要素 y_j との対応付け $j = u_i$ ($i = 1, \dots, I$) を最適化する問題である.

従来の DP マッチング (以下, 従来法) でこの最適化問題を解く場合, 図1(a)のような $I \times J$ 個の格子点を持つトレリスを考える. これを I 段 J 状態の多段決定過程 [7] と見なす. そして各状態 j での局所的な経路選択を $i = 1$ から I まで順に繰り返すことで, 最終的にトレリス上の最適経路 $u_1, \dots, u_i, \dots, u_I$ として2パターン間の最適弾性マッチングを求める. この最適経路はマッチングパスや歪み関数とも呼ばれる. 以上の処理は探索幅 J , 深さ I の幅優先探索と見ることもでき, 従って計算量は $O(IJ)$ となる.

経路すなわちマッチングは, 各格子点 (i, j) の局所距離 (図1(b))

$$d_i(j) = |x_i - y_j| \quad (1)$$

を, その経路が通過する格子点 (i, u_i) について累積した値 $\sum_{i=1}^I d_i(u_i)$ により評価される. 局所距離は局所的なマッチング誤差, すなわち対応付けられる2つの特徴ベクトルの差異であり, その累積値は X と Y 全体でのマッチング距離となる.

3. 解析的 DP マッチング

3.1 問題の定式化

解析的 DP マッチングは次の目的関数の最小化問題として定式化される.

$$F(u_1, \dots, u_I) = \sum_{i=1}^I d_i(u_i) + \sum_{i=2}^I \lambda(u_i - u_{i-1})^2 \quad (2)$$

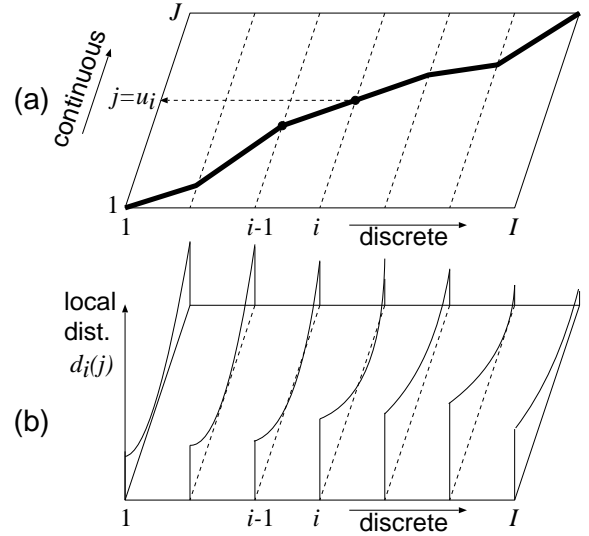


図2 解析的 DP マッチング. (a) 最適経路探索空間. (b) 局所距離.

第1項は局所距離の累積値である. また, 第2項 $(u_i - u_{i-1})^2$ は対応付けの連続性評価項であり, λ は重み係数である^(注1). 境界条件として, $u_1 = 1$, $u_I = J$ を仮定する. 以下では, F を最小化する u_i を \bar{u}_i と表記する. なお, この定式化では, 従来仮定されることの多い単調性 $u_{i-1} \leq u_i$ は保証されない.

3.2 局所距離の関数近似表現

本手法では, 最適化に先立ち, 第 i 段での局所距離 $d_i(1), \dots, d_i(j), \dots, d_i(J)$ を次のように2次関数で近似表現しておく.

$$d_i(j) = \alpha_i j^2 + \beta_i j + \gamma_i \quad (3)$$

すなわち各 i で1つの2次関数を定義し, そのパラメータ $\alpha_i, \beta_i, \gamma_i$ をあらかじめ決めておくことになる. 従来法では局所距離 (1) が整数格子点 (i, j) において離散的に定義されていたのに対し, 解析的 DP マッチングにおいては, 非整数の j においても $d_i(j)$ が与えられることになる. 従って, (2) の最小化において u_i が整数値である必要がなくなる.

図2は解析的 DP マッチングの概要を表したものである. 上述のように, 局所距離は2次関数として各 i 毎に定義される. 本手法の目的は, 連続性評価項 $(u_i - u_{i-1})^2$ に配慮しながら, なるべくこの2次関数の頂点に近いところを通過するようなマッチングパスを求めることになる. 従来法 (図1) においてこのパスは整数格子点 (i, j) を通過する必要があった. 一方, 解析的 DP マッチングにおいては, 上述の通り j および u_i は実数であり, 整数格子点を通過する必要はない.

局所距離を2次関数近似する場合, 下に凸なものに限定する必要がある (すなわち $\alpha_i \geq 0$). 上に凸の2次関数を用いると, $j = u_i$ が $\pm\infty$ に近づくことで $d_i(j)$ の値はいくらでも小さくなり, 結果的に $\bar{u}_i = \pm\infty$ となり不都合である.

(注1): 単調増加性に配慮し, 連続性評価項を $(u_i - u_{i-1} - 1)^2$ と定義してもよい. ただし, $u_1 = 1$, $u_I = J$ という境界条件の下では, 実質的には $(u_i - u_{i-1})^2$ との差はほとんどない. 実際, 後述する認識実験において両者に大きな差異は認められなかった.

Input:

Coefficients of local distance functions:

$$\{\alpha_i, \beta_i, \gamma_i \mid i = 1, \dots, I\}$$

Boundary condition: $u_1 = 1, u_I = J$ Weight: $\lambda \in \mathbf{R}^+$ **Output:**Optimal correspondence: $\{\bar{u}_i \in \mathbf{R} \mid i = 1, \dots, I\}$ Minimum distance $\min F$ **Step 1: Initial condition**

$$a_I = \lambda$$

$$b_I = -2\lambda u_I$$

$$c_I = d_I(u_I) + \lambda u_I^2$$

Step 2: DP recursionfor $i = I - 1$ downto 2 do

$$a_i = \frac{\lambda(\alpha_i + a_{i+1})}{\alpha_i + \lambda + a_{i+1}}$$

$$b_i = \frac{\lambda(\beta_i + b_{i+1})}{\alpha_i + \lambda + a_{i+1}}$$

$$c_i = -\frac{(\beta_i + b_{i+1})^2}{4(\alpha_i + \lambda + a_{i+1})} + \gamma_i + c_{i+1}$$

Step 3: Termination

$$\min F = d_1(u_1) + a_2 u_1^2 + b_2 u_1 + c_2$$

Step 4: Backtrack

$$\bar{u}_1 = u_1 = 1$$

for $i = 2$ to $I - 1$ do

$$\bar{u}_i = \frac{\lambda}{\alpha_i + \lambda + a_{i+1}} \bar{u}_{i-1} - \frac{\beta_i + b_{i+1}}{2(\alpha_i + \lambda + a_{i+1})}$$

図 3 解析的 DP マッチングのアルゴリズム

3.3 DP 漸化式

前節で与えた局所距離の 2 次関数表現により，目的関数 (2) は以下のように書き換えられる．

$$F(u_1, \dots, u_I) = d_1(u_1) + \sum_{i=2}^I [\alpha_i u_i^2 + \beta_i u_i + \gamma_i + \lambda(u_i - u_{i-1})^2] \quad (4)$$

Angel [8] に従い，ここで次のような関数を考える．

$$f_i(u_{i-1}) = \min_{u_i, \dots, u_{I-1}} \sum_{k=i}^I [\alpha_k u_k^2 + \beta_k u_k + \gamma_k + \lambda(u_k - u_{k-1})^2] \quad (5)$$

この定義より， F の最小値は

$$\min F = d_1(u_1) + f_2(u_1) \quad (6)$$

となる．

最適性の原理から (5) は次のように表せる．

$$f_i(u_{i-1}) = \min_{u_i} [\alpha_i u_i^2 + \beta_i u_i + \gamma_i + \lambda(u_i - u_{i-1})^2 + f_{i+1}(u_i)] \quad (7)$$

上式は DP 漸化式 (もしくは関数方程式，関数再帰方程式) と呼ばれ， i に関する再帰的關係を表現している．

従来法では，この DP 漸化式に基づいた幅優先探索を行なう．すなわち， u_{i-1} を整数とし，各 $u_{i-1} \in [1, 2, \dots, J]$ について $f_i(u_{i-1})$ の値を求め，テーブル化しておく．そして次に，この $f_i(u_{i-1})$ の値を用いながら，やはり J 通りの u_{i-2} について， $f_{i-1}(u_{i-2})$ の値を求め，テーブル化する．この処理を $i = I$ から 1 まで (逆も可) 続けることで，最小値 $f_2(u_1)$ を求めることになる．これに対し本手法では，このような幅優先探索は行なわず，微分により最適な $u_i = \bar{u}_i$ を u_{i-1} により一意的かつ解析的に求める．以下にその詳細を述べる．

3.4 解析的解法

関数 $f_i(u_{i-1})$ は，その 2 次関数的性質から，変数 a_i, b_i, c_i を用いて

$$f_i(u_{i-1}) = a_i u_{i-1}^2 + b_i u_{i-1} + c_i \quad (8)$$

と書ける．この正当性については付録 1. で証明する．上式 (8) を (7) の右辺に代入すると，

$$f_i(u_{i-1}) = \min_{u_i} [\alpha_i u_i^2 + \beta_i u_i + \gamma_i + \lambda(u_i - u_{i-1})^2 + a_{i+1} u_i^2 + b_{i+1} u_i + c_{i+1}] \quad (9)$$

となる．右辺 [] 内を Q とおくと， Q を最小にする $u_i = \bar{u}_i$ は， $\partial Q / \partial u_i = 0$ を満たす必要がある．これを解いて，

$$\bar{u}_i = \frac{\lambda}{\alpha_i + \lambda + a_{i+1}} u_{i-1} - \frac{\beta_i + b_{i+1}}{2(\alpha_i + \lambda + a_{i+1})} \quad (10)$$

この $u_i = \bar{u}_i$ を (9) に代入して整理すると，

$$f_i(u_{i-1}) = \frac{\lambda(\alpha_i + a_{i+1})}{\alpha_i + \lambda + a_{i+1}} u_{i-1}^2 + \frac{\lambda(\beta_i + b_{i+1})}{\alpha_i + \lambda + a_{i+1}} u_{i-1} - \frac{(\beta_i + b_{i+1})^2}{4(\alpha_i + \lambda + a_{i+1})} + \gamma_i + c_{i+1} \quad (11)$$

が得られる．この (11) と (8) の比較により，

$$\left. \begin{aligned} a_i &= \frac{\lambda(\alpha_i + a_{i+1})}{\alpha_i + \lambda + a_{i+1}} \\ b_i &= \frac{\lambda(\beta_i + b_{i+1})}{\alpha_i + \lambda + a_{i+1}} \\ c_i &= -\frac{(\beta_i + b_{i+1})^2}{4(\alpha_i + \lambda + a_{i+1})} + \gamma_i + c_{i+1} \end{aligned} \right\} \quad (12)$$

となる．上式 (12) は， $a_{i+1}, b_{i+1}, c_{i+1}$ から a_i, b_i, c_i を逐次計算する方法を与えている．

この逐次計算の開始値 (a_I, b_I, c_I) は，(5) 式において $i = I$ とすることで求まる．すなわち，

$$f_I(u_{I-1}) = \alpha_I u_I^2 + \beta_I u_I + \gamma_I + \lambda(u_I - u_{I-1})^2 = \lambda u_{I-1}^2 - 2\lambda u_I u_{I-1} + (d_I(u_I) + \lambda u_I^2)$$

これと，(8) において $i = I$ とした

$$f_I(u_{I-1}) = a_I u_{I-1}^2 + b_I u_{I-1} + c_I$$

を比較することで，次のように (a_I, b_I, c_I) が求まる．

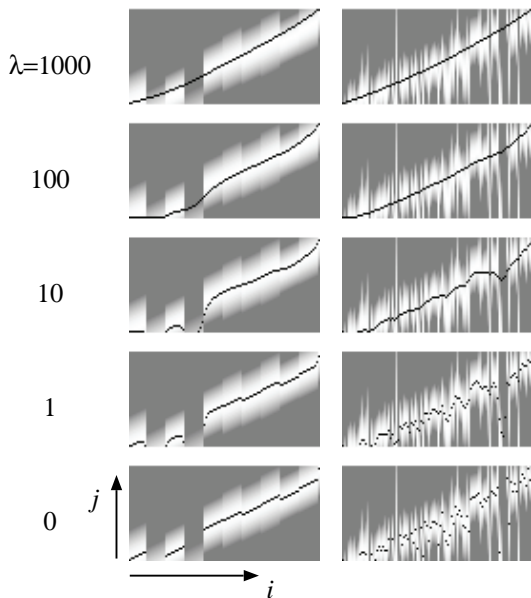


図4 シミュレーション実験の結果

$$\left. \begin{aligned} a_I &= \lambda \\ b_I &= -2\lambda u_I \\ c_I &= d_I(u_I) + \lambda u_I^2 \end{aligned} \right\} \quad (13)$$

以上により、(4) の最適化問題が解けたことになる。すなわち、(13) の後に (12) を反復適用することで a_2, b_2, c_2 を求め、さらにそれを (6) に代入することで、目的関数の最小値

$$\begin{aligned} \min F &= d_1(u_1) + f_2(u_1) \\ &= d_1(u_1) + a_2 u_1^2 + b_2 u_1 + c_2 \end{aligned} \quad (14)$$

が求まる。また、 X と Y 間の最適対応 $u_i = \bar{u}_i$ については、境界条件 u_1 ならびに (10) から求まる。

図3はこうして導出された解析的 DP マッチングのアルゴリズムである。注意すべきは、このアルゴリズムで求まるのは、目的関数 (4) の大域的最適解であるという点である。すなわち、目的関数を (2) から (4) に書き換える時点で局所距離 $d_i(j)$ の2次関数近似が行なわれているが、その後の計算過程において一切近似は無く、(4) の厳密な最適解を求めている。

4. 実験

4.1 シミュレーション実験

図4は、人工的に生成した局所距離 $d_i(j)$ に対して解析的 DP マッチングを行なった結果である。同図は各 (i, j) での $d_i(j)$ の値を表した濃淡画像の上に、マッチングパス $\bar{u}_1, \dots, \bar{u}_i, \dots, \bar{u}_I$ を黒い点列として表したものである。濃淡画像の各画素は白いほど $d_i(j)$ の値が小さいことを意味する。

まず $\lambda = 0$ の場合、連続性評価項が無くなるため、パスは各 i で常に $d_i(j)$ の極小点を通り、不連続なものとなっている。一方、 λ が大きくなると、連続性評価項が支配的になるため、パスは直線的になっている。その中間の λ では、 $d_i(j)$ の極小点付近を滑らかに通過できていることがわかる。

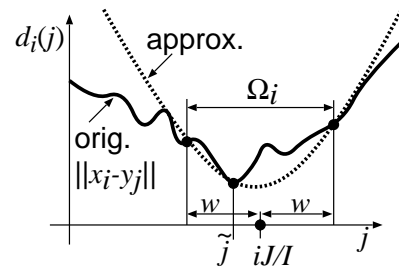


図5 局所距離の2次関数近似

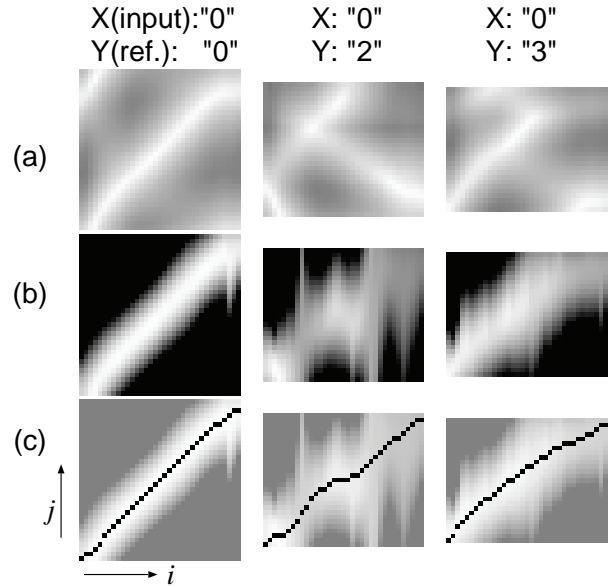


図6 3組のオンライン文字データの解析的 DP マッチング結果。(a) 真の局所距離。(b) 2次関数近似した局所距離。(c) (b) 上にマッチングパスを重畳したもの。

4.2 オンライン文字認識

解析的 DP マッチングの実パターンへの応用可能性を示すことを目的として、オンライン文字認識実験を行なった。Ethem Alpaydin Digit [9] のオンライン孤立数字データセット (全 10 文字種) のうち、テストパターン全 3498 サンプルを認識対象 X とした。各テストパターンについて、1 画化、線形正規化ならびにリサンプリングを行った。また手で合計 21 個 (各文字種に 1~4 個) の標準パターン Y を準備した。各入力パターン X と標準パターン Y について解析的 DP マッチングを行い、得られた $\min F$ に基づく最短距離識別を行なった。

4.2.1 局所距離の2次関数近似

局所距離 $d_i(j)$ を2次関数近似する際、オンライン文字認識という問題の性質を考慮した。すなわち、正解標準パターンとのマッチングにおいては、 $X \sim Y$ であるため、マッチングパスは直線的になるべき、すなわち \bar{u}_i が区間 $\Omega_i = [iJ/I - w, iJ/I + w]$ 内に存在するべきと考えた。ここで w は区間幅を表す正定数である。この考えに基づき、区間 Ω_i 内の真の (すなわち 2次関数近似前の) 局所距離値 $\|x_i - y_j\|$ を重視して、2次関数 $d_i(j)$ を定めることにした。

具体的には、図5に示す方法で各 i の2次関数 $d_i(j)$ を決定した。まず Ω_i の両境界点および Ω_i 内で最小の局所距離を持

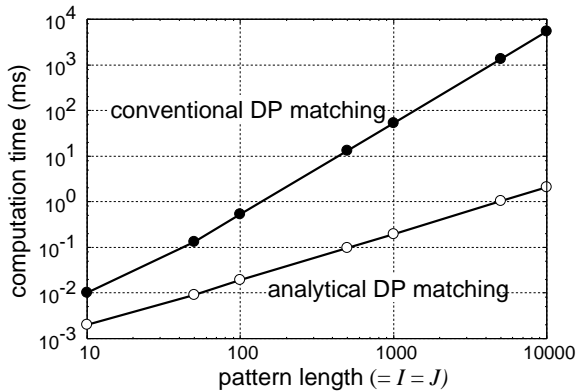


図7 計算時間の比較

つ点 \bar{j} の合計 3 点を定める．次にそれら 3 点で真の局所距離値 $\|x_i - y_j\|$ を持つ 2 次関数 $d_i(j)$ ，すなわち $\alpha_i, \beta_i, \gamma_i$ を求めた．2 次関数近似のパラメータ w は，従来法の整合窓と同じく実験的に定められるべき性質のものである．本実験では 3 とした．

4.2.2 マッチング結果

図 6 に本手法による 3 組のオンライン文字パターンのマッチング結果を示す．連続性評価項の重み λ は，実験的に 10 とした．正解標準パターンとのマッチング (同図左列) では， (i, j) -平面の対角線付近に 2 次関数近似された局所距離の鋭いピーク (白に近い領域) が現れている．最適化されたマッチングパスはこのピーク付近を滑らかに通過していることがわかる．

不正解標準パターンとのマッチング (同図中央および右列) でも同様にピーク付近を通過する滑らかなパスが求まっている．ただし，ピークは鋭くなく，従ってパス通過点での局所距離 $d_i(u_i)$ は比較的大きな値となっている．結果的に $\min J$ も大きくなるので，不正解標準パターンが認識結果として選出されにくくなるのがわかる．

4.2.3 認識率

全テストサンプルに対する解析的 DP マッチングの認識率は 93.5% であった．これに対し，線形伸縮マッチング (すなわち $\lambda = \infty$) による認識率は 91.1% であった．このことから，解析的 DP マッチングの実問題への有効性が確認できる．また， $\lambda = 0$ とした場合，すなわち \bar{u}_i が 2 次関数のピーク位置に一致する場合の認識率は 92.8% であった．このことから，連続性評価項の必要性がわかる．

比較のため従来法によるオンライン文字認識も行なった．重み λ は 0 とし，単調連続性条件 $u_i - u_{i-1} \in \{0, 1, 2\}$ によりパスを直接制約した．また整合窓すなわち条件 $\|i - u_i\| \leq W$ による制約も併用した．その結果，整合窓幅 $W = 5$ の時に最高認識率 94.5% を得た．本手法の認識率のほうがやや低い，厳密な単調連続性条件下での従来法と大きな差が無かったことは，本手法で求めたマッチングパスが多くの場合良好なものであったことを示唆している．

4.3 計算時間

解析的 DP マッチングおよび従来法により，2 つの 1 次元パターン X, Y をマッチングするための時間を実測した．具体的には，両パターンの長さは同じ (すなわち， $I = J$) という条件

の下，この長さを 10 から 10000 まで変えながら，1 回のマッチングに要する平均計算時間を計測した．プログラムは C 言語で実装され，1.7GHz CPU 搭載の計算機上で実行された．

結果を図 7 に示す．比較を容易にするために，本図の計算時間には局所距離の計算時間を含めていない．これは，4.2.1 の 2 次関数近似を採用する場合， $w = W$ ならば局所距離計算に要する時間は両マッチング法でほぼ同じ ($O(IW) = O(Iw)$) となるためである．また，従来法については整合窓を用いていない場合 (すなわち $W = J$) の計算時間を示している．従来法では整合窓を用いることで計算量を $W/J = W/I$ 倍に低減できる．ただし，多くの場合 $W = 0.1I$ 程度が最小幅であり，従って整合窓による計算量低減効果は高々 $1/10$ 程度である．

図 7 より，本手法の計算量的優位性がわかる．ただし，上述のように整合窓を用いることで従来法の計算量を $1/10$ にできることを考えると， $I = J \sim 100$ 程度までならば，両手法に大きな計算量の差は無いと言える．しかしながら，それ以上のパターン長では， J 状態の幅優先探索を行なう必要がある従来法に比べ，その必要のない本手法のほうが格段に少ない計算量で済むことがわかる．このように 1 次元パターンのマッチング問題においても本手法の計算量的優位性が実証されたことは，状態数が指数的に増加する 2 次元パターンのマッチング問題に対して，本手法がさらに有力な解法となりうることを示唆している．なお，局所距離計算 (本手法の場合 $\{\alpha_i, \beta_i, \gamma_i\}$ の決定) を除外した場合，本手法および従来法の計算量はそれぞれ $O(I)$ と $O(IJ) = O(I^2)$ であり，図 7 のグラフの傾きの差異を裏付けている．

5. 議 論

5.1 解析解法としての DP

DP の本質はその再帰的計算過程にあり，それは従来法のような離散的な問題だけでなく，本手法のような連続的な問題にも適用しうるものである [10]．しかしながら，デジタル化された対象が多いパターン認識・画像処理の分野において，DP は主として離散化された問題に適用されてきた．実際，連続的な問題に対して DP に基づく解析的解法を与えた例は，1 次元および 2 次元データの最適補間問題への適用例 [8] 程度と少ない．

5.2 マッチングにおける関数近似

マッチング問題を解析的 DP で解くために，本手法では局所距離を 2 次関数近似する．この考え方は Etoh [11] に着想を得ている．同文献では，2 画像のアフィン変換マッチング問題を局所距離の 2 次関数近似により解析的に解く手法を提案している．局所距離すなわちマッチング誤差ではなく，パターンそのものを関数近似する試みもある [12], [13]．例えば大町らは [12] パターンを多項式近似することで，線形伸縮を含むマッチング問題を効率的に解く方法を提案している．このように関数近似の導入自体には従来より検討例があるが，対象としている問題が非線形な弾性マッチングである点と，解法に DP を用いる点で，本手法と従来試みとは大きく異なっている．

5.3 連立解法との関係

目的関数 F の最小化問題は，解析的 DP 以外の方法でも解

くことができる。すなわち、(4) で与えられる目的関数 F の極値条件 $\partial F/\partial u_i = 0$ を u_1, \dots, u_{I-1} について連立し、得られた $(I-1)$ 元一次連立方程式を解くという方法 (以下、連立解法と呼ぶ) が考えられる。詳細は略すが、この場合の連立方程式の係数行列は 3 重対角になる。従って逆行列を求めることなく逐次代入処理で解くことができる [14]。

1 次元パターンを対象とすれば、連立解法と本手法の計算量はほぼ同じである。しかし 2 次元の問題への拡張を考えた場合、両者の違いは顕著になる。すなわち、変数 u_i が u_{i_1, i_2} のように 2 次元化し、さらに u_{i_1, i_2} が連続性評価等により近傍の変数 $u_{i_1 \pm 1, i_2 \pm 1}$ と互いに影響するような場合、連立解法はもはや 3 重対角行列には基づかず、決定論的弛緩法 [15] に見られるような大規模な行列に基づくことになろう。一方、解析的 DP には、Angel [8] の例のように、問題のマルコフ性を活かして、 u_{i_1, i_2} を i_1 毎にまとめたカラムワイズな処理 (すなわち 2 次元パターンを列ベクトルの 1 次元系列と見なした処理) が適用できる。その結果大規模な逆行列計算が不要な最適化過程を実現できるものと考えられる。

5.4 局所距離が本質的に 2 次関数となる場合

以上では一貫して、解析的 DP マッチングを従来法の近似解法として扱ってきた。しかしながら、目的関数が元々 (4) の形式で書ける問題、すなわち局所距離 $d_i(j)$ が本質的に 2 次関数となる問題に対しては、本手法はもはや近似解法ではなく、その大局的最適解を与える有力な手法となる。例えば、2 パターン間に予め与えられた幾つかのラフな対応関係を最適補間する問題としてマッチング問題が定式化された場合、まさに (4) の形式の目的関数が直接導出され、本手法はその解法となる。

ところでこの補間問題は標準正則化理論 [15] に基づく一種の曲線当てはめの問題 (マッチングパスが曲線に相当) と等価である。この事実は、本手法が正則化理論に基づく様々な問題に適用可能なことを示唆する。すなわち、マッチング問題に限らずとも、事前知識によりペナルティ項が 2 次関数の和として定式化されるような正則化問題であれば、本手法と同様の考え方によりその大局的最適解を効率よく求めることができる。

6. む す び

解析的 DP マッチングを提案した。本手法は、幅優先探索に基づく従来の DP マッチングとは異なり、DP の解析的解法としての側面を利用した方法になっている。この解析的解法を可能としたのは、局所距離を 2 次関数近似し、それに応じて状態を連続化することで、これまでのような有限個の状態を個々に考える必要を無くしたためである。従って、従来の DP マッチングでは困難であった状態が非常に多い問題に対して本手法は特に有効と思われる。本論文では 1 次元パターン間の弾性マッチング問題を用いて、本手法の基本的なアルゴリズムの導出をおこなった。また、オンライン文字パターンのマッチングを行なうことで、実際の問題に応用可能であることを示した。

1. でも述べたように、1 次元パターン間の弾性マッチング問題では状態はそう多くなならない。従って本手法の計算量的な利点を示すには不十分と考えている。今後は [8] 等を参考に、本

論文の議論を進展させ、指数関数的な状態数を要する 2 次元パターンの弾性マッチング問題を解析的 DP マッチングにより解く予定である。また、5.4 で述べたような、本手法を直接適用し得る各種の正則化問題についても検討する予定である。

文 献

- [1] 迫江博昭, 千葉成美, “動的計画法を利用した音声の時間正規化に基づく連続音声認識,” 日本音響学会誌, vol. 27, no. 9, pp. 483-490, 1971.
- [2] H. Sakoe and S. Chiba, “A dynamic programming algorithm optimization for spoken word recognition,” IEEE Trans. Acoust. Speech and Signal Proc., vol. ASSP-26, no. 1, pp. 43-49, 1978.
- [3] H. Sakoe, “Two-level DP-matching algorithm—a dynamic programming based pattern matching algorithm for continuous speech recognition,” IEEE Trans. Acoust. Speech and Signal Proc., vol. ASSP-27, no. 6, pp. 588-595, 1979.
- [4] 大田友一, 山田博三, “動的計画法によるパターンマッチング,” 情報処理, vol. 30, no. 9, pp. 1058-1066, 1989.
- [5] 内田誠一, 迫江博昭, “動的計画法に基づく単調連続 2 次元ワーブ法の検討,” 信学論, vol. J81-D-II, no. 6, pp. 1251-1258, 1998.
- [6] D. Keysers and W. Unger, “Elastic image matching is NP-complete,” Pattern Recog. Lett., vol. 24, no. 1-3, pp. 445-453, 2003.
- [7] 茨木俊秀, 組合せ最適化の理論, 電子情報通信学会, 1979.
- [8] E. Angel, “Dynamic programming for noncausal problems,” IEEE Trans. Automatic Control, vol. AC-26, no. 5, pp. 1041-1047, 1981.
- [9] ftp://ftp.ics.uci.edu/pub/machine-learning-data-bases/pendigits/
- [10] 鍋島一郎, 動的計画法, 数学ライブラリー 7, 森北出版, 1968.
- [11] M. Etoh, “Promotion of block matching: parametric representation for motion estimation,” Proc. ICPR, vol. 1 of 2, pp. 282-285, 1998.
- [12] 大町方子, 大町真一郎, 齋藤恒雄, “関数を用いた認識のためのパターン表現法,” 信学技報, PRMU2005-84, 2005.
- [13] J. Kybic and M. Unser, “Fast parametric elastic image registration,” IEEE Trans. Image Proc., vol. 12, no. 11, pp. 1427-1442, 2003.
- [14] J.R. Westlake 著, 戸川隼人訳, コンピュータのための線形計算ハンドブック, 培風館, 1972.
- [15] 坂上勝彦, 横矢直和, “弛緩法と正則化,” 情報処理, vol. 30, no. 9, pp. 1047-1057, 1989.

付 録

1. 式 (8) の妥当性

ここでは式 (8) のように $f_i(u_{i-1})$ が常に u_{i-1} に関する 2 次式で表現できることを帰納法で証明する。まず $i = I$ の時、 $f_i(u_{i-1})$ の定義 (5) から

$$f_I(u_{I-1}) = \alpha_I u_I^2 + \beta_I u_I + \gamma_I + \lambda(u_I - u_{I-1})^2$$

であり、右辺は u_{I-1} に関する 2 次式となっている。

次に $i+1$ ($i < I$) の時に成り立つと仮定して、 i の時も成り立つことを示す。この仮定は $f_{i+1}(u_i)$ が

$$f_{i+1}(u_i) = a_{i+1} u_i^2 + b_{i+1} u_i + c_{i+1}$$

と表現できることを意味する。この式を (7) の右辺に代入すると、(9) が得られる。従って結局 $f_i(u_{i-1})$ は (11) に帰着し、これは u_{i-1} に関する 2 次式になっている。以上より、証明終。