カラー画像テンプレートマッチングとその高速化

佐藤辰雄
機械電子部

Template-matching in color-image, and its fast algorithm

Tatsuo SATO
Mechanics & Electronics Division

要旨

テンプレートマッチングは，何らかの方法により画像間の距離を定義し，どれだけ近いか（一致しているか）を調べるもので，その手法としては正規化相関が良く知られている．また，この手法はあらかじめわかっているマークなどを用いれば位置合わせを可能にするため，印刷などの用途での位置合わせに多く実用されている．ところがこの正規化相関は白黒画像について定義されており，例えば金属の表面などのように濃淡の微妙な場合にはうまく行かないことが多い．

この問題を解決すべく，性能向上や適用範囲の拡大をするためには，限られた解像度で情報量を多くする意味からカラー画像への拡張が有効と考えられる．そこで本研究では，正規化相関を多次元に拡張することでカラー画像に適用し，実験でその有効性を確かめた．

また，正規化相関は計算量が非常に多いため，処理が遅いという問題がある．この点についてその高速アルゴリズムについても検討し，そのいくつかについて実験して，高速化の見通しを得た．

１．はじめに

画像処理やパターン認識は自動化装置や無人監視のためのセンサとして産業ニーズは非常に高くなっており，また市販の装置も様々に開発されてきている．その中でもテンプレートマッチングは古くから研究されてきており，いろいろな手法がある．

はじめの頃は取り込んだ画像を２値化し，２値画像の上でマッチングが行われていたが，現在では半導体やLSIの技術とマイクロコンピュータの進歩に伴いハードウェアもソフトウェアもローコスト化してきたため，また実際に使ってみると対象物と背景とをうまく分離するような２値化ができなかったり環境が少し変わっただけで照明やスレッショルドなどを調整し直したりといった厄介な問題をさけるため，さらにマッチング精度や適用範囲の改善のため，モノクロの濃淡画像を使うことも多くなってきた．またその応用分野としては位置決めや探索の用途がもっとも一般的なようである．

実現手法はいろいろなものが提案[1) 2)]されているが，そのなかでも相互相関に基づく正規化相関（相互相関係数）はよく知られている．

一方入力に用いられるCCDの画素数などによって制約された解像度の範囲でマッチング精度や対象物の適用範囲を改善するには，入力の情報量を何らかの方法で増やす必要がある．

その方法の一つにカラー画像が考えられる．しかし正規化相関はそのままではカラー画像には適用できない．

カラー画像とモノクロ画像との本質的な違いを考えると，モノクロ画像はスカラー量の２次元マップであるが，カラー画像は(r,g,b)３次元のベクトルの２次元空間マップであることがあげられる．そこで本研究では正規化相関の手法を多次元に拡張しカラー画像への適用をはかった．

これによりモノクロ濃淡画像では照明や光学フィルターなどを工夫しなければならなかったり，それでもコントラスト比が低くてうまくマッチングがとりづらい様な場面で，もし色が違ってさえいれば適用可能となる．その意味で適用範囲が広がると考えられ，またカラー画像を使うことで情報量が増えることからマッチングの精度の改善も期待できる．

2. テンプレートマッチング

テンプレートマッチングは，入力パターンと，あらかじめ記憶してある標準パターンとを重ね合わせ，最もよく一致する入力パターンを調べる手法である.

一致の度合いを表現するためには，パターン間の距離を定義する必要があり，ユークリッド距離を用いる場合や，マハラノビス距離を用いる方法などが検討されてきた. それらの一つに，正規化相関がある.

2.1 正規化相関

正規化相関は，パターンとしては画像そのものを直接扱う方法であり，白黒濃淡画像や2値画像について定義されている.

いま，画面の座標を (u,v) で，また重ね合わせる位置を (U,V) で表し，入力画像を $f(u,v)$，テンプレート画像を $g(u,v)$ とすると，正規化相関は

$$\frac{\sum\sum\{g(u,v)-\bar{g}(u,v)\}\{f(u+U,v+V)-\bar{f}(u+U,v+V)\}}{\sqrt{\sum\sum\{g(u,v)-\bar{g}(u,v)\}^2}\sqrt{\sum\sum\{f(u+U,v+V)-\bar{f}(u+U,v+V)\}^2}} \tag{1}$$

と表される.

2.2 カラー化

このように定義された正規化相関の手法をカラー画像に適用するには何らかの拡張が必要である. ここではカラー画像のデータ表現に着目し，多次元への拡張を試みた.

カラー画像ではそれぞれの画素の値は三原色の明るさで表されている. そこでこれらをまとめればベクトルとして表現できる. すなわち

$$f(u,v) \equiv (r(u,v), g(u,v), b(u,v)) \tag{2}$$

これを(1)式に適用すればよい. ここでベクトルの積は内積をとることにする. すると正規化相関は次のように表せる.

$$\frac{\sum\sum\{g(u,v)-\bar{g}(u,v)\}\cdot\{f(u+U,v+V)-\bar{f}(u+U,v+V)\}}{\sqrt{\sum\sum|g(u,v)-\bar{g}(u,v)|^2}\sqrt{\sum\sum|f(u+U,v+V)-\bar{f}(u+U,v+V)|^2}} \tag{3}$$

3. 実験と結果

実験は画像取り込みボードを付加した汎用ワークステーションにCCDカメラを接続したものを使い，カメラを屋外の景色を撮影するように固定して640×480画素の画像を午前中1枚午後1枚の計2枚取り込み，最初の一枚からその一部(30×30画素)を切り出してテンプレートとし，もう一枚の画像の中で一致する場所を探して，切り出したテンプレートの座標と一致するかどうかを調べた.

午前中はよく晴れており，午後は薄曇りとなったので，見かけ上はほぼ同じ写真のように見えるが，明るさやコントラストなどの面で少し異なっていると思われる.

また(3)式では正規化相関の値が負の値を取ることがあるが，これはちょうどネガ画像とのマッチングに相当し，負の値はこの実験では必要ないので負になったらすべて0にした. その方がマッチング結果を画像表示するときに見安い写真が得られたためである.　探索手法については今回は特に考えず，単純に一番大きなスコアが得られた点の座標を虱潰しに調べる方法を採った.

実験に使った画像の例を図1に，カラー画像によるマッチング結果を図2に，同じ画像を白黒に変換して，従来の正規化相関を行った結果を図3に示す.

またこれらについての探索結果はすべて切り出した点の座標値と同じになっており，マッチングは良好であると考えられる.

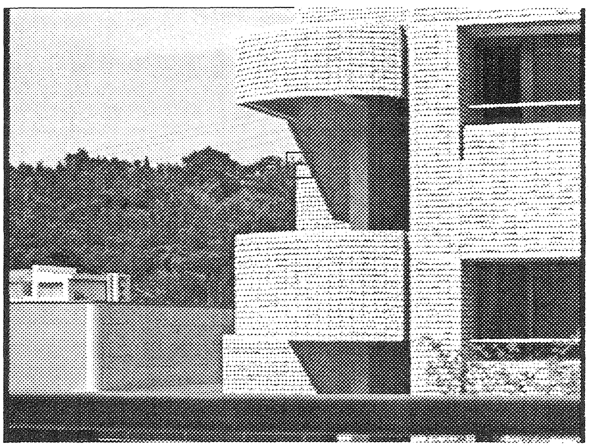

図1　テンプレート画像
(図中の赤い枠はテンプレートとして切り出した部分.)

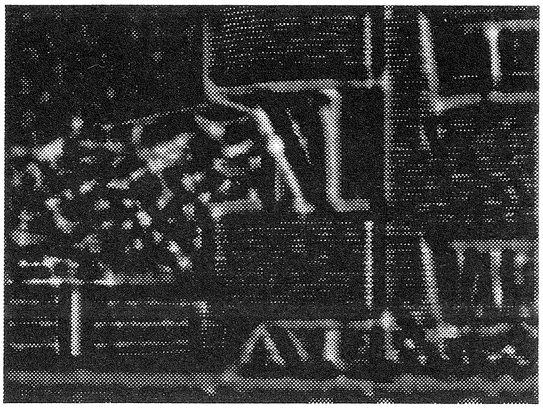

図2　カラー画像によるマッチング結果

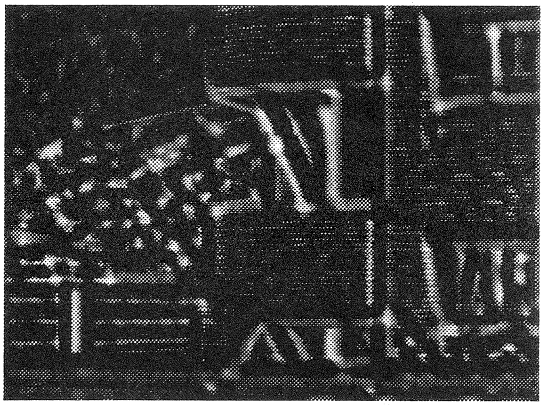

図3　白黒濃淡画像によるマッチング結果

4.　高速化

4.1 畳み込み(*Convolution*)と相関関数(*Correlation*)

相関関数と畳み込みは次のような関係である.

- 相関　$\sum_{n=0}^{N-1} h_n x_{l+n}$　(4)
- 畳込み　$\sum_{n=0}^{N-1} h_n x_{l-n}$　(5)

・両者は入力数列のうちの添字式の符号が違うだけ
・計算するときにその順番を逆順に並べ替えれば 同じアルゴリズムでよい
・従って相関関数は畳込み演算として計算できる

4.2 高速フーリエ変換

離散フーリエ変換(DFT)は複素指数関数の単位根を用いて次のように定義される

$$X(k) = \sum_{n=0}^{N-1} x(n) W_N^{nk}, \quad k = 0,1,\cdots,N-1$$

$$x(n) = N^{-1} \sum X(k) W_N^{-nk}, \quad n = 0,1,\cdots,N-1$$

$$W_N = e^{-j(2\pi/N)} \tag{6}$$

この変換を高速に実現するアルゴリズムの一つがFFTである. 高速アルゴリズムはこの他にもいくつかが知られている.
このFFT(DFT)を用いれば, 次の手順で相関関数が高速に計算できる.

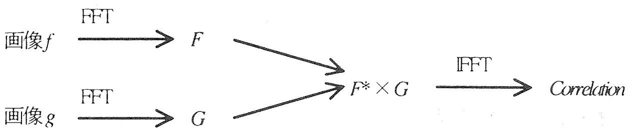

図4　FFTを用いた相関関数計算手順

4.3 数論変換(NTT)

DFTが複素指数関数の単位根を用いた変換として定義されるのに対して, NTTは1の原始N乗根(整数)を用いた変換で, すべての演算は整数を法として定義される.

$$X(k) = \sum_{n=0}^{N-1} x(n) \alpha^{kn} \pmod P$$
$$x(n) = N^{-1} \sum_{k=0}^{N-1} X(k) \alpha^{-kn} \pmod P \tag{7}$$

これを用いた畳み込みは次の手順で実現される.

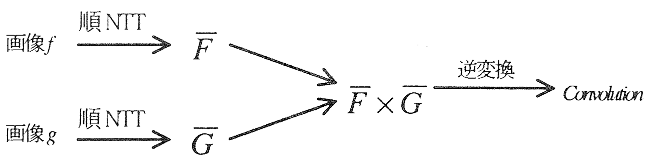

図5　数論変換による相関関数計算手順

その特徴は以下の通り.
・丸め誤差が生じない
・原始N乗根を2のべき乗にとれば乗算がシフト演算

に置き換えられ，高速になる．
- 変換の長さを2のべき乗にとればFFT型のアルゴリズムが利用できて高速にできる．
- 変換長を自由にとれず，演算語長によって変換長が制限される．
- オーバフローが重大な誤差となるため，ダイナミックレンジの制約がきつい．

①Mersenne数変換

*Mersenne*数(2^p-1, p: 素数)を法とするNTTを*Mersenne*数変換と呼ぶ．

その特徴は以下の通り．
- (2^p-1)を法とする演算はよく知られた1の補数演算であり，2進ハードウェアで容易に実現できる．
- FFTのような高速アルゴリズムが無い．
- pビットの場合，pと$2p$の２種類の変換長しかとれない．

②Fermat数変換(FNT)

*Fermat*数($2^{2^t}+1$)を法とするNTTを*Fermat*数変換と呼ぶ．

③複素数論変換

拡張体 $GF(q^2)$ 上での数論変換を複素数論変換と呼ぶ．

④疑似数論変換

疑似*Mersenne*数 2^p-1, p: 合成数や，疑似*Fermat*数(2^v+1, $v \neq 2^t$)を法とする数論変換を疑似数論変換と呼ぶ．

4.4 多項式変換

多項式を法とする多項式の演算は環をなし，多項式環と呼ばれる．数論変換を多項式環上の演算に基づいて拡張したとき，これを多項式変換と呼ぶ．

サイズ $N \times N$ の二次元巡環畳み込み，

$$y_{u,l} = \sum_{m=0}^{N-1}\sum_{n=0}^{N-1} h_{n,m}\, x_{u-n,l-m} \qquad u,l = 0,\cdots,N-1 \tag{8}$$

がある時，次のように多項式代数で表現すれば一次元多項式の畳み込みと見なせる．

$$Y_l(z) \equiv \sum_{m=0}^{N-1} H_m(z)\, X_{l-m}(z) \qquad \bmod\ (Z^N-1)$$
$$H_m(z) = \sum_{n=0}^{N-1} h_{n,m} z^n, \qquad m = 0,\cdots,N-1$$
$$X_r(z) = \sum_{s=0}^{N-1} x_{s,r} z^s, \qquad r = 0,\cdots,N-1 \tag{9}$$

ここで $y_{u,l}$ はN個の多項式

$$Y_l(z) = \sum_{u=0}^{N-1} y_{u,l} z^u, \qquad l = 0,\cdots,N-1 \tag{10}$$

から z^u の係数として得られる．

ここでこの計算のために，$P(z)$ を法とした多項式演算に基づく多項式変換を定義する．

$$\overline{X}_k(z) \equiv \sum_{r=0}^{p-1} X_{1,r}(z)^{rk} \qquad \bmod\ P(z) \qquad k = 0,\cdots,p-1$$
$$X_{1,r}(z) \equiv X_r(z) \qquad \bmod\ P(z) \tag{11}$$

同様に逆変換は

$$X_{1,r}(z) \equiv \frac{1}{p}\sum_{k=0}^{p-1} \overline{X}_k(z)\, z^{-rk} \qquad \bmod\ P(z),$$
$$z^{-rk} \equiv z^{(p-1)-rk}, \qquad r = 0,\cdots,p-1 \tag{12}$$

で定義される．

このときサイズ $p \times p$ の二次元畳込みは次のように計算される．

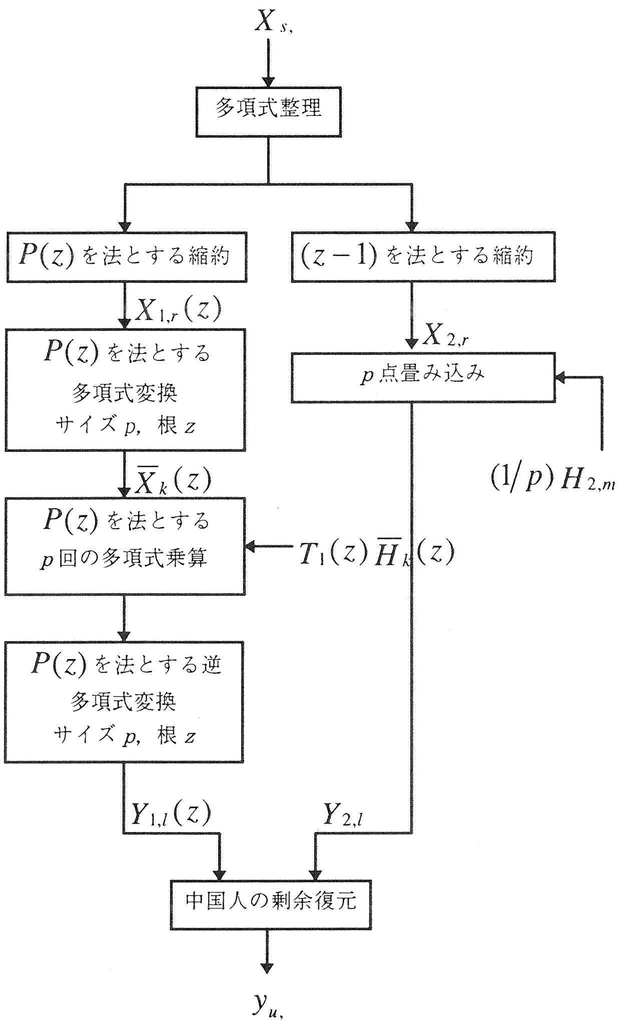

図6　多項式変換を用いた相関関数計算手順

4.5 実験と結果

実験は，これまでと同じ画像について行い，計算時間を計測した．ただし，カラー画像については，プログラム作業が間に合わなかったため白黒画像で行った．

その結果，計算時間は以下の通りとなり，高速化できることが確認された．

表1　計算時間比較

逐次計算	FFTを用いた計算	多項式変換を用いた計算
21分51秒	4分7秒	(未計測)

5. サブピクセルサイズでの位置探索

自己相関関数は偶関数で，変位量が小さい範囲では変位量に従って単調に減少する．したがって，相互相関も，マッチング点の近傍ではほぼ単調に減少すると考えられる．

このことから，マッチング点の近傍では直線近似可能で，下図のようなあてはめを行い，その頂点の位置をサブピクセルサイズの分解能で求めればよい．

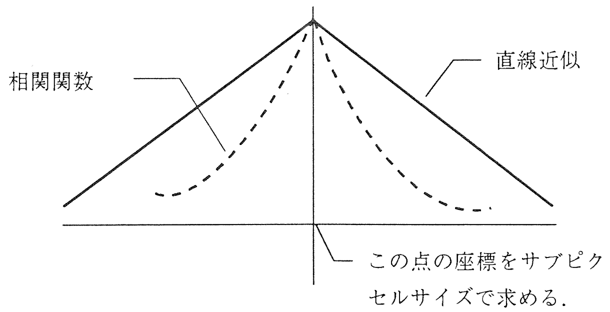

図7 サブピクセルサイズでの位置探索手順

6. まとめと今後の課題

① 正規化相関を拡張してカラー画像に適用し，白黒画像の場合より性能が優れていることを確認した．

② 相関関数の高速計算アルゴリズムについて，FFT法，数論変換，多項式変換等研究し高速化を確認した．．

③ 実験の結果，FFT法で約5倍の高速化が確認できた．

④ 多項式変換では多くのアルゴリズムがあり更に高速にするため他のアルゴリズムを研究する

⑤ 実用化のためサブピクセルサイズ位置探索について研究する

なお，本研究は大分県地域人材不足対策技術開発研究事業の中で行ったものである．

参考文献

1）佐藤辰雄，阿南正明：″カラー画像のテンプレートマッチング″，信学技報，PRU94-47(1994-10)

2）佐川雅彦，貴家仁志：″高速フーリエ変換とその応用″，昭晃堂，(1993)

3）佐藤辰雄，三浦喜穂孝，遠藤勉：“正規化相関の高速計算法に関する検討”，電子情報通信学会1996年総合大会，(1996-3)