

連想記憶による画像検索  
Image Retrieval with Associative Memory

谷 正之 谷藤 真也  
Masayuki TANI Shinya TANIFUJI

(株)日立製作所 日立研究所  
Hitachi Research Laboratory, Hitachi, Ltd.

Abstract

An application of associative memory to image retrieval is studied. Autoassociative memory by orthogonal projection is used. A preprocessing of images by one-dimensional Laplacian is proposed. With this preprocessing, computational steps and memory space necessary for inverse processing can be reduced drastically compared with the conventional preprocessing by two-dimensional Laplacian. A computer simulation shows this preprocessing to be very effective when a small fragment of a memorized image is used as a key for retrieval.

1. はじめに

大量の画像データを記憶し必要な画像を簡単に検索できる画像データベースの実現が望まれている。画像データベースでは画像の内容を反映した情報をキーとして検索できると便利である。

ある情報をキーとしてそれに関連する情報を取り出す処理を一般に連想処理と呼ぶ。現在の記憶検索システムではひとつひとつの情報ごとに記憶場所を確保する。読み出しは記憶場所を指定して行なう。そのため連想処理を実現するには、記憶場所を順次指定して記憶情報を読み出し、キー情報と比較する作業を繰り返す必要がある。

一方、人間の脳では、見たり聞いたりした事柄からそれに関連する記憶事項を即座に想起できる。このような人間の記憶、想起の機構をモデル化したものとしてT. Kohonen, 中野らの連想記憶モデルがある。これらのモデルは、(1) あいまいな情報や部分的な情報をキーとして記憶情報を検索できる、(2) 情報を記憶媒体に分散し多重に記憶しているため、情報が失われにくい、(3) 比較などの論理演算を用いず乗算と和算だけで情報を検索できる、などの特徴を備えており画像検索に適用できる可能性がある。

本論文では、代表的な連想記憶モデルである直交射影型モデルを用いて、多値画像を記憶し、記憶画像の一部をキーとして検索する方法について論じる。

特に、想起精度を改善するため、1次元ラプラスアンによる前処理方法を提案し、本前処理方法が従来の2次元ラプラスアンによる前処理方法に比べ、計算量及び計算に必要なメモリ量が少なく済むことを示す。

2. 直交射影型モデル

直交射影型モデルでは、N個の要素からなるK個の記憶項目、

$$y^k \cong [y_1^k y_2^k \dots y_N^k]^T (k=1, 2, \dots, K) \quad (1)$$

と、それと対をなすL個の要素からなるキー項目

$$x \cong [x_1^k x_2^k \dots x_L^k]^T (k=1, 2, \dots, K) \quad (2)$$

が与えられたとき、行列

$$\tilde{M}(K) \cong Y(K)X(K)^+ \quad (3)$$

を記憶する。但し

$$Y(K) \cong [y^1 y^2 \dots y^K] \quad (4)$$

$$X(K) \cong [x^1 x^2 \dots x^K] \quad (5)$$

・<sup>T</sup> | ベクトル及び行列の転置  
・<sup>+</sup> | 行列の擬逆行列

である。式(3)で定義される行列 $\tilde{M}(K)$ を記憶行列と呼ぶことにする。

想起は、キー入力 $x$ と行列 $\tilde{M}(K)$ との乗算によって行なう。すなわち、想起出力 $y$ を

$$y = \tilde{M}(K)x \quad (6)$$

により決定する。但し

$$y = [y_1 y_2 \dots y_N]^T \quad (7)$$

$$x = [x_1 x_2 \dots x_N]^T \quad (8)$$

とする。

記憶行列 $\tilde{M}(K)$ は、式(4)、(5)で定義される行列 $Y(K)$ 、 $X(K)$ が与えられたとき、 $M$ を未知行列とする下記行列方程式の最良近似解である。

$$MX(K) = Y(K) \quad (9)$$

ここでいう最良近似解とはユークリッド行列ノルム

$$\|Y(K) - MX(K)\| = \sqrt{\sum_{k=1}^K \|y^k - Mx^k\|^2} \quad (10)$$

を最小にする解のことである。特に、キー項目  $x^k$  ( $k=1, 2, \dots, K$ ) が線形独立ならば、 $\hat{M}(K)$  は式(9)の厳密解となる。すなわち

$$y^k = \hat{M}(K) x^k \quad (k=1, 2, \dots, K) \quad (11)$$

が成立し、キー入力として  $x^k$  を入力すると記憶項目  $y^k$  が正確に想起される。

## 2. 2 画像検索への適用

( $m_1 \times m_2$ ) 画素からなる  $K$  枚の記憶すべき多値画像を

$$f^k(i, j) \begin{cases} i=1, 2, \dots, m_1 \\ j=1, 2, \dots, m_2 \\ k=1, 2, \dots, K \end{cases}$$

とする。但し、 $f^k(i, j)$  は  $k$  番目の記憶画像の  $i$  行  $j$  列目の画素値を表す。この記憶画像を走査して 1 次元数値列に並べ換えたものを、式(1)の記憶項目とする。すなわち

$$y^k = [y_1^k y_2^k \dots y_N^k]^T \\ = [f^k(1,1) f^k(1,2) \dots f^k(1,m_1) \\ f^k(2,1) f^k(2,2) \dots f^k(m_1, m_2)]^T \quad (12)$$

とする。ここで、直交射影型モデルの特別な場合として、キー項目  $x^k$  を記憶項目  $y^k$  自身としたモデルを考える。このモデルを特に自己想起モデルと呼ぶ。自己想起モデルにおける記憶行列を  $\hat{M}(K)$  とすると、式(3)より

$$\hat{M}(K) \cong Y(K) Y(K)^T \quad (13)$$

となる。想起は式(6)と同様に、キー入力  $x$  と記憶行列  $\hat{M}(K)$  との乗算によって行なう。すなわち、想起出力  $y$  は

$$y = \hat{M}(K) x \quad (14)$$

である。式(13)で定義される行列  $\hat{M}(K)$  は、記憶画像  $y^k$  ( $k=1, 2, \dots, K$ ) が張る  $K$  次元空間への直交射影演算子であり、式(14)で求められる想起出力  $y$  は、キー入力  $x$  を記憶画像  $y^k$  の線形結合で最良近似(最小二乗の意味で最良)したものである。すなわち、想起出力  $y$  は

$$\|x - \sum_{k=1}^K C_k y^k\|^2 \quad (15)$$

を最小とする係数  $C_k$  を用いて

$$y = \sum_{k=1}^K C_k y^k \quad (16)$$

と表わせる。従って、キー入力  $x$  が記憶画像のひとつ  $y^r$  と等しいと、 $C_k = 1$  ( $k=r$ )、

$C_k = 0$  ( $k \neq r$ ) となり、想起出力  $y$  は  $y^r$  に等しくなる。一方、キー入力  $x$  が、記憶画像の部分像である場合や、記憶画像  $y^r$  にノイズが重畳されたものである場合には、式(16)において  $C_k \neq 0$  ( $k \neq r$ ) となり、想起出力  $y$  に、 $C_r y^r$  以外の項

$$w = \sum_{k \neq r} C_k y^k \quad (17)$$

がノイズとして含まれる。式(17)で定義される  $w$  をクロストークノイズと呼ぶ。キー入力  $x$  が、記憶画像  $y^r$  と十分な相関がある場合には、式(16)において  $C_r y^r$  の項が式(17)のクロストークノイズ  $w$  に比べ優勢となり、キー入力  $x$  から記憶画像  $y^r$  が想起できる。

なお、式(13)で定義される記憶行列  $\hat{M}(K)$  は、記憶画像が線形独立ならば、 $y^k$  の張る空間の正規直交基底を  $\hat{y}^k$  ( $k=1, 2, \dots, K$ ) として

$$\hat{M}(K) = \hat{Y}(K) \hat{Y}(K)^T \quad (18)$$

$$\text{但し } \hat{Y}(K) = [\hat{y}^1 \hat{y}^2 \dots \hat{y}^K] \quad (19)$$

と表わせる。正規直交基底  $\hat{y}^k$  は次式に示す Schmidt の直交化法を用いて、新しい記憶画像が入力されるごとに求めることができる。

$$\hat{y}^1 = y^1 / \|y^1\| \quad (20)$$

$$\hat{y}^k = \frac{y^k - \sum_{j=1}^{k-1} (\hat{y}^j T y^k) \hat{y}^j}{\|y^k - \sum_{j=1}^{k-1} (\hat{y}^j T y^k) \hat{y}^j\|}$$

$$= \frac{y^k - \hat{M}(K) y^k}{\|y^k - \hat{M}(K) y^k\|} \quad (k \geq 2) \quad (21)$$

## 3 前処理による想起精度の改善

前章で述べたように、直交射影型モデルにおいて記憶画像の部分像をキーとして用いる場合には、想起出力に式(17)のクロストークノイズが重畳される。これは、Schmidt の直交化法により画像全体は直交化されるものの、任意の対応する部分像間は直交化されないためである。そこで、本章では、クロストークノイズを低減し想起精度を改善する方法について述べる。

従来、クロストークノイズを低減する手法として、

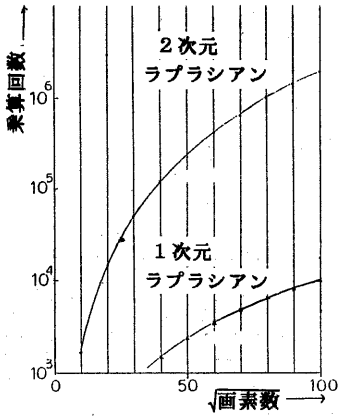


図1 乗算回数の比較

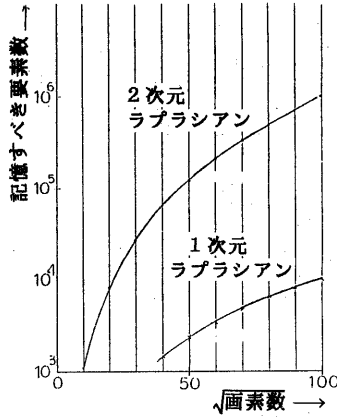


図2 メモリ量の比較

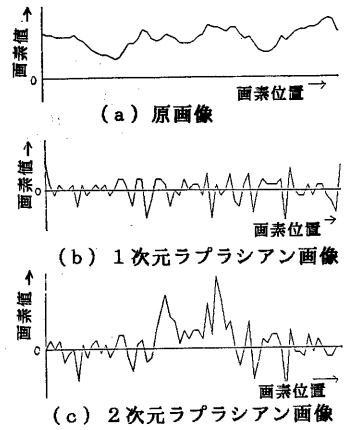


図3 特性の比較

2次元ラプラシアンによる前処理方法が提案されている<sup>4)</sup>。これは、2次元ラプラシアンにより、画像の高い空間周波数成分を強調し、任意の対応する部分間の直交性を高めることにより、クロストークノイズを低減する手法である。

この手法では、高周波成分の強調された想起出力から原画像を復元するため、後処理として2次元ラプラシアンの逆変換が必要となる。この逆変換は、画素数と等しい次元の連立方程式を解く必要があり、多大な計算量とメモリ量が必要となる。そこで、2次元ラプラシアンに比べ計算量、メモリ量が少なく済む1次元ラプラシアンによる前処理方法を提案する。

### 3. 1 1次元ラプラシアンによる前処理

第2章で述べたように、直交射影型モデルでは実質的にベクトル(1次元数値列)情報に対して連想処理する。その記憶想起のメカニズムは画像が本来保持している2次元構造には依存しない。従って、クロストークノイズを低減するためには、空間周波数を操作する必要はなく、式(12)のような1次元数値列としての高周波成分を強調すればよい。

1次元数値列の高周波成分を強調する簡便な演算として1次元ラプラシアンがある。

数値列  $f_1 f_2 \dots f_N$  に対する1次元ラプラシアンは次式により定義される。

$$g_i = f_{i+1} + f_{i-1} - 2f_i \quad (i=1, 2, \dots, N) \quad (22)$$

(但し、 $i \leq 0, i \geq N$  のとき  $f_i = 0$  とす)

式(22)は行列を用いて

$$g = D f \quad (23)$$

と表わせる。但し、Dは

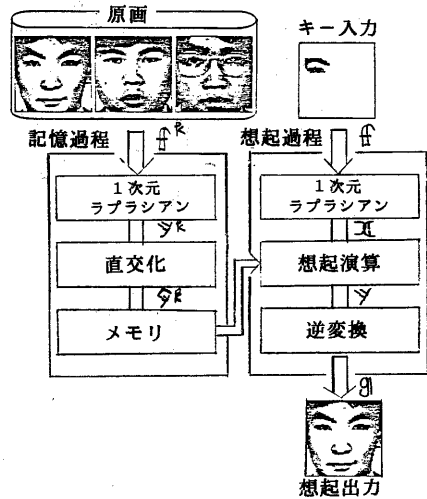


図4 シミュレーションの方法

$$D \cong \begin{bmatrix} -2 & 1 & & & 0 \\ & 1 & -2 & & \\ & & & \ddots & \\ & & & & 1 & -2 \\ 0 & & & & & & 1 \end{bmatrix} \quad \dots (24)$$

なるN次正方行列であり

$$f \cong [f_1 \ f_2 \ \dots \ f_N]^T \quad (25)$$

$$g \cong [g_1 \ g_2 \ \dots \ g_N]^T \quad (26)$$

である。

想起出力は、式(23)の変換が施された画像であり、原画像に復元するために式(23)の逆変換を行なう必要がある。この逆変換は  $f$  を未知ベクトルとする連立方程式(23)の解法に帰着する。(付録参照)

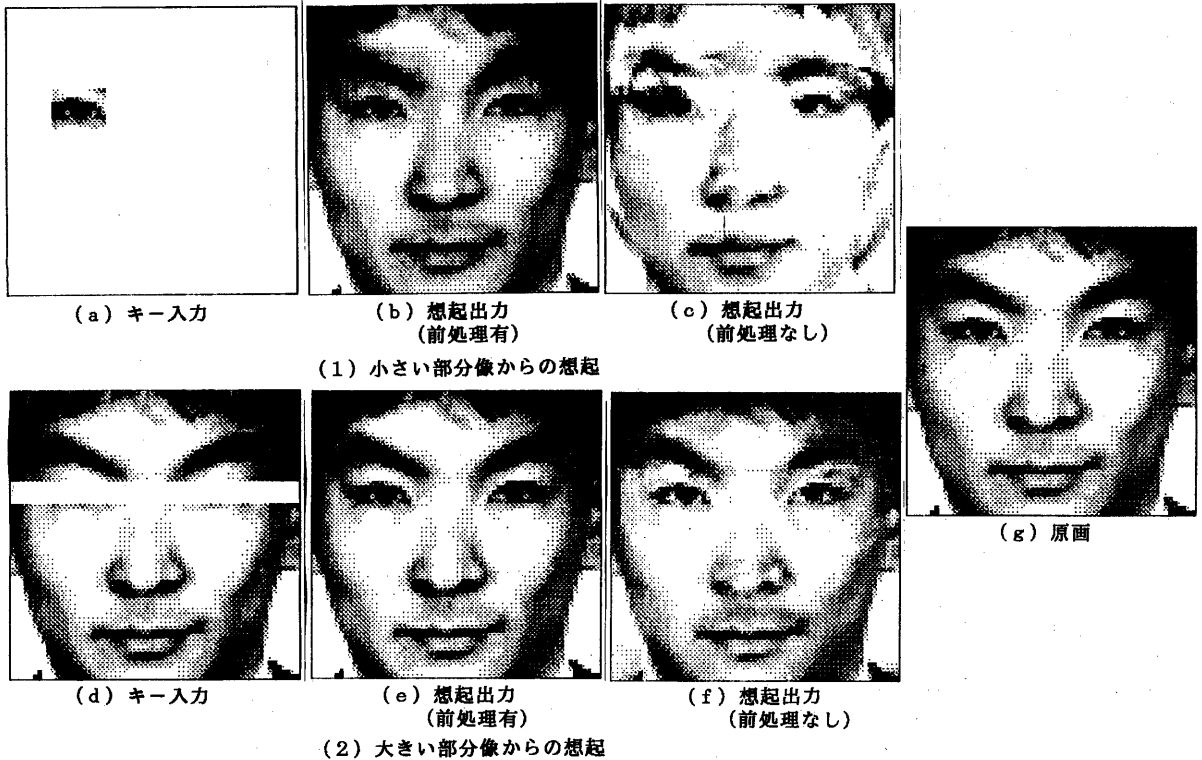


図5 前処理の有無による想起精度の比較

### 3. 2 1次元ラプラシアンと2次元ラプラシアンとの比較

前節でのべた1次元ラプラシアンによる前処理と従来の2次元ラプラシアンによる前処理とを、計算量、メモリ量、特性について比較する。なお、メモリ量、計算量の導出は付録に示す。

(1) 計算量の比較：乗算回数を比較する。簡単のために画像の縦方向の画素数 $M_1$ と横方向の画素数 $M_2$ は等しいとする。すなわち、全画素数を $N$ として、 $M_1=M_2=\sqrt{N}$ とする。前処理における乗算回数は等しいので逆変換についてのみ比較する。それぞれの乗算回数は

1次元ラプラシアン： $(2N-1)$ 回

2次元ラプラシアン： $(2N\sqrt{N}-2N+2\sqrt{N}-1)$ 回

となり、1次元ラプラシアンでは2次元ラプラシアンに比べ約 $1/\sqrt{N}$ 倍の乗算回数で済む(図1)

(2) メモリ量：逆変換では、計算に必要な変数を一時的にメモリに保持する必要がある。このメモリ量を(1)と同様の仮定のもとに比較する。1次元ラプラシアンでは $(2N)$ 個の要素の記憶場所が必要である。一方、2次元ラプラシアンの逆変換では $(N\sqrt{N}+\sqrt{N})$ 個の要素の記憶場所が必要となる。従って、1次元ラプラシアンの方が2次元ラプラシアンに比べ約 $2/\sqrt{N}$ 倍のメモリ量で済む。(図2)

(3) 特性の比較：実際の画像に対して1次元ラプラシアン、及び2次元ラプラシアンによる前処理を施した例を図3に示す。この図は、画像の1行分の

画素値を折れ線表示したものである。同図(a)は前処理前の画像、(b)は1次元ラプラシアン処理後の画像、(c)は2次元ラプラシアン処理後の画像である。(b)、(c)ともに高周波成分が強調されているが、(b)の方が零交差数が多い。それだけ、画像の対応する部分像間の直交性を高める効果があると考えられる。

## 4. 計算機シミュレーション

### 4.1 シミュレーションの方法

直交射影型モデルと1次元ラプラシアンによる前処理とを組み合わせる画像検索のシミュレーションを行なった。本シミュレーションの構成を図4に示す。

(1) 原画：白黒写真をスキャナから読み込んだ全12枚の多値画像である。大きさは $(64 \times 64)$ 画素、各画素256階調である。但し、表示は16階調に変換し、1画素当たり $(4 \times 4)$ ドットの網点を用いている。

(2) 記憶過程：入力画像 $f^k$ の1次元ラプラシアンをとり $y^k$ とする。1次元ラプラシアンは式(22)に示すたたみ込み和により計算する。次に、式(20)、(21)のSchmidtの直交化法により $y^k$ を正規直交基底 $\phi^k$ に変換し $\phi^k$ を記憶する。すなわち、式(19)の行列 $\hat{Y}(k)$ を次式のように更新する。



であり、

$$f = [f(1, 1) f(1, 2) \cdots f(2, 1) f(2, 2) \cdots f(m_1, m_2)]^T \quad (\text{付6})$$

$$g = [g(1, 1) g(1, 2) \cdots g(2, 1) g(2, 2) \cdots g(m_1, m_2)]^T \quad (\text{付7})$$

である。

逆変換ではガウスの消去法を用いて得られる漸化式(付 )~(付18)を逐次計算する。

$$(i) \alpha_1 = 1/4, \quad r_1 = g_1 \quad (\text{付8})$$

$$(ii) 2 \leq i \leq m_2$$

$$\alpha_i = 1 / (4 - \alpha_{i-1}) \quad (\text{付9})$$

$$\beta_{i, j} = \begin{cases} \alpha_{i-1} \beta_{i-1, j} & : j = m_2 + 1, m_2 + 2, \dots, m_2 + i - 1 \\ \alpha_{i-1} & : j = m_2 + i \end{cases} \quad (\text{付10})$$

$$r_i = g_i + \alpha_{i-1} r_{i-1} \quad (\text{付11})$$

$$(iii) m_2 + 1 \leq i \leq N$$

$$\alpha_i = 1 / \left\{ 4 - \left( \sum_{j=1}^{m_2-1} \alpha_{i-j} \beta_{i-j, i} \right) - \alpha_{i-m} \right\} \quad (\text{付12})$$

$$\beta_{i, i+j} = \begin{cases} 1 + \left( \sum_{k=1}^{m_2-j+1} \alpha_{i-k} \beta_{i-k, i+j} \right) + \alpha_{i-m_2+j} & : j=1 \\ \left( \sum_{k=1}^{m_2-j+1} \alpha_{i-k} \beta_{i-k, i+j} \right) + \alpha_{i-m_2+j} & : j=2, 3, \dots, m_2-1 \end{cases} \quad (\text{付13})$$

$$r_i = g_i + \sum_{k=1}^{m_2} \alpha_{i-k} r_{i-k} \quad (\text{付14})$$

但し、式(付10)は  $i+j \leq N$  まで計算する

$$(iv) f_N = -\alpha_N r_N \quad (\text{付15})$$

$$(v) N-1 \geq i \geq N-m_2+1$$

$$f_i = \alpha_i \left\{ \left( \sum_{j=i+1}^N \beta_{i, j} f_j \right) - r_i \right\} \quad (\text{付16})$$

$$(vi) N-m_2 \geq i \geq m_2$$

$$f_i = \alpha_i \left\{ \left( \sum_{j=1}^{m_2-1} \beta_{i, i+j} f_{i+j} \right) + r_{i+m_2} - r_i \right\} \quad (\text{付17})$$

$$(vii) m_2-1 \geq i \geq 1$$

$$f_i = \alpha_i \left\{ r_{i+1} + \left( \sum_{j=m_2+1}^{i+m_2-1} \beta_{i, j} r_j \right) + r_{i+m_2} - r_i \right\} \quad (\text{付18})$$

まず、 $\alpha_i, \beta_{i, j}, \bar{r}_i$  を式(付8)~(付14)に従い  $i=1$  から順に  $N = m_1 \times m_2$  まで計算する。次に、解  $f_i$  を式(付15)~(付18)に従い

$i=N$  から逆順に  $i=1$  まで計算する。

(2) 乗算回数:  $\alpha_i, \beta_{i, j}$  は、画像の大きさ  $m_1, m_2$  にしか依存しないので一度計算してメモリに記憶しておけばよい。 $\bar{r}_i$  は画像  $B$  に依存するので逆変換するたびに計算する。従って、逆変換するたびに式(付11), (付14), (付15)~(付18)を計算する必要があり、総計  $(2N\sqrt{N} - 2N + 2\sqrt{N-1})$  回の乗算が必要となる。

(3) メモリ量: 逆変換の際には、 $\alpha_i, \beta_{i, j}, \bar{r}_i$  を一時的に記憶しておく必要がある。そのため  $(N\sqrt{N} + \sqrt{N})$  個の要素の記憶場所が必要となる。