



画像処理と線形代数

—— 山下研究室 ~ 開発システム工学科 ——



山下 幸彦 助教授

現在、科学技術の発達を最も感じさせるものは何かと問うたら情報工学を思いつく人は多いだろう。その情報工学が対象とするものの一つに、画像処理というものがある。画像の持つ情報を有効利用するために、画像の変形や画像を特徴ごとに分類するものがそれだ。これからますます情報化が進められる中で画像データを扱うことも多くなり、それにともなって画像処理の重要性も増すことは明らかである。そこで今回は、主に画像処理の研究を行っている山下研究室を訪問させていただくことにした。



機械に文字はどう見えるのか

まずは画像処理の中でもパターン認識というものについて触れよう。パターン認識というのは図形などの空間的なものの形の特徴を判別し、それらを同じカテゴリーに対応づける操作である。もしそのようなことが文字に対してできたなら機械に手紙を読ませるような応用ができるだろう。しかし形が似ている似ていない、ということはとても漠然としたものであり、コンピュータが処理しやすい数値などに置き換えるのはとても困難なのである。したがって、それ自体が十分な研究対象となり得るものとなる。そこで、一般的にどのような方法と基準で、図形が似ている似ていないかの判断をしているのかを、基本的なパターン認識である文字識別を例に取り上げて説明しよう。

文字識別では当然、画像としての文字の入力過程から始まる。その後、汚れなど文字本体に関係のないものを取り除く過程があり、つぎに文字の特徴とはあまり関係ない線の太さを識別しやすい様に加工する過程を経る。

さて次がメインの特徴抽出である。特徴抽出は簡単に説明すると、文字をその文字の特徴を表すベクトルに変換することである。具体的な方法は

いったん、文字を適当な大きさに分割する(図1を参照)。その後、分割した画像それぞれについて、エッジ、つまり「へり」や「輪郭線」がどれだけ割合あるのかを数値化する。この数値化は上下、左右、左上右下、左下右上の四つの方向に対して行われる。こうすると分割された文字はそれぞれ別個の数字のデータを持つ。この数字の集まりは、文字がある部分においてどの方向にどの程度だけエッジを持っているかを示すベクトルと見なせる。そしてこのベクトルどうしの類似性をもとに文字が似ている似ていないの判断をしているのだ。したがって、効率よくベクトルの類似性を見つけることは効率よく文字を分類することに直結する。では、ベクトルが類似しているという判断はどの様に行うのだろうか。

まずは代表的なクラフィック法について説明しよう。ここではわかりやすいように十個の文字で済む数字を例にして識別の方法について書く。

まず、クラフィック法では0から9のそれぞれの文字について、その文字が持つ特徴を表すベクトルの特徴を抽出する行列を作る。

特徴を抽出する行列とは、図2の様に、ある数

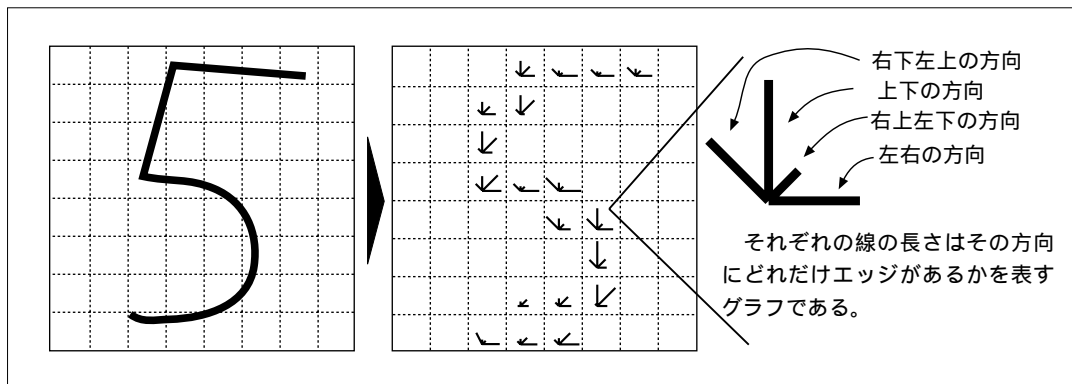


図1 ベクトル化

字を0から9に対応する行列を作用させた場合、3に対応する行列が最も強く反応したら、は3だと判断できる行列のことである。

この行列は、類内抽出作用素とよばれる。クラフィック法ではその類内抽出作用素の作り方にKL変換を使っている。簡単に説明するとKL変換による類内抽出作用素は図3(a)の式を元にして作られる。(a)の式が表しているものは、あるカテゴリ*i*に属する全てのベクトル f_1, f_2, \dots, f_n に対して $P^{(i)}$ をかけて、それをベクトル自身に対して減算をおこなったもののノルム(ベクトルの大きさ)をさらに2乗し、それら f_1, \dots, f_n について総和をとったものである。そして $P^{(i)}$ のランクを固定したとき、この(a)の値を最も小さくする $P^{(i)}$ を類内抽出作用素として扱う。つまり $P^{(i)}$ は平均的に $P^{(i)}f - f$ のノルムの2乗が小さくなるように決められたものなのだ。

(a)の値を $P^{(i)}$ が小さくするためにはベクトルの特徴に対してうまく $P^{(i)}$ の内部の要素が対応しな

くてはならない。

それでは、類内抽出作用素の具体的な使い方を説明するでしょう。0, ..., 9の類内抽出作用素を X_0, X_1, \dots, X_9 とする。また、ある数字をその数字の特徴を表すベクトルにしたものを f とし、この f をそれぞれ X_0, X_1, \dots, X_9 に対して掛ける。そして、新しくできた10個のベクトルそれぞれのノルムをとる。そしてこのノルムの大きさを比べて、たとえば X_3f のノルムが大きければその入力された画像は3であると判断する。

クラフィック法はこの流れで識別を行う。

しかしクラフィック法にも構造的な問題がある。この方法では、類内抽出作用素は、その作用素が反応するカテゴリのベクトルのみから作られている。これは作用素があるカテゴリに属するベクトルの特徴の平均を取って作られていることを意味する。つまり、ある文字が持つ特徴をほかの文字も持っていた場合ノルムの大小の差が出にくい。そのためクラフィック法では近い形の文字どうしても誤認識が生じ易いのだ。

その問題を解決するために、山下研が提案したのが相対KL変換を用いた相対KL変換法である。この変換で作られる用素は自分のカテゴリだけに含まれる特徴を抽出し、他のカテゴリも持っている特徴を抽出しない性質がある。その作用素がどう作られているのかを説明すると図3(b)のようになる。

(a)の式との違いは第2項が加えられていることである。第1項はKL変換と同じく自分のカテゴリの平均二乗誤差を表すのものであるが、第2項は他のカテゴリの類内特徴の平均を表してい

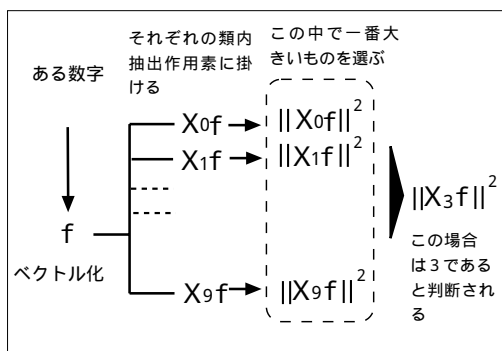


図2 類内抽出作用素

(a) K L 変換

$$\sum_f \sum_i \left\| f - P^{(i)} f \right\|^2$$

カテゴリー数を K、特徴空間の次元を N とする。カテゴリー i (i=1,2,...,k) の K L 変換 P⁽ⁱ⁾ は

$$\text{rank} (P^{(i)}) = M (M - N)$$

という条件のもとで (a) の値を最小にする作用素 P⁽ⁱ⁾ として定式化される。

ここで f は N 次元パターン、⁽ⁱ⁾ はカテゴリー i に属するパターンの集合、 $E_f^{(i)}$ は⁽ⁱ⁾ に関する平均を表す。

(b) 相対 K L 変換

$$\sum_f \sum_i \left\| f - X^{(i)} f \right\|^2 + \sum_i \sum_j \sum_f \left\| X^{(i)} f \right\|^2$$

カテゴリー数を K、特徴空間の次元を N とする。カテゴリー i (i=1,2,...,k) の K L 変換 X⁽ⁱ⁾ は

$$\text{rank} (X^{(i)}) = M (M - N)$$

という条件のもとで (b) の値を最小にする作用素 X⁽ⁱ⁾ として定式化される。

ここで f は N 次元パターン、⁽ⁱ⁾ はカテゴリー i に属するパターンの集合、 $E_f^{(i)}$ は⁽ⁱ⁾ に関する平均を表す。

また、 $E_{i,j}$ は i 以外のカテゴリーに関する平均を表す。

図 3 K L 変換と相対 K L 変換

る。クラフィック法で(a)の値を小さくするものとして P⁽ⁱ⁾ をとったのと同様に考えていくと、(b)の値を小さくするには第 1 項を小さくすると同時に第 2 項の値も小さくしないと行けない。ここで第 2 項の性質を考えると f の特徴が他のカテゴリーが持つ特徴と同じである場合、X がそれに対応してしまったのでは第 2 項は大きくなる。(b)の値を小さくするには X は f の独自の特徴に対応しなければならない。つまりこのときに作られる X は、自分のカテゴリーに含まれ、かつ他のカテゴリーには含まれない類内特徴を抽出することができる。

実際のパターン認識では認識対象のパターンの

性質によって(b)の第 2 項に適当なパラメータを掛けてさらに精度を上げている。

ここでクラフィック法と相対 K L 変換法のそれぞれの成果を具体的に説明しよう。

郵政省郵政研究所が作成したコンテスト用の手書きアラビア数字のデータを用いた認識実験における結果は、クラフィック法が認識率 96.71 % に対して、相対 K L 変換法のそれは 97.83 % であった。これは高々 1 % 強の差でしかないかと思うかもしれないが、何千何万の郵便物を仕分けするような場合に、多くの効果を期待できる。ゆえに、相対 K L 法の有効性は証明されたといえよう。



少ないデータでよりよい画像を

今度は画像圧縮の問題について触れよう。画像はそのままデジタルデータにすると、とても大きなデータ量になってしまう。それを避けるために、画像の性質を利用することによって画像を表現するデータ量を減らすのが画像圧縮である。また画像圧縮の基本的な事柄としておさえておいて欲しいことに無歪み方式と有歪み方式の違いがある。無歪み方式とは、画像を圧縮しさらに元に戻すと全く同じ画像が得られる方式のことだ。有名どころだと GIF などがある。それに対

し有歪み方式とは一度画像を圧縮した後さらに画像に変換し直すと得られる画像は元の画像の近似でしかないのである。有名どころだと JPEG などがそれである。無歪み方式は画像がきれいであるが圧縮率はあまり良くなく、それにたいして有歪み方式は画質はあまり良くないが圧縮率を良くすることが可能だ。当然、画像データは少なくできることに越したことはないが、かといって画質を落とすことも避けたいものである。

そこで、山下研で行っている画質の落ちがすく

ないJPEGの研究を紹介しよう。

まずは、ざっと従来のJPEGの画像圧縮の方法と問題点を説明しよう。画像を圧縮したデータにする手順としては、はじめブロック化、次にDCT(discreet cosine transform)、その後量子化、最後にエントロピー符合化の順番となる(図4)。

ブロック化は画像を8×8ドットを一まとまりとした単位に分けることである。以下の処理はこのブロックを単位として処理される。DCTは、まず画像をx, y軸それぞれについて様々なコサイン波の合成に変換する。DCTによって得られる係数は互いの相関が低くなり、圧縮が効率的に行えるようになる。次の量子化はDCTの波を離散的な値を取る波にする課程である。分かりやすくいうと端数を切り捨てることである。ここでもデータを減らすことができる。最後のエントロピー符号化は図5に示したハフマン符号化とおなじ原理で、伝える可能性が高いものを表現するデータ量は減らし伝える可能性が低いものを表現するデータ量を増やすことにより全体のデータ量を減らすことである。

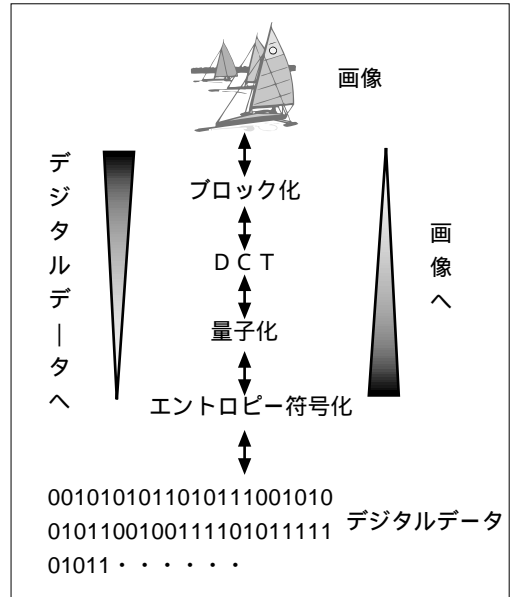


図4 JPEGの流れ

しかし画像をより小さく圧縮できるJPEGにも画質に関しては幾つか問題点がある。その一つにモスキート雑音というものがある。これは元の画像のへりの部分が再生後にそのエッジの部分において縞模様が浮き上がってしまうことがそれだ。

そこで山下研では上記の問題を改善できる新しいJPEGを提案している。モスキート雑音の解決には、先にふれたDCTの部分に選択的にKLTを使用するという方式を取り入れる。元々、DCTというのは、画素の相関が等方的で非常に大きい場合の極限におけるKLTから導かれたものなのだ。

KLTは画素の変化が一次的に起こっている場合に、その方向の変化を多く取り入れるように係数を掛けて画像を波に変えることができる。

しかしこれによりエッジの部分はエッジとして扱うので縞模様が出にくくなる。これによって今まで以上にクリアな画像を得ることが可能なのである。

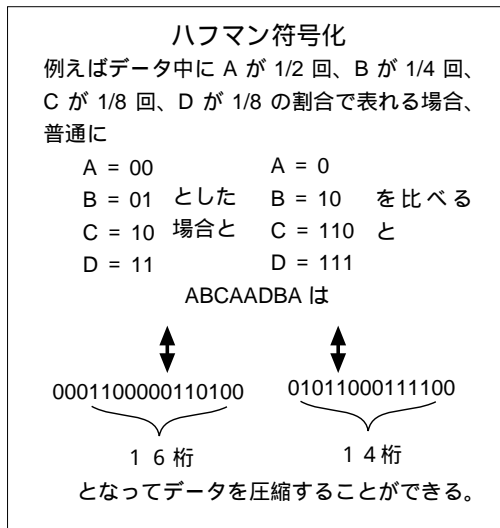


図5 ハフマン符号化

ここでは紹介できなかったが、山下研究室ではこの他にも様々な研究がなされている。新しいプロセッサの開発や4次元空間シミュレータの研究などだ。LANDFALLの取材でこの様に多くのこ

とを研究対象としている研究室を訪問したのは初めてである。どの研究も多くの成果をあげられることを祈りつつ筆を置く。

(小林 大介)