コンピュータビジョン　88－5－1
アルゴリズム　38－8－1
（1994．3．18）

計算幾何学的手法の画像処理への応用

浅野哲夫（大阪電気通信大学）　加藤直樹（神戸商科大学）

画像処理に関する研究は最近の入出力機器の高性能化に伴って質的な変革を遂げようとしている。計算機の能力が非常に限定されていた頃には 64×64 画素の画像が実験に使われた時代もある。今や 1000×1000 の画像ですら当り前になって来ている。この入力サイズの爆発は計算時間と記憶スペースの面でも深刻な影響をもたらしている。画像のサイズが一定であれば、小手先の修正（たとえば、ある部分を機械語で置き換えるなど）で効果を上げることもできたが、対象画像のサイズがこのように大きくなって来ると計算量のオーダーを解析してアルゴリズムを設計することが必要になるものと思われる。ここでは、2値画像から直線成分を検出する問題に計算幾何学の分野で考え出されたアルゴリズム設計の方法論が役に立つことを示したい。

1　問題の定式化

サイズ $N\times N$の入力のディジタル画像に対応する整数格子点の集合を $G=\{(x_i,y_i)|\ x_i=0,1,\ldots,N-1,\ y_i=0,1,\ldots,N-1\}$ とし、整数格子点 (x_i,y_i), $0\le x_i<N$, $0\le y_i<N$に エッジ点が存在する場合には $g(x_i,y_i)=1$ と定め、存在しない場合には $g(x_i,y_i)=0$ と定める。

直線 $y=ax+b$ に対応する格子点の集合（ディジタル像）の定義方法はいろいろと考えられるが、ここでは、この直線までの水平距離または垂直距離が 0.5 以内であるような格子点からなる集合として定義する。

エッジ点の集合 Pに対して、Pを（集合として）含むディジタル像を与える直線が存在するとき、Pは**直線成分**であるという。さらに、Pを真部分集合として含むどのようなエッジ点の集合についてもそのような直線が存在しないとき、Pを**極大直線成分**という。

上の定義に基づいて次の問題を考える。

[問題] n 個のエッジ点を含むサイズ $N\times N$の ディジタル画像 G と整数 t が入力として与えられたとき、G に含まれるサイズ t 以上のすべての極大直線成分に対応する直線の式を出力せよ。

2　素朴な方法

ディジタル画像から直線成分を抽出するもっとも単純な方法は、入力に含まれるエッジ点の中から2点を選ぶすべての組み合わせに対して、その2点を通る直線から水平または垂直距離が 0.5 以内のエッジ点の集合を求め、そのサイズが予め定めたしきい値を超えていれば、対応する直線の式を求めて出力するというものである。2点を定めると、後の処理は $O(n)$ の時間でできるから、この方法の計算時間は $O(n^3)$ である。考え方は単純であるが、直線成分に対応する直線が常にその中の2点を通るわけではないので、これでは全ての極大直線成分を検出することはできない。

3　Hough 変換に基づく方法

直線成分の抽出に対して現在最も一般的に用いられているのは、Hough 変換に基づく方法である。色々なバリエーションが考えられているが、本質的には、エッジ点 (x_i,y_i) を変換式

$$\rho=x_i\cos\theta+y_i\sin\theta$$

によって$\rho-\theta$平面の曲線に変換し、多数の曲線が通過する部分をバケット法で求めるというものであろう。バケットの個数を $O(N^2)$ とすれば、計算時間は $O(nN+N^2)$ であり、実用的な方法として重宝されている。この方法の性能はバケットの決め方に大きく依存するが、どのようにバケットを定めてもすべての極大直線成分を検出することは困難であると思われる。

4　L_1双対変換に基づく方法

$|u|\le 1$のとき、点 (x_i,y_i) から直線 $y=ux+v$ への垂直距離と水平距離を比較すると、必ず垂直距離の方が小さく、$|y_i-ux_i-v|$ と表わすことができる。これは次のアルゴリズムを示唆している。すなわち、直線の傾き u を定めたとき、各エッジ点 (x_i,y_i) について $v=y_i-ux_i$の値を計算し、vの値に関してエッジ点をソートして vの値の差が1以内であるような極大のエッジ

点集合を求めればよい（筆者らはこの変換式、$v = y_i - ux_i$を L_1双対変換と呼んでいる）。

傾き vとして $N \times N$の格子面の 2 点を通る直線の傾きだけを考えても極大直線成分をすべて検出できることが示されている。また、それらの傾きは分母が N以下で値が 1 以内の有理数の列（Farey 数列）として表現できる。重要な点は、上記の数列を傾きとして直線成分を列挙するとき、各傾きに対するソーティング操作は $O(n)$ 時間でできることである。傾きを固定したとき、vの値に関してソートして、ソート列をスキャンしながら vの値の差が 1 以内の集合を求めていくが、この操作は明らかに $O(n)$ 時間でできる。また、このスキャンは点の挿入と削除を繰り返す形で実行できるから、最小 2 乗法で近似直線を求めるための諸量を逐次更新しておけば、極大直線成分が得られたときに近似直線の式を $O(1)$ 時間で出力することができる。調べるべき傾きは $O(N^2)$ 通りあるから、全体の計算時間は $O(nN^2)$ となる。

上では傾きの絶対値が 1 以内の曲線成分を求める方法について述べたが、傾きの絶対値が 1 以上の場合についても同様の方法が考えられる。

5 計算幾何学の技法に基づく方法

5.1 平面走査法に基づく方法

傾きの絶対値が 1 以内の直線のディジタル像は、その直線を上下に 0.5 だけずらした 2 本の平行直線（定義直線と呼ぶ）で囲まれた領域に含まれる格子点の集合として定義される。したがって、任意の直線成分に対してそれを含むディジタル像を与える直線が存在するが、内に含まれるエッジ点の集合を変えることなく適当に上下に平行移動することにより、定義直線上にエッジ点を乗せることができる。このことより、すべてのエッジ点について、この点を通る直線の傾きを−1 から 1 まで変化させながら、この直線とそれを 1 だけ上にずらした直線で囲まれる領域に含まれるエッジ点の集合を求めれば、全ての直線成分を検出することができる。1 個のエッジ点につきこの回転走査は $O(n \log n)$ 時間で実行できるから、全体の計算時間は $O(n^2 \log n)$ である。

5.2 トポロジカル・ウォークに基づく方法

エッジ点 (x_i, y_i) が直線 $y = ax + b_i$から垂直方向に 0.5 以内の範囲に存在するための必要十分条件は

$$-0.5 \le y_i - ax_i - b \le 0.5、$$

すなわち、

$$-x_i a + y_i - 0.5 \le b \le -x_i a + y_i + 0.5$$

が成り立つことである。これを、$x-y$平面の点 (x_i, y_i) から $a-b$ 平面において 2 本の平行線 $b = -x_i a + y_i - 0.5$ と $b = -x_i a + y_i + 0.5$ によって囲まれた領域への変換として考えることができる。この領域を点 (x_i, y_i) の支配領域と呼ぶ。t 個のエッジ点の支配領域が共通部分をもつなら、それら t 個の点はその共通部分の任意の点 (a, b) に対応する直線 $y = ax + b$ のディジタル像に含まれる。したがって、問題は $a-b$ 平面において t 個以上の支配領域が重なった部分を求めることに帰着される。

$2n$ 本の直線は $a-b$ 平面を $O(n^2)$ 個の小領域（セルと呼ぶ）に分割する。上の議論より、各セルがそのセルに対応する直線のディジタル像に含まれるエッジ点の集合に対応しているから、隣接セルを次々に訪問するという形でこれらのセルをすべて訪問することができれば、サイズが t 以上の直線成分をすべて検出することができる。$O(n^2)$ の記憶領域を用いても $O(n^2)$ 時間でそのような訪問を実現することは自明ではないが、計算幾何学の分野で知られているトポロジカル・ウォーク [1] という技法を用いると $O(n^2)$ 時間、$O(n)$ 記憶領域で可能である。

トポロジカル・ウォークでは次々に隣接セルを訪問するため、対応するエッジ点の集合は点の挿入と削除の系列で管理することができる。したがって、L_1双対変換に基づく方法と同様、最小 2 乗法で近似直線を求めるための諸量を逐次更新していくことができる。

6 円や楕円の検出

直線成分だけでなく円や楕円の成分を検出することも重要な問題である。筆者らは曲線成分を検出する一般的な方法を確立することに成功した。紙面の都合上、ここでは詳しく述べないが、アルゴリズム研究会において発表しているので参照されたい。

参考文献

[1]T. Asano, L.J. Guibas, and T. Tokuyama: "Walking in an Arrangement Topologically", International J. of Computational Geometry and Applications, to appear.