

デジタル幾何学における図形と
その Computational Complexity について

広島大学 工学部 中村 昭 (Akira NAKAMURA)

広島大学 工学部 会沢 邦夫 (Kunio AIZAWA)

1. はじめに

Image processing 或いは Computer graphics の基礎理論として、最近デジタル幾何学が定義され、興味ある話題を提供している ([1], [2])。これらは、主に平面画像の中に現れるパターンの分類に関係している。この分類は、パターンの幾何学的性質をもとに行われる。しかし、これをデジタル計算機で処理するためには、パターンはデジタル化されていなくてはならない。このため、この分類は、与えられた格子点の有限集合が、ある幾何学的性質を持つ実際の図形から得られるか否かを判定する問題に変換される。一般に、このような判定は自明ではなく、直線に対する議論は文献 [3] - [5] に見られる。ここでは、あるデジタル化の方法をもとに、円と長方形を認識するアルゴリズムと、その Computational Complexity について述べる。

2. 準備

任意の二点 p, q に対して pq は p と q を結ぶ線分を表し、 $|pq|$ は pq の長さを表すとす。 Q を、平面の任意の有限部分とする。デジタル計算機での処理のため、普通 Q は格子

点の有限集合で表される。この集合は、 Q のデジタル像と呼ばれ、 Q からそのデジタル像への写像をデジタル化と呼ぶ。ここで用いられるデジタル化は、主に曲線に対して使われているもので、次のように定義される。

定義 1. (Kim's grid digitization)

格子点の有限集合 D は、次の二つの条件が成り立つならば、曲線 f のデジタル像であるといわれ、 $D = IK(f)$ と表される。

(i) 格子点 $d = (h, k)$ が D の要素であれば、 f は次のような点 $z = (x, y)$ で格子を横切る： $\max\{|h-x|, |k-y|\} < 1$ かつ d は f の右側にある。

(ii) 格子点 $d = (h, k)$ が D の要素でないならば、上の条件を満足する横断点 $z = (x, y)$ は存在しない。

上の定義では、 f はある方向を持つと仮定され、この方向によって、 f の“右側”が定められている。今後、円と長方形の外周は、時計回りの方向を持つと仮定する。

定義 2.

(i) デジタル線分

格子点の有限集合 D は、 $D = IK(f)$ なる線分 f が存在するときまたそのときに限り、デジタル線分と呼ばれる。

(ii) デジタル円

格子点の有限集合 D は、 $D = IK(c)$ なる円 c が存在するときまたそのときに限りデジタル円と呼ばれる。

(iii) デジタル長方形

格子点の有限集合 D は、 $D = IK(r)$ なる長方形 r が存在するときまたそのときに限り、デジタル長方形と呼ばれる。

デジタル円とデジタル長方形を上のように定義した場合、一点またはたがいに接する数個の点からなる退化した円や長方形が生ずる可能性がある。しかしここでは、このような図形の認識はそれほど重要でないとして考えないことにする。

3. 円を認識するアルゴリズム

今、 $d = (p, q)$ を平面上の任意の点とし、 $x = p$, $y = q$, $y = x + q - p$, $y = -x + q + p - 1$ によって分割される8つの楔形の部分平面をOctants と呼ぶ。それらは、直線 $y = q$ の $x > 0$ なる部分のすぐ上にあるものから順に、反時計回りに、ローマ数字が付されているものとする。このとき、格子点の有限集合 D と点 d に対する次のような条件を考える。

- (1) R の各要素は、少なくとも二つの8-隣接点をもつ。そして、
 (2) $m = (p_i, q_i)$, $n = (p_j, q_j)$ を次のような格子点とする：

$$\begin{aligned} |dm|^2 &= \max \{ |da|^2 \mid a \in R \}, \\ |dn|^2 &= \min \{ |da|^2 \mid a \in R \}. \end{aligned}$$

この n に対して、 n' を次のように定める。

$$n' = \begin{cases} (p_j+1, q_j) & n \in \text{Octant I または VIII のとき,} \\ (p_j, q_j+1) & n \in \text{Octant II または III のとき,} \\ (p_j-1, q_j) & n \in \text{Octant IV または V のとき,} \\ (p_j, q_j-1) & n \in \text{Octant VI または VII のとき.} \end{cases}$$

ならば、 $|dm|^2 < |dn'|^2$ が成り立つ。

今後、上の条件をCDC-K と呼ぶことにする。このとき、次の補題を示すことができる。

補題 1. 与えられた格子点の有限集合 R と、与えられた一点 d に対して、 R と d が条件 CDC-K を満たすときまたそのときに限り $R = I K (q)$ で中心が d であるような円 q が存在する。

証明: まず、 d を中心とし $R = I K (q)$ なる円 q が存在するならば、 R は条件 CDC-K を満たすことを証明する。条件 CDC-K (1) が成り立たないと仮定する。ならば、図 1 に示すような点が R に含まれることになり、明らかに R はデジタル円ではない。つぎに、条件 CDC-K (2) が成り立たないとする。Octant I では、 q が格子を横切るとの二点を結ぶ線分も -1 より小さい傾きを持つ。従って、文献 [3] に示されるように、 R のどの点も q から格子幅の $\frac{1}{2}$ 以上水平方向に離れることができない。同様に、Octant II では、格子幅の $\frac{1}{2}$ 以上垂直方向に離れることができない。このような性質が、各 Octant に対して成り立つので、もし CDC-K (2) が成り立たないならば m と n をともに $I K (q)$ に含む円 q は存在しない。逆に、条件 CDC-K が R に対して成り立っているならば、 $|dm| < r < |dn'|$ を満足する任意の r にたいして、 d を中心とし半径 r の円 q は $R = I K (q)$ なる性質を持つ。なぜなら、もし $I G (q)$ に含まれるが R には含まれないような格子点 a が存在するならば、 $|da|$ は $|dm|$ より大きいかまたは $|dn'|$ より小さい。これは m と n の定義に矛盾する。また、もし R には含まれるが $I G (q)$ には含まれない格子点 a が存在するならば、前の議論から $R \neq I G (q)$ が成り立つ。しかし、このような R に対して条件 CDC-K は成り立たない。■

次に、条件 CDC-K を用いて中心 d を有限時間内に発見し、 R がデジタル円であるか否かを判定するアルゴリズムを示す。

$d_1 = (p, q)$, $d_2 = (p+1, q)$, $d_3 = (p+1, q+1)$, $d_4 = (p, q+1)$ なる格子点を 4-centers と呼び、図 2 のように Octants を定める。

このとき、図 3 に示すアルゴリズム RDC-K を考えるならば次の定理を証明することができる。

定理 1. 格子点の有限集合 R は, R がアルゴリズム RDC-K によって受理されるときまたそのときに限り, デジタル円である。

証明: まず, アルゴリズム RDC-K が必ず停止することを示す。 R の内部には有限個の格子点しかないので, (a) - (d) の繰り返しは (b) - (c) の繰り返しが停止するならば, 必ず停止する。そこで, (b) - (c) の繰り返しが無限に続き得ると仮定する。ならば, 垂直二等分線は有限本しかないので, step (c) で同じものが一度以上使われなくてはならない。しかし, そのときは d と m は垂直二等分線を挟んで反対側にあるので, 新しい S は空になる。これは仮定と矛盾する。故に, (b) - (c) の繰り返しは必ず停止する。次に, RDC-K の正当性を示す。もし R が step (b) で受理されるならば, R と d は条件 CD C-K を満たす。補題 1 より R はデジタル円である。逆に, R が step (d) で棄却されると仮定する (最初で棄却される場合は明らか)。ならば, 正方形 $d_0d_1d_2d_3$ とすべての垂直二等分線によって定義されるどの多角形の重心も, $q = IK(R)$ なる円 q の中心にならない。ところで, 点 m, n, n', d の定義から, どの多角形に含まれる d に対しても R と d が条件 CD C-K を満たさないならば, CD C-K を満たす点が R の内部に存在しないことがわかる。故に, 格子点の有限集合 R はデジタル円ではない。■

4. 長方形を認識するアルゴリズム

一本のデジタル線分 S に含まれる一つの格子点 a を定めたとき, a を通り $S = IK(f)$ なる線分 f のとり得る最大の傾きを $m(a)$ とし, 最小の傾きを $n(a)$ とする。また, a を通り傾き $m(a), n(a)$ 持つ線分で S の原像になるものを, それぞれ $LM(a), Ln(a)$ と呼ぶことにする。

補題 2. R を任意のデジタル長方形とする。 R を構成する四つのデジタル線分上に格子点 $a_i (1 \leq i \leq 4)$ を一点ずつ定める。ならば, 少なくとも一辺が $Lm(a_i), Ln(a_i)$

($1 \leq i \leq 4$) のどれか一つに一致し、かつ $R = IK(r)$ なる長方形 r が存在し、その逆も成り立つ。

証明： R はデジタル長方形なので $R = IK(r')$ なる長方形 r' が存在する。もし r' が $L_m(a_i), L_n(a_i), (1 \leq i \leq 4)$ のどれかを辺として持つならば、 $r = r'$ とおけばよい。そうでないなら、 r を回転移動すればよい。逆は明らか。■

このとき、格子点の有限集合 R に対して、次のような条件を考える。

- (1) R のそれぞれの要素は、少なくとも二つの 8-隣接点をもつ。
- (2) $R = P_1 \cup P_2 \cup P_3 \cup P_4$, ここで、 $P_i (1 \leq i \leq 4)$ はデジタル線分であるが、 $P_i \cup \{d\} (d \notin P_i \text{ かつ } d \in R)$ はデジタル線分ではない。
- (3) $P_h (1 \leq h \leq 4)$ は、ある一つの $P_i (i \neq h)$ 以外の二本の線分 P_j, P_k と出会って、 P_h の両端点はそれぞれ P_j と P_k の端点である。
- (4) 以下の条件を満足する四つの格子点 $a_1, a_2, a_3, a_4, (a_i \in P_i)$ が存在する。
 - ・ P_i と P_j が、出会っていないならば、
 区間 $[n(a_i), m(a_i)]$ と $[n(a_j), m(a_j)]$ には共通部分が存在する。この共通部分の最大を $m_{ij}(a_h)$, 最小を $n_{ij}(a_h), (h = i, j)$ とする (図4参照)。
 - ・ ある同じ傾きを持つ線分 (例えば) $L_{m13}(a_1), L_{m13}(a_3)$ に対して次の不等式が成り立つ (図5参照)。

線分 $a_1a_2, a_2a_3, a_3a_4, a_4a_1$ の中点をそれぞれ c_1, c_2, c_3, c_4 とする。

$$\begin{aligned} \text{ならば, } & |c_4b_2|^2 \leq |c_4a_4|^2 \leq |c_4b_1|^2, \\ & |c_1b_3|^2 \leq |c_1a_1|^2 \leq |c_1b_4|^2, \\ & |c_2b_6|^2 \leq |c_2a_2|^2 \leq |c_2b_5|^2, \\ & |c_3b_7|^2 \leq |c_3a_3|^2 \leq |c_3b_8|^2. \end{aligned}$$

今後、上の条件をCDR-K と呼ぶことにする。

定理 2. 格子点の有限集合 R は、条件CDR-K を満たすときまたそのときに限り、デジタル長方形である。

証明: まず、 $R = IK(r)$ なる長方形 r が存在するならば、 R は条件CDR-K を満たすことを示す。そのような長方形が存在するならば、 R は明らかにCDR-K (1) - (3) を満たす。また、補題1よりCDR-K (4) を満たすことも容易に判る。逆に、条件CDR-K が R に対して成り立っていると仮定する。CDR-K (1) - (3) から、 R は四つのデジタル線分からなる中空の図形であることが容易に判る。また、CDR-K (4) より、図5に示されるような $Lm13(a1)$ に垂直でそれぞれ $a2, a4$ を通る二本の線分と、 $Lm13(a3)$ に垂直でそれぞれ $a2, a4$ を通る二本の線分を引くことができる。 $Lm13(a1)$ と $Lm13(a3)$ は平行なので、これらの線分は長方形を構成する。■

5. 時間計算量について

デジタル円 R が $N \times N$ の大きさの平面に含まれているならば、円周上の点は $O(N)$ 個ある。従って垂直二等分線は $O(N^2)$ 本あり、(b) - (c) の繰り返しは $O(N^2)$ 回の計算を必要とする。さらに、 R の内部には $O(N^2)$ 個の格子点があるので、結局アルゴリズムRDC-K は $O(N^4)$ の時間計算量を持つ。しかし、 R の中心を含む正方形 S の候補をあらかじめ定数個に定めることが可能である。例えば、中心は最も左 (又は右) にある R の格子点と同じ行にある R の格子点との中点の近く ($\pm 1/2$) の列にあるはずである。また同時に、最も上 (又は下) にある R の格子点と同じ列にある R の格子点との中点の近くの行に存在しなくてはならない。このような方法で、高々四個の S の候補を定めることができる。従って、RDC-K を以上の方針で修正するならば、新しいアルゴリズムは $O(N^2)$ の時間計算量を持つはずである。

次に、デジタル長方形 R が $N \times N$ の大きさの平面に含まれているならば、 R の点は $O(N)$ 個ある。条件CDR-K (1)と(3)を判定するには明らかに $O(N)$ 回の計算で十分である。また、文献〔5〕で述べられているように、CDR-K (2)の判定も $O(N)$ 回でできる。デジタル線分の傾きは $O(N)$ で求められ、CDR-K (4)の不等式の判定は定数オーダーである。しかし、CDR-K (4)の判定は、すべての a_1, a_2, a_3, a_4 に対して行われるので、結局CDR-Kは $O(N^5)$ の時間計算量を持っている。

References

- 〔1〕 A. Rosenfeld: Picture Languages, Academic Press, New York (1979), Chapter 2, Digital Geometry.
- 〔2〕 E. F. Krause: Taxicab Geometry, Addison-Wesley, Menlo Park (1975).
- 〔3〕 A. Rosenfeld: Digital straight line segments, IEEE Trans. Computer 23 (1974), pp. 1264-1269.
- 〔4〕 C. E. Kim: On cellular straight line segments, Computer Graphics and Image Processing 18, (1982), pp. 369-381.
- 〔5〕 A. Rosenfeld and C. E. Kim: How a digital computer can tell whether a line is straight, The American Mathematical Monthly, Vol.81, No.4 (1982), pp.230-235.
- 〔6〕 A. Nakamura and K. Aizawa: On digital circles, TR-1193, Computer Science Center, Univ. of Maryland (1982).
- 〔7〕 M. I. Shamos: Geometric complexity, Proc. 7th ACM Symposium on the Theory of Computing (1975), pp.224-233.

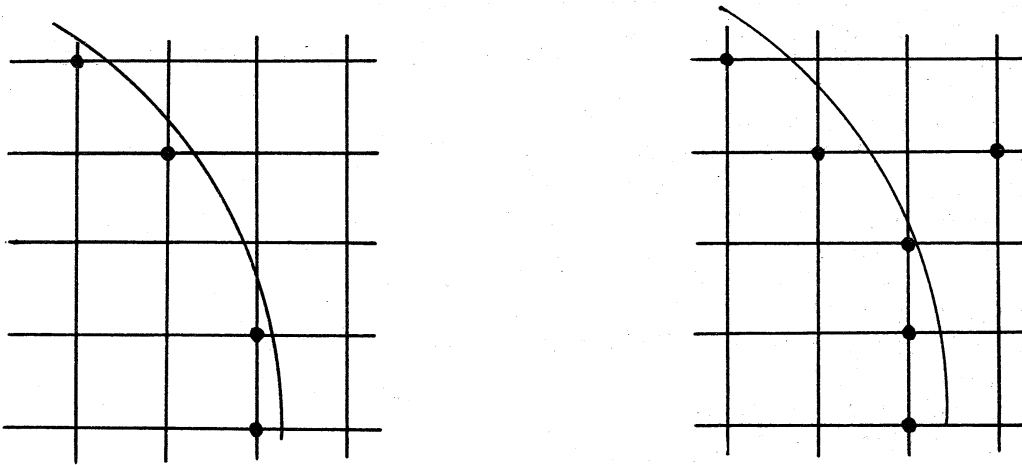


图 1

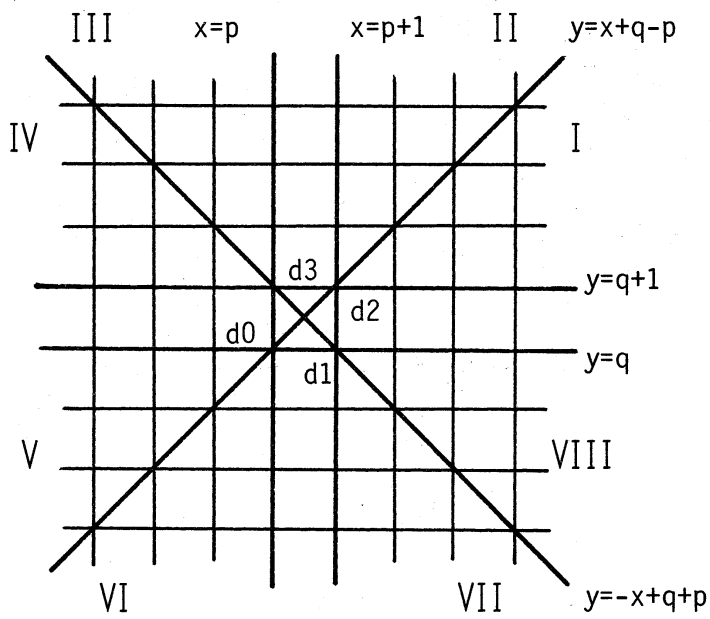


图 2

Algorithm RDC-G :

- if CDC-K (1) holds for R then go to step (a) ;
 else R is rejected ; stop ; end
 (a) Take new 4-centers d_0, d_1, d_2, d_3 at any interior lattice points of R and call it polygon S ;
 (b) Let d be the barycenter of polygon S ;
 if CDC-K (2) holds for R with d then
 begin R is accepted ; stop ; end
 else go to step (c) ;
 (c) Let $f(x) = ax+b$ be the perpendicular bisector of mn' ;
 if m is in the half-plane F defined by $f(x) > 0$ then
 Set new S as $S \cap F$;
 if m is in the half-plane G defined by $f(x) < 0$ then
 Set new S as $S \cap G$;
 if f is null then go to step (d) ;
 else go to step (b) ;
 (d) if there exist new 4-centers then go to step (a) ;
 else begin R is rejected ; stop ; end

图 3

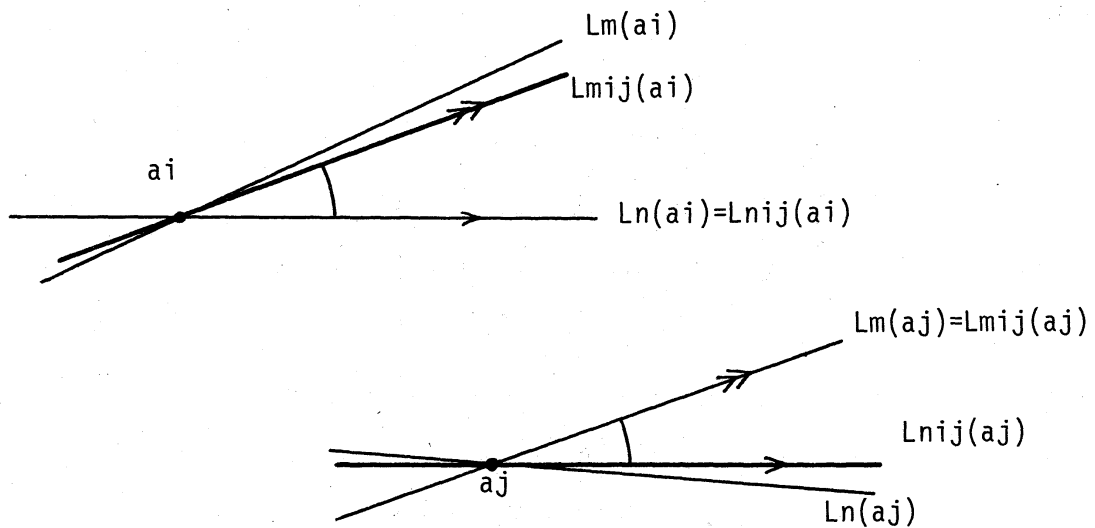
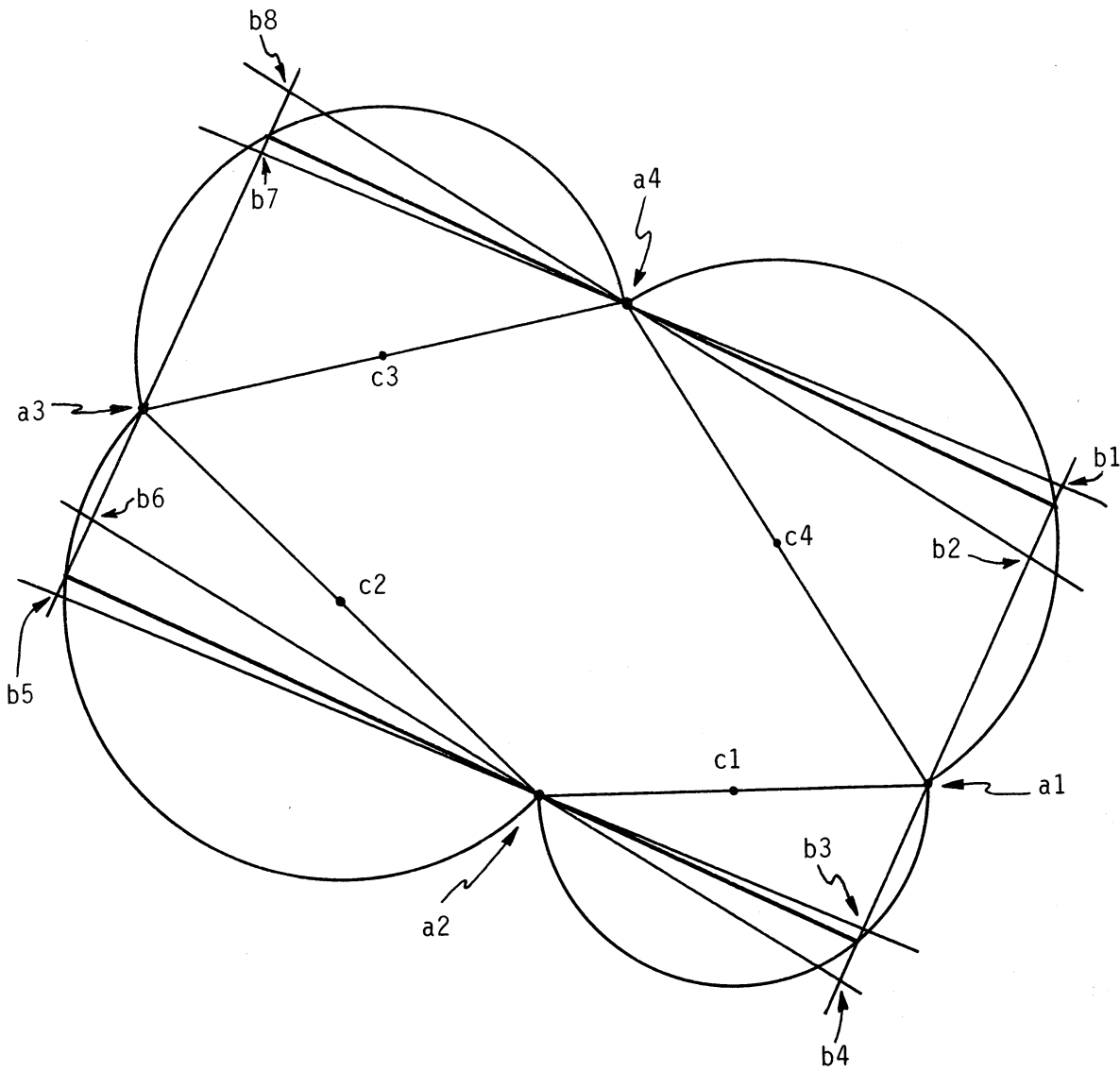


图 4



☒ 5