

## 文字輪郭線の円弧と直線とによる近似†

山崎 一生\*\* 井村 博徳\*\*\*

与えられた線図形をデジタル的に正確に表現しようとする、線図形が与えられている領域を細かく縦横に分割することが必要となり、取り扱うべき情報量が龐大なものとなってしまふ。元の図形に忠実に、しかも情報量少なく近似する方法の一つとして、線図形上の点を適当な間隔で標本化して記憶しておき、他の部分は適当な関数によって補間することが考えられる。本稿では円弧と直線とによるデジタル文字輪郭線の近似について検討を行う。元の線図形と近似図形との近さは、曲線の法線方向に測った量が人間の感覚との整合性がよい。与えられた任意の曲線から法線方向に一定距離離れた線が満たすべき式を導く。法線方向に一定距離離れた線の式は、与えられた曲線が初等的な関数の場合、直線あるいは円弧のときにしか陽の形には求められない。碁盤目状の領域に表現されている平仮名「あ」と漢字「超」とについて、直線あるいは円弧による近似を試みた結果、十分なデータ圧縮機能のあることがわかった。800×800の碁盤目状の領域に表現されている「あ」の場合、元の輪郭線データの約5%の標本点があれば、円弧あるいは直線による補間によって、与えられた元の輪郭線と目視ではほとんど区別できない程度の近似図形を得ることができる。人間の感覚と整合性のよい元の曲線の法線方向に測った誤差と、計算の容易な座標軸に沿って測った平均2乗誤差との間には、2乗の関数関係があるので平均2乗誤差を近似のよさの評価尺度とすることが可能である。

## 1. ま え が き

キャラクタディスプレイ、ドットプリンタ、レーザプリンタ、CRT 写真植字機等においては、文字パターンはドットマトリクスによって表現されている。これらの文字パターンはデジタル文字と一般に呼ばれている。JIS 化されている24×24ドット文字<sup>1)</sup>のように解像度の低い文字の場合には、すべての文字パターンはビットマップあるいはドットマトリクスの形で記憶されているのが普通である。

一方、解像度が高く、高品位の印字出力が要求される場合には、文字の輪郭線を記憶しておいてドットマトリクスの形に展開するか、スプライン等の関数近似によってドットマトリクスの形の文字を計算機で直接生成している。後者の代表的な例にKnuthのMETAFONTがある<sup>2)</sup>。輪郭線の記憶方式には大別すると次の二つの方式がある。一つはランレンクス法で、ラスタ走査によって文字パターンを生成する装置でよく用いられている<sup>3)</sup>。もう一つは輪郭線の座標を適当な間隔で標本化して記憶しておき、他の部分は補間によって発生する方式である。後者の方式で代表的なものはベクトル方式と呼ばれているもので、標本点の間を直線で繋いで輪郭線を発生するものである<sup>3),4)</sup>。

文字パターンの輪郭線を含む線図形あるいは平面曲

線を直線あるいは円弧を用いて近似する方法はきわめて一般的なものである<sup>3),5),6)</sup>。たとえば文献5),6)では、直線あるいは直線を円弧とを用いて与えられた平面曲線のおおよその形を表現して情報量を減らす方法について検討している。これらの文献では与えられた曲線を忠実に近似することはほとんど意図されておらず、また与えられた曲線と近似曲線との近さについては主観的な評価が定性的になされているだけである。一方、文献3)では文字輪郭線を直線と円弧とを用いて近似することによって、直線のみで近似する場合よりも自然な形が得られることが定性的に述べられているにすぎず、標本点の選定方法、データ圧縮の程度についてはまったく触れられていない。

本稿では与えられた高分解能で高品質のデジタル文字の輪郭線を直線と円弧とで忠実に近似して情報量を減らす方法について検討する。文字輪郭線を含む平面曲線の近似においては、近似のよさの尺度の定義法と標本点の選定方法とが中心的な課題である。次の2章では与えられた曲線と近似曲線との間の近さの評価尺度について述べる。3章では与えられた文字輪郭線データのある許容誤差限界以内で、円弧あるいは直線で近似して標本点を求める方法について述べる。最後の4章では許容誤差限界と標本点数あるいは平均2乗誤差との関係を、800×800の碁盤目状の領域に表現されている平仮名「あ」とゴシック体の漢字「超」の場合について示し、許容誤差限界と平均2乗誤差との間の関係の検討を行う。

† Curve-Fitting for Character Outlines with Circular Arcs and Straight Lines by ISSEI YAMASAKI (Faculty of Engineering, Niigata University) and HIRONORI IMURA (Nippon Seiki Co., Ltd.).

\*\* 新潟大学工学部情報工学科

\*\*\* 日本精機(株)

### 2. 近似の良さの評価尺度

いま、2次元  $xy$  平面を考え、与えられた曲線を  $y=g(x)$  (1)

とし、その近似曲線を

$$y=f(x) \quad (2)$$

とするとき、通常関数近似においては、ある点  $P$  における誤差  $\varepsilon_v$  は式(1), (2)から

$$\varepsilon_v = |f(x) - g(x)| \quad (3)$$

で定義される。この  $y$  方向誤差  $\varepsilon_v$  は視覚的な近さを反映した量とは一般にはいえない。すなわち、 $\varepsilon_v$  は点  $P$  の近傍における曲線の勾配が大きい場合には、視覚的な近さよりも大きな値となり、勾配が緩くなるに従って視覚的な近さに合う値となってくる。線図形の近似問題の場合には、上に定義された  $y$  方向誤差ではなく、点  $P$  における接線の法線方向に測った誤差が、視覚的な近さの評価量として適切であるといえる。

さて、与えられた曲線からその接線の法線方向に  $w$  だけ離れた点の軌跡を表す式を導こう。与えられた曲線は式(1)であるとするとき、この曲線上のある点  $P: (x, y)$  における接線ベクトルは  $(1, g'(x))$  と表される。この接線ベクトルと直交する単位法線ベクトル  $\mathbf{a}$  は、

$$\mathbf{a} = \left( \mp \frac{g'(x)}{\sqrt{1+g'(x)^2}}, \pm \frac{1}{\sqrt{1+g'(x)^2}} \right) \quad (4)$$

となる。したがって、曲線  $y=g(x)$  の上の点  $P: (x, y)$  から  $w$  だけ離れた点の座標  $(\xi, \eta)$  は

$$\begin{cases} \xi = x \mp (g'(x)/\sqrt{1+g'(x)^2})w \\ \eta = y \pm (1/\sqrt{1+g'(x)^2})w \end{cases} \quad (5)$$

と表される。

与えられた曲線  $y=g(x)$  が何らかの関数の形に書き表せかつ微分可能であれば、上の式(5)によって曲線  $y=g(x)$  から  $w$  だけ離れた点の軌跡  $(\xi, \eta)$  を求めることができる。与えられた曲線で2次以上の多項式、3角関数、指数関数、楕円弧等の場合には、軌跡  $(\xi, \eta)$  を初等的な関数の形に書き表すことはできない(曲線  $y=g(x)$  が導関数を有する場合には、式(5)を用いて軌跡  $(\xi, \eta)$  を逐次計算することは可能である)。軌跡  $(\xi, \eta)$  を初等的な形に求めることができるのは、与えられた曲線が直線あるいは円弧の場合のみである。このことは平面曲線の近似に直線あるいは円弧を用いることにより、人間の感覚に合う近似を行うことが可能なことを示している。

いま、与えられた直線と円弧とをそれぞれ

$$\begin{cases} y = ax + b \\ y = \pm \sqrt{r^2 - (x - x_0)^2} + y_0 \end{cases} \quad (6)$$

とすると、これらの直線あるいは円弧から  $w$  だけ離れた点の軌跡の式は、式(5)から、それぞれ

$$\begin{cases} \eta = a\xi + b \pm w\sqrt{1+a^2} \\ (\xi - x_0)^2 + (\eta - y_0)^2 = (r \pm w)^2 \end{cases} \quad (7)$$

となる。式(7)を直線あるいは円弧の許容誤差限界線(の式)と呼び、これらの直線あるいは円弧の二つの許容誤差限界線によって挟まれた領域を許容誤差限界内と呼ぶことにする。

### 3. 文字輪郭線の近似

#### 3.1 文字輪郭線データ

文字輪郭線は図1に示すように、2次元  $xy$  平面上に与えられる。この図における輪郭線は、文字図形を水平( $y$ )方向に走査して表示する場合の、黒部分の開始点と終点とを表している。したがって、 $y$  軸に平行な水平の文字輪郭線は存在しない。

黒部分の開始あるいは終了を表す一繋がりのデータ点をブロックと呼ぶことにする。図1に示してある文字「あ」の輪郭線は20個のブロックから構成されている。各ブロックはつねに  $x$  に対する1価関数となっている。これらのおおののブロックのデータ点を円弧あるいは直線で近似することによって、情報の圧縮を行うことを検討するものとしよう。

#### 3.2 近似

あるブロックを構成するデータ点を  $P_i: (x_i, y_i)$  ( $i=1, 2, \dots, n$ ) で、標本点を  $S_j: (x_j, y_j)$  ( $j=1, 2, \dots,$

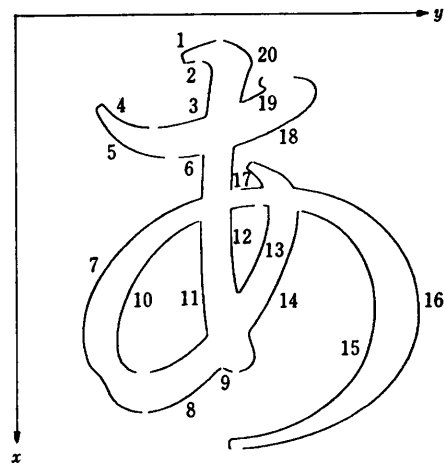


図1 文字輪郭線のブロック分け  
Fig. 1 Decomposition of character outline.

m) で書き表すものとする。標本点  $S_j$  は与えられたデータ点  $S_i$  のいずれかであり、ブロックの始点と終点とはつねに標本点となる。

各ブロックのいずれの端から標本点を取り始めるかによって、一般には得られる標本点は異なるが、ここでは与えられたブロックの一方の端 ( $x$  座標の小さいほう) から順次標本点を取っていくものとする。

### (1) 直線近似

最初の標本点  $S_1$  はブロックの最初のデータ点  $P_1$  とする。第 2 番目以後の標本点は以下のように決める。標本点  $S_{j-1}$  まで決まっているとき、次の標本点  $S_j$  は下の二つの条件を満たすように選ばれる。

- i) 標本点  $S_{j-1}$  と  $S_j$  とを結ぶ直線の許容誤差限界内に  $S_{j-1}$  と  $S_j$  との間のデータ点はすべて収まる。
- ii) 標本点  $S_j$  は前の標本点  $S_{j-1}$  と最も離れている。

### (2) 円弧・直線近似 (先の点非考慮)

ブロックの始めの部分では、始点を含む三つの標本点  $S_1, S_2, S_3$  を  $x$  座標が等間隔となるようにとり、これらの三つの点を通る円弧で点  $S_1$  と  $S_3$  との間のデータ点を近似する。三つの標本点  $S_1, S_2, S_3$  は次の二つの条件を満たすように選ばれる。

- i) 三つの標本点  $S_1, S_2, S_3$  で決まる円弧の許容誤差限界内に、標本点  $S_1$  と  $S_3$  との間のデータ点はすべて収まる。
- ii) 標本点  $S_3$  は  $S_1$  と最も離れている。

第 4 番目以後の標本点の決め方は以下のとおりである。標本点  $S_{j-2}, S_{j-1}$  がすでに決まっているものとし、新たに標本点  $S_j$  を選んだとする。これらの三つの標本点を通る円弧で標本点  $S_{j-1}$  と  $S_j$  との間のデータ点を近似するものとする。このとき、新しい標本点  $S_j$  は次の二つの条件を満たすように選ばれる。

- i) 三つの標本点  $S_{j-2}, S_{j-1}, S_j$  で決まる円弧の許容誤差限界内に点  $S_{j-1}$  と  $S_j$  との間のデータ点がすべて収まる。
- ii) 標本点  $S_j$  は  $S_{j-1}$  と最も離れている。

ただし、ブロック内の最初あるいは途中で、三つの標本点が一直線上に並ぶ場合には、直線で近似するものとする。

### (3) 円弧・直線近似 (先の点考慮)

文字輪郭線を構成するあるブロックにおいて、各標本点の間を結ぶ近似曲線は、おのおの独立なものではなく、一つの連続した文字輪郭線の一部である。したがって、各標本点間を結ぶ近似曲線を決めるとき、そ

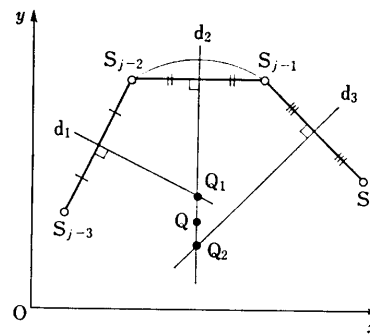


図 2 円弧・直線近似 (先の点を考慮する場合)  
Fig. 2 Curve-fitting with circular arcs and straight lines (considering a farther point).

の前後の標本点をも考慮することが望ましい<sup>7)</sup>。そこで、近似曲線を当てはめようとしている標本点間のさらに先にもう一つ標本点を選び、これを考慮して近似曲線を決めることにしよう。

ブロックの始めの部分における標本点の取り方は、先の(2)の場合と同様である。ただし、三つの標本点  $S_1, S_2, S_3$  が一直線上に並んでいるときには直線で近似するものとする。

第 4 番目以後の標本点のとり方は以下のとおりである。図 2 に示すように標本点  $S_{j-3}, S_{j-2}, S_{j-1}$  がすでに取り出されているものとし、新たに標本点  $S_j$  を選んだとする。線分  $\overline{S_{j-3}S_{j-2}}, \overline{S_{j-2}S_{j-1}}, \overline{S_{j-1}S_j}$  の垂直二等分線をそれぞれ  $d_1, d_2, d_3$  とし、 $d_1$  と  $d_2$  との交点を  $Q_1$ 、 $d_2$  と  $d_3$  との交点を  $Q_2$  とし、さらに  $Q_1$  と  $Q_2$  との中点を  $Q$  とする。そして、点  $Q$  を中心とする標本点  $S_{j-2}, S_{j-1}$  の 2 点を通る円弧で、 $S_{j-2}$  と  $S_{j-1}$  との間のデータ点を近似するものとする。このとき新しい標本点  $S_j$  は次の二つの条件を満たすように選ばれる。

- i) 上のようにして決めた円弧の許容誤差限界内に点  $S_{j-2}$  と  $S_{j-1}$  との間のデータ点がすべて収まる。
- ii) 標本点  $S_j$  は  $S_{j-1}$  と最も離れている。

ブロックの最後の部分では、点  $Q_2$  を中心とする点  $S_{j-1}$  と  $S_j (=S_m=P_m)$  とを通る円弧で近似する。

上には円弧が決まる場合についてのみ述べてきた。先に述べた交点  $Q_1$  あるいは  $Q_2$  が存在しない場合、および存在しても線分  $\overline{S_{j-2}S_{j-1}}$  に対して交点  $Q_1$  と  $Q_2$  とが互いに反対側にある場合には、標本点  $S_{j-2}$  と  $S_{j-1}$  との間を直線で近似するものとする。

ところで、標本点  $S_{j-1}$  は標本点  $S_{j-3}$  と  $S_{j-2}$  との間のデータ点を許容誤差限界内で近似するように選ばれているので、 $S_{j-1}$  が決まる段階では標本点  $S_{j-2}$

と  $S_{j-1}$  との間のデータ点はまったく考慮されていない。したがって、標本点  $S_j$  を可能な範囲内のどこに選んでも  $S_{j-2}$  と  $S_{j-1}$  との間のデータ点を許容誤差限界内で近似できないことが発生する可能性がある。そこで、このような場合には標本点  $S_{j-1}$  を始点に近いほうに選びなおして、標本点  $S_{j-3}$  と  $S_{j-2}$  との間への近似曲線の当てはめを再び行い、標本点  $S_j$  が選べるようになるまで  $S_{j-1}$  を始点の方へ近づける。この操作があるので先の点を考慮する円弧・直線近似では、標本点の選定に時間がかかることになる。

3.3 復元

直線近似の場合には、ブロックごとに各標本点を直線で繋いで近似輪郭線を得る。

先の点を考慮しない円弧・直線近似の場合には、各ブロックごとに始点のほうから三つずつ重なりがあるように標本点  $S_j, S_{j+1}, S_{j+2}$  ( $j=1, 2, \dots, m-2$ ) をとり、これらの三つの点で決まる円弧によって  $S_{j+1}$  と  $S_{j+2}$  との間を繋ぐ。ただし、ブロックの始めの部分では3点  $S_1, S_2, S_3$  を通る円弧で、標本点  $S_1$  と  $S_3$  との間を繋ぐ。ここで、もし三つの標本点が一直線上に並んでいるときには、標本点  $S_{j+1}$  と  $S_{j+2}$  との間を直線で繋ぐものとする。

先の点を考慮する円弧・直線近似の場合には、以下の手順で近似輪郭線を得る。

ブロックの始めの部分では、標本点  $S_1, S_2, S_3$  を通る円弧あるいは直線で  $S_1$  と  $S_3$  との間を繋ぐ。

ブロックの途中では、四つの標本点  $S_j, S_{j+1}, S_{j+2}, S_{j+3}$  ( $j=2, 3, \dots, m-3$ ) をとり、次のようにして  $S_{j+1}$  と  $S_{j+2}$  との間を円弧あるいは直線で繋ぐ。三つの線分  $\overline{S_j S_{j+1}}, \overline{S_{j+1} S_{j+2}}, \overline{S_{j+2} S_{j+3}}$  のそれぞれの垂直二等分線  $d_1, d_2, d_3$  から、 $d_1$  と  $d_2$  との交点  $Q_1$ ,  $d_2$  と  $d_3$  との交点  $Q_2$  を求め、さらに交点  $Q_1, Q_2$  の中点  $Q$  を求める。そして、標本点  $S_{j+1}, S_{j+2}$  を通る点  $Q$  を中心とする円弧で  $S_{j+1}$  と  $S_{j+2}$  との間を繋ぐ。ここで、交点  $Q_1, Q_2$  が線分  $\overline{S_{j+1} S_{j+2}}$  に関して反対側にある場合、 $Q_1$  あるいは  $Q_2$  が定まらない場合には、標本点  $S_{j+1}$  と  $S_{j+2}$  との間を直線で繋ぐ。

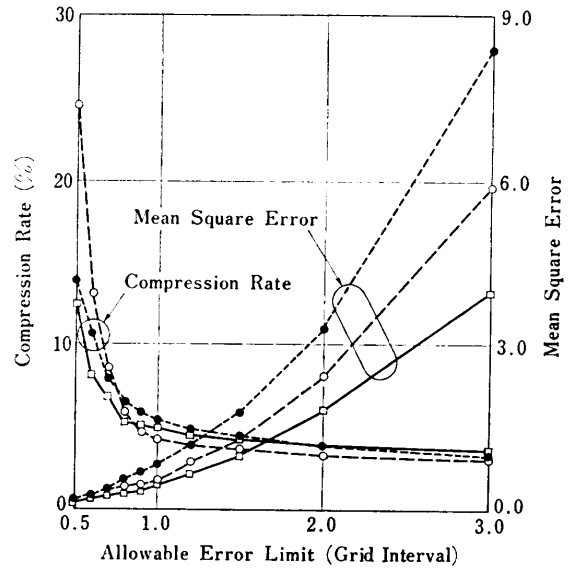
ブロックの最後の部分では、三つの標本点  $S_{m-2}, S_{m-1}, S_m$  によって決まる円弧あるいは直線で  $S_{m-1}$  と  $S_m$  との間を繋ぐ。

4. データの圧縮例とその吟味

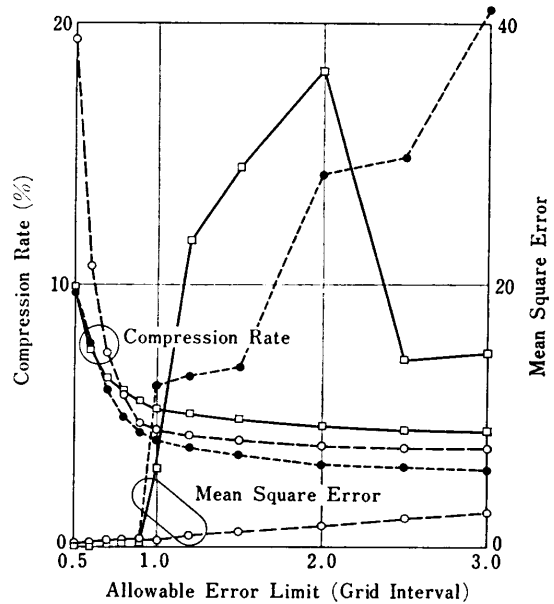
4.1 圧縮例

ここでは碁盤目状の領域に表現された平仮名「あ」

とゴシック体の漢字「超」との輪郭線データの圧縮について述べる。「あ」を例として取り上げたのは、この文字が種々の傾斜の曲線部からなっており、データの圧縮効果を確認するのに適当であると考えられるからである。また、ゴシック体の「超」は直線と曲線と



(a) 「あ」 800×800



(b) 「超」 800×800

図3 圧縮例

Fig. 3 Examples of data compression. (a) The first Japanese alphabet “あ”, and (b) a gothic Chinese character “超” represented on a matrix of 800×800.

●: fitting with straight lines, ○: fitting with circular arcs, □: fitting with circular arcs (considering a farther point)

が混在する文字であることから例として取り上げた。与えられた文字データは文字のデザイナーにより修正を施されたデジタル文字パターンである。碁盤目状の領域(800×800)に表現された文字「あ」の文字輪郭線は、先の図1に示したように20個のブロックに分けられていて、データ点の総数は3,260である。また「超」の輪郭線のデータ点総数は3,574である。

図3(a), (b)は許容誤差限界 $w$ を変化させたときの圧縮率と平均2乗誤差との変化である。圧縮率 $C$ は、

$$C \equiv \frac{\text{標本点数}}{\text{元のデータ点総数}} \times 100(\%) \quad (8)$$

で定義される。平均2乗誤差 $\epsilon^2$ は、元のデータ点総数を $N$ 、第 $i$ 番目の元のデータ点と復元データ点との $y$ 座標をそれぞれ $y_i, Y_i$ とするとき

$$\epsilon^2 \equiv (1/N) \sum_{i=1}^N (y_i - Y_i)^2 \quad (9)$$

で定義される。

図において点線は直線近似、破線は先の点を考慮しない円弧・直線近似、実線は先の点を考慮する円弧・直線近似の場合の結果を示している。

圧縮率は、直線近似の場合も円弧・直線近似の場合も、許容誤差限界が0.8格子間隔よりも小さくなると急激に大きくなる。また、許容誤差限界が1格子間隔よりも大きくなると、いずれの近似法においても圧縮率はほとんど変化しなくなる。このような関係は文字の種類にほとんど依存しないことも図3(a), (b)よりわかる。ここにはデータとしては示さないが、文字パターンが表現されている碁盤目の粗さにかかわらず、許容誤差限界と圧縮率との間には上に述べた関係が成立している。

種々の傾斜をもつ曲線から構成されている文字「あ」の場合には、いずれの近似法でも、平均2乗誤差は許容誤差限界のほぼ2乗に比例して増大する。ゴシック体の漢字「超」では、直線近似の場合を除いてこのような関係はない。これは水平( $y$ 軸方向)に近い直線部が存在しているため、与えられた曲線の法線方向の許容誤差限界に比して水平( $y$ )方向の誤差が極端に大きくなってしまったことによる。このことから、水平( $y$ )方向に近い傾斜をもつ曲線が多く存在する文字輪郭線の場合には、計算の容易な平均2乗誤差は人間の感覚とかなり異なる大きな値となってしまうことに注意する必要がある。

許容誤差限界に対する圧縮率と平均2乗誤差との変

表1 許容誤差限界が1格子間隔の場合の文字「あ」の圧縮率  
Table 1 Compression rate of character "あ", when allowable error limit=1.0 (grid interval). (unit: %)

Matrix size	800×800	400×400	200×200
Straight line	5.83	8.15	12.35
Straight line/ circular arc			
Not considering farther point	4.75	6.62	12.10
Considering farther point	5.00	8.09	12.96

表2 許容誤差限界 $w=1.0$ (格子間隔)の場合の圧縮時間と復元時間(「あ」800×800)  
Table 2 Processing time when allowable error limit  $w=1.0$  (grid interval) ("あ" 800×800)

Type of approximation	Compression time (sec)	Reproduction time (msec)
Straight line	4.48	353
Straight line/ circular arc		
Not considering farther point	8.23	601
Considering farther point	111.54	681

化から、いずれの近似法においても、許容誤差限界は0.8~1.0格子間隔が適切であるといえる。許容誤差限界が1.0格子間隔の場合の各近似法による復元図形の例を付録に示す。

碁盤目の粗さが変わったとき、これに伴って圧縮率が如何に変化するかを文字「あ」についてみてみよう。表1は許容誤差限界を1.0格子間隔としたときの800×800, 400×400, 200×200の粗さにおける圧縮率を示す。碁盤目が粗くなるに従って、いずれの近似法においても圧縮率は大きくなっている。碁盤目が粗くなると各要素の占める面積一重要さ一が、細かい場合に比べて相対的に大きくなるので、これは当然の結果である。

各近似法における圧縮時間と復元時間とを、許容誤差限界が1.0格子間隔の場合について表2に示す。模擬実験に使用した計算機はMELCOM COSMO 700 IIで、プログラムはFORTRANで書かれている。圧縮時間には、1語に収められている $x, y$ 座標値の分割処理時間、および標本点の $x, y$ 座標値を1語に合成するための処理時間も含まれている。

この表を見ると、圧縮時間は直線近似の場合が最も短く、先の点を考慮する円弧・直線近似の場合が最も

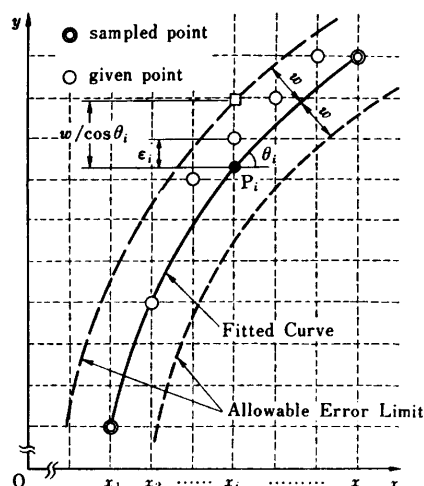


図4 近似曲線の決め方

Fig. 4 A fitted curve that determined by given data points.

長く、直線近似の場合の約60倍であることがわかる。先の点を考慮する円弧・直線近似の場合、標本点の再選定を行っているため処理に時間を要するものと思われる。また、直線近似の場合の復元時間は円弧・直線近似の場合の約1/2であることもわかる。

#### 4.2 許容誤差限界と平均2乗誤差との関係

標本点  $S_j$  ( $j=1, 2, \dots, m$ ) が与えられたとき、標本点間一区間は直線あるいは円弧で繋がる。近似曲線は最終的には離散値に変換されるが、ここでは簡単のためにアナログ値をとる曲線のまま取り扱うものとする。図4に示すような  $\nu$  個のデータ点より成る区間を考えよう。図において白丸はデータ点、2重丸は標本点、実線は近似曲線、破線は許容誤差限界  $w$  の線である。

横軸  $x=x_i$  における近似曲線上の点  $P_i$  (近似値) と与えられたデータ点との  $y$  方向の誤差  $\epsilon_i$  は、平均値0、分散  $\sigma_i^2$  の正規分布しているものと仮定すれば、その確率密度関数  $p(\epsilon_i)$  は

$$p(\epsilon_i) = (1/\sqrt{2\pi}\sigma_i) \exp(-\epsilon_i^2/2\sigma_i^2) \quad (10)$$

で与えられる。

いま、点  $P_i$  における近似曲線の接線と  $x$  軸とのなす角を  $\theta_i$  とすれば、この点における  $y$  方向の最大許容誤差は  $w/\cos\theta_i$  となる。ところで、近似曲線はその許容誤差限界内に、与えられたデータ点がすべて収まるように決められていること、ならびに、誤差は正規分布であると仮定しているため、 $\pm 3\sigma_i$  のなかに  $y$  方向誤差の 99.73% が入ることから

$$3\sigma_i \approx w/\cos\theta_i \quad (11)$$

とおくものとする。

さて、点  $P_i$  における  $y$  方向の2乗誤差の期待値  $\overline{\epsilon_i^2}$  は、式(11)を考慮すれば

$$\overline{\epsilon_i^2} \equiv \int_{-\infty}^{\infty} \epsilon_i^2 p(\epsilon_i) d\epsilon_i = \frac{w^2}{9 \cos^2 \theta_i} \quad (12)$$

となる。すなわち、ある点  $P_i$  における2乗誤差は許容誤差限界  $w$  の2乗に比例し、その点における接線が  $x$  軸と成す角  $\theta_i$  の余弦の2乗に反比例する。したがって、ある区間における平均2乗誤差  $\overline{\epsilon s^2}$  は、式(12)から

$$\overline{\epsilon s^2} \equiv \frac{1}{\nu} \sum_{i=1}^{\nu} \overline{\epsilon_i^2} = \frac{w^2}{9\nu} \sum_{i=1}^{\nu} \frac{1}{\cos^2 \theta_i} \quad (13)$$

となる。近似曲線は決まっているので各  $\theta_i$  も確定している。また、区間内のデータ点数  $\nu$  も決まっているので、平均2乗誤差  $\overline{\epsilon s^2}$  は許容誤差限界  $w$  の2乗に比例していることがわかる。このことから、与えられたデータ点と近似曲線との近さの評価を、平均2乗誤差を用いて行うことの妥当性が結論される。また、近似曲線の一部が  $y$  軸にほぼ平行、すなわち  $\theta_i$  が  $\pi/2$  に近い場合には、 $1/\cos\theta_i$  は非常に大きな値となり、その部分の誤差が平均2乗誤差の値に大きく寄与してしまうことも式(13)からわかる。

#### 5. あとがき

円弧と直線とによってデジタル文字の輪郭線を近似する方法について述べた。与えられた文字輪郭線の法線方向の許容誤差限界を1格子間隔以下にとれば、元の輪郭線を円弧あるいは直線によって十分な精度で近似することができる。800×800の基盤目状領域に表現されている文字「あ」の場合、許容誤差限界が1格子間隔のときの標本点数は、元の文字輪郭線のデータ点総数3,260の約5% (4.23~5.40%) である。直線近似、円弧・直線近似とも十分なデータ圧縮機能もっている。

上に述べた三つの近似法の特徴は次のとおりである。直線近似は圧縮、復元時間も短く効率が良いが、難点は復元図形の湾曲部において角ばりが目立つことである。先の点を考慮しない円弧・直線近似は圧縮、復元に多少時間を要するが、復元図形はほぼ満足できるものが得られる。先の点を考慮する円弧・直線近似の最大の難点は圧縮に大きな時間を要することであるが、元の図形に最も近い復元図形が得られる。

標本点を記憶しておいて文字パターンを生成する方

式では、輪郭線の生成を実時間で行わなければならないので、この処理時間が小さいことが要請されるが、圧縮は実時間で行う必要がないので圧縮に多少時間を要しても許容できると考えられる。先の点を考慮する円弧・直線近似は、元の文字パターンを忠実に復元したいときに有効である。

文字輪郭線を含む一般の線図形の場合には、元の線図形と近似図形との近さは、曲線の法線方向に測った量が人間の感覚との整合性がよい。与えられた任意の曲線から法線方向に一定の距離離れた線が満たすべき式(5)を2章において導いた。法線方向に一定距離離れた線の式は、与えられた曲線が初等関数の場合、直線あるいは円弧のときにしか陽の形には求められない。

近似によって生成された輪郭線の平均2乗誤差は、許容誤差限界の2乗に比例していることを4.2節で明らかにし、実験結果の裏づけを行った。法線方向の許容誤差限界と平均2乗誤差との間には2乗の関数関係があることから、種々の傾斜をもつ曲線の場合には、計算の容易な平均2乗誤差を線図形の近似度の評価に用いてもよいといえる。

輪郭線上の屈折点を抽出し、これを標本点とすれば近似精度が上がる可能性がある。また、直線あるいは円弧の代りに区分的多項式等を用いて既存のフォントの文字輪郭線を近似する方法の検討については今後を期したい。

謝辞 式(5)に関して有益な討論をしてくれた竹内寛君(昭和55年度卒研生・現在日本電気(株))、直線

近似の計算処理等を手伝ってくれた小柳樹枝官に感謝する。

### 参考文献

- 1) 日本工業標準調査会(日本規格協会): JIS C 6234, ドットプリンタ用24ドット字形(1983).
- 2) Knuth, D. E.: *TEX and METAFONT*, Digital Press, Bedford (1979).
- 3) Bigelow, C. A.: *The Principles of Digital Type, The Seybold Report on Publishing Systems*, Vol. 11, No. 11, pp. 4-22 (1982).
- 4) US Patent 905451, Character Generation Method and Apparatus (1978. 5. 15).
- 5) 佐藤: 平面曲線の最適折線近似, 信学論(D), Vol. J65-D, No. 9, pp. 1145-1150 (1982).
- 6) 名倉: 手書き線図形の直線と円弧による近似, 信学論(D), Vol. J64-D, No. 9, pp. 839-845 (1981).
- 7) 山崎, 飯島: 計算機による標準文字図形の発生, 情報処理, Vol. 12, No. 3, pp. 168-175 (1971).

### 付録 近似によって生成された文字図形の例

直線近似, 先の点を考慮しない場合と考慮する場合との円弧・直線近似によって発生された文字パターン「あ」を図5(a), (b), (c)にそれぞれ示す。許容誤差限界は1格子間隔である。先の点を考慮する円弧・直線近似によって発生された文字図形が最も癖のない美しい形をしている。文字のデザイナーによる評価は良い方から(c), (b), (a)の順であったことを付言しておく。

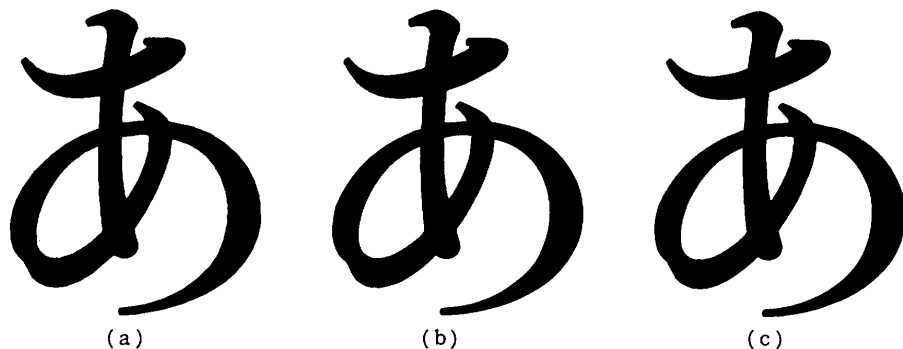


図5 復元図形の例。(a)直線近似,(b)先の点を考慮しない円弧・直線近似,(c)先の点を考慮する円弧・直線近似

Fig. 5 Example of character patterns generated by curve-fitting with (a) straight lines, (b) circular arcs and straight lines (not considering a farther point), and (c) circular arcs and straight lines (considering a farther point). Allowable error limit is 1.0 grid interval.

(昭和59年5月21日受付)

(昭和60年2月21日採録)