ＡＴＲテクニカルレポート表紙

〔非公開〕

ＴＲ－Ｃ－００４２

３次元形状の再構成手法について

西田　臣一　TAMIKAZU NISHIDA

ダニエル・リー　DANIEL LEE

岸野　文郎　FUMIO KISHINO

１９９０．２．２８

ＡＴＲ通信システム研究所

目　次

1．はじめに

本報告は、1988年3月から1990年2月までの2年間、（株）ATR通信システム研究所知能処理研究室において、3次元形状の再構成について行なった検討結果をまとめたものである。

近年、コンピュータ・グラフィックス，ソリッド・モデラー等の発展にともない、イメージされるさまざまな形状をより高度に表現できるようになってきた。また、計測技術の進歩により、自然物体の3次元形状を自動的に計測するさまざまな方法が提案されている。しかし、それらの計測手法の多くは、計測精度を上げるとデータ量が多くなるという問題を持っており、形状データから物体を再構成する最適な方法はまだ見つかっていない。
過去に、計測データをもとに立体形状をどう表現すればよいかを検討した例は少なく、Ｓｃｈｍｉｔｔ らにより提案されている、分割手法を用いた物体再構成の方法[1][2]は非常に良い方法だと思われるが、まだまだ多くの問題点を持っている。

そこで本研究では、複雑な物体の3次元形状をより精度良くデータ入力し、より少ないデータ量で形状の複雑さに適応した物体モデルを構築するため、光切断法で得られた形状データをもとに、特徴点を抽出して三角パッチで物体を再構成する方法と、subdivision の手法を利用して曲面パッチで物体を再構成する方法について検討した。
また、その結果を人間の目で確認するため、Ｗａｖｅ Ｆｒｏｎｔ，ｉｒｉｓ４Ｄ／８０ＧＴ（GT Graphics Library）を利用してシェーディング等を行ない、再構成された物体を評価した。

本文では、以上のテーマについて、3章に分けて検討結果を報告する。

まず、第2章では、光切断法の3次元形状計測原理と、円柱座標系（r, θ, z）を使用した3次元形状データの作成方法について述べる。このシステムを利用すると、円柱座標系のθ方向，z方向に対して、任意のサンプリング間隔で形状データを作成することができ、Ｗａｖｅ Ｆｒｏｎｔ を用いて表示することが可能である。

次に、第3章では、特徴のある部分を抽出して三角パッチを生成するという手法を用いて、少ないデータ数で立体形状を再構成する方法について述べる。
表面形状が滑らかに変化していて、ある部分で大きく形状が変化したと考えると、z軸に垂直な断面の面積も大きく変化すると考えられる。そこで、断面積が大きく変化する部分を取り出し、さらにその断面形状を近似することにより、3次元形状の近似表現が可能になる。

次に、第４章では、光切断法とＢｅｚｉｅｒ曲面の特徴をいかした階層的に分割を行なう Subdivision の手法を用いて、曲面パッチで立体形状を再構成する方法について述べる。

平面パッチによる物体の再構成は、単純な物体に対しては非常に良い結果が得られるが、複雑な物体に対しては、滑らかさがなくなりデータ数が多くなる。これを解決するため、Subdivision の手法を用いて曲面パッチで立体形状を再構成した。普通に Subdivision の手法を用いると、曲面間の接続が困難であるとか、分割する必要のない曲面を分割してしまうとか、厄介な問題が存在する。この問題を光切断法とＢｅｚｉｅｒ曲面の特徴をいかして解決した。

最後に、第５章では、全体のまとめと今後の課題について述べる。

２． 立体形状計測システム

２．１ 計測原理[3][4]

スリット光が回転テーブルの回転軸に投影されるようにスリット光発生装置を設置し、それと約３０度離れた位置にＴＶカメラを置く。その時の位置の補正は、計算機上でできるようになっている。（既知の点を数点入力し、パラメータを決定するという方法で行なう。）　そして、回転テーブルの中央付近に対象物体を置き、スリット光投影部分を抽出して、三角測量の原理でそれを３次元座標に変換する。（図２．１参照）

現在では、図２．２に示すように、物体表面のスリット光投影部分がカメラから観測できない部分（以後、死角と呼ぶ。）をできるだけ回避するため、スリット光に対して反対側にもう１台のカメラ（ＴＶカメラ２）を設置し、もとのカメラ（ＴＶカメラ１）からは死角のため得ることができなかったデータをＴＶカメラ２で補うことができる。

また、現時点での光切断法の分解能は、カメラの垂直成分に対しては１ピクセル、水平成分に対してはスリット光投影部分の濃淡画素値を利用して 1/4 ピクセルとなっている。

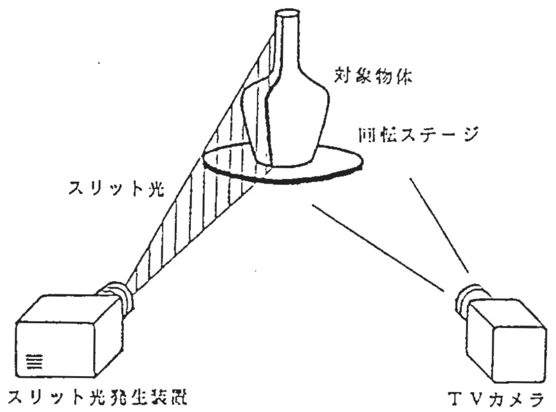


図２．１　計測原理

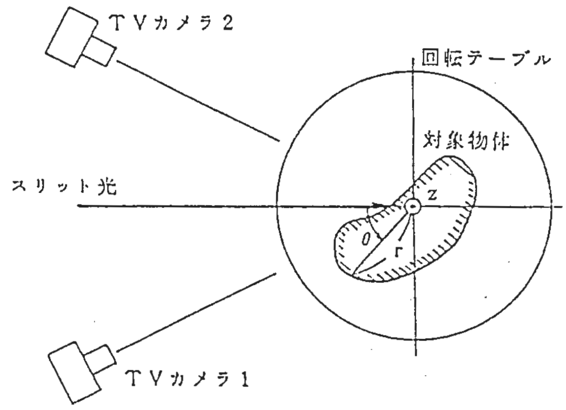


図２．２　形状入力システム

2．2　3次元形状データの作成

2．2．1　概要

光切断法から得られた3次元形状データにおけるサーフェイス・モデルの最適化を検討するために、円柱座標系（r，θ，z）のθ方向，z方向に対して、それぞれ任意のサンプリング間隔で頂点座標データを作成する。作成したデータは、.smp ファイルと呼ばれる中間ファイルにおとし、3，4章で述べる物体再構成システムの入力データとして使用する。

そして、この .smp ファイルから Wave Front フォーマットの .obj ファイルを作成する。この .obj ファイルを使用することにより、ワイヤーフレイム・モデル，シェーディング・モデル等の表示が可能となり、人間の目で物体を確認することができ、3次元形状データの簡単なチェックを行なうことができる。
あまり精度を要求しないモデリングに対しては、このシステムで充分である。

2．2．2　.smp ファイルの効果

.smp ファイルの効果を以下に示す。

(1) 本研究所の光切断法で作成された .coo ファイルと呼ばれる3次元画像データファイルと比較すると、データ量は 1/5 程度になる。
（同程度の精度を持つ .coo ファイルと .smp ファイルで比較した。）

(2) θとzを指定することにより頂点座標を求めることができ、座標データの取り扱いが容易となる。

(3) 頂点の位置関係が明確になり、.obj ファイルの作成が容易となる。

2．2．3　.smp ファイルの作成方法

.smp ファイルの作成手順を以下に示す。

但し、θ方向，z方向のサンプリング間隔をそれぞれ θs（度），zs（mm）とする。

(1) 1度（回転テーブルの回転角），1ピクセル（スリット光投影部を撮影するＴＶカメラの分解能，細線化処理を行なっているので高さ方向の分解能と一致する。）の精度で .coo ファイルを作成する。
この .coo ファイルのデータは、ＴＶカメラにより撮影されたスリット光投影部の各ピクセルの中心に対して、3次元座標（直交座標系を使用）を求めたものである。

(2) 図２．３に示すように、θ方向にθs（度）の間隔でデータを取り出し、高さ方向に最も小さい最上位部を見つけ、サンプリングスタート位置（z = 0）とする。

(3) 高さ方向に最も大きい最下位部を見つけ、サンプリングエンド位置とし、サンプリング領域を決定する。

(4) サンプリングスタート位置から下方にzs（mm）の間隔で、サンプリング領域内で頂点座標を求める。

・図２．３を見ればわかるように、サンプリング位置に必ずしも .coo ファイルの頂点が存在するとは限らない。存在しない場合は、高さ方向の上下２点の直線補間により頂点座標を求める。

・頂点座標は円柱座標系（r，θ，z）を使用する。

ここで、θ = i は i＊θs（度）を表わし、z = j はサンプリングスタート位置から j＊zs（mm）だけ下方の位置を表わす。（図２．４を参照）

※ ここで使用する円柱座標系はz軸が下方を向いていることに注意。

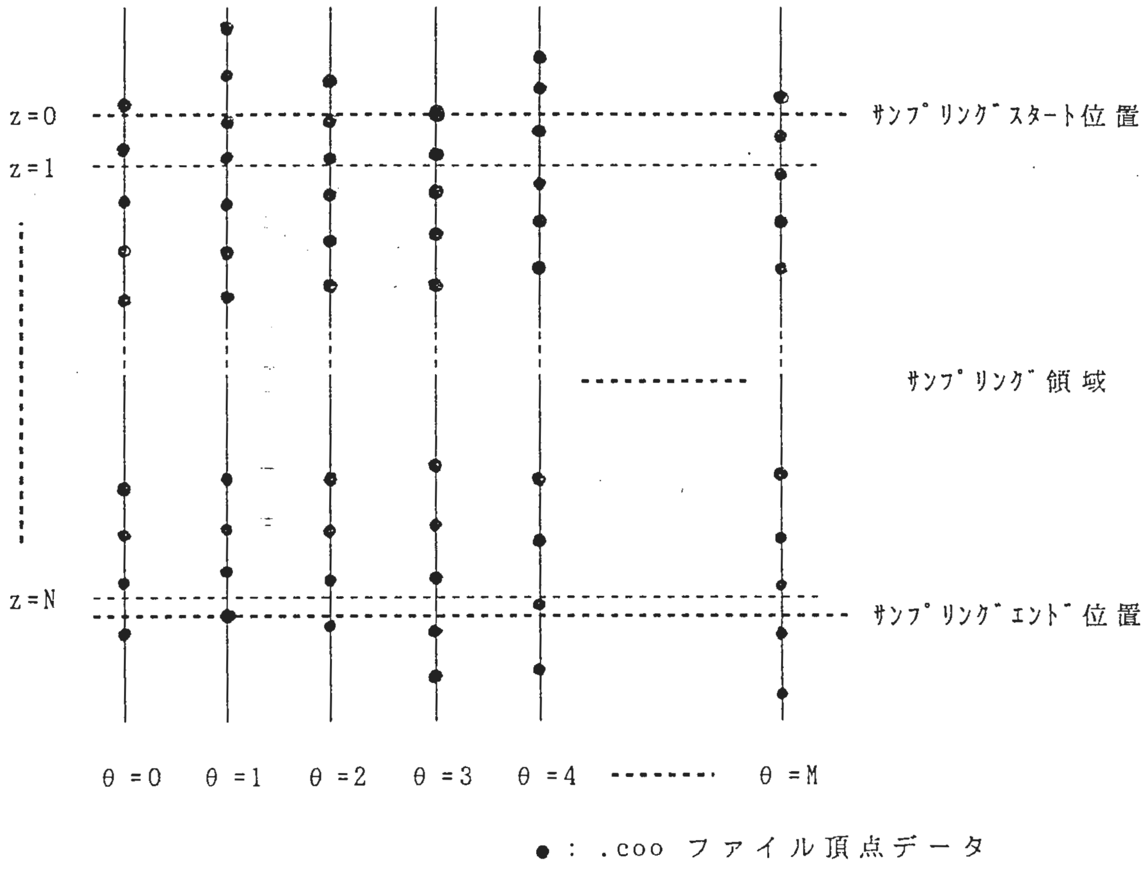


図２．３　.smp ファイルの作成状況

以上の手順により作成された頂点座標のデータ構造を図２．４に示す。

	θ(0)	θ(1)	---------	θ(i)	---------	θ(M)
z(0)	r(0,0)	r(1,0)	---------	r(i,0)	---------	r(M,0)
z(1)	r(0,1)	r(1,1)	---------	r(i,1)	---------	r(M,1)
⋮	⋮	⋮	---------	⋮	---------	⋮
z(j)	r(0,j)	r(1,j)	---------	r(i,j)	---------	r(M,j)
⋮	⋮	⋮	---------	⋮	---------	⋮
z(N)	r(0,N)	r(1,N)	---------	r(i,N)	---------	r(M,N)

r(a,b) ： θ =a,z=b のｒ成分

$M = 360 / \theta_s - 1$, $N = z_{max}$

図２．４　データ構造

２．２．４　.obj ファイルと表示例

.obj ファイルの作成手順の概要を以下に示す。

(1) 図２．５に示すように、隣り合ったθ間に三角パッチを生成する。

(2) 最上・下部の頂点の中心にそれぞれ新たに頂点を発生させ、最上・下部の頂点との間で蓋となる三角パッチを生成する。

(3) 生成された三角パッチを Wave Front のフォーマットに変換し、.obj ファイルを作成する。

以上の手順で作成した .obj ファイルに対してシェーディングを行なった例を図２．６に示す。
但し、この例のサンプリング間隔は、
θ方向 5度, z方向 5mm である。

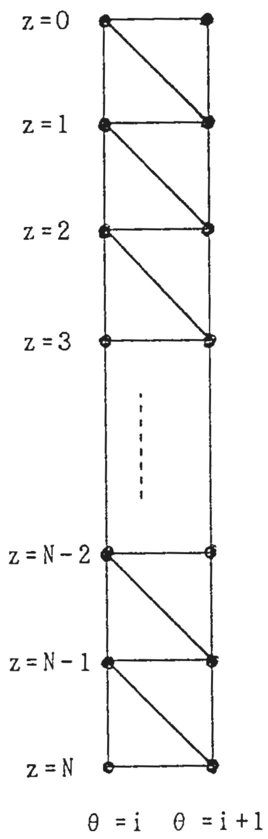


図２．５　三角パッチ

（a）正十二面体

（b）ウイスキーのボトル

（c）ビーナス

（d）小児立像

図２．６　シェーディングの例

３．断面積を利用した物体再構成[5]

３．１　概要

ｚ軸に垂直な断面の面積（断面積）をｚ軸方向に積分すると体積となる。ここで、体積誤差を物体再構成の評価基準に選ぶと、断面積を忠実に近似すればするほど体積誤差が小さくなり、よりよい物体再構成が行なえる。
また、表面形状が滑らかに変化していて、ある部分で大きく形状が変化したと考えると、断面積も大きく変化すると考えられる。そこで、断面積が大きく変化する部分を順次取り出すという方法で断面積を近似し、さらにその取り出された断面の形状を近似することにより、物体再構成を行なった。

３．２　アルゴリズム

アルゴリズムの手順を以下に示す。

(1) 光切断法からの情報をもとに、θ方向 1度，ｚ方向 1mm の間隔で頂点データを作成する。

(2) 図２．４に示す全てのｚについて断面積を求め、最大を１，最小を０として正規化を行なう。

(3) 図３．１に示すように、横軸にｚ，縦軸に正規化された断面積をとり、誤差パラメータ：ε_1［％］で区分的直線近似[6]を行なう。
ここで得られた点を分割点と呼ぶことにし、物体再構成を行なうために取り出された断面を意味する。

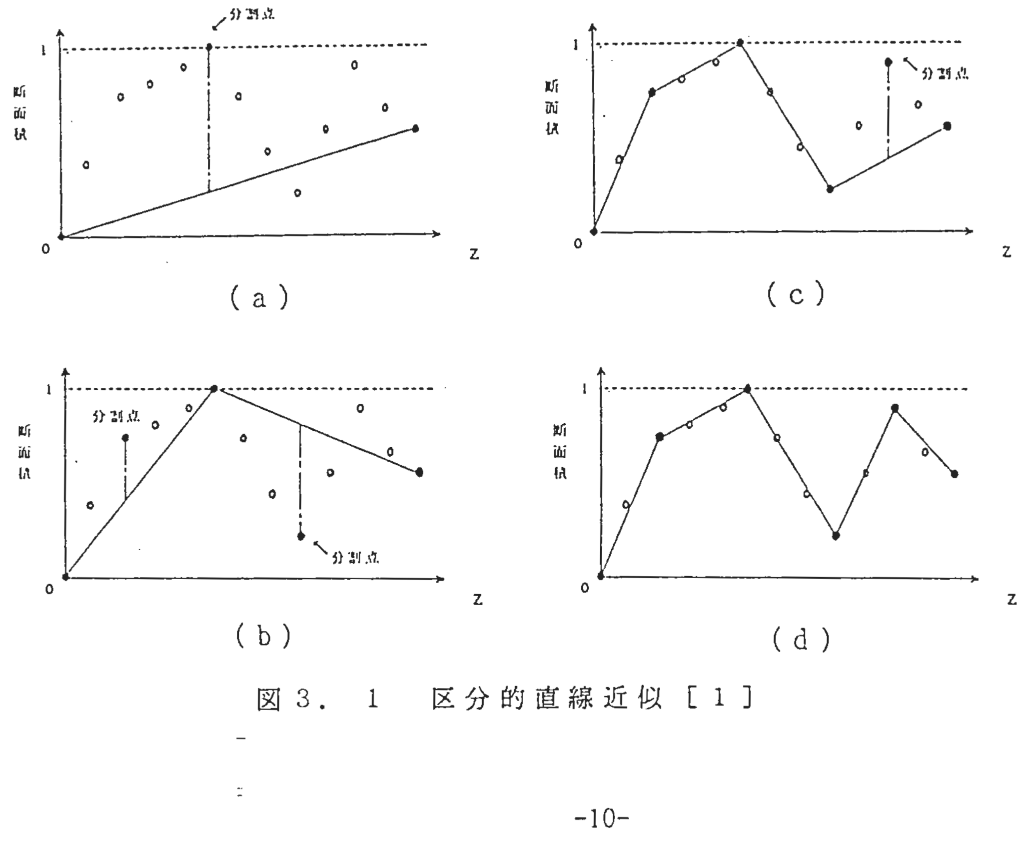


図３．１　区分的直線近似［１］

(4) (3)で取り出された断面を　$z=z_i$ ($i=1,2,\cdots$)　とすると、すべてのz_iについて断面を近似する。

図3．2に示すように、$z=z_i$ の平面上の点でr成分が最大値をとる点を見つけ、この平面を3分割する。そして、その各々の領域に対して、r成分の誤差が一定値以下になるように、誤差パラメータ：ε_2 [mm] で区分的直線近似を行なう。

但し、ここで得られた分割点は、(3) と区別するため節点と呼ぶことにする。

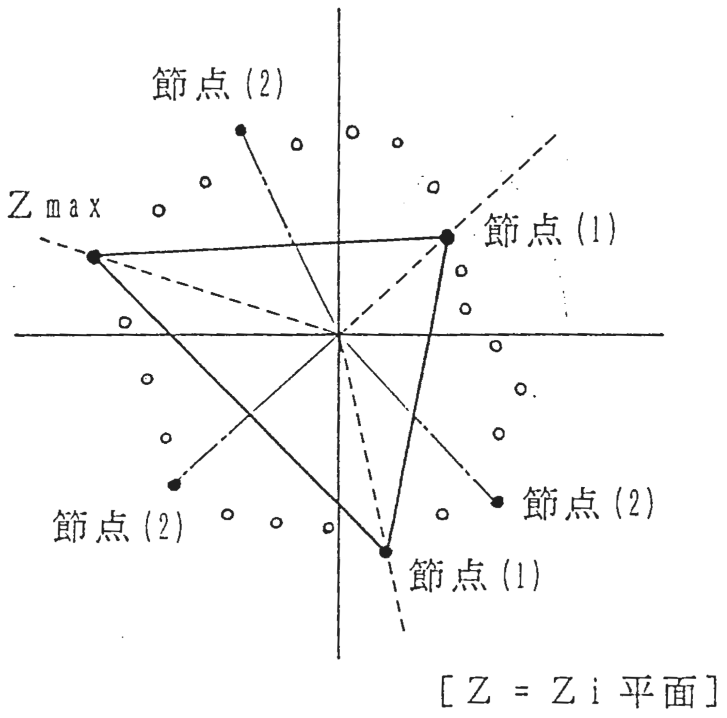


図3．2　区分的直線近似 [2]

(5) (4)で得られた節点を頂点とし、隣り合った z_i平面上の点に対して、$\theta-z$ 平面で近い点どうしを結ぶことにより三角パッチを生成する。

３．３　実験例

ウイスキーのボトルに対して、上記アルゴリズムを適応した結果を図３．３，図３．４，図３．５，図３．６に示す。

図３．３はアルゴリズム(2)の結果であり、図３．４は最終結果の頂点の個数を、横軸をε_1，パラメータをε_2として表わしたものである。

また、図３．５は、$\varepsilon_1 = 0.1$，$\varepsilon_2 = 1.0$の時のワイヤーフレイム・モデルとそのシェーディング結果であり、図３．６は、$\varepsilon_1 = 0.5$，$\varepsilon_2 = 3.0$の時のそれである。

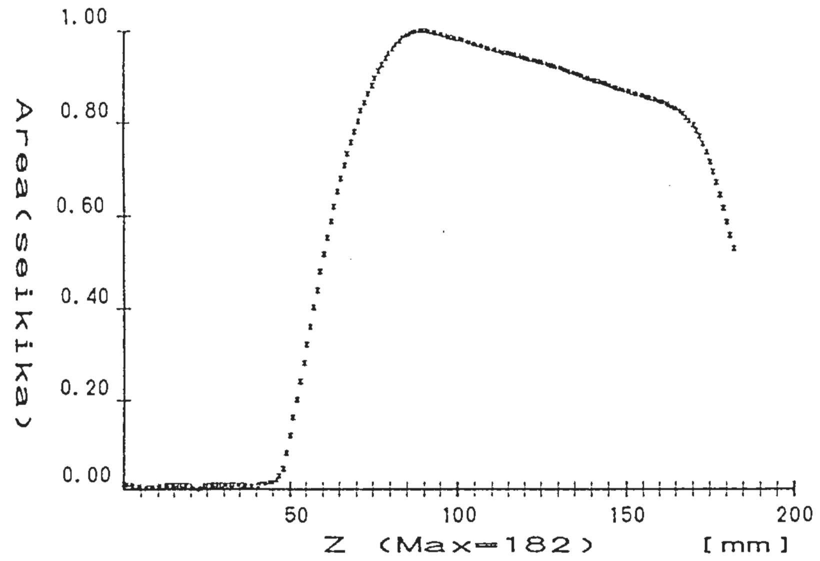


図３．３　正規化された断面積

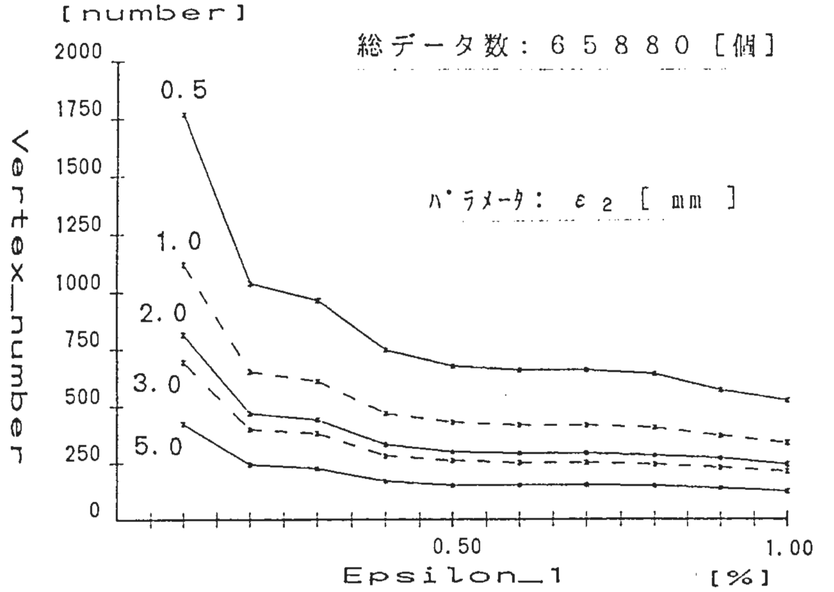


図３．４　頂点データの数

(a) ﾜｲﾔｰﾌﾚｲﾑ･ﾓﾃﾞﾙ

(b) ｼｪｰﾃﾞｨﾝｸﾞ

頂点の個数： 1123（個）
三角パッチの個数： 2242（個）

図3．5 結果例1（ε_1 = 0.1 %，ε_2 = 1.0 mm）

(a) ﾜｲﾔｰﾌﾚｲﾑ･ﾓﾃﾞﾙ

(b) ｼｪｰﾃﾞｨﾝｸﾞ

頂点の個数： 263（個）
三角パッチの個数： 522（個）

図3．6 結果例2（ε_1 = 0.5 %，ε_2 = 3.0 mm）

3．4　検討結果

上記の実験例に対して、体積誤差とデータ圧縮比について検討した結果を以下に示す。

3．4．1　体積誤差

$z = z_m,\ \theta = \theta_n$ の物体表面上の点を $SR_{m,n}$ 三角パッチ上の点を $PR_{m,n}$ とした時の体積誤差：E を次式で定義する。

$$E = \frac{\sin(1^\circ)}{2} \sum_{m=0}^{z_{max}-1} \sum_{n=0}^{359} \left| \frac{SR_{m,n} * SR_{m,n+1} + SR_{m+1,n} * SR_{m+1,n+1}}{2} - \frac{PR_{m,n} * PR_{m,n+1} + PR_{m+1,n} * PR_{m+1,n+1}}{2} \right| \quad \cdots ⓐ$$

但し、n=360 と n=0 は同じである。また、絶対値の中の前半部分は物体の正確な体積を表わし、後半部分は近似された多面体の体積を表わしている。

式ⓐで計算した体積誤差を百分率に換算し、結果を図3．7に示す。

但し、図中の体積（基準）は、上式の正確な体積を示している。

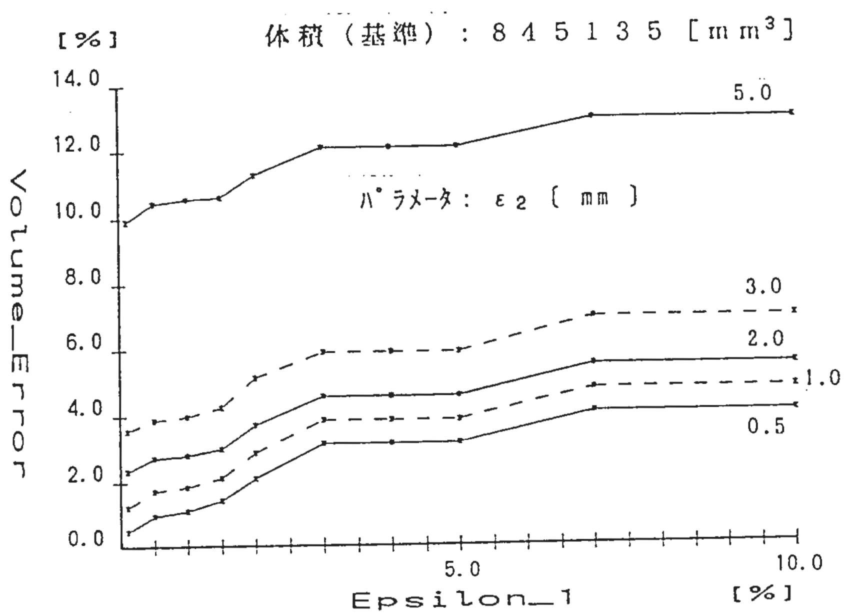


図3．7　体積誤差

ε_1を 0.1 から 1.0 まで変化させた場合、体積誤差に対するε_1の影響は 1% 以下であった。（図３．７には $\varepsilon_1 = 0.1, 0.5, 1.0$ の値しか示していないが、実際には、0.1 間隔のε_1でデータをとり確認した。）
これは、断面積をｚ軸方向に積分すると体積になると考えると、ε_1の体積に与える影響は最大でε_1(%) であり、予想と一致している。
しかし、ε_1を 0.1 から 10.0 まで変化させても、図３．７の結果は 3% 程度しか変化していない。これは、図３．３の断面積の変化状況を見ればわかるように、表面形状が大きく変化する所が３ヶ所（$z = 45, 80, 170$ 付近）あり、そこに分割点を発生させることができたら、ε_1の体積に与える影響が小さくなるためだと思われる。
また、ｒ成分の２乗平均の平方根を $R_{r.m.s}$ とし、各々のｒ成分にε_2の誤差を見込んだ時のそれを $R_{r.m.s} + \varepsilon_2$ だと仮定すると、ε_2の断面積に与える影響は最大で

$$\frac{(R_{r.m.s} + \varepsilon_2)^2 - R_{r.m.s}^2}{R_{r.m.s}^2} \quad \cdots ⓑ$$

となる。上と同様に考えると、これは最大体積誤差と考えることができる。
ここで、実測値より $R_{r.m.s}$ の値を計算すると、

$R_{r.m.s} = 38.394515$

となる。ちなみに、体積（基準）$= \pi \cdot R_a^2 \cdot z_{max}$ として R_a を計算すると、

$R_a = 38.44611$

となり、ほぼ $R_{r.m.s}$ の値と等しくなった。すなわち、これは $R_{r.m.s}$ の値を使って体積を論じても良いということを示唆している。
ε_1 が 1.0 以下の値に対して ε_1 による体積誤差が零だと仮定して、式ⓑ により計算した誤差の値と図３．７の結果を比較すると、図３．７の結果は計算値の 1/3 以下の値を示すという結果が得られた。

３．４．２　データ圧縮比

物体の存在するｚの範囲は $z = 0 \sim z = 182$ であるから最初のデータ数は 65880 (360 * 183) 個であり、データ圧縮比は図３．４の縦軸を 65880 で割った値となる。
ちなみに、$\varepsilon_1 = 0.5$, $\varepsilon_2 = 3.0$ （図３．６）の時のデータ圧縮比は

$(263 - 2) / 65880 = 1 / 252.4 \cdots$

となる。（上下に蓋を作成するため、新たに頂点を２点追加したので、それを差し引いて計算した。）

ここで、体積誤差，データ圧縮比，ε_1とε_2，の関係が明確になれば、希望の体積誤差，最終頂点個数とかデータ圧縮比を入力することにより、自動的に最適な ε_1とε_2 の組合せを選択し、３次元形状を再構成することが可能となる。

３．５　表示

ウイスキーのボトルとビーナスの石膏像に対して、前記アルゴリズムを使用して サーフェイス・モデル を作成し、フラット・シェーディング と スムース・シェーディング を行なった。その結果を図３．８と図３．９に示す。

ボトルのような形状に対しては、スムース・シェーディングが有効である。フラット・シェーディングではごつごつとしたボトルになるが、スムージングを行なうことによりボトルらしくなり、データ数を減らしてもボトルらしく見える。また、ビーナスのような形状に対しては、一様なスムージングはただのっぺりとした感じになるだけであまり有効ではなく、部分的にスムージングを行なうとかの方法を考える必要がある。

以上より、結論としては、複雑な形状をよりリアルに表示するには、疑似曲面化手法の能力には限界があり曲面の情報が必要である。

３．６　まとめ

任意の近似誤差で３次元形状を、少ないデータ数で再構成する方法について述べた。この方法は、ボトルのような形状に対しては大変有効的だといえる。また、ねじれているような物体に対しては、断面積を分割して本記アルゴリズムを適応することにより、ねじれが検出でき形状を再構成を再構成することが可能である。複雑な物体に対しては、曲面パッチによるモデルの最適化を考える必要がある。

（a） ﾌﾗｯﾄ・ｼｪｰﾃﾞｲﾝｸﾞ

（b） ｽﾑｰｽ・ｼｪｰﾃﾞｲﾝｸﾞ

ε_1 ＝ 0.1 （%），	ε_2 ＝ 0.5 （mm）
頂点数： 1771	パッチ数： 3538

（c） ﾌﾗｯﾄ・ｼｪｰﾃﾞｲﾝｸﾞ

（d） ｽﾑｰｽ・ｼｪｰﾃﾞｲﾝｸﾞ

ε_1 ＝ 0.5 （%），	ε_2 ＝ 3.0 （mm）
頂点数： 263	パッチ数： 522

図３． 8　ボトルの表示例

（a） ﾌﾗｯﾄ･ｼｪｰﾃﾞｨﾝｸﾞ

（b） ｽﾑｰｽ･ｼｪｰﾃﾞｨﾝｸﾞ

ε_1 = 0.1 (%) ,	ε_2 = 1.0 (mm)
頂点数： 7210	パッチ数： 14416

（c） ﾌﾗｯﾄ･ｼｪｰﾃﾞｨﾝｸﾞ

（d） ｽﾑｰｽ･ｼｪｰﾃﾞｨﾝｸﾞ

ε_1 = 0.3 (%) ,	ε_2 = 3.0 (mm)
頂点数： 1606	パッチ数： 3208

図３．９　ビーナスの表示例

４．Ｂｅｚｉｅｒ曲面を利用した物体再構成[7]

４．１　概要

３章で述べたように、平面パッチによる物体再構成は、単純な物体に対しては非常に良い結果が得られたが、複雑な物体に対しては、なめらかさがなくなりデータ数が多くなるという問題が生じた。

そこで、この問題を解決する方法として、光切断法で得られたデータから、光切断法とＢｅｚｉｅｒ曲面の特徴をいかした階層的に分割を行なう副分割手法を用いて、立体形状を再構成する手法を提案し、それについて検討した結果を報告する。

４．２　設定目標

物体の形状を損なわないために、細かい間隔で形状データを入力する。そして、任意の誤差：ε を設定し、すべての入力データに対して 誤差 $\leqq \varepsilon$ を満足するとともに、表面形状の複雑さに適応したサーフェイス・モデルを構築することを目標とする。

但し、ここでいう誤差とは、入力データと近似モデルの円柱座標系における半径方向を示すｒ成分の差であり、サーフェイス・モデルはＢｅｚｉｅｒ曲面で構築することとする。

また、構築された曲面のサーフェイス・モデルに対して、シェーディング方法を考える。

４．３　対象物体の設定

円柱座標系で物体表面の点を $S(r,\theta,z)$ とした時、Ｓが存在するすべての (θ,z) に対してＳが一意的に決まるように座標系を設定することができ、ＣＡＤ等で容易に作成することができないようなものを対象とする。具体的には、ビーナスの石膏像と小児立像を対象とする。

4．4　アルゴリズム

一般的に、n次のBezier曲線は任意の点で2つのn次Bezier曲線に分割することができるという特徴を利用して、Subdivision（副分割）手法を用いてBezier曲面を階層的に分割することにより物体の表面近似が可能になると考えた。そのアルゴリズムの要約を以下に示す。

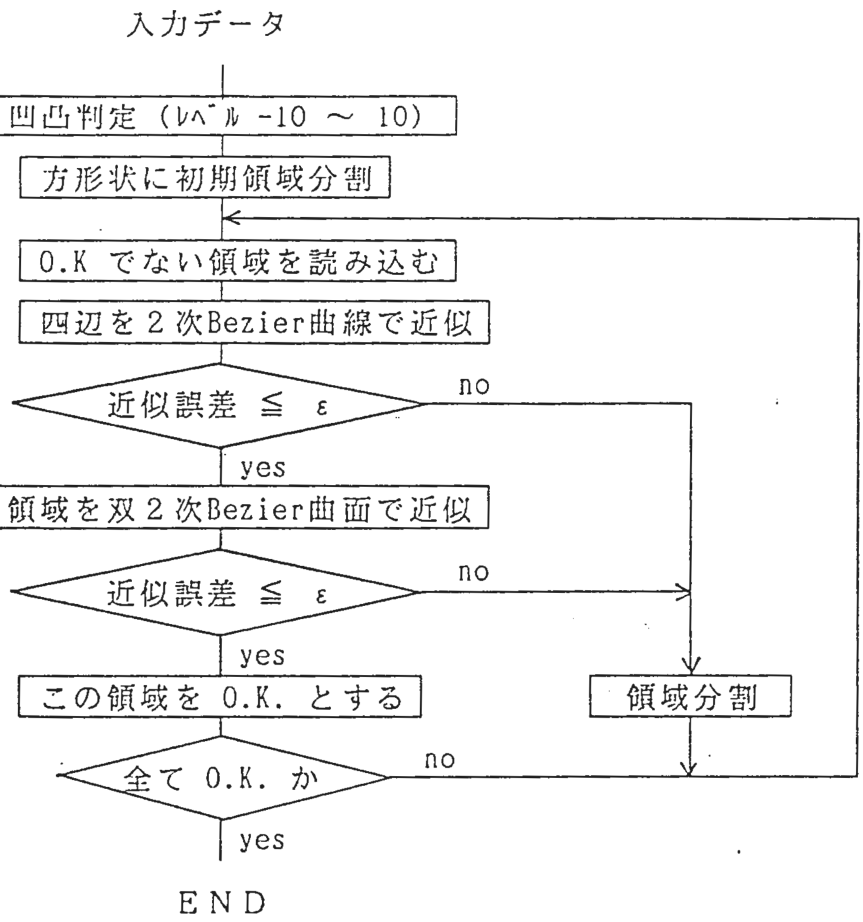


図4．1　アルゴリズム

［Bezier曲面を選択した理由］

・ｸﾞﾗﾌｨｯｸ・ﾜｰｸｽﾃｲｼｮﾝ でサポートされている。

・エッジ等の表現が容易に行なえる。

・取り扱いが容易である。

［副分割手法を選択した理由］

・Systematic に考えることができる。

・応用範囲が広い。

・アルゴリズムが簡単になる。

4．4．1 入力データ

入力システムとその誤差等を考慮して、θ方向 1度，z方向 1mm のサンプリング間隔で形状データを作成する。（図2．4のデータ構造を参照）

4．4．2 凹凸判定

初期領域分割を行なうため、入力データに対して凹凸情報を付加する。その手順を以下に示す。

(1) 孤立点を除去し、エッジ等の情報を得やすくするため、3*3 のメディアンフィルタを使ってフィルタリングの処理を行なう。

（但し、処理されたデータを使うのは、凹凸判定部だけである。）

(2) 図4．2に示すように、θ方向，z方向にそれぞれkだけ離れた点を見つけ、それを制御点として双2次Bezier曲面を作成し、その曲面上の中点（u = v = 0.5 ）を計算する。その値を rb(i,j) とすると、rb(i,j) は次式に u = v = 0.5 を代入することにより求めることができる。

$$rb(i,j) = S(0.5,0.5)$$

$$S(u,v) = [\ 1\ \ u\ \ u^2\ ]\, B\, P\, B^T [\ 1\ \ v\ \ v^2\ ]^T$$

但し、$B = \begin{bmatrix} 1 & 0 & 0 \\ -2 & 2 & 0 \\ 1 & -2 & 1 \end{bmatrix}$　　$P = \begin{bmatrix} r(i-k,j-k) & r(i-k,j) & r(i-k,j+k) \\ r(i,j-k) & r(i,j) & r(i,j+k) \\ r(i+k,j-k) & r(i+k,j) & r(i+k,j+k) \end{bmatrix}$

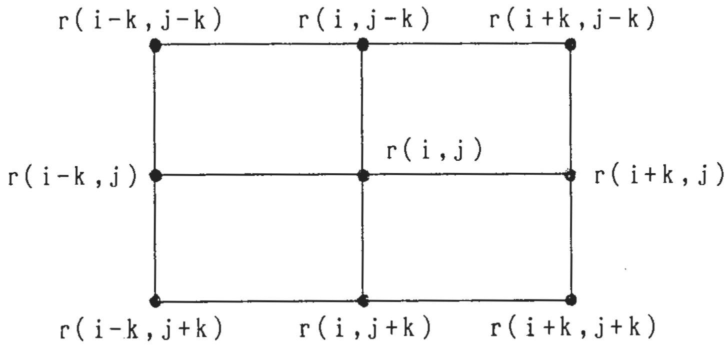


図4．2　制御点

(3) ｜r(i,j) − rb(i,j)｜が最大となるものを見つけその値を d_{max} とする。そして、(r(i,j) − rb(i,j)) / d_{max} を計算し、それを 10 倍して四捨五入した値をその点の凹凸情報とする。この値は凹凸の程度を表わし、その符号は +: 凸, −: 凹 を表わしている。

4．4．3　初期領域分割

上で求めた凹凸情報をもとに、初期領域分割を行なう。その手順を以下に示す。

(1) 凹凸情報の値の絶対値が 3 以上ならその点を凹凸点とする。

(2) 凹凸点に対して細らせ処理を行なう。
具体的には、凹凸点を見つけ、その周囲 8点がすべて凹凸点ならその点を凹凸点とする。

(3) (2)で求めた凹凸点に対して太らせ処理を行なう。
具体的には、凹凸点を見つけ、その周囲 8点をすべて凹凸点とする。

(4) (3)で求めた凹凸点に対して、4 連結の手法を用いてラベリングを行なう。
すなわち、θ 方向，z 方向で隣り合う凹凸点は同じラベルがつけられており、同ラベルの凹凸点の集まりを 1 つの凹凸領域とする。

(5) $\theta = i$ の点列がいくつの凹凸領域と交差するかをすべての i について求め、その情報をもとに θ 方向の初期領域分割を行なう。
（詳細は実験例のところで述べる。）

(6) $z = j$ の点列がいくつの凹凸領域と交差するかをすべての j について求め、その情報をもとに z 方向の初期領域分割を行なう。
（詳細は実験例のところで述べる。）

4．4．4　四辺の２次Ｂｅｚｉｅｒ曲線近似

各々の辺について、その端点を制御点として最小２乗法で２次Ｂｅｚｉｅｒ曲線近似を行なう。すなわち、辺を形成する点列を R(0),R(1),･･･,R(n) とし、のこる１つの制御点を RP とすると、点列の i 番目の点 R(i) における近似誤差：ED_i は

$$ED_i = a_i * RP + b_i - R(i)$$

a_i, b_i：定数（R(0),R(n),i により決まる。）

但し、$a_0 * RP + b_0 - R(0) = a_n * RP + b_n - R(n) = 0$ である。

となり、近似誤差の２乗の総和：ES は

$$ES = \sum_{i=0}^{n} \{a_i * RP + b_i - R(i)\}^2$$

となる。次に、$dES / dRP = 0$ となるように RP を決定し、点列のすべての点に対して ED_i を計算する。

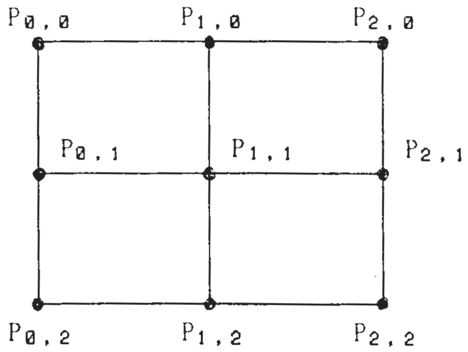
そして、$ED_i > \varepsilon$（設定値）となる点が１点でも存在すれば、その四辺で囲まれた領域を４分割する。

但し、θ 方向の辺を取り扱う時は、点列に対して角度補正を行なう必要がある。

4．4．5　領域の双２次Bezier曲面近似

領域に含まれる点列（頂点データ）に対する双２次Bezier曲面近似の概要を図4．3に示す。

$$
\begin{array}{ccccc}
R_{0,0} & R_{1,0} & R_{2,0} & \cdots & R_{m,0} \\
R_{0,1} & R_{1,1} & R_{2,1} & \cdots & R_{m,1} \\
R_{0,2} & R_{1,2} & R_{2,2} & \cdots & R_{m,2} \\
\vdots & \vdots & \vdots & & \vdots \\
R_{0,n} & R_{1,n} & R_{2,n} & \cdots & R_{m,n}
\end{array}
\quad \Rightarrow
$$



(a) 領域に含まれる点列　　(b) 双２次Bezier曲面

図4．3　近似の概要

すでに、四辺が２次Bezier曲線で近似されているので、$P_{1,1}$ 以外の制御点は決っており、$P_{1,1}$ を決定することにより、領域を双２次Bezier曲面で近似することができる。$P_{1,1}$ の決定方法を以下に示す。

点列の i,j 番目の点 $R_{i,j}$ における近似誤差：$ED_{i,j}$ は

$$ED_{i,j} = a_{i,j} * P_{1,1} + b_{i,j} - R_{i,j} \qquad a_{i,j}\ ,\ b_{i,j}\ :\ \text{定数}$$

となり、近似誤差の２乗の総和：ES_F は

$$ES_F = \sum_{i=0}^{m} \sum_{j=0}^{n} \{a_{i,j} * P_{1,1} + b_{i,j} - R_{i,j}\}^2$$

となるので、$d\,ES_F \,/\, d\,P_{1,1} = 0$ となるように $P_{1,1}$ を決定する。

そして、点列のすべての点に対して $ED_{i,j}$ を計算し、$ED_{i,j} > \varepsilon$（設定値）となる点が１点でも存在すればその領域を４分割する。

但し、点列に対しては角度補正を行なう必要がある。

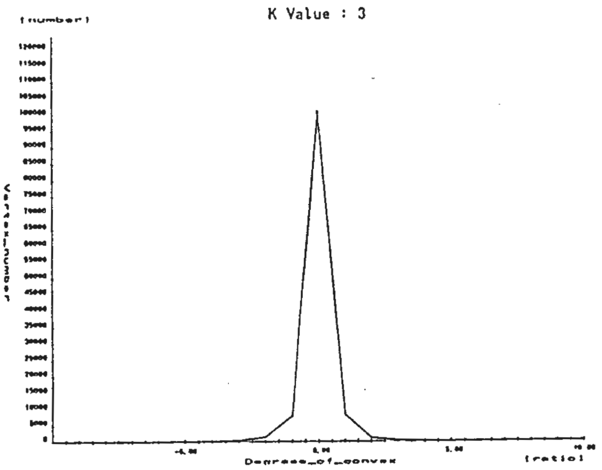
4．5　実験例

ビーナスの石膏像対して上記アルゴリズムを実行させた結果を以下に示す。

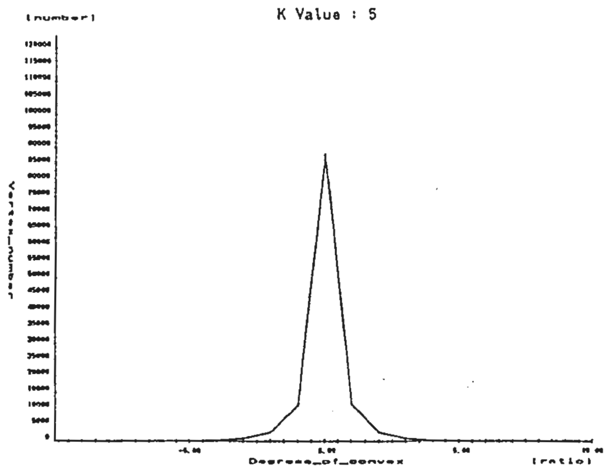
4．5．1　kの決定

アルゴリズムの説明の所（4.4.2（2））で述べたkの値を決定するため、kの値を変化させてデータを取った。そのうち、k＝3，5，7,10 の時の結果を図4．4に示す。図4．4で、横軸は凹凸情報，縦軸は頂点の個数を示している。
また、kの値を大きくすればするほど、全体の頂点の個数は減っているが、これはz軸方向の最上部と最下部で、凹凸を判定することができないからである。

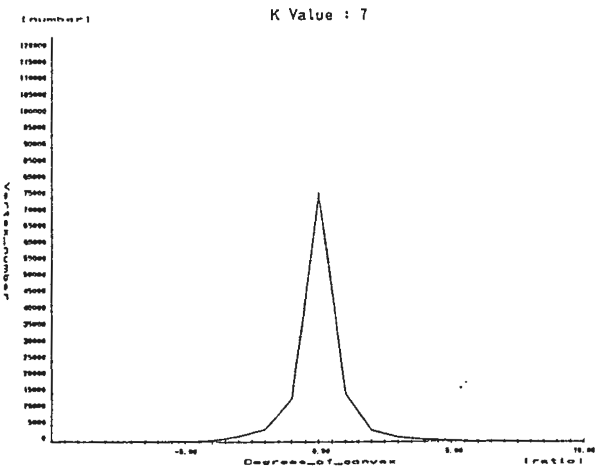
図4．4より、kの値を変えてもあまり結果は変わらず、kの値を大きくすればするほど凹凸の判定ができない部分が多くなることがわかる。
よって、ここではkの値を5とする。



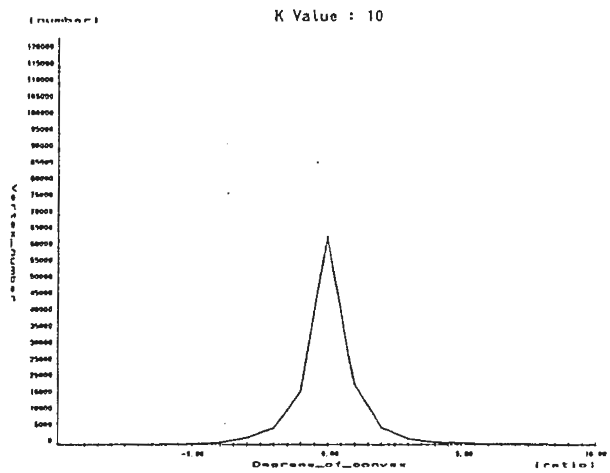
（a） k ＝ 3



（b） k ＝ 5



（c） k ＝ 7



（d） k ＝ 10

図4．4　凹凸情報に対する頂点の個数

4．5．2　初期領域分割

(1) θ方向の初期領域分割

θ＝ｉの頂点の列がいくつの凹凸領域と交差するかをすべてのｉについて求め、その結果を図4．5に示す。

図4．5より、以下の手順にもとずいてθ方向の初期領域分割を行なう。

ⓐ 1以上が連続する部分を１つのグループとする。
但し、θ=0（θ=360）をまたいで存在しても１つのグループとする。
また、グループの大きさは、それが存在するθの大きさとする。

ⓑ 一番小さいグループとそれに最も近いグループを１つにまとめる。
（グループ間をも含めて１つにまとめる。）

ⓒ １つのグループの大きさが３０度以上になるまでⓑを繰り返す。

ⓓ グループの中心（θの値）を求める。

ⓔ どのグループにも属していないものを見つけ、中心値が最も近いグループに属させる。

以上の手順により、θ方向を 46-141，141-237，237-312，312-46 と４つの領域に初期分割することができる。但し、領域の境界部は両方の領域に属することとする。

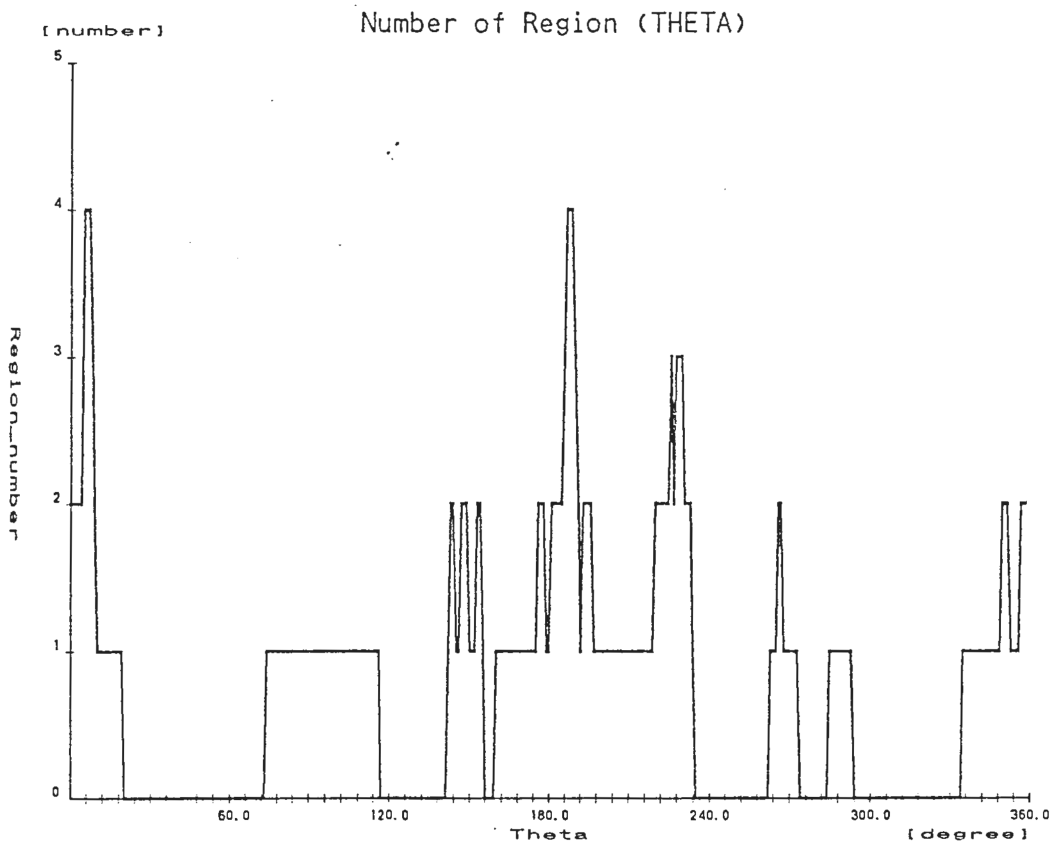


図4．5　凹凸領域の数（θ方向）

(2) z方向の初期領域分割

$z = j$ の頂点の列がいくつの凹凸領域と交差するかをすべての j について求め、その結果を図4．6に示す。

図4．6より、$z = 0$，$z = z_{max}$ があるグループの端だとして、(1)と同じ手順でz方向の初期領域分割を行なった。その結果、z方向を0-95，95-191，191-241，241-z_{max} と4つの領域に初期分割することができた。但し、領域の境界部は両方の領域に属することとする。

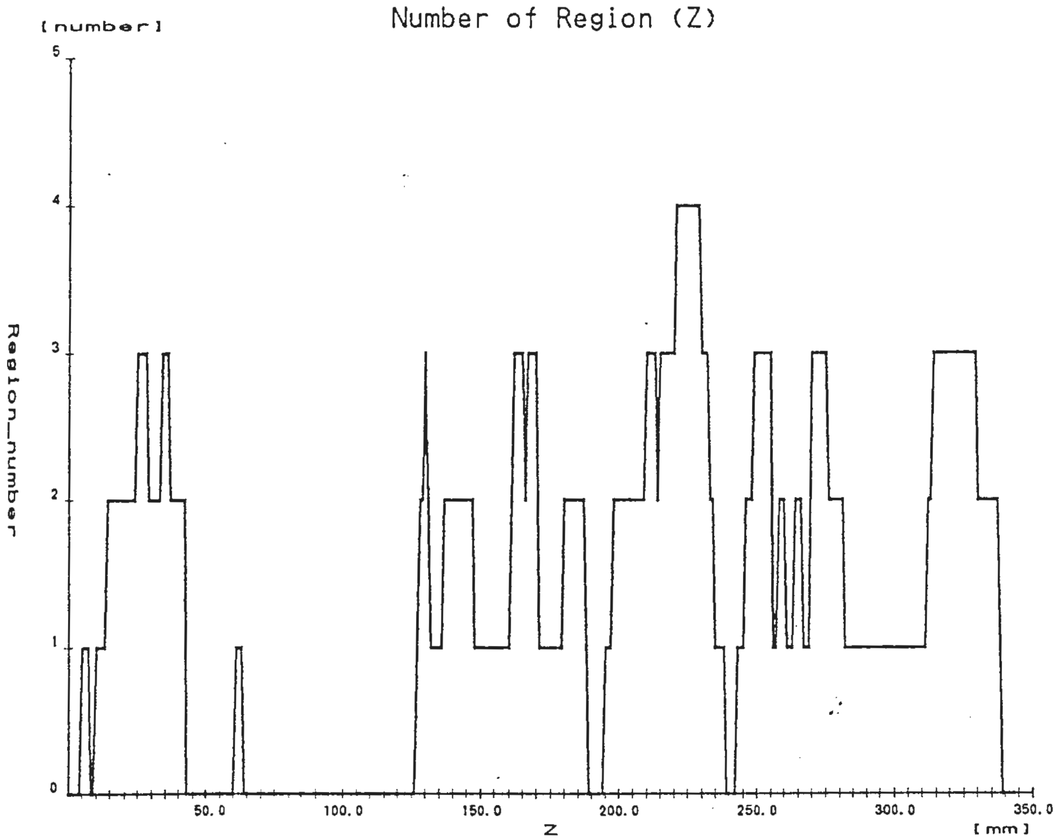


図4．6 凹凸領域の数（z方向）

(1)(2)の結果を図4．7に示す。

この図は、見やすくするためθ方向に -46 度回転させたものであり、緑色の線は領域の境界部を示し、赤色の部分は凹凸領域を示す。

この図より、凹凸の部分とフラットな部分をだいたい区別することができたといえる。

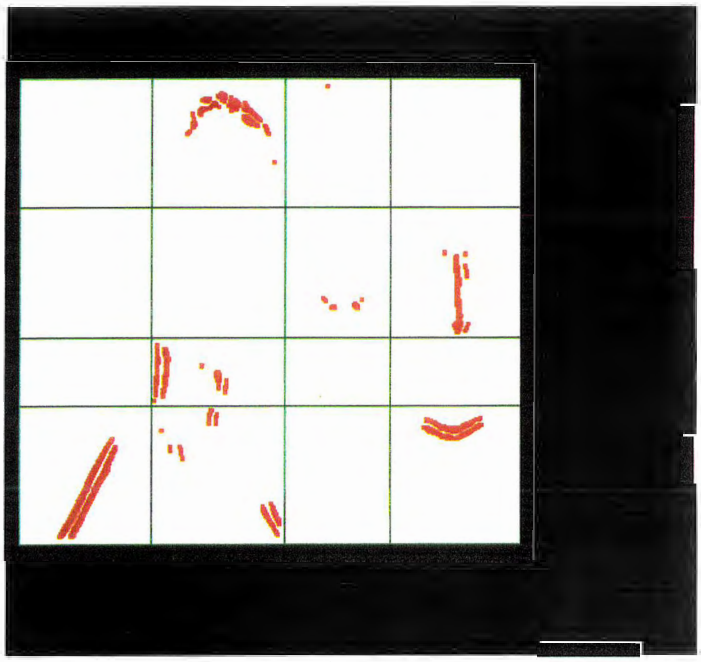


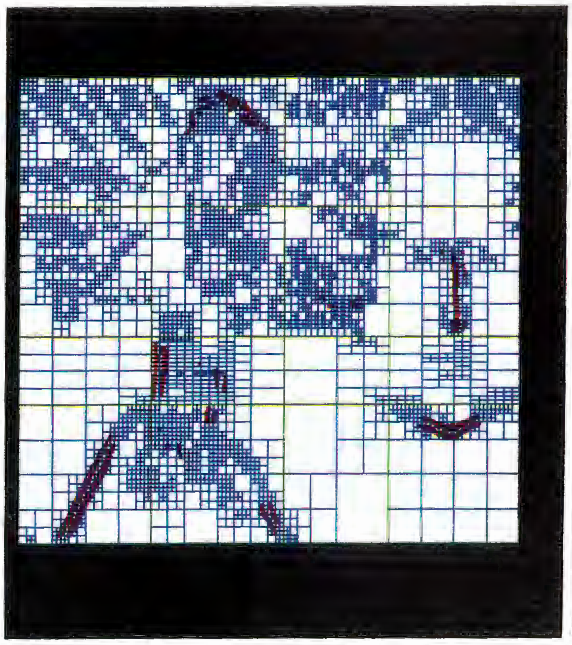
図4．7　初期領域分割

4. 5. 3 Subdivision（副分割）

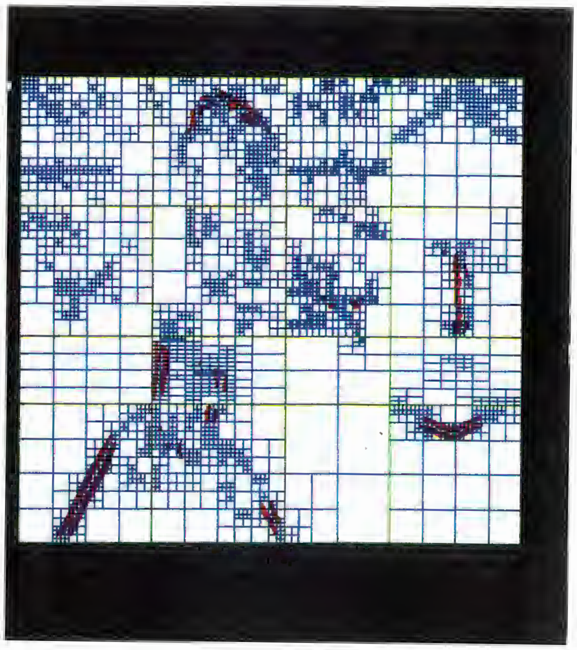
アルゴリズムの所（4.4.4，4.4.5）で述べた方法を用いて、初期領域を分割した結果を図4．8（a）（b）に示す。

図4．8（a）は ε ＝2.5 mm，図4．8（b）は ε ＝5.0 mm と設定した時の結果であり、方形状に分割された領域はそれぞれ双2次Bezier曲面を意味している。また、双2次Bezier曲面の制御点の3次元座標を与えることにより、物体のサーフェイス・モデルを作成することができる。

図4．8（a）（b）より、図中の赤色部分は凹凸領域を表わしており、凹凸領域付近が細かく分割されていることがわかる。



（a） ε ＝2.5 mm

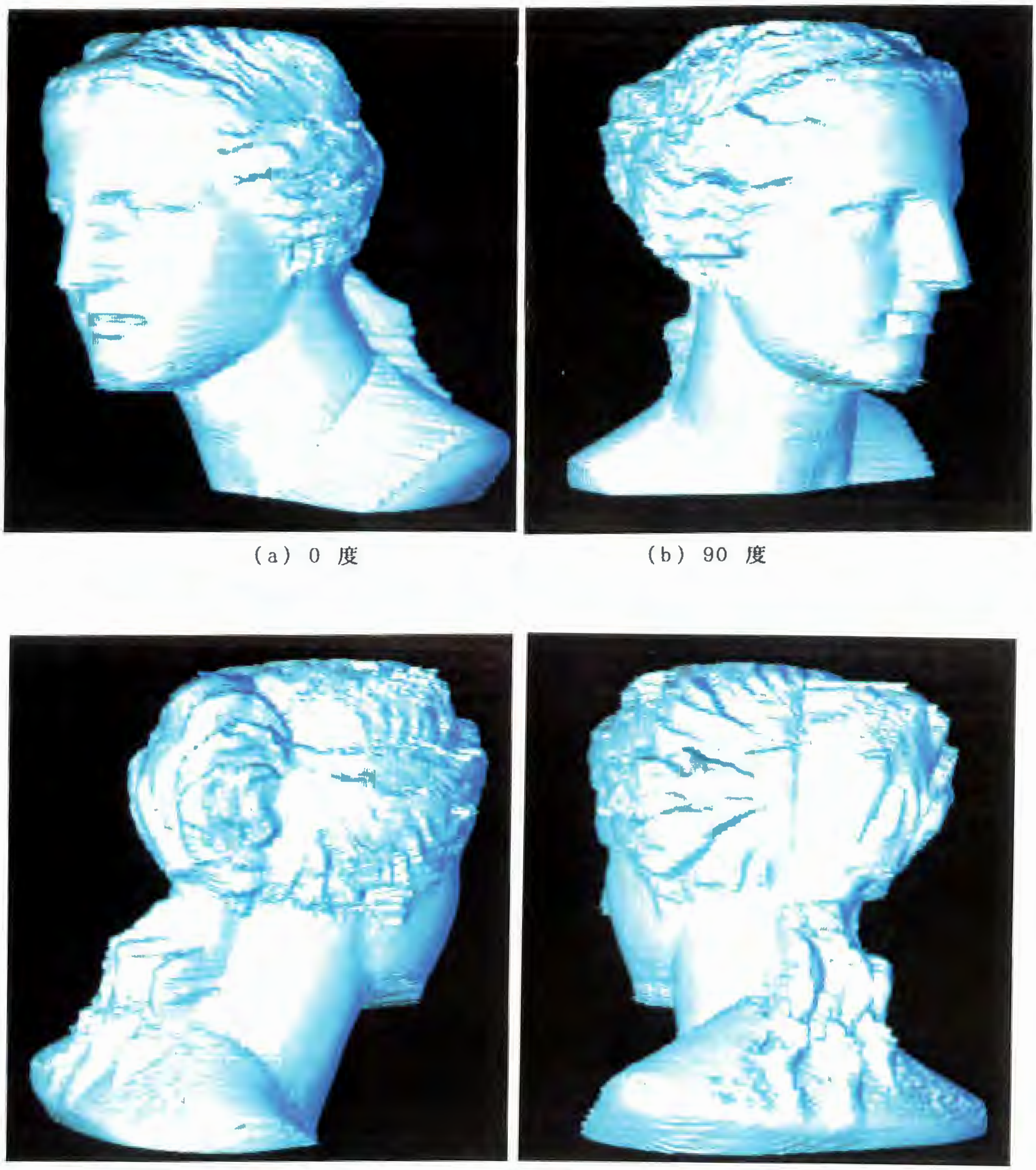


（b） ε ＝5.0 mm

図4．8 分割結果

4. 5. 4 表示

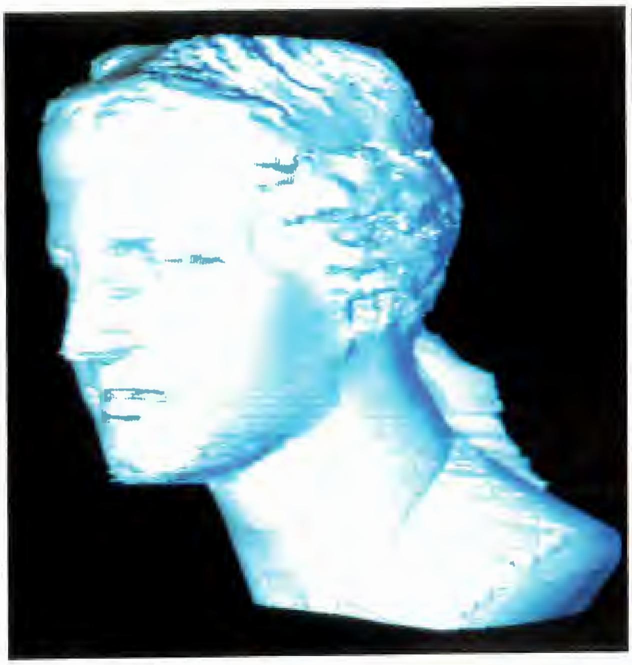
上記の手順でビーナス石膏像のサーフェイス・モデルを作成し、それをθ方向に90度づつ回転させてシェーディングを行なった結果を、図4. 9，図4. 10, 図4. 11に示す。ここで、図4. 9は ε = 1.0 mm ,図4. 10は ε = 2.5 mm, 図4. 11は ε = 5.0 mmの時の結果である。



(a) 0 度　(b) 90 度

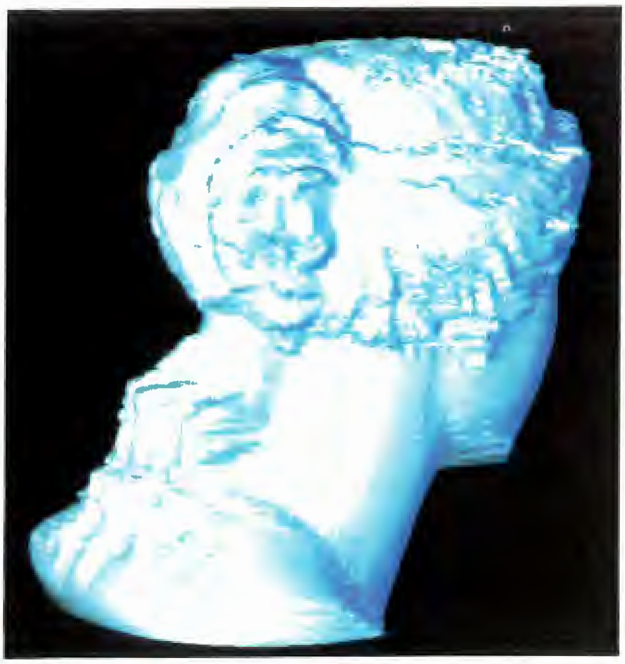
(c) 180 度　(d) 270 度

図4. 9 シェーディング (ε = 1.0 mm)



(a) 0 度

(b) 90 度



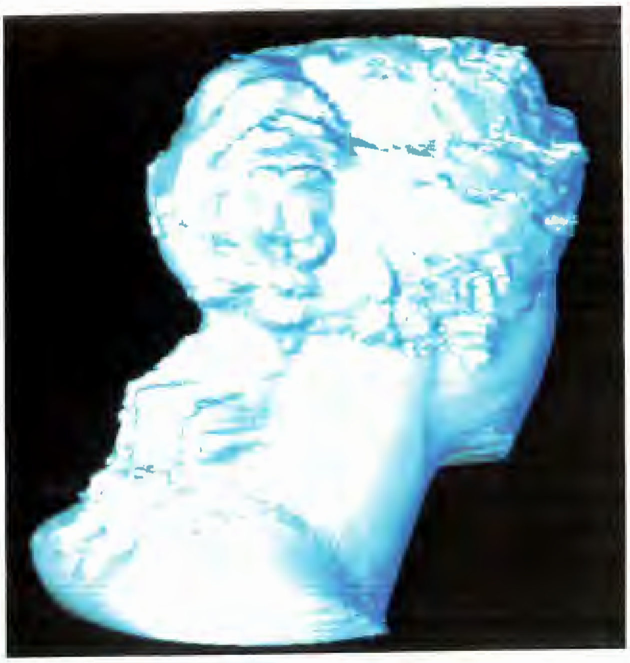
(c) 180 度

(d) 270 度

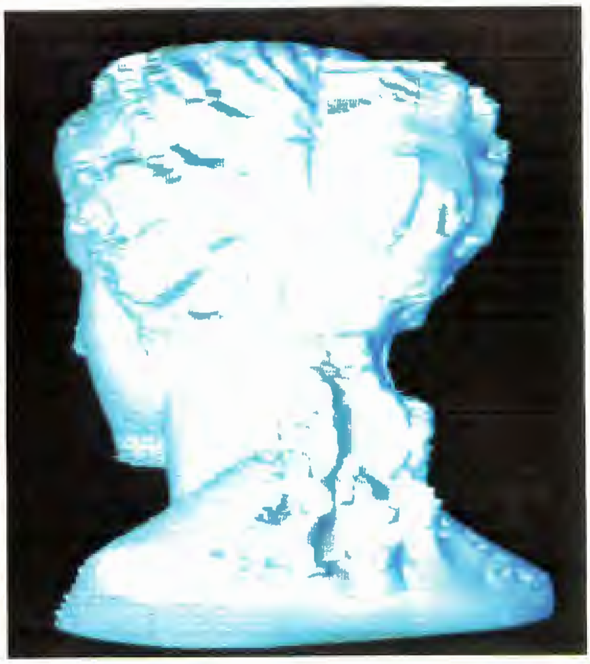
図４．１０ シェーディング（ ε ＝ 2.5 mm ）

(a) 0 度

(b) 90 度



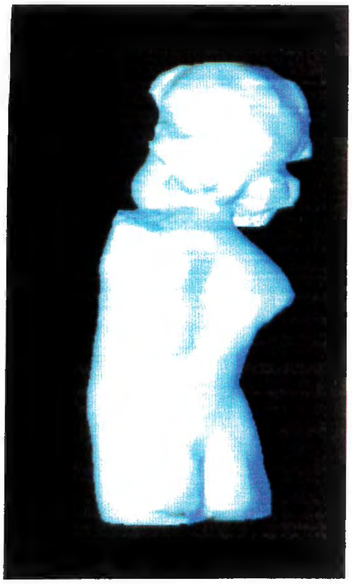
(c) 180 度



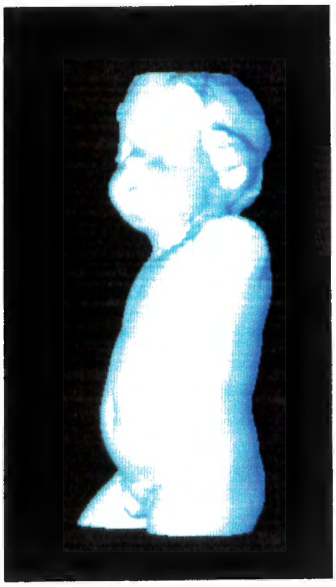
(d) 270 度

図4．11　シェーディング（ ε ＝ 5.0 mm ）

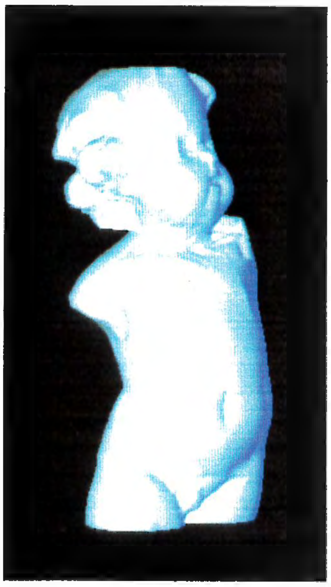
同様の手法で小児立像のサーフェイス・モデルを作成し、シェーディングを行なった結果を図4．１２に示す。但し、こでは、$\varepsilon = 2.5$ mmの時の結果である。



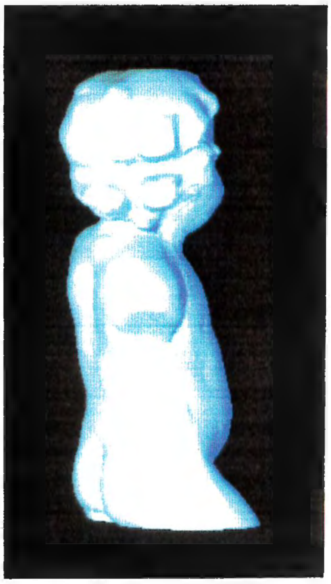
(a) 0 度



(b) 90 度



(c) 180 度



(d) 270 度

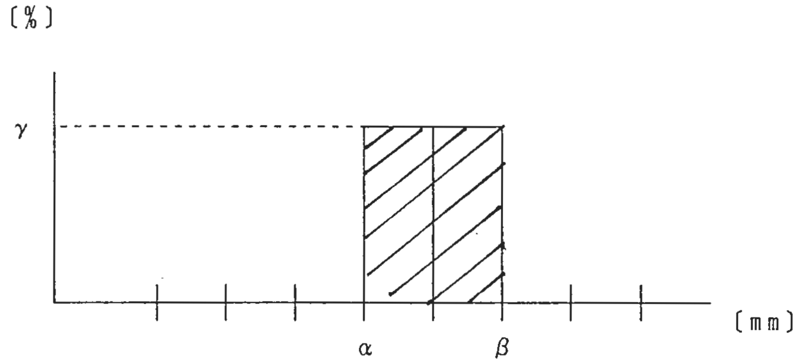
図4．１２　小児立像のシェーディング例

4．6　検討結果

ビーナス石膏像の実験例に対して、r成分の誤差とデータ圧縮比について検討した結果を以下に示す。

4．6．1　r成分誤差

すべての入力データ（360 * 344 = 123840 個）に対して、r成分の誤差を求めた。ここで、r成分の誤差とは、同じ（θ，z）に対する入力データのr成分マイナス Bezier曲面上のr成分 とする。
ε = 2.5 mm の時の結果を図4．13に、ε = 5.0 mm の時の結果を図4．14に示す。ここで、



は、α～β〔mm〕の範囲の誤差を持つデータが γ〔%〕存在することを意味している。また、誤差が ±2ε より大きいものをまとめて ± 2.2ε のところに表示している。

そして、誤差を解析した結果を表4．1に示したが、非常に良い結果が得られている。

	ε = 2.5 mm	ε = 5.0 mm
誤差の平均（mm）	-0.00039	-0.00333
誤差の絶対値の平均（mm）	0.48717	0.65133
誤差のr．m．s（mm）	1.05885	1.23384
ε以下の誤差（%）	97.60	99.22
2ε以上の誤差（%）	0.73	0.13

表4．1　誤差の解析結果

※ 理論的には、2ε以上の誤差は存在しないが、近似誤差等の計算誤差により、2ε以上の誤差が発生したものと思われる。

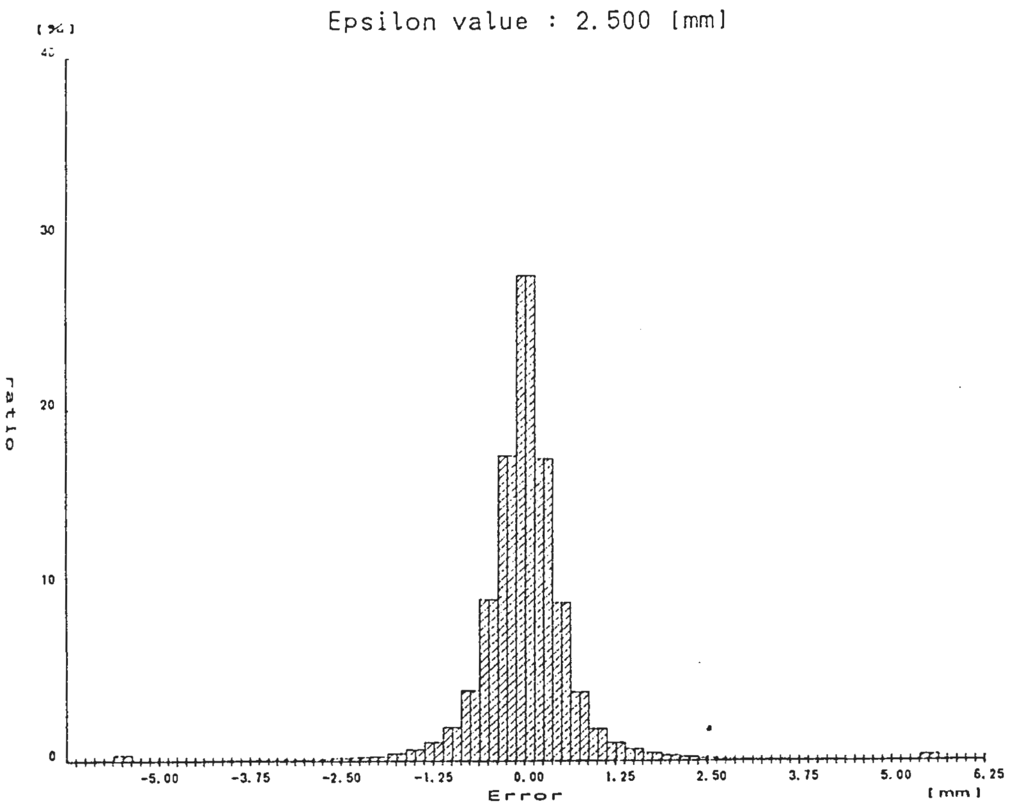


図４．１３　ｒ成分誤差（ε ＝ 2.5 mm ）

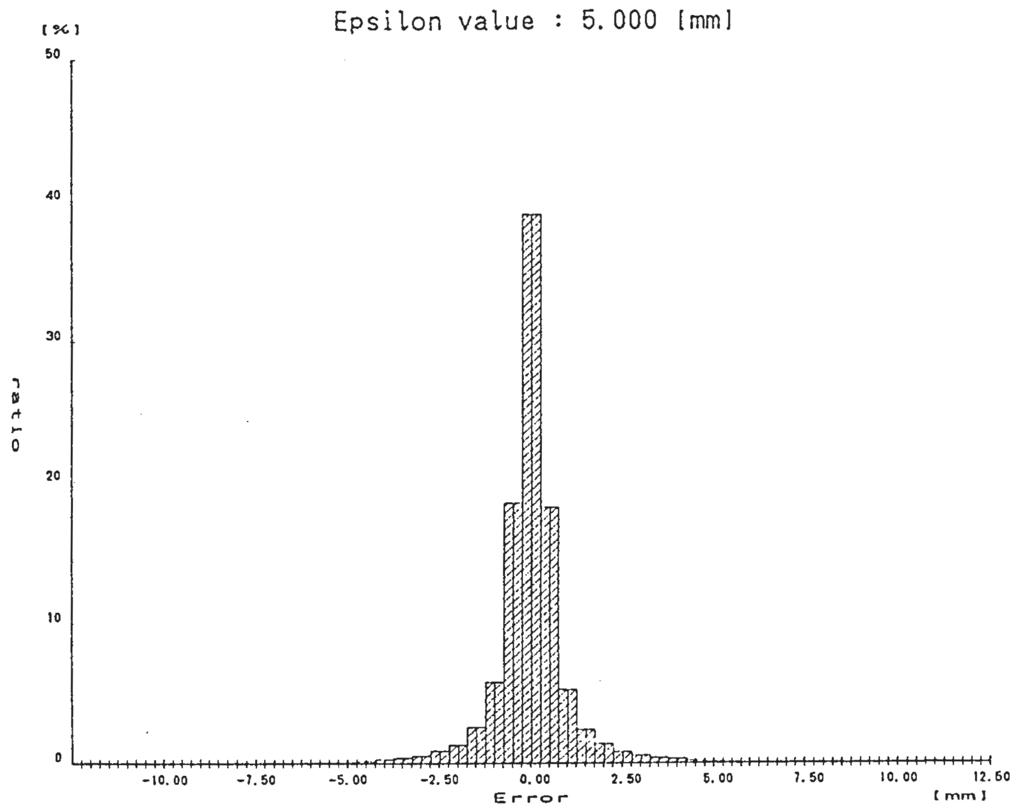


図４．１３　ｒ成分誤差（ε ＝ 5.0 mm ）

4．6．2　データ圧縮比

εの値を 1.0 mm ～ 5.0 mm まで 0.5 mm 間隔で変化させ、最終的に作成されたBezier曲面の数と制御点の数を求めた。それをそれぞれ図4．14と図4．15に示す。

図4．15の縦軸を入力データの総数（360 * 344 = 123840）で割ったものがデータ圧縮比となる。

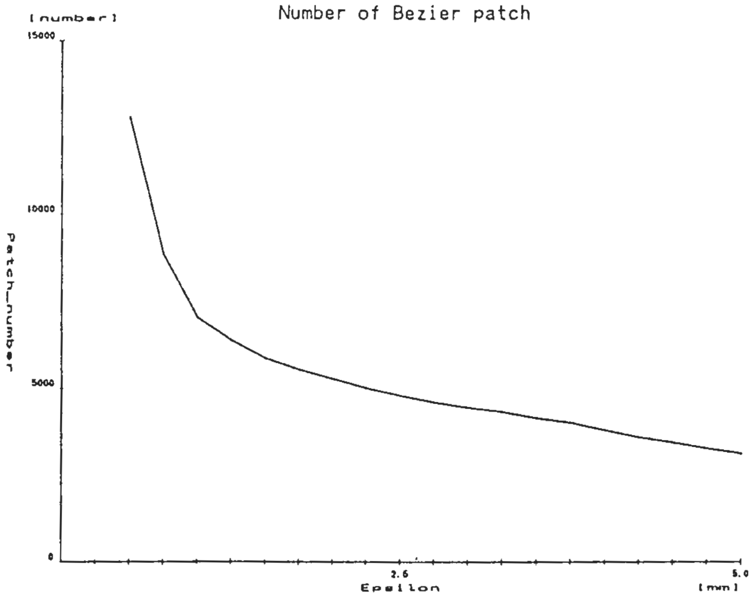


図4．14　Bezier曲面の数

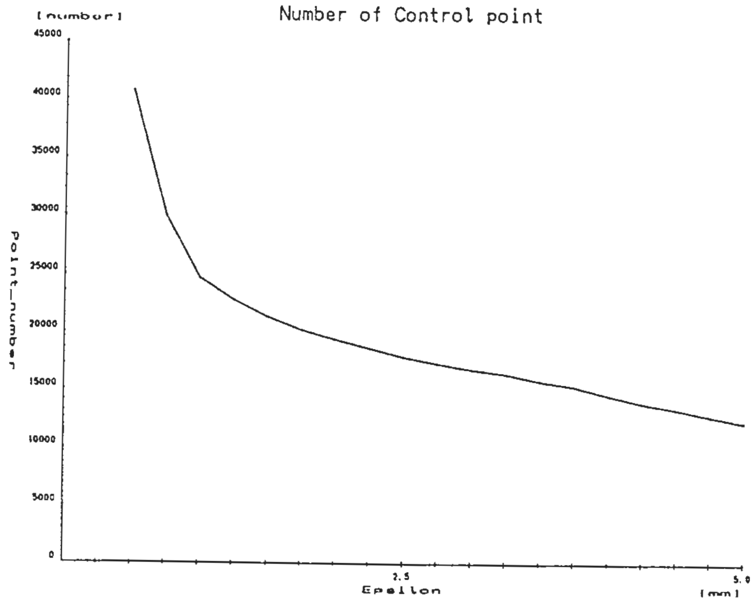


図4．15　制御点の数

４．７　まとめ

普通に曲面分割を行なえば、分割不用な曲面を分割したり、曲面と曲面の接続に苦労するが、本手法を用いると分割すべき曲面は自動的に分割され、曲面間の接続は考えなくてすむ。そして、前章で述べた平面パッチによる物体表現のシェーディング例と比較すると、見た目にも非常に良い結果が得られ、データ数を減らしても違和感の増す割合が鈍くなったといえる。
また、アルゴリズムを３次のＢｅｚｉｅｒ曲線・曲面に拡張することにより、より少ない情報量でより精度の良い近似が可能になると考えられる。

現在のグラフィック・ワークステイション（G.W.S）では、まだ平面パッチが主流を締めており曲面については取り扱いにくいところが多々あるが、今後 G.W.S のハードウェアで曲面をサポートしようという動きがあり、曲面がサポートされていくにしたがい曲面の優位性が増加していくと思われる。

５．おわりに

本報告では、光切断法から得られた立体形状データに対して、断面積を利用した物体再構成の方法とＢｅｚｉｅｒ曲面を利用した物体再構成の方法について検討した結果を述べた。

断面積を利用じた平面パッチによる物体再構成の方法は、ボトルのような形状に対しては、少ないデータ数で物体の再構成を行なうことができ大変有効的だといえる。しかし、ビーナスのような形状に対しては、表示が角張ったものになり曲面の情報が必要であることがわかった。
また、Ｂｅｚｉｅｒ曲面を利用した曲面パッチによる物体再構成の方法は、曲面の取り扱いが簡単なうえに、表示例，誤差ともに非常に良い結果が得られ、今後アルゴリズムを３次のＢｅｚｉｅｒ曲面に拡張することにより、より良い結果が得られると思われる。

謝辞

最後に、本研究を進めるにあたり、いろいろと御指導頂いた（株）ＡＴＲ通信システム研究所葉原耕平会長、有益な御提案を頂いた山下紘一社長に深く感謝致します。また、有益な御助言・御討論を頂いた前知能処理研究室長小林幸雄氏、前知能処理研究室主任西野治彦氏はじめ関係諸氏、並びに、ソフトウェアの作成・実験に御協力頂いた浦真吾氏に感謝します。

参考文献

[1] F.Schmitt, B.Gholizadeh: "Adaptative Polyhedral Approximation of Digitized Surfaces", SPIE Vol.595 Computer Vision for Robots(1985)

[2] F.Schmitt, B.Barsky, W.Du: "An Adaptive Subdivision Method for Surface-Fitting from Sampled Data", SIGGRAPH'86, Volume20,Number4 (1986)

[3] 西野, 秋山, 小林: "光切断法による3次元立体形状自動入力に関する一検討", 第36回情処全大, 5W-7(1988)

[4] 西野, 肥塚, 秋山, 小林: "光切断法による3次元立体形状入力と形状合成", テレビ学技法, 13,6(1989)

[5] 西田, LEE, 小林: "3次元距離画像入力に対するサーフェイスモデルの一検討", 1989年信学春季全大, SD-3-6(1989)

[6] 長谷川, 輿水, 中山, 横井: "画像処理の基本技法", 技術評論社, P.80-81

[7] 西田, LEE, 岸野: "Bezier曲面を利用した3次元物体再構成に関する一考察", 1990年信学春季全大, (1990)