数値情報の画像表示のための処理

東京大学生産技術研究所　正会員　村井俊治
千葉大学工学部　正会員　○建石隆太郎

1. 研究の目的

われわれが土木に関する研究あるいは計画を行うとき、常に数値情報を扱っている。これらの数値情報を画像表示することは、その全体的傾向や特徴を一目瞭然にとらえる上で不可欠である。本研究は、数値情報を画像表示するためのアルゴリズムを確立することを目的としている。

ここでいう画像表示とは、数値情報を濃淡または色のついた画素の集合で表現することで、画像出力のために最終的には細かい格子状に配列された画像データファイルを作成しなければならない。画像出力機器としてはラインプリンタからCRTディスプレイ、ドラムレコーダなど順次走査型のものを対象としている。

2. 数値情報の分類

ここでは、座標の関数として与えられる数値情報を取り扱うことにする。これらの数値情報を、画像表示のためのアルゴリズムを確立する立場から分類すると以下のようになる。

(1)　$t=f(x,y)$　　数値情報 t が平面座標 (x,y) の関数として与えられている場合
例．FEMの二次元解析の結果、国土数値情報、リモートセンシングデータ

(2)　$t=f(\alpha)$　　数値情報 t が平面内の多角形 α の関数として与えられている場合
例．行政界毎の情報

(3)　$t=f(x,y,z)$　　数値情報 t が空間座標 (x,y,z) の関数として与えられている場合
例．FEMの三次元解析の結果、地形面上の情報

(4)　$t=f(\beta)$　　数値情報 t が空間内の多角形平面 β の関数として与えられている場合
例．構造物壁面の情報

3. 画像表示のための処理

2．で分類した数値情報のうち(1)(2)の平面的な場合は、離散的に与えられた数値情報から内挿補間することにより、細かい格子状に配列された数値情報のデータファイルを作る問題となる。一方、(3)(4)の空間的な場合は、平面上に投影変換をするのと並行して隠れ点処理を施し、その投影平面において細かい格子状に配列された数値情報のデータファイルを作る問題となる。2．で分類した各々の場合について、その画像表示のためのアルゴリズムについてのべよう。

3－1　平面的処理

(1)　$t=f(x,y)$ の場合

i) 図1に示すように点 (x,y) が正方形あるいは長方形の格子状に配置されている場合。格子内の点Pの値 t は、周囲の4格子点の値を用いて共一次式により、または周囲16格子点の値を用いて共三次式により内挿される。

①共一次内挿の場合（図2(a)参照）

$$t=(1-y\quad y)\begin{pmatrix}t_a & t_b\\ t_c & t_d\end{pmatrix}\begin{pmatrix}1-x\\ x\end{pmatrix}$$

②共三次内挿の場合（図2(b)参照）

$$t=(g_1(y)\ g_2(y)\ g_3(y)\ g_4(y))\begin{pmatrix}t_{a1} & t_{b1} & t_{c1} & t_{d1}\\ t_{a2} & t_{b2} & t_{c2} & t_{d2}\\ t_{a3} & t_{b3} & t_{c3} & t_{d3}\\ t_{a4} & t_{b4} & t_{c4} & t_{d4}\end{pmatrix}\begin{pmatrix}g_1(x)\\ g_2(x)\\ g_3(x)\\ g_4(x)\end{pmatrix}$$

ここで、x および y は、格子間隔を単位とし格子ごとに定めた座標系における、横および縦方向の点Pの座標である。t_S は、点Sにおける t の値を示す。関数 g は、つぎのように定義される。

$$g_1(u) = -\frac{1}{2}u^3 + u^2 - \frac{1}{2}u$$

$$g_2(u) = \frac{3}{2}u^3 - \frac{5}{2}u^2 + 1$$

$$g_3(u) = -\frac{3}{2}u^3 + 2u^2 + \frac{1}{2}u$$

$$g_4(u) = \frac{1}{2}u^3 - \frac{1}{2}u$$

関数 g は、内挿された値がなめらかに連続するように考えられた関数である。

ii) 図3に示すように点 (x, y) が閉多角形領域内にランダムに分布している場合。内挿したい点Pが閉多角形内に含まれるかどうかを調べ、点Pに近い数点を選び出し、これらの点からあてはめられた関数 $t = f_p(x, y)$ により、点Pの値を求める。

iii) 図4に示すように点 (x, y) がランダムに分布しており、かつ要素を形成している場合。多角形の要素をすべて三角形に分割し、三角形内の点Pの値は、三角形の頂点の値から一次式により内挿される。細かい格子状の点を内挿するためには、これらの点がどの三角形に属するかを探索するアルゴリズムも必要となる。

(2) $t = f(\alpha)$ の場合

この場合は、点Pが閉多角形 α 内に含まれるかどうかを調べる問題となる。

3－2　空間的処理

まず、空間的処理に必要な投影変換と隠れ点処理についてのべ、つぎに 2. で分類した(3) (4) の場合の画像表示のためのアルゴリズムについてのべよう。

ここで考える投影変換は、平行投影変換と中心投影変換である。隠れ点処理とは、見えない部分の数値情報を表示しないようにするための処理である。点画表現の隠れ点処理は、原理的には、重ね書きをすることによってなされる。すなわち、図5に示すように、1本の投射線に対して、視点より距離の遠い方から順に投影変換をしてゆけば、結果的に一番近い点の数値情報だけが投影面に残り、隠れ点処理がなされる。実際の隠れ点処理は、図6に示すように視点を含む平面K（平行投影の場合は、視線を含む平面K）内においてなされる。すなわち、平面Kによる断面上の点を投影変換して投影面上で重なった場合、視点からの距離の大小を論理的に比較して隠れ点処理を施す。言いかえれば、投影面上における直線（投影面と平面Kとの交線 ℓ）の上で隠れ点処理を施すことになる。この平面Kのとり方は決ったものではないが、交線 ℓ が平行等間隔になるように平面K群をえらべば、画像出力用のデータファイル作成には好都合である。

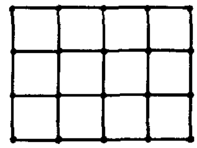

図1　グリッド

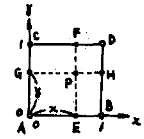

図2(a)　共一次内挿

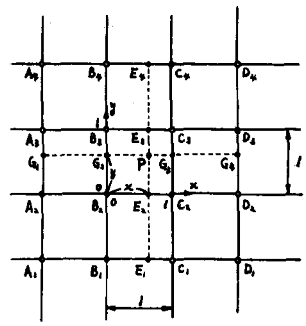

図2(b)　共三次内挿

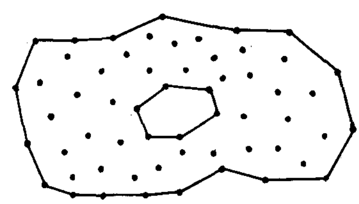

図3　ランダム・ポイント

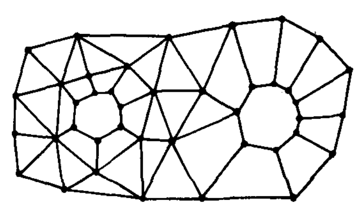

図4　ランダム・エレメント

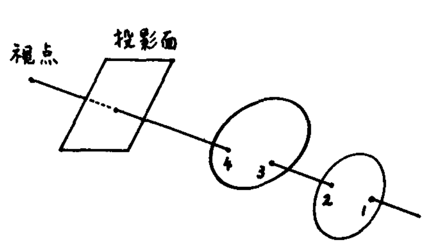

図5　隠れ点処理の原理

つぎに、2．で分類した(3)(4)の場合についてのべる。

(3) $t=f(x,y,z)$ の場合

i) 曲面 $z=h(x,y)$ が一価関数の場合

一価関数の特徴を生かすためには、平面Kとして、投射線を含む鉛直面を考えると都合がよい。図7は、平行投影変換の場合で、図8は、中心投影変換の場合を示している。

処理は、つぎの3段階に分けることができる。

a-① 投射線を含む平面Kによる断面上の点を発生させる。

a-② 投影変換と隠れ点処理

a-③ 画像出力用データファイル作成のために、投影面上において、細かい格子状の点を内挿計算により求める。

(イ) 平行投影変換の場合（図7参照）

点(x,y)が格子状に配置されている場合、処理a-①において投射線方向をデータの並び方向に一致させれば、内挿計算は不要となり、直接、データの1ラインごとに処理a-②がおこなえる。点(x,y)がランダムに配置されている場合は、内挿計算が必要となる。処理a-③は、平行等間隔に並んだ直線ℓ上で内挿計算を行なえばよい。

(ロ) 中心投影変換の場合（図8参照）

処理a-①において、xy平面上で扇状に広がった直線上の点を内挿計算により求める必要がある。処理a-③において、投影面上で扇状に広がった直線ℓ上の点から、細かい格子状の点を内挿計算により求める必要がある。

以上(イ)(ロ)からわかるように、平行投影変換の場合、ラインごとの処理であるため、多量のデータでも処理可能である。これに対し、中心投影変換の場合は、扇状に配列されたデータの内挿計算を含むため、多くのメモリを必要とする。

上の方法は、図6で説明すると、まず平面Kによる断面上の点、(例：A,B,C,D)を発生させて、直線ℓ上に投影する方法である。これに対して、特に平面Kによる断面を考えないで投影変換をし、投影面上の点(例：P,Q)から直線ℓ上の点(例：R)を内挿して求めれば、結果として平面Kによる断面上の点を考えていることになる。図9は、この方法を利用した中心投影変換の概念図である。平面αは、視点を含まない鉛直面である。互いに平行等間隔となる直線群ℓと視点とを含む平面群が平面K群に相当する。

この処理は、つぎの4段階に分けることができる。

b-① 投射線を含まない平面αによる断面上の点を発生させる。

b-② 投影変換

b-③ 投影面上において、i番目の断面(A_j)と$i+1$番目の断面(B_j)とから直線ℓ上の点(例：R,S)を内挿計算で求

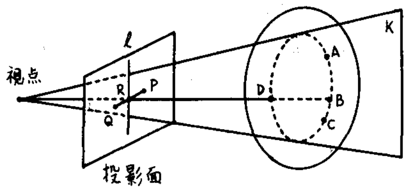

図6　平面K内での隠れ点処理

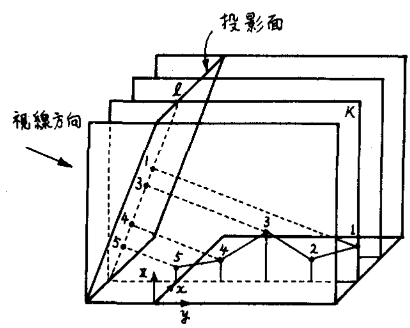

図7　平行投影変換と隠れ点処理

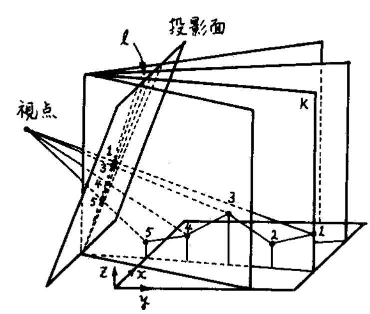

図8　中心投影変換と隠れ点処理
方法1

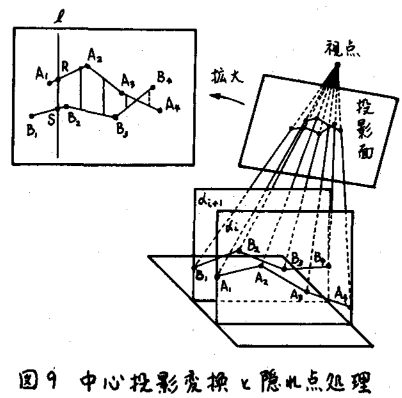

図9　中心投影変換と隠れ点処理
方法2

める。つぎに直線ℓ上で隠れ点処理をする。

b-④　画像出力用データファイルの作成

点（x, y）が格子状に配置されている場合、処理b-①は不要となる。この方法も、ライン単位で完結する処理ではないため、多くのメモリを必要とする。

ii）曲面$f(x, y, z) = 0$が多価関数の場合

平面Kとして、直線ℓが平行等間隔になるようにえらべば、画像出力用データファイル作成に都合がよい。図10は、中心投影変換の場合の概念図である。処理の手順は、曲面が一価関数で表わされる場合の手順a-①、a-②、a-③と同じである。処理a-②の隠れ点処理において、視点からの遠近を論理的に判断するために、投影変換をしてゆく点の順序を工夫しなければならない。

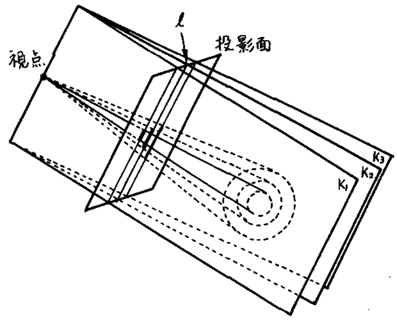

図10　多価関数曲面の中心投影変換

(4)　$t = f(\beta)$の場合

多角形平面β内の各空間座標が同じ数値情報tをもつと考えれば、(3)と同様のアルゴリズムで画像表示が行なえる。

4. 応用例

(1) トンネルの応力解析（写真1）

有限要素法によるトンネルの応力解析の結果である。第1主応力の大きさが2.の(1)の形で与えられている。

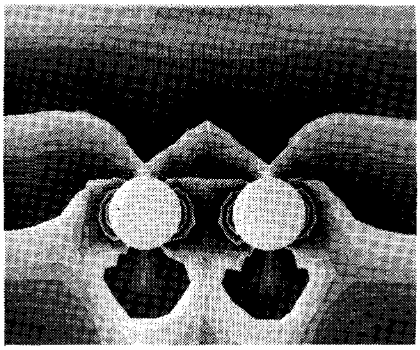

写真1　トンネルの応力解析

(2) 標高メッシュデータからの地形景観図（写真2）

標高データがメッシュ状に与えられており、点（x, y, z）の数値情報として、標高データ自身を色で表示している。2.の(3)の場合で中心投影である。

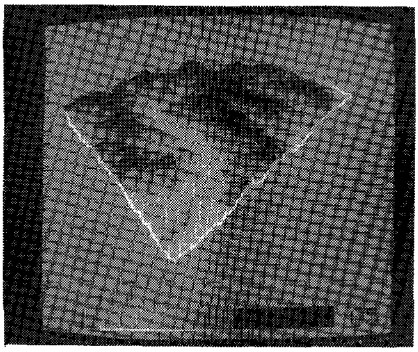

写真2　地形景観図

(3) LANDSATデータによる三次元景観図（写真3）

LANDSATデータと標高メッシュデータとを結合させて作成したものである。2.の(3)の場合で平行投影である。

(4) 三次元構造物の応力図（写真4）

有限要素法による解析結果で第1主応力の大きさを表示している。2.の(4)の場合で平行投影である。

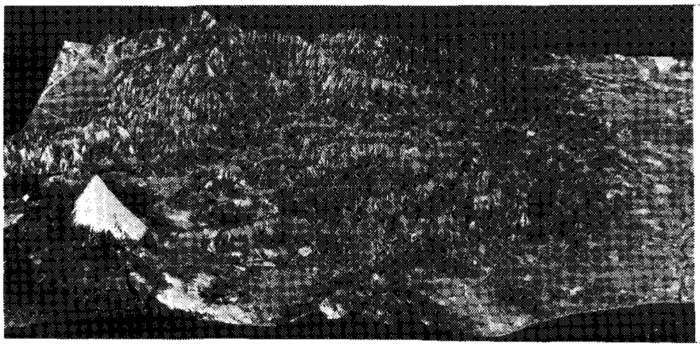

写真3　LANDSATデータの三次元景観図

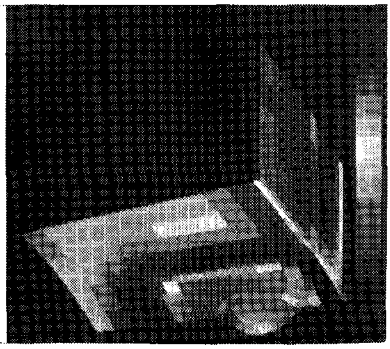

写真4　三次元構造物の応力図