

## 三次元物体のカラー陰影表示の一手法

西田 友是

中前 栄八郎

(福山大学工学部) (広島大学工学部)

### 1. はじめに

三次元物体の透視図をCRT上に表示することは、機器、建築物のCADに有用な手段であり、また教育訓練用シミュレータ、アニメーション等へも応用される。三次元物体を二次元平面上に表示する方法には、隠れた部分を消去して線画で表示する隠線消去(Hiddenline elimination)<sup>1~3)</sup>、および濃淡表示(Half tone presentation)<sup>4~10)</sup>があり、近年では、より写実的に表現できるカラー陰影表示の研究が中心となっている。本稿ではこのカラー陰影表示の一手法について述べる。

Half tone問題に関しては多くの手法が開発されており、次のように分類できる。すなわち三次元図形の記述に関しては、面記述および立体記述<sup>11)</sup>があるが、面記述が多く、曲面を対象としたものが多い。他の面に隠されて見えない部分を判定する隠面消去(Hidden surface removal)については、図形の複雑さに従って細かく画面分割を行ない、そのエリア単位で処理する方法<sup>5)</sup>、および図形をテレビのように走査し、その走査線と面との交線に関して前後判定をし、可視部分を表示する方法<sup>6~10)</sup>がある。前者は走査タイプの表示装置には不向きであり、後者は交線単位で前後判定を行なうので時間を要す。後者においては、時間短縮のため面単位ではあるが走査前に前後関係の順位を求めておく方法も開発されている<sup>11,12)</sup>。また影付に関しては、面の構成稜線を影の生じる面へ投影し、その境界を求める方法、輪郭線分を投影する方法、およびShadow polygonを利用するものなどがある<sup>14)</sup>。

筆者らは、平行光線による濃淡表示について以下の特徴を有する方法を先に提案した<sup>15)</sup>。すなわち、(1)三次元図形を凸多面体の集合として取扱うことによる、入力データの簡易化。(2)凸多面体の輪郭線を利用した隠面消去、影付および物体単位の判定による高速化である。しかし、これらはプロッターあるいはラインプリンターの重ね印字による表示なので十分な画質を得ることができなかった。今回はこの考えをさらに発展させ、カラーディスプレイ装置による表示方法について開発したので報告する。なお、本プログラムの主要な特徴は以下の如くである。

(1)複数光源；平行光線、点光源および照射範囲を限定した点光源の三種類を複数個取扱うことができ、各光源の光度、色および配光特性を任意に指定できる。

(2)物体の自動作成；文献<sup>15)</sup>の機能のほかに、物体の自動複製、自動分割、自動組立機能をもつ。

(3)図柄処理；密等の物体上の図柄や、花柄、レンガ等の模様および材質感の表現を可能とし、より現実的な画面を得ることができる。

(4)WINDOW処理；視点あるいは光源を任意な位置に設定できる高能率なWINDOW処理ができる。

(5)図形のファイリング；計算結果の画像情報をDISKにファイルすることにより、色および濃淡レベルの修正、図柄の追加等の処理が可能である。これはデザイン変更を容易にし、かつその処理時間を短縮する。

## 2. 定義および準備

- (1)光源：全方向を照らす点光源、ある方向を円錐状に照らす点光源（仮にスポットライトと称す）および平行光線の3種類の光源を取扱う。入力データとしては、点光源は光源位置、スポットライトは光源位置、指向点および照射角（図3参照）、平行光線はその方向( $\theta, \varphi$ )により与える。また光源の色は基本的には白色光とし、色成分( $r, g, b$ )を指定することにより色光源とすることができる。なお、スポットライトの場合必要に応じて配光特性（但し、軸対称のもの）を与えることができる。
- (2)透視面：視点から見る場合には注視点を通り視線に垂直な平面、また光源から見る場合注視点（スポットライトの場合は指向点）を通り光線に垂直な平面。
- (3)物体：本稿で取扱う物体は凸多面体（以下単に物体と称す）とし、面の法線ベクトルは物体外部を向くように定義する。また面は物体外部から見て右廻りの頂点番号列で定義する。
- (4)図柄：物体上の色の異なる凸多角形を図柄とし、その図柄の属す面番号、構成頂点座標（右廻り）により定義する。また任意な形状の模様や材質感を表現することをTEXTUREといい、TEXTUREはある領域を $n \times m$ に量子化し、各点の色を与える。
- (5)物体の色：物体の各面および図柄の色は単位白色光を当てた際の色により定義する。但し、使用する色を予め( $R, G, B$ )で与え、入力順に番号付けをしておき、その色番号を用いて各面の色指定をする。
- (6)輪郭線：視点または光源を向いている面を、それぞれ表の面(front-face)、その他の面を裏の面(back-face)とする。また表と裏の面との稜線を輪郭線分、そのつながりを輪郭線(contour line)とする。したがって視点または光源から見た際の表の面および輪郭線は右廻りで、かつ凸多角形となる性質がある。
- (7)陰影：光の当たらない面を陰の面(Shade)とし、他の物体にさえぎられて生じる部分を影(Shadow)とする。

## 3. アルゴリズム

物体と視点および光源との関係には次の性質がある。

- (1)不可視面か否か、あるいは陰の面か否かは視点あるいは光源に対する面の向きで求まる。
- (2)物体が重なって見えるか否かは、視点からみた透視面上において、物体の輪郭線が重なるかどうかで求まる。
- (3)影が生じるか否かは、光源から見た透視面上において、物体の輪郭線同士が重なるかどうかで求まり、その影の境界は隠す側の輪郭線で求まる。

以上の性質を利用した本アルゴリズムは、大別して次のステップから成る。

- [ステップⅠ]：入力パラメータから頂点座標、面の構成頂点番号等のデータ構造を作成する。また必要なら自動複製、自動分割、自動組立等を行なう。
- [ステップⅡ]：視野と各物体の位置関係により物体を分類し、視点より前方の頂点を透視面上へ座標変換する。さらに面の向き、輪郭線を求める。
- [ステップⅢ]：視点から見た透視面上においては、輪郭線の交差判定を行ない、交差する場合は物体の前後関係を求める。また光源から見た透視面上では、輪郭

線の相互関係から面上の影の領域を求める。

[ステップⅣ]: 透視面上の図形を上から下へと走査し、走査線上の物体を求め隠面消去を行なう。次に各点の照度および色を求めてCRTへ出力する。

また、必要ならステップⅣの後にTEXTURE 処理等を施す。

図1に本アルゴリズムの概略流れ図を示す。図でもわかるようにステップⅡ,Ⅲは視点から見た場合および光源から見た場合について、すなわち( $N_s+1$ )回繰返す。さらにステップⅡ~Ⅳは $k_0$ 回繰返す。但し、 $N_s$ は光源数で、 $k_0$ は視点または光源の移動数である。

#### 4. 主要な処理方法

本プログラムにおける主要な処理について説明を加える。

##### 4.1 物体の自動作成

使用頻度の多い形状(直方体, 円柱等)は標準化され、必要なパラメータにより自動作成されるが、他に次の処理をする。

① 自動分割; 物体同志を相貫させて入力した場合、自動的に凸多面体に分割する。また虚物体を考え、この虚物体と実物体を相貫させることにより物体に穴をあけたり、物体を切断することができる。

② 自動組立; 使用する部品をDISKに保管しておき、例えばどの物体のどの面にどの物体を接合させるといった記述をすることにより、任意な部品を用いて容易に物体を組立てることができる。

##### 4.2 WINDOW 処理

視野を図2に示すような四角錐と考えこれを視野ピラミッドと称する。またスポットライトに関して視野の場合と同様な考えから図3に示すような

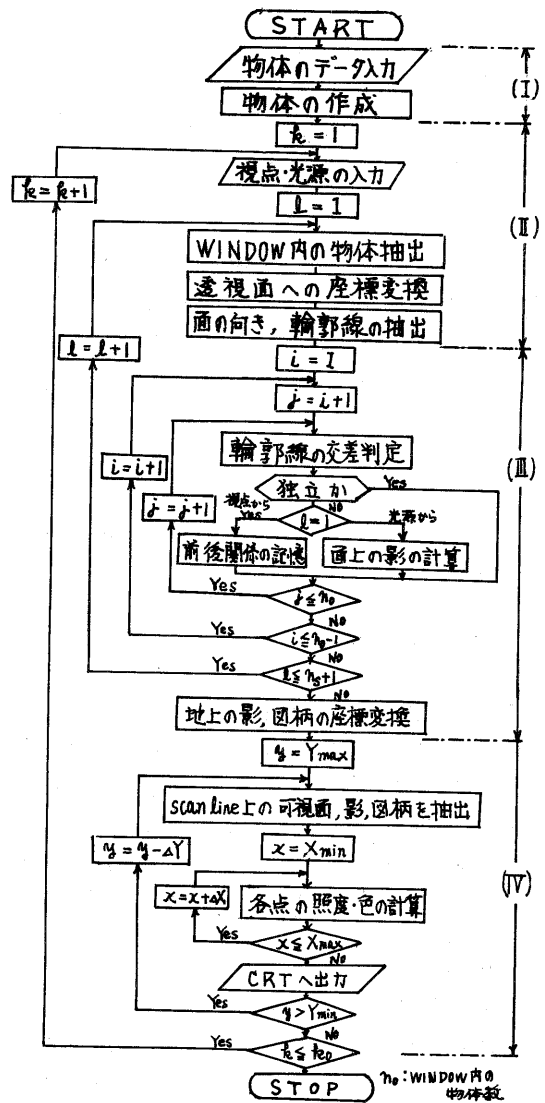


図1. 流れ図

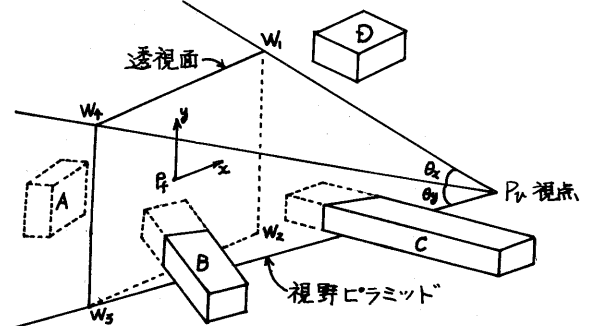


図2. 視野と物体との関係

照射領域が考えられる。これを光円錐と称することにする。但し、点光源の場合は照射角が $180^\circ$ と考えればよい。

これらの視野(照射領域)と物体との関係から物体を次のように分類し、それぞれに応じた処理を行なう。

#### (1) 物体の分類

① 視点に対して；全頂点が視野ピラミッド内にあるものをA種、全頂点が視点の前方に存在し、かつ物体の一部が視野ピラミッド内のものをB種、物体の一部は視点後方にあり、かつ一部視野ピラミッド内のものをC種とする。なお、B、C種の物体については、その面について、それぞれ物体の場合と同様にA～Dに分類する。

② 光源に対して；視点の場合と同様に光円錐との関係により物体をA～D種に分類する。ここで光の到達距離は有限と考え、ある距離より遠い物体はD種と考える。但し、この距離は光度およびディスプレイ装置の輝度レベルの能力により決定される。なお、点光源の場合は注視点と指向点とを考え、平行光線の場合はすべての物体はA種となる。

#### (2) 処理方法

① 可視判定：D種物体は当然不可視であり、A～C種物体の表の面が見える可能性のある面であるから、A～C種に関してのみ輪郭線を利用して交差判定する。

② 影の判定：光源に対するA～C種物体が視点に対するA～C種物体(光源に対するD種を除く)に影を落とすか否かを求めればよい。

ここで視野ピラミッドと物体との関係は後述の(2)式を、光円錐との関係は(T)式を用いて求める。また視点より後方か否かの判定は(付2)式の分母で求まる。

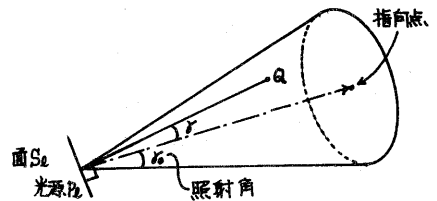


図3. 光円錐

### 4.3 面の向き判定

面 $S[P_1, P_2, P_3, \dots]$ を考えた場合、凸多面体の性質から、最初の3頂点のみを用いて面の向きが求まる。視点より前方の面の向きは透視面上の式(付3)の符合により求めることができる。但し、C種物体のように透視面へ投影できないものについては次式により求める。面の向きは視線と面の法線ベクトルとの内積で求まるから、次式が正なら表の面である。但し、視点 $P_v(X_v, Y_v, Z_v)$ とする。

$$F_s(P_v) = \frac{(P_v - P_1) \cdot (P_1 - P_2) \times (P_2 - P_3)}{|(P_1 - P_2) \times (P_2 - P_3)|} = \Delta A X_v + \Delta B Y_v + \Delta C Z_v + \Delta d \quad (1)$$

ここで

$$\Delta A = -\frac{1}{\Delta} \begin{vmatrix} Y_1 & Z_1 \\ Y_2 & Z_2 \\ Y_3 & Z_3 \end{vmatrix} \quad \Delta B = -\frac{1}{\Delta} \begin{vmatrix} X_1 & Z_1 \\ X_2 & Z_2 \\ X_3 & Z_3 \end{vmatrix} \quad \Delta C = -\frac{1}{\Delta} \begin{vmatrix} X_1 & Y_1 \\ X_2 & Y_2 \\ X_3 & Y_3 \end{vmatrix} \quad \Delta d = -\frac{1}{\Delta} \begin{vmatrix} X_1 & Y_1 & Z_1 \\ X_2 & Y_2 & Z_2 \\ X_3 & Y_3 & Z_3 \end{vmatrix} \quad \Delta = \sqrt{\Delta A^2 + \Delta B^2 + \Delta C^2}$$

### 4.4 物体の交差判定および前後判定

隠面消去および影を求めるために輪郭線の交差判定を行なうが、視点(光源)より前方の物体に対しては透視面上で行なう<sup>3)</sup>。一方、判定する物体の一方でもC種物体の場合は文献3)の方法を三次元へ拡張し、次の判定を行なう。

① 交差判定；本方法は輪郭線同志に限らず、面と輪郭線、面同志の判定にも利

用できるから、ここでは面  $A [P_{i1}, P_{i2}, \dots, P_i, P_{i+1}, \dots, P_n]$  と輪郭線  $B [P_{j1}, \dots, P_j, P_{j+1}, \dots, P_m]$  について説明する。視点  $P_v$  と  $P_i, P_{i+1}$  を右廻りに定義した面を  $S_i$  とすると、次式が成立する。

- i)  $F_{Si}(P_j) \leq 0$  ( $j=1 \sim m$ ) なる線分  $[P_i, P_{i+1}]$  があれば  $A$  と  $B$  は独立
- ii)  $F_{Si}(P_j) \geq 0$  ( $i=1 \sim n, j=1 \sim m$ ) なら  $A$  が  $B$  を囲む
- iii)  $(\text{sgn } F_{Si}(P_{j+1}) - \text{sgn } F_{Si}(P_j)) \cdot (\text{sgn } F_{Sj}(P_{i+1}) - \text{sgn } F_{Sj}(P_i)) = -4$  なら  $[P_i, P_{i+1}]$  と  $[P_j, P_{j+1}]$  は交差する

iii) の場合  $[P_j, P_{j+1}]$  上の交点  $P_{tj}$  は次式により求まる。

$$P_{tj} = \frac{F_{Si}(P_{j+1}) \cdot P_j - F_{Si}(P_j) \cdot P_{j+1}}{F_{Si}(P_{j+1}) - F_{Si}(P_j)} \quad (3)$$

② 前後判定；交差した場合、面  $A$  は表の面であることから、(1)式に  $P_{tj}$  を代入した際の符号から前後関係が求まる。但し、 $A$  が輪郭線の場合は  $[P_i, P_{i+1}]$  を含む表の面を用いて判定する。

ここで(1)、(3)式は三次元であるが、処理時間を短縮するため、例えば  $X-Y$  平面 ( $z=0$ ) に投影して二次元関数として判定する。

また、ii) の  $A$  が  $B$  を囲む場合は  $B$  の任意の頂点  $P_j$  を用いて  $F_A(P_j)$  により求まる。

#### 4.5 面上の影

光源から見た輪郭線が交差した場合、隠される側の物体に影を生じる。したがって隠される物体の構成面のうち、視点および光源に対して表の面について影の領域を求めておく。影の領域の抽出法は文献(5)の方法を用いるが、記憶形式は多種光源を取扱うことから次のように改良する。

ここでは、図5に示すように輪郭線  $B$  が面  $A$  に影を落とす例について説明する。本例では影の領域は  $Q_{IN}, Q_{j+1}, Q_{OUT}$  によって囲まれた部分であり、これらの点を(付1, 2)式により透視変換した二次元点列の形式で記憶する。ここで  $Q_{IN}, Q_{OUT}$  は(3)式で求まり、 $Q_{j+1}$  は  $P_{j+1}$  を面  $A$  へ投影した点であり次式により求まる。ただし光源を  $P_L$  とする。

$$Q_{j+1} = \frac{F_A(P_{j+1}) \cdot P_L - F_A(P_L) \cdot P_{j+1}}{F_A(P_{j+1}) - F_A(P_L)} \quad (4)$$

これらの影の構成頂点を記憶し、ステップIVにおいて走査する。なお、この影は透視面上でも必ず凸となる性質がある。

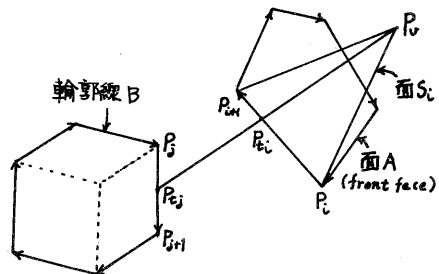


図4. 交差判定

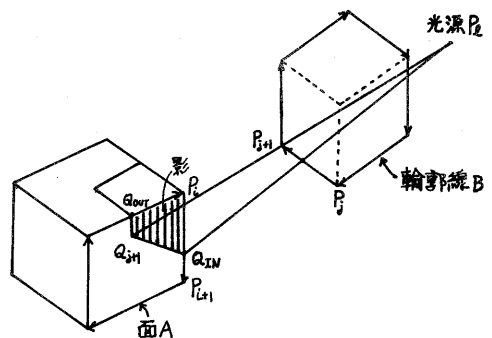


図5. 面上の影の領域

#### 4.6 図形の走査

図形の走査は、面を単位として三次元上で行なう方法が多く発表されているが、本手法は物体を単位として走査を行なう。しかも前述のように、ほとんどの処理は二次元平面上で行なう。また対象図形(輪郭線, 図柄, 影, 面)はすべて凸多角形であるから処理が簡単であり、処理の高速化が期待できる。

本手法は、まず輪郭線を用いて scan line 上の物体を求め、次にその物体上の表の面, 図柄, 影の順で scan line との交点を求める。ここで凸多角形の性質から交点は必ず2個であり、次の scan line ( $y_s = y_s - \Delta Y$ ) と交差する線分は容易に予測できる。また物体の走査を容易にするため、遠方から順に番号づけされた物体番号のテーブル DP を考え、ある物体の Y 座標の最少値以下になるとその物体を消去する。したがって物体を走査する個数は画面下方になるほど減少する。

図形と scan line との交差判定は、A および B 種物体については文献 [5] の方法を用いるが、C 種物体については次の方法で行なう。物体については輪郭線を視野ピラミッドでクリッピング後、透視面上へ投影し、二次元上で走査するが、面に関しては記憶容量等(クリッピングによる新頂点増加の制約)から三次元上で判定する。すなわち scan line と視点  $P_v$  によって構成される面を scan plane  $S_p$  とし、図 6 の  $P_L, P_U, P_R$  のように右廻りに定義すると、表の面  $S_i [P_i \dots P_i, P_{i+1} \dots]$  を走査する場合の関係は、

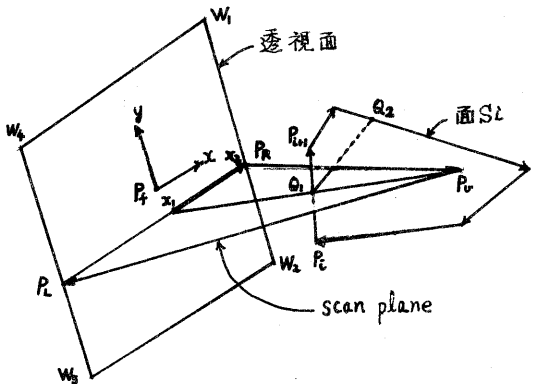


図 6. 三次元上での図形走査

$$\begin{cases} F_{sp}(P_i) > 0, F_{sp}(P_{i+1}) < 0 \text{ なら } [P_i, P_{i+1}] \text{ は } S_p \text{ と交差し、その交点は } Q_1 \\ F_{sp}(P_i) < 0, F_{sp}(P_{i+1}) > 0 \text{ なら } [P_i, P_{i+1}] \text{ は } S_p \text{ と交差し、その交点は } Q_2 \end{cases} \quad (5)$$

となる。ここで  $Q_1$  が  $P_v W_1 W_2$  平面より外側、あるいは  $Q_2$  が  $P_v W_3 W_4$  平面より外側なら面  $S_i$  は視野外である。それ以外の  $[Q_1, Q_2]$  を視野ピラミッドでクリッピングし透視変換を行えば scan line 上の交点が求まる。

#### 4.7 隠面消去および影の走査

scan line 上の面のうちどの部分が可視であるかの判定は次のようにして求める。scan line 上の画素数(今回は 512)に対応したスキャンテーブル  $LS1$  を考える。scan line 上の物体はテーブル DP の順に求まるから、その順に面番号を  $LS1$  に記憶することにより最終的には各画素における可視面が求まる。ここで scan line 上で物体の前後に関する順位(depth priority)が一意に決まるのは凸多面体だからである。

影についても、面の場合と同様に scan line 毎に求める。これはステップ III で求めておいた影の領域との交差判定により求まる。面の場合と同様にテーブル  $LS2$  を考え、影の範囲を記憶する。本稿では複数光源を考えているから、同一点でも各光源に対して影が否かは異なる。したがって光源  $l$  に対して  $x_1 \sim x_2$  間が影なら、 $x_1 \sim x_2$  番地の  $l$  ビットを 1 とする形式を用いて記憶する。

#### 4.8 各点の照度および色の算出

透視面上の点 $P(x, y)$ の照度および色は、 $P$ 点を含む面の法線ベクトル、その面のもつ色、 $P$ 点を照らす光源の位置、光度および光源の色により決定される。scanline上のある画素の面および影の情報はテーブル $LS1, LS2$ より求まるから、以下の方法により照度を求め、次に色を求める。

①点光源の照度： $P$ 点を含む面を $S$ とし、 $P$ 点を面 $S$ 上へ逆透視変換した点を $Q(X_q, Y_q, Z_q)$ とする。光源の位置 $P_l(X_l, Y_l, Z_l)$ および光度 $I_l[I_r, I_g, I_b]$ が与えられ、 $Q$ 点の光源 $l$ に対する照度 $E_l[E_r, E_g, E_b]$ は次式となる。

$$E_l = \frac{\cos \alpha}{r^2} \cdot I_l = \frac{F_s(P_l)}{r^3} \cdot I_l \quad (6)$$

ここで、 $r = |P_l Q|$ ,  $\alpha$ は面の法線ベクトルと光線のなす角である。なお(6)式の分子は同一面、同一光源に対しては定数となる。

②スポットライトの照度； $Q$ 点が光円錐内なら(6)式を用い、光円錐外なら $E_l = 0$ とする。但し、 $Q$ 点が光円錐内か否かは(7)式と $\cos \theta_0$ との大小関係により求まる。

$$\cos \theta = \frac{(Q - P_l) \cdot (P_0 - P_l)}{|Q - P_l| \cdot |P_0 - P_l|} = \frac{F_{sl}(Q)}{r} \quad (7)$$

ここで、 $P_l$ を通り $P_0 - P_l$ に垂直な面を $S_l$ とする(図3参照)。また、配光曲線を指定する場合は光度は $\theta$ の関数 $I_l(\theta)$ となる。

③平行光線の照度；面 $S$ 上のどの点の照度も同じであり、次式により求まる。

$$E_l = I_l \cos \alpha = (\Delta A \lambda_x + \Delta B \lambda_y + \Delta C \lambda_z) I_l \quad (8)$$

ここで $(\Delta A, \Delta B, \Delta C)$ は面 $S$ の単位法線ベクトルで、 $(\lambda_x, \lambda_y, \lambda_z)$ は平行光線の方角余弦である。

④色の算出；以上の方法により各光源に対する照度が求まる。したがってその総和を $E$ とすると $P$ 点の色成分 $D[D_r, D_g, D_b]$ が次式により求まる。

$$D = k(E + H) \cdot C$$

ここで $C[C_r, C_g, C_b]$ は面 $S$ のもつ色であり、 $H$ は環境による照度である。この $H$ は色成分を有すが、白色光の場合スカラーとなる。また $k$ は最も明るい場合でも表示装置の許容輝度レベルを超えないようにするための補正係数である。

#### 4.9 TEXTURE 処理

花柄等任意な形状の模様や、レンガ、畳等の材質を表現(但し、光沢感を除く)するためのTEXTURE 処理について述べる。

①TEXTURE の記述方法；模様は面上のある範囲内であるパターンが繰返されているものとし、その基本パターンを $n \times m$ に量子化し、各点の色を記憶しておく。図7に示すように、模様のある面上において $(x, y)$ 座標系を定義する。 $(\Delta x_2, \Delta y_2)$ 幅で量子化されたパターンが、 $P_0(x_0, y_0)$ を原点とし $(\Delta x_1, \Delta y_1)$ 間隔で繰返されているものとする。

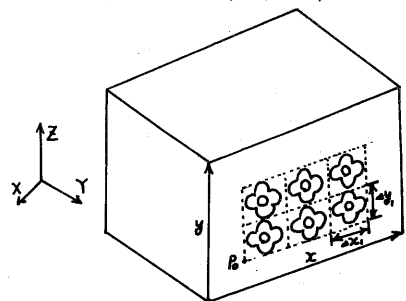


図7. TEXTURE 処理

②処理方法；本手法は任意な色・形状を表現

できる反面、多大の記憶容量を要することからDISKに図形(各点の面番号, 照度)を一時ファイルし、他のJOBで再処理する方式をとる。なお次の手順により処理する。i) DISK内の画面を上から下へと走査し、その点がTEXTURE処理を施す面であれば、逆透視変換によりその点の三次元上の点 $Q(X, Y, Z)$ を求める。ii)  $Q$ 点をその面上の座標系 $q(x, y)$ に変換する。iii)  $Q$ 点に対応するパターン要素 $(i, j)$ を(10)式により求める。iv) その点の照度とパターンの色情報によりその点の色を求める。

$$\begin{cases} i = \text{mod}(x - x_0, \Delta x_1) / \Delta x_2 + 1 \\ j = \text{mod}(y - y_0, \Delta y_1) / \Delta y_2 + 1 \end{cases} \quad (10)$$

## 5. 適用例

図9に適用例を示す。(a)はトランスのCAD例であり、コイル部の円筒は虚物体円柱を用いて作成したものである。さらに構造を理解し易くするために自動分割処理によりカットモデルとし、影の処理を省略している。(b)~(d)は実在する建築物の例である。(b)は鳥瞰図であり、(c)は接近した際の図である。(c)については現実感を増すために空を水平線に近づくほど白っぽくした。(d)は夜間照明の例であり、光源は玄関、通路、芝生照明用の三点光源とし、各光源の色・光度は異なる。また芝生用電球および一部の窓は発光色として処理した。(e)は室内の例を示す。本例は必要部材を土台から順に自動組立により作成したものである。また複、畳、掛け軸にはTEXTURE処理を施し、光源は上部が暗くなる配光特性を用いている。(f)はコンピュータアートへの応用であり題は「生花」である。これは学部祭に展示した学生の作品中、最高得票を得たものである。

本例の計算にはOKITAC 40/50およびグラフィカM508カラーCRT(512×512画素, 256輝度レベル)を使用した。なお、本プログラムはFortranで約5000ステップである。

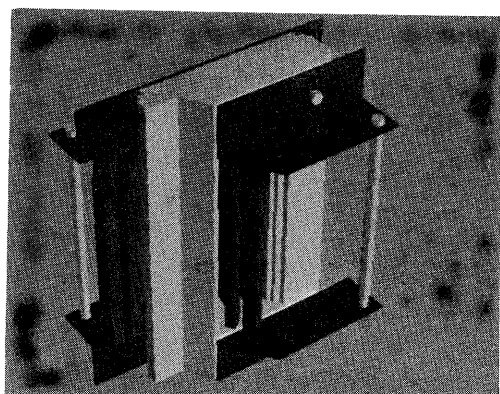
## 6. むすび

適用例にも示したように、本システムは物体の記述、アレンジが容易であるので機器、建築物等の形状設計に便利であり、また照明設計、色デザインにも有用である。本稿は白黒表示で残念であるがかなりリアルな画像を得ることができた。

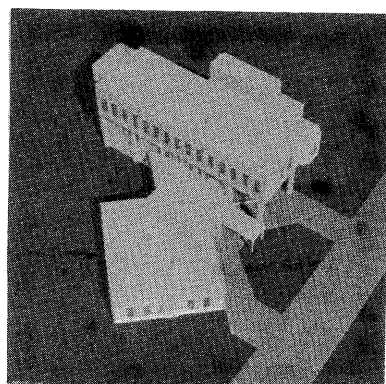
処理時間に関して以下の特徴を持つ。①計算式が(1),(3),(付3)式の如く簡単でしかもその値を重複して利用できる。②凸多面体の輪郭線を利用することにより物体の前後判定、影の算出、図形の走査が物体を単位として処理され、かつ殆んどの判定を二次元上で行なうことにより処理時間が減少する。例えば交差判定を面毎に行なう場合、面数を $n$ とすると $n^2$ 回必要であるが、物体単位の場合は、直方体を例にとると $n$ に減少する。③この種の問題では図形の走査に最も時間を要すが、前後関係および影は前もって求めておくから走査時間の短縮が計れる。例えば pint by pint で図形を走査する場合、画面サイズの二乗に比例した時間を要すが、本手法を適用した場合は、画面サイズの1.4乗程度の結果が得られた。

なお、今後の課題としては、より実用的なシステムにするために非常に多くの物体と光源が処理できるプログラムを開発する必要があると思う。

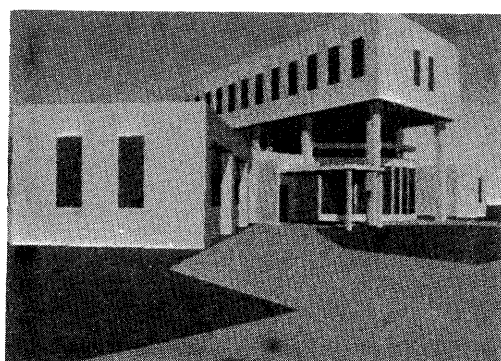




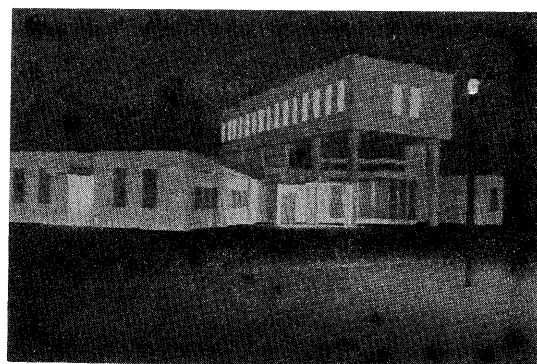
(a)



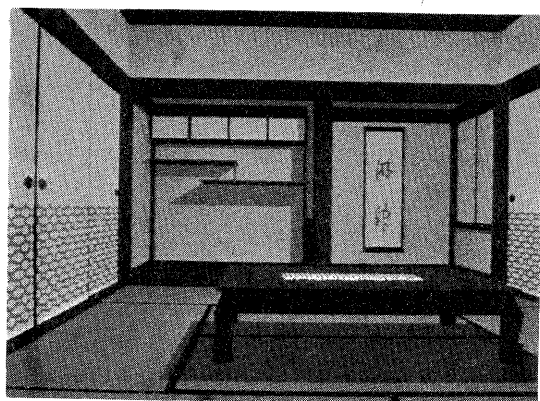
(b)



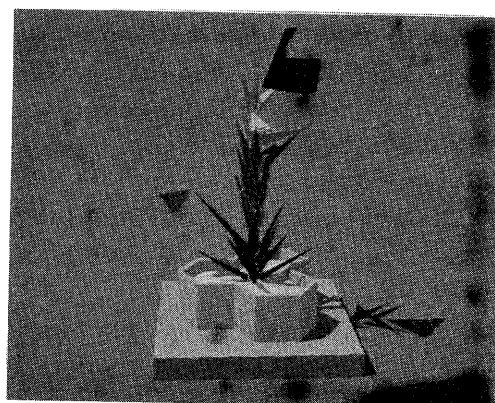
(c)



(d)



(e)



(f)

图9. 适 用 例

[付録]

i) 透視変換：三次元上の点  $P(X, Y, Z)$  を透視面上の点  $P'(x, y)$  へ投影する

$$\begin{bmatrix} X' \\ Y' \\ Z' \end{bmatrix} = \begin{bmatrix} -\sin \theta & \cos \theta & 0 \\ -\cos \theta \cdot \sin \varphi & -\sin \theta \cdot \sin \varphi & \cos \varphi \\ \cos \theta \cdot \cos \varphi & \sin \theta \cdot \cos \varphi & \sin \varphi \end{bmatrix} \cdot \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \quad (\text{付1})$$

$$\begin{bmatrix} x \\ y \end{bmatrix} = \frac{R}{R-Z'} \begin{bmatrix} X' \\ Y' \end{bmatrix} \quad (\text{付2})$$

但し、視点あるいは光源を極座標  $(R, \theta, \varphi)$  で与えたとき

ii) 三頂点  $P_i, P_{i+1}, P_j$  の右廻り判定 (面の表裏判定でも使用)

$$f_{i,j} = - \begin{vmatrix} x_i & y_i & 1 \\ x_{i+1} & y_{i+1} & 1 \\ x_j & y_j & 1 \end{vmatrix} \begin{cases} > 0 & P_i P_{i+1} \text{ の右手に } P_j \text{ が存在} \\ < 0 & P_i P_{i+1} \text{ の左手に } P_j \text{ が存在} \\ = 0 & \text{同一直線上に存在} \end{cases} \quad (\text{付3})$$

[参考文献]

- 1) P. P. Lourel: A solution to the hidden-line problem for computer drawn polyhedra, IEEE Trans. Computers, C-19 (March 1970).
- 2) 大森: 多面体の隠線消去問題, 情報処理, 12, 9, P534 (1971).
- 3) 中前・西田: 多面体の隠線消去の一手法, 情報処理, 13, 4, P239 (1972).
- 4) C. Wylie, G. Romney, D. Evans, R. Erdahl: Half-tone perspective drawings by computer, Fall Joint Computer Conference, (1967).
- 5) J. E. Warnock: A hidden surface algorithm for computer generated halftone pictures, Dep. Comput. Sci., Univ. of Utah, Tech. Rep., 4-15, (Jun. 1969).
- 6) A. Appel: On calculating the illusion of reality, Hardware, p. 945 (1969).
- 7) W. J. Bouknight: A Procedure for Generation of Three Dimensional Half-toned Computer Graphics Presentation, Comm. ACM, 13, 9, (Sep. 1970).
- 8) Schumaker, R. A.: Study for Applying Computer generated Images to Visual Simulation AFHRLTR-69-14 U.S. Air Force Human Resources Laboratory, (Sep. 1969).
- 9) Watkins: A Real-Time Visible Surface Algorithm, Dep. of Computer Science, Univ. of Utah, (June. 1970).
- 10) R. Goldstein, R. Nagel: 3-D visual simulation, Simulation, p. 25 (Jan. 1971).
- 11) H. Gouraud: Continuous shading of curved surfaces, IEEE Trans. Computers, p. 623 (June 1971).
- 12) Newell: A new Approach to the Shaded picture Problem, Proc. of the ACM, (1973).
- 13) D. P. Greenberg: Computer Graphics in architecture, Scientific American, (May. 1974).
- 14) F. C. Crow: Shaded Computer Graphics in the Entertainment industry, Computer Graphics, Vol. 11, No. 2, (Summer. 1977).
- 15) 西田・中前: 三次元物体の陰影表示の一手法, 情報処理, 15, 1, P. 18 (Jun. 1973).