

## 高速画像生成装置における画像生成ソフトウェア

秋本 高明      玉邑 嘉章      三ツ矢 英司

NTT 電気通信研究所

画像生成技術の実験を効率的に行うための研究環境の整備を主な目的として、高速画像生成装置の開発を進めている。本文では、本装置のための画像生成ソフトウェアについて、その開発方法、使用する画像生成手法及び具体的なインプリメントの仕方について報告する。まず、画像生成処理で頻繁に現れる演算及び処理を、高速な実行が可能な基本演算マクロ及び基本処理モジュールとして作成し、それを使用して画像生成ソフトウェアを組み立てる。また、有用な基本処理モジュールを蓄積することで、各種の画像生成手法を容易にインプリメントできる環境を形成する。

### A Software Configuration of a High-speed Image Generation System

Taka-aki AKIMOTO, Yosiaki TAMAMURA and Eiji MITSUYA

NTT Electorical Communications Laboratories  
(1-2356 Take Yokosuka-Shi Kanagawa 238-03 Japan)

We are developing a picture synthesizing system that has the ability to generate images very quickly. In this paper, we report on the image generation method used in this system, and also how to develop its software. Software must have flexibility and extensibility, because our purpose of developing this system is to study various image generation algorithms. Therefore, the image generation procedure is divided into modules and the software is constructed by collecting these modules.

## 1. まえがき

コンピュータを用いて、リアルな画像を生成する技術が盛んに使われている。そして、いろいろな形状・材質を持つ物体のリアルな画像が容易に生成できるようになりつつある。しかし、画像生成技術はまだ完全なものではなく、現在もなお多くの人々により研究がなされている。

筆者らも、画像生成技術の一つの研究課題である画像生成専用高速ハードウェアの検討と、画像生成技術の実験を効率的に行うための研究設備の整備をかねて、高速画像生成装置の開発を進めている<sup>1)</sup>。本画像生成装置は、浮動小数点演算の並列処理と高速データ転送が可能なプロセッサユニットを複数持つ、マルチプロセッサシステムである。本文では、本画像生成装置にインプリメントする画像生成ソフトウェアについて報告する。

## 2. ハードウェアの概要

図1に本高速画像生成装置の全体構成を示す。このように本装置は、装置全体を制御する制御プロセッサ(CP)と、複数のプロセッサユニット(PU)からなるマルチプロセッサシステムである。

PUは画像生成処理のほとんどを受け持つプロセッサで、そのブロック図を図2に示す。その中の演算部を図

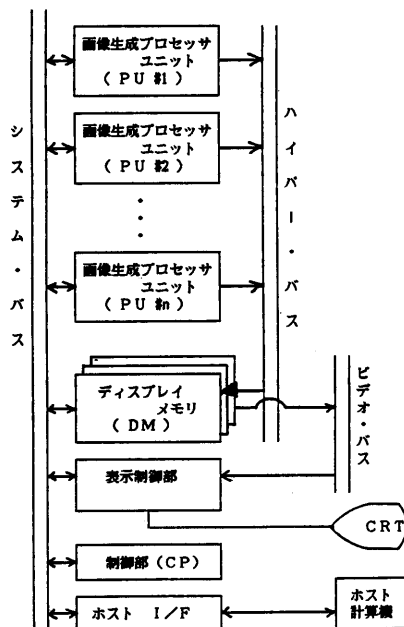


図1 画像生成装置の構成

3に示す。このように、PUは3つの高速浮動小数点演算器(ベクトル演算器:VFPP)と高速浮動小数点演算器と算術論理演算器の組(スカラー演算器:SP)により、3~4次元ベクトルの要素ごとの演算が1ステップで行える。またVFPP中の3つの演算器は互いにたすきがけ状のデータバスを持っており、レジスタを介することなくベクトルの積和演算などが実行できる。さらに、SPには関数演算用テーブル(LUT)が付加されており、逆数、平方根、三角関数を高速に計算する。

データメモリは4つのバンクに分かれており、4つの演算器とのデータ転送を並列に行える。

アドレス生成器は、32個のベースレジスタ(BR)・インデックスレジスタ(IR)・アドリビュートレジスタ(AR)と一つのアドレス計算器からなる。これは2次元アドレッシングモードを持っており、画像や表形式のデータを高速にアクセスできる。

PU全体は256ビットの水平型マイクロプログラムで制御される。これにより4つのFPPを独立に制御できる。

## 3. ソフトウェアの開発方法

このように、本画像生成装置を構成するPUのハードウェアの処理能力は非常に高い。しかし、多数の演算器と複雑なデータバス等のハードウェアの構造を考慮しつつプログラムを作成することは困難である。

一方、3次元の形状データを与えて、カラー陰影画像を生成する処理の中では、座標や3原色についての処理のために、3~4次元ベクトル及び行列の演算が非常に

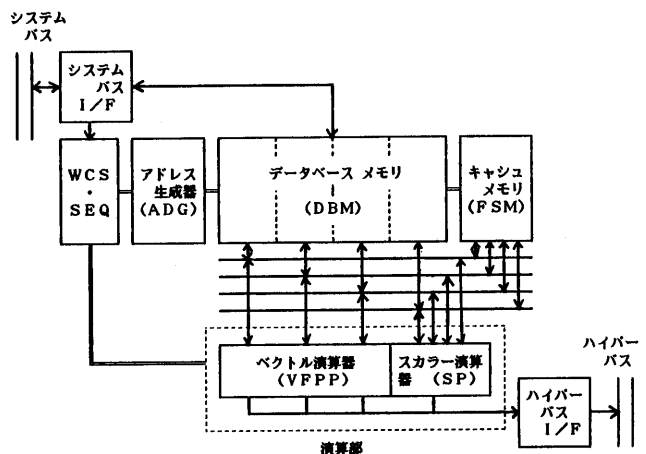


図2 画像生成プロセッサユニットの構成

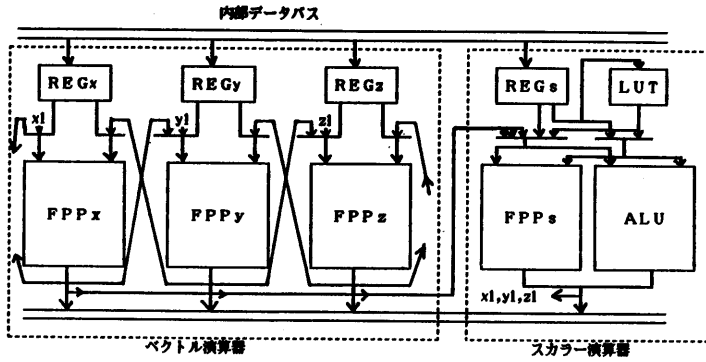


図3 演算部の構成

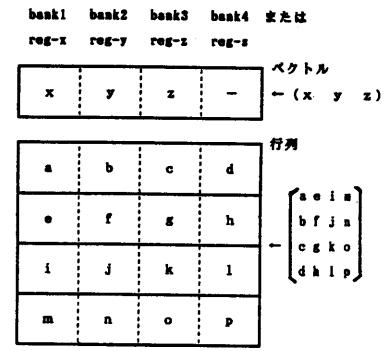


図4 ベクトル・行列データの置き方

多い。それらの頻繁に出てくる演算を非常に高速に行えば、かなりの処理の高速化が見込める。また、それらの個々の演算をプログラム中で容易に記述できるようにしておけば、そのような演算を使用する処理のためのプログラムを容易に作成することができる。

そこで、PUのマイクロコードを生成するためのアセンブラとして、ベクトルや行列演算命令を分かりやすく書くことのできるマクロ・アセンブラを作成する。そして、画像生成処理に必要な演算命令マクロを用意する。これを基本演算マクロと呼ぶ。

マクロ・アセンブラは読み込んだ命令をマイクロコードの各フィールドを直接記述しているメタ・アセンブラの命令に変換する。マクロ・アセンブラの命令は、一般の計算機の命令も含む。

さらに、画像生成処理でよく現れる部分を切り出して上述したマクロ・アセンブラやメタ・アセンブラで記述し、モジュールとして蓄積しておけば、各種の画像生成処理でそれを流用することができ、ソフトウェアの作成が容易になる。これを基本処理モジュールと呼ぶ。

この様にしてソフトウェアを作成することにより、各種の画像生成手法を容易に本装置にインプリメントすることができ、かつそのソフトウェアはハードウェアをある程度有効に使った高速なものにすることができる。

4. 基本演算マクロ

表1に基本演算マクロとその実行ステップ数を示す。ただし、データはレジスタに置き、演算結果もレジスタに書き込む場合の実行ステップ数である。

スカラー演算マクロと関数演算マクロは主にSPで実行される。ベクトル演算マクロと行列演算マクロはVFPとSPの両方で実行される。演算を効率よくするた

	REG1	REG2	REG3	REG4	
R1	x	y	z	-	ベクトル
R2	x'	y'	z'	-	正規化後のベクトル

STEP	FPPx	FPPy	FPPz	FPPs	
1	X=X	Y=Y	Z=Z	-	R1=R1
2	X=X+Y	-	-	Z=Z	加算
3	-	-	-	X=X+Y+Z	加算
4-12	-	-	-	-	平方根
13-17	-	-	-	-	逆数→S
18	X=S	Y=S	Z=S	-	乗算、R2へ

図5 ベクトルの正規化の実行のようす

表1 基本演算マクロ

分類	演算	実行ステップ数
スカラー演算マクロ	加算	1
	減算	1
	乗算	1
	除算	6
関数演算マクロ	平方根	9
	逆数	5
	SIN	15
ベクトル演算マクロ	要素ごとの和	1
	要素ごとの差	1
	要素ごとの積	1
	要素ごとの商	16
	ベクトルとスカラーの積	1
	ベクトルとスカラーの商	6
	ベクトルの内積	3
	ベクトルのノルム	12
ベクトルの正規化 (大きさを1にする)	18	
行列演算マクロ	ベクトル×行列	12
	行列×行列	64

めにベクトルと行列はメモリ又はレジスタに図4のように置かれる。例として、ベクトルの正規化の実行の様子を図5に示す。

## 5. 画像生成手法

一般的な画像生成手法としては、光線追跡法、スキャンライン法、Zバッファ法、優先順位法等がある<sup>2)</sup>。基本処理モジュールの選定と本装置の評価のため、以下の理由でこの中から光線追跡法とスキャンライン法を選び、インプリメントを進めることとした。

光線追跡法<sup>3)</sup>は、画面のある画素を通り視点に届いた光線の強度を、視点から逆に追跡して求めてその画素の輝度を計算し画像を生成する方法である。この方法は鏡面や透明物体により反射屈折される光線を追跡することで、鏡面や透明物体を正確に表示できる。しかし画像の生成に非常に長い時間がかかるので、高速化手法の研究や専用ハードウェアの開発が盛んに行われている。本画像生成装置も光線追跡法の高速処理が一つの目的である。

スキャンライン法、Zバッファ法、優先順位法は、共に陰面消去と透視変換により画面上のどの位置にどの物体(面)を表示するか決めて、その部分をその物体(面)の色を使って陰影付けすることにより画像を生成する方法である。鏡面や透明物体の表示は疑似的にしかできないが、光線の追跡のような時間のかかる処理をしないので光線追跡法より極めて高速である。この中でZバッファ法と優先順位法は重ね書きにより陰面消去を行うので、高速塗りつぶし機構を持っている装置には都合がよいが、本装置ではディスプレイメモリへの書き込みがボトルネックになる可能性がある。そこで、スキャンライン法をインプリメントし、画像生成の応答時間が重視される用途に使用することにした。

## 6. 光線追跡法を用いた画像生成ソフトウェア

光線追跡法の詳細は各種文献に述べられている。ここでは、本画像生成装置へのインプリメントの仕方について具体的に述べる。

### 6.1 データ構造

光線追跡法に必要な主なデータはシーンを構成している物体のデータである。シーンは図6のようにツリー状に定義する。プリミティブは形状の基本要素で、現在は平面と2次曲面を使用する。各物体の形状はプリミティブの集合演算により定義する。物体群はいくつかの物体

(群)の集まりで、複数の物体をまとめた方が都合がよいとき(通常一緒に移動するような物体)にそれらをまとめて物体群とする。各物体(群)はそれぞれ自分のローカル座標系で定義されている。それが上位の物体群の座標系の中のどこに配置されるかはそれに付加された変換行列で指示される。この変換行列を変化させることで物体の集まりを回転・移動させることができる。さらに、物体(群)にはそれを取り囲む球状のエクステントを設定して階層的なエクステントを構成し、交点計算回数の低減を図る。

実際のデータ構造を図7に示す。このように、物体(群)のデータをポインタで結ぶことにより木構造を表す。プリミティブ、エクステント、変換行列及び属性(色・反射率等)は別の場所にまとめて置き、物体(群)データ又はプリミティブデータからのポインタにより参照する。これらのデータは、処理が効率的に行えるようにメモリに図8のようにおかれる。物体データはそれを構成しているプリミティブへのポインタと、集合演算による形状定義式を含む。この形状定義式は容易に評価できるようにポーランド形式で表す。なお画像を生成する前に、シーンを定義する木構造の各ノードの変換行列をプリミティブごとに合成し、合成された行列でプリミティブを変換してしまうことにより、座標変換しながら交点計算することを避ける。

### 6.2 全体の処理の流れと基本処理モジュール

光線追跡法による画像生成処理の流れを図9に示す。基本処理モジュールの切り出しは、例えば改良された新しいジェーディングモデルへの変更がいかにか容易にできるか、プリミティブの種類の追加が容易か、少しの修正だけで新しい高速化手法を組み込むことができるかなどソフトウェアの柔軟性と拡張性を考慮して、次に示す処理を基本処理モジュールとした。

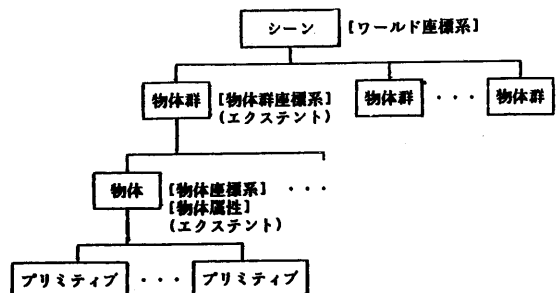


図6 シーンの定義のようす

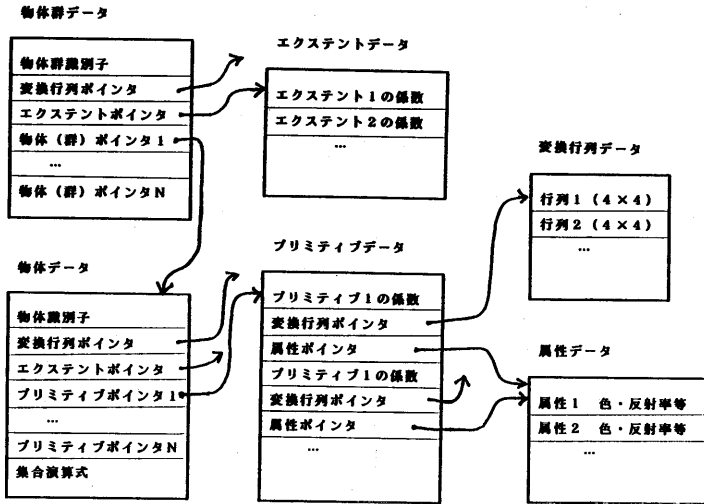


図7 光線追跡法における物体データの構造

bank1	bank2	bank3	bank4
平面	属性	変換行列	-
識別子	ポイント	ポイント	
$n_x$	$n_y$	$n_z$	$d$

$$n_x X + n_y Y + n_z Z + d = 0$$

(a) 平面プリミティブ

bank1	bank2	bank3	bank4
2次曲面	属性	変換行列	-
識別子	ポイント	ポイント	
$p_x$	$p_y$	$p_z$	$r$
$q_{11}$	$q_{21}$	$q_{31}$	-
$q_{12}$	$q_{22}$	$q_{32}$	-
$q_{13}$	$q_{23}$	$q_{33}$	-

$$\begin{pmatrix} X-p_x & Y-p_y & Z-p_z \end{pmatrix} \begin{pmatrix} q_{11} & q_{12} & q_{13} \\ q_{21} & q_{22} & q_{23} \\ q_{31} & q_{32} & q_{33} \end{pmatrix} = r$$

(b) 2次曲面プリミティブ

bank1	bank2	bank3	bank4
色-赤	色-緑	色-青	-
$I_a$	$K_d$	$K_s$	$n$
$K_r$	$K_t$	屈折率	-
赤透過率	緑透過率	青透過率	-

$$I_a \sim K_t : \text{反射率等}$$

(c) 属性データ

図8 プリミティブデータと属性データのメモリ中での配置

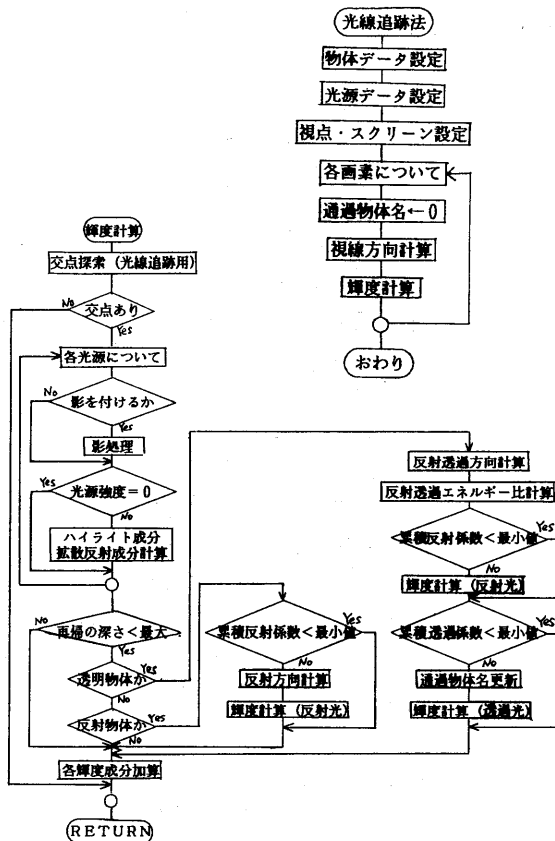


図9 光線追跡法による画像生成処理

基本処理モジュール一覧 (主に光線追跡法用)

- (1) 物体データ初期化
- (2) 視線方向計算
- (3) 拡散反射・ハイライト成分計算
- (4) 反射方向計算
- (5) 反射透過方向計算
- (6) 反射透過光エネルギー比計算
- (7) 輝度成分加算
- (8) エクステント交差判定
- (9) 平面交点計算
- (10) 2次曲面交点計算
- (11) 平面IN/OUT判定
- (12) 2次曲面IN/OUT判定
- (13) 交点有効性判定
- (14) 交点探索 (光線追跡用)
- (15) 交点探索 (影処理用)

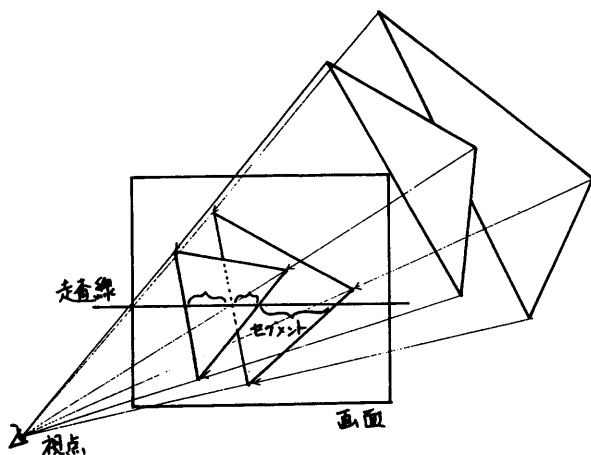


図 10 スキャンライン法

このようなモジュールに分けることにより、新しいプリミティブの追加はそのプリミティブに対する交点計算モジュールとIN/OUT判定モジュールを追加し、交点の有効性判定モジュールと交点探索モジュールの若干の修正だけで対処できる。また、シェーディングモデルの変更は拡散反射・ハイライト成分計算モジュールのみを取り替えるか修正することで対処できる。テクスチャマッピングを取り入れるときは、交点座標に基づいてマッピング用画像からマッピングデータを取り出すか、ある手続きによりそれを作り出すモジュールを作成し、それから得られたデータを拡散反射・ハイライト成分計算モジュールなどへ渡すことで対処できる。

## 7. スキャンライン法を用いた画像生成ソフトウェア

### 7.1 スキャンライン法の概要

スキャンライン法は、物体形状を多角形の集まりで表現しておき、その多角形の辺を画面に透視変換し、投影された画面上の辺と走査線（スキャンライン）との交点を求め、各交点間でその区間に表示すべき多角形を決めて、その多角形の色でその区間を塗ることにより画像を生成する方法である<sup>2)</sup>。ここでは、多角形はすべて三角形として物体データの構造を単純にする。図10にスキャンライン法による画像生成の様子を示す。スキャンライン法は、各種のコヒーレンスをうまく利用した高速な画像生成手法である。

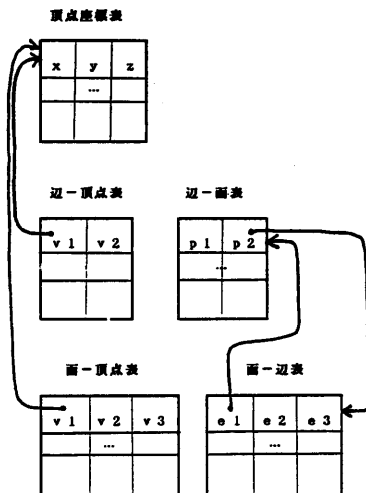


図 11 スキャンライン法における物体データの構造

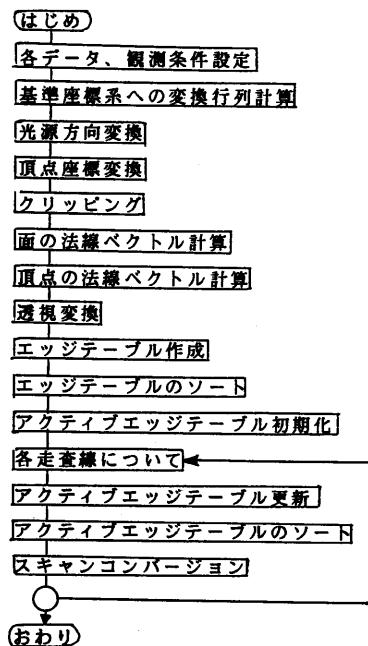


図 12 スキャンライン法による画像生成処理

## 7. 2 データ構造

スキャンライン法で使用する物体データを図11に示す。頂点座標と属性データは、それに関する処理が高速に行えるようにベクトルデータの置き方または光線追跡法の属性データの置き方と同じようにメモリ中に配置する。

## 7. 3 全体の処理の流れと基本処理モジュール

図12にスキャンライン法による画像生成処理の全体の流れを示す。スキャンライン法は古くから研究されかなり完成された画像生成手法である。そこでプログラムの作成のしやすさと処理の高速性を考慮して、モジュール内の処理がなるべく単純になるようモジュールを切り出した。以下にスキャンライン法のための基本処理モジュールを示す。

### 基本処理モジュール一覧(主にスキャンライン法用)

- (1) 基準座標系への変換行列計算
- (2) 光源方向変換
- (3) 頂点座標変換
- (4) クリッピング
- (5) 面の法線ベクトル計算
- (6) 頂点の法線ベクトル計算
- (7) 透視変換
- (8) エッジテーブル作成
- (9) エッジテーブルのソート
- (10) アクティブエッジテーブルの初期化
- (11) アクティブエッジテーブルの更新
- (12) アクティブエッジテーブルのソート
- (13) 輝度計算

スキャンライン法はZバッファ法のような重ねがきにより陰面消去を行う方法に比べ処理が複雑である。しかし、上に示したモジュールはほとんど単純な処理であり高速なプログラムを容易に作成することができ、これらのモジュールをブラックボックスとして使用することにより、複雑な部分のソフトウェアの作成が助けられる。

## 8. むすび

以上、開発を進めている画像生成装置のソフトウェアについて概要を述べた。画像生成処理で頻繁に現れる演算及び処理を、高速に実行する基本演算マクロ及び基本処理モジュールとして蓄積することで、各種の画像生成手法を容易にインプリメントでき、そのソフトウェアは局所的に高速にすることができる。

現在、本装置のソフトウェア開発環境の整備、および本文で述べたソフトウェアのインプリメントを進めている段階である。基本演算やモジュール類の実行時間等の評価については、別途報告することとしたい。

ここで述べた画像生成ソフトウェアは、画像が生成できる最低限の処理しか含めていない。今後、各種の処理アルゴリズムや高速化手法の基本となるモジュールを蓄積してゆく予定である。

[謝辞] 日頃ご指導頂く画像通信研究部画像処理研究室小杉信室長、知的画像処理研究グループ末永康仁リーダーに感謝致します。また、討論頂いた間瀬健二研究主任ならびに画像処理研究室、知的画像処理研究グループの皆様へ感謝致します。

### [文献]

- 1) 玉邑,三ツ矢,秋本,"高速画像成処理装置の基本構成",TV学会画像処理・画像応用研究会,IPA-86,(1986.10)
- 2) J.D.FOLEY,A.VAN DAM,"Fundamentals of Interactive Computer Graphics",Addison-Wesley Publishing(1982)
- 3) T.Whitted,"An Improved Illumination Model for Shaded Display",C.ACM, vol.23,no.6,(1980)