

## 注視点に依存するポリゴンモデルの適応的表示手法 Adaptive Focus-Dependent Display Method of Polygon Models

松本 哲也† Mitsunori Makino  
Tetsuya Matsumoto

### 1 はじめに

近年、没入型立体視システムの利用分野が広がり、表現対象となる物体形状の複雑化が進んでいる。物体形状が複雑になるほど、表示に要する処理時間が増大し、没入感を損ねる恐れが増大する。今後の没入型立体視システムの普及のためにも、没入型立体視システムでの物体表示に要する処理時間のより短縮が求められる。これまで、物体の詳細度を適応的に簡略化することで、見かけ上の精度を維持し、かつ、高速表示を実現する手法の研究が一般的な PC を対象にされてきた[1][2]。本研究では、没入型立体視システムでの利用を考慮した物体の高速表示を実現する手法を提案する。没入型立体視システムでは、複数のディスプレイによって使用者が囲まれるため、通常の PC に比べてディスプレイの表示対象となる範囲が広がる。そのため、物体がディスプレイの表示対象となるかの判定では簡略化が難しい。そこで、使用者の視野における重要性を考慮した適応的表示を行う。

提案手法では、視野における物体の位置から、人間の視野特性を利用して表示詳細度を決定する。この詳細度に応じて物体を簡略化することで、注視点付近は詳細に、周辺では簡略に物体を表示する。これにより使用者にとっては重要な（注視している）物体の詳細度を維持し、かつ、表示対象物体数を低減して高速表示を実現する。本手法を没入型立体視システム CAVE[3]で実装し、ポリゴンで定義された物体を対象に有効性を示す。

### 2 提案手法

#### 2.1 視野特性

人間の視野は、およそ上方  $60^\circ$ 、下方  $70^\circ$ 、外側  $100^\circ$  の楕円形をしており、中心視野と周辺視野にわけることができる。中心視野とは黄斑部中心窩を中心とする  $30^\circ$  までの視野、周辺視野とは中心  $30^\circ$  以外の周辺の視野である。視野内の物体の鮮明度は中心視野では高く、周辺部ほど低い[4]。

#### 2.2 視野における物体位置の判定と詳細度

視野における物体の位置により詳細度を決定する。物体が中心視野・周辺視野・視野外のどの範囲に属するか判定する。物体位置を、左右方向の角度  $\alpha^\circ$  ( $0 \leq \alpha \leq 180$ )、上下方向の角度  $\beta^\circ$  ( $-90 \leq \beta \leq 90$ )、注視点から物体までの距離を  $D$  とする (図 1)。

物体  $m$  の位置の上下方向の角度を  $m_\alpha$ 、左右方向の角度を  $m_\beta$  とする。 $\alpha$  を横軸、 $\beta$  を縦軸にとり、視野特性をもとに作成した図 2 上の座標  $m(m_\alpha, m_\beta)$  の位置により、中心視野・周辺視野・視野外のいずれに属するかを判定する。ここで、原点から周辺視野と視野外の境界までの距

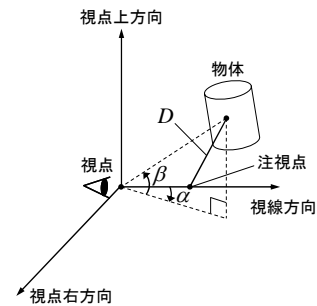


図 1: 物体位置

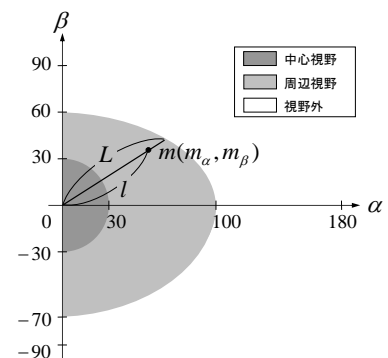


図 2: 視野分類の判定

離を  $L$ 、原点から点  $m$  までの距離を  $l$  とする。

中心視野内にある物体は重要性が高いので詳細にする。周辺視野内にある物体は周辺部に近いほど簡略化する。視野外にある物体は重要性が低いので更に簡略化する。また、注視点と物体の距離  $D$  に応じて物体を簡略化する。

#### 2.3 ポリゴンモデルの適応的表示

ポリゴンで定義された物体を対象にした場合の詳細度の適応的決定法を示す。中心視野内での物体のポリゴン数を  $P_1$ 、周辺視野内でのポリゴン数を  $P_2$ 、周辺視野内での最小ポリゴン数を  $P_2'$ 、視野外でのポリゴン数を  $P_3$  とする。周辺視野内のポリゴン数  $P_2$  は、物体が周辺部に近づくにつれ、線形的に減少させる (図 3)。  $P_2$  は式 (1) で決定する。

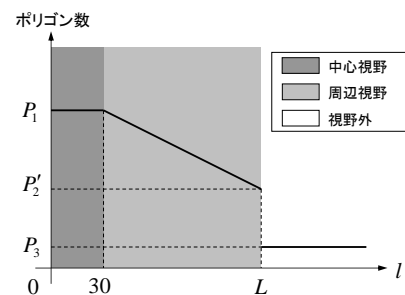


図 3: 各視野のポリゴン数

$$P_2 = P_1 - (P_1 - P'_2) \left( \frac{l-30}{L-30} \right) \quad (1)$$

更に、中心視野内と周辺視野内にある物体は、注視点から物体までの距離  $D$  に応じて、ポリゴン数を線形的に減少させる。各視野での物体  $m$  の最終的なポリゴン数  $P_m$  は式 (2) で決定する。ここで、 $D_\infty$  は注視点から十分に遠方にある点までの距離とする。

$$P_m = \begin{cases} P_1 \cdot \left( 1 - \frac{D}{D_\infty} \right) & (\text{中心視野}) \\ P_2 \cdot \left( 1 - \frac{D}{D_\infty} \right) & (\text{周辺視野}) \\ P_3 & (\text{視野外}) \end{cases} \quad (2)$$

### 3 シミュレーション

没入型立体視システム CAVE を用いて、表 1 の環境でシミュレーションを行った。仮想空間内に Stanford Bunny (ポリゴン数 69451) [5] を 12 個 (合計ポリゴン数 833412) 配置し、視点や視線方向の移動を行い、表示処理時間を 1000 回測定し、最大値、平均値、最小値を求めた (表 3)。メッシュ簡略化には、Garland らの手法 [6] を用いた。表 2 に各パラメータ設定値を示す。また、実装時の正面 (視線方向) を撮影した画像を図 4 に示す。

表 1: シミュレーション環境

PC	hp workstation x4000 3 台
CPU	Intel Xeon 2.40GHz×2
メモリ	2GB (PC800 ECC RDRAM)
ビデオカード	3Dlab Wildcat III 6210
OS	Microsoft Windows 2000 Professional

表 2: パラメータの設定

$P_1$	69451 (オリジナルのポリゴン数)
$P'_2$	20835 ( $P_1 \times 0.3$ )
$P_3$	3472 ( $P_1 \times 0.05$ )

表 3: 処理時間

		提案手法あり	提案手法なし
平均ポリゴン数		171684	833412
位置判定・ 形状再定義 [s]	最大	0.172	0
	平均	0.025	0
	最小	0.015	0
描画処理 [s]	最大	0.219	0.344
	平均	0.095	0.330
	最小	0.031	0.312
合計 [s]	最大	0.266	0.344
	平均	0.119	0.330
	最小	0.062	0.312

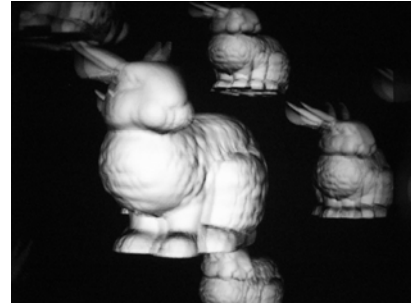


図 4: 実装時の画像

表 3 より、提案手法を用いた場合の表示処理の最大時間は、用いない場合の最小時間よりも小さいことがわかる。これより、表示処理時間の短縮のために、提案手法が有効性を頑健に示せた。また、今回は  $P_1$  を物体のオリジナルのポリゴン数と等しく設定したので、提案手法を用いない場合と比較しても見かけ上の差はほとんど感じられない。今回ほどには見かけを重視しなければ、各パラメータの設定を変えることで、表示処理を更に高速化可能である。

### むすび

本論文では、人間の視野特性を利用し、没入型立体視システムでの利用も考慮した物体の高速表示を実現する手法を提案した。CAVE システムにおいて、ポリゴンモデルを対象にし、見かけ上の精度を維持したまま表示処理時間を短縮でき、本手法の有効性を示した。

今後の課題として以下が挙げられる。

- 視点・視線方向の移動の頻度も考慮する。
- 他の適応的表示手法との組み合わせる。
- ポリゴンモデル以外のボリュームデータなどで表現された物体を対象とする。

### 謝辞

本研究の一部は中央大学理工学研究所共同研究、中央大学特定課題研究の補助を受けた。

### 参考文献

- [1] J. Jang, W. Ribarsky, C. Shaw and P. Wonka, "Appearance-Preserving View-Dependent Visualization", IEEE Visualization 2003, pp. 473-480, October 2003.
- [2] H. Hoppe, "View-Dependent Refinement of Progressive Meshes", ACM SIGGRAPH 97, pp. 189-198, 1997.
- [3] C. Cruz-Neira, D.J. Sandin, T.A. DeFanti, R.V. Kenyon and J.C. Hart, "The CAVE: Audio Visual Experience Automatic Virtual Environment", Communications of the ACM, Vol. 35, No. 6, pp. 65-72, June, 1992.
- [4] 大野重昭, 澤充, 木下茂編, "標準眼科学", 医学書院, 2004.
- [5] The Stanford 3D Scanning Repository, <http://graphics.stanford.edu/data/3Dscanrep/>.
- [6] M. Garland, P.S. Heckbert, "Surface Simplification Using Quadric Error Metrics", ACM SIGGRAPH 97, pp. 209-216, 1997.