

複数のディスプレイを用いた3次元図形表示システム

— マルチディスプレイ —

2P-6

平池 龍一 篠原 克也

(日本電気(株) C&Cシステム研究所)

1. はじめに

近年、設計/シミュレーションをはじめとする様々な分野でCGが利用されている。特に、設計という創造的作業を支援する場合には、同時に多量の図形データを表示できる設計環境が望まれる。大画面ディスプレイは、このような設計環境を実現するために非常に有効である。

しかし、会議などで利用されている大型スクリーン用プロジェクトでは、輝度/コントラストの点でCRTディスプレイに劣る。逆にCRTディスプレイでは、安全性の面からも大画面化が困難である。また、メモリアクセス速度の点から両者共に高解像度化には限界がある。

そこで我々は、大画面と高解像度を同時に実現する3次元図形表示システムとして、CRTをマトリックス状に配置するマルチディスプレイ方式を提案する。本稿では、マルチディスプレイ表示システムの構成と表示図形の分散処理方式について述べた後、EWSとイーサネット構成した実験システムによる実行結果を示す。

2. マルチディスプレイ表示システム

2.1 システム構成

図2.1にマルチディスプレイ方式に基づく表示システムの基本的なシステム構成を示す。

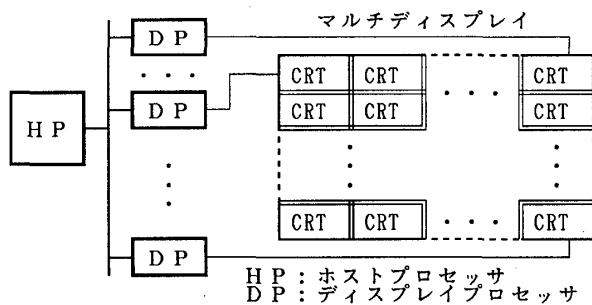


図2.1 基本システム構成

本システムは、マトリックス状に配置したN台のディスプレイと、それらを個々に制御するN台のプロセッサ(DP)、およびこれらのプロセッサを管理する1台のプロセッサ(HP)で構成される、2レベルのマルチプロセッサシステムである。

本システムでは、単に大画面システムとしてだけでなく、縦長/横長など目的に応じて任意のディスプレイ構成を実現可能である。

2.2 処理方式

3次元図形表示処理は、多角形の各頂点を処理対象とする前段階(座標変換、輝度計算、クリッピング)と、多角形内部の各点を処理対象とする後段階(隠面消去、シェーディング)とからなる。本システムでは、ディスプレイ単位で表示領域が分割されている点に着目して、前段階に比べて圧倒的に処理量が多い後段階における処理を、各ディスプレイに対応する複数のDPで並列に行う。また、前段階の処理をHPで行う。(図2.2参照)

HPは、処理後の多角形データをそれが表示されるディスプレイを判定して該当するDPへ転送する。DPは、受理したデータから順次、Zバッファ法による隠面消去処理、グローシェーディング処理を施していく。図2.2に示すように、1つの多角形が複数のディスプレイにまたがって表示される場合には、クリッピング処理で多角形を分割した後、各々の部分多角形に対して必要な座標変換を施して当該DPへ転送する。本システムでは、DP-DP間で並列処理を行っていると同時に、HP-DP間でパイプライン処理を実現している。

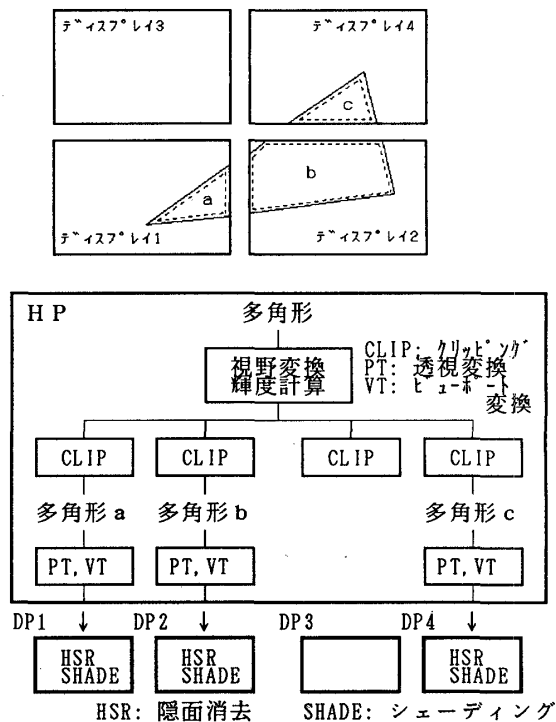


図2.2 多角形データの処理

3. 実験システムによる評価

3.1 システム構成

マルチディスプレイ方式を評価するための実験システムとして、HP 1台とDP 4台、2×2構成の表示専用ディスプレイ（写真3.1）、およびHPに接続した制御用ディスプレイからなるシステムを構成した。プロセッサには NEC 製の EWS4800(7MIPS) を、またプロセッサ間の通信路にはイーサネットを用いた。プロセッサ間の通信は、UNIXのソケット機構を用いたプロセス間通信として実現した。

入力データは、3次元形状の表面を分割して得られた多角形データ（三角形あるいは四角形）である。具体的には、多角形の各頂点における座標値および輝度値からなる可変長データである。図形表示は、バッファクリア命令、複数の多角形描画命令、イメージ転送命令の各データをHPからDPへ転送することで行う。最大データサイズは、四角形がクリッピング時に視野ピラミッドのすべての構成面と交差して生成された8角形分のデータをもつ多角形描画命令のときに得られる（ $130\pi^2$ ）。

3.2 実行結果

960個の多角形（三角形／四角形）で構成された3次元形状を、 $1280 \times 1024 \times 4$ （ 2560×2048 ）画素の画面に表示する場合（写真3.1）の実行時間を測定した。

この結果、全体の実行時間（全プロセッサにおける処理が終了するまでの時間）は32.0秒であり、またHP／DPの実行時間は各々26.8秒／15.4秒であることから、処理のオーバーラップにより約24%の時間短縮を実現していることがわかる。

単一プロセッサによる処理

同画素数の画像を単一プロセッサで生成した場合の実行時間は160.6秒で、4台のプロセッサで生成した場合の約5倍の時間を要しており、マルチプロセッサによる並列処理の効果が顕著である。また、1ディスプレイ分の画像生成に必要な時間は20.9秒であり、4倍以上の時間を要している。これは、小容量の主記憶上に大画面分のZバッファやカラーバッファを確保したために、ディ

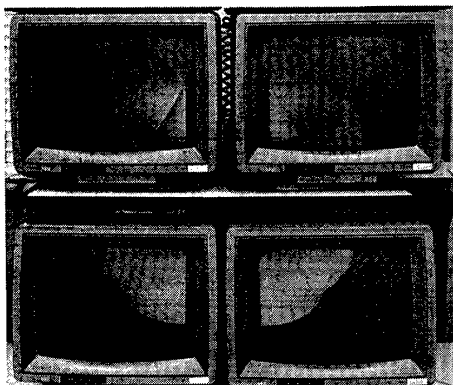


写真3.1 実験システム（マルチディスプレイ）

スクアクセスが頻繁に発生したからである。

静的負荷分散による高速化

3次元図形として隣接する多角形は、表示の際にも隣接するため、マルチディスプレイにおいて同一のディスプレイに表示される可能性が高い。一般に隣接多角形は、データ構造上でも隣接しており、時間的に隣接して処理されるため、特定のDPに負荷が集中する。このため、あるDPのみが稼働中であり他のDPが待ち状態となるのみでなく、繁忙DPがデータを受理しないためにHPも待ち状態に入り、並列処理の著しい効率低下を招く。そこで、HPでは定義空間内で隣接していない多角形を順番に処理してDPに転送することで高速化を図った。その結果、HPでの実行時間が20.0秒に短縮され全体の実行時間は25.2秒となり、20%以上の速度向上が得られた。本実験システムでは、配列中の多角形データを一定間隔でアクセスすることで実現しており、負荷分散による処理の増加は無視できる程度である。

データバッファによる高速化

DPはHPに比べて処理負荷が大きいので、HPは転送済みの多角形の処理をDPが終了するのを待っており、並列処理の効率低下を招いている。そこで、各DPにデータバッファ（ $4096\pi^2$ ）を設けて、HPが待ち状態にある時間を短縮した。転送データは大部分が三角形／四角形の多角形描画命令であり、平均データサイズを $64\pi^2$ として64多角形分の容量である。測定の結果、HPでの実行時間が11.0秒に短縮され全体の実行時間は18.6秒となり、さらに25%以上の高速化を実現できた。

本システムでは上述の高速化を図ることにより、単一プロセッサで1ディスプレイ分の画像を生成するのと同程度の時間で、4ディスプレイ分の画像を生成することができた。図3.1に実行時間の測定結果のまとめを示す。

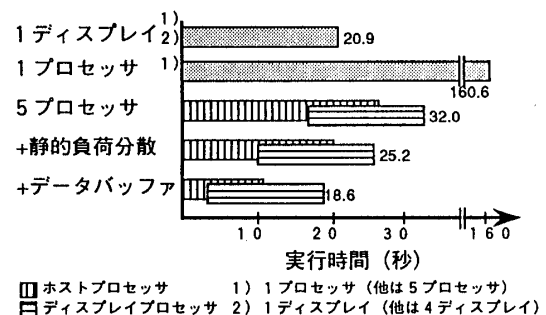


図3.1 実行時間

4. おわりに

現在、クリッピング処理の高速化を図っている。今後は、景観シミュレーションなどで要求される高品質画像生成のために、付影処理や他のシェーディング手法の組み込みについても検討していく予定である。