

〔非公開〕

TR-C-0076

運動視における表示遅れの影響

佐藤 隆宣  
Takanori SATO

伴野 明  
Akira TOMONO

1 9 9 2 . 5 . 2 5

A T R 通信システム研究所

# 運動視における表示遅れの影響

佐藤 隆宣

伴野 明

# 目次

概要	.....	3
まえがき	.....	4
運動視表示システムの構成	.....	4
画像生成の並列処理	.....	5
運動視における表示遅れの評価		
表示遅れによる影響	.....	6
タスク実験	.....	7
画像更新レートによる影響	.....	8
高速画像生成法		
頂点数と処理時間	.....	10
視点を利用した頂点削除	.....	10
視線を利用した頂点削除	.....	13
むすび	.....	14
参考文献	.....	14
付録	.....	15
実験用ソフトウェア	.....	16

## 概要

コンピュータ・グラフィックスを用いて、運動視を実現する場合、利用者の視点位置を検出して、そこから見える透視投影画像を表示する。その場合、視点位置を検出してから表示するまでの遅れ時間が画像へ及ぼす影響について述べている。また、画像劣化を与えない遅れ時間を求め、その許容時間内で高品質な画像を生成する手法について提案している。表示遅れの評価は、視点位置を磁気センサで検出し70インチプロジェクタに表示して実験を行なった。実験項目は、(1)視点移動開始、及び、停止時の画像変化の表示遅れ、(2)表示遅れが運動視を用いたタスクの作業量に及ぼす影響、(3)視点移動中の画像変化の滑らかさと画像更新レートの3点である。その結果、利用者に遅れの影響を与えないためには、表示遅れが100msec以下、画像更新レートが90Hzであることが望ましい。

高速な画像生成では、視点と物体の距離、又は、注視点からの偏角によって精細度が階層的に変化する3次元モデルを用いる。その結果、計算量を増すことなく高品質と感じる画像表示が可能となることを明らかにした。

The goal of our research is teleconferencing with realistic sensation. Generating images with motion parallax is one of the key technologies which will make it possible to further human to human communication. To generate images with realism by CG, the number of vertexes of an object should be increased. However, to generate images with motion parallax, the images should be generated quickly. So, there is a tradeoff between generating highly realistic images and generating them quickly. In this paper, we evaluate the effect of delay time according to three parameters: response time of generating images for head movement and frequency of generating them, and amount of task in virtual space. We also present a method to generate CG images with realism at high speed. The method is to select from a database of different detailed models, one model based on the distance between the viewpoint and object, or between gazing point and object.

## 1. まえがき

ATRでは、仮想的な3次元空間（仮想空間と呼ぶ）をCGによって作り出し、実際には遠隔地に居る人々があたかもその仮想空間の中に会したかのような感覚で協調作業することができる臨場感通信会議システムの構想を提案している[1]。このシステムでは、利用者の居る空間と表示装置画面の中の空間が連続して、かつ自然に知覚されるようにすることが本質的に重要である。このためには、人が3次元空間を知覚する心理的な要因をできるだけ多く取り入れること、具体的には、広視野（大画面）・高精細で、運動視、立体視を実現することが必要と考えている[2][3]。このうち、視点の動きに応じた画像を実時間で生成する運動視表示は、利用者が空間を知覚するために積極的な観察姿勢をとることを可能にするため効果的である[4]。

CGを用いて運動視を実現するためには（以下、運動視CGと呼ぶ）、利用者の視点位置を常時検出し、その位置から見た3次元世界モデルをスクリーンに投影する、いわゆる透視投影変換による画像生成表示が必要である。この手法では、視点位置の検出、モデルの変形処理等に時間が必要なため、表示遅れが発生する。表示モデルの精細度が増すと画像生成処理時間が増える。大画面・高精細にしようとする表示モデルの頂点数は増さなければならず、遅れは、急激に増加する傾向にある。しかし、表示遅れが大きくなると、運動視は不自然なものになるため、運動視の質を上げる観点からは遅延時間を短くすることが必要である。

このように現状の運動視CG技術では、実時間性を確保しつつ、大画面・高精細化を達成することは困難な状況にあり、新しいアルゴリズムが望まれている。

ところで、運動視CGの表示遅れを決定するパラメータとしては、以下の2つが考えられる。

a) 視点の移動開始、停止、方向変換時に知覚され易い画像変化の表示遅れ（以下、画像の表示遅れと呼ぶ）

b) 視点移動中の画像変化の滑らかさを決定する画像の更新レート（画像更新レートと呼ぶ）

これまで、a)については、比較的小さな画面を用い、視点移動範囲が狭く、視点から表示した物体までの距離が近い場合については、調べられているが[5]、大画面で運動視CGを前提とした場合の画像の表示遅れについては、明らかにされていない。また、b)については、視点を固定してディスプレイ上の移動物体を見る場合の滑らかな動きに必要な画像更新レートについての報告はあるが[6][7]、視点を随意的に移動させ

た場合の画像更新レートは調べられていない。

このように、運動視CGにおいて、許容される遅延時間については、定量的なデータが少ない。

そこで、本稿では、先ず、上記2種類の遅延パラメータが画質に与える影響について調べ、表示遅れの許容時間を設定し、次にその許容時間内で高速に画像を生成する方法について検討する。2. では、運動視CGを実現するため装置構成について述べる。視点の検出には、磁気センサを用い、画像生成には、汎用のグラフィックス・ワークステーションを用いた。3. では、運動視における表示遅れについて解説し、4. では、表示遅れの影響と許容される遅延時間について実験し考察する。a) の表示遅れについては、頭部の移動方向に着目した実験と運動視を用いたタスク実験を行った。b) の画像生成周期の影響では、頭部移動速度、視点と物体間の距離に着目した。4. では、計算量を増やさずに精細と感じられる新しい画像生成アルゴリズムを提案し、その有効性を明らかにする。本手法は、視点の位置情報、または視線情報を利用して精細度を階層化された表示対象の3次元モデルが用いられる。

## 2. 運動視表示システムの構成

図1に本実験で用いた運動視表示システムの構成を示す。表示画像の生成方法について以下に述べる。

(1) 利用者が相対するスクリーンの前後に仮想的な

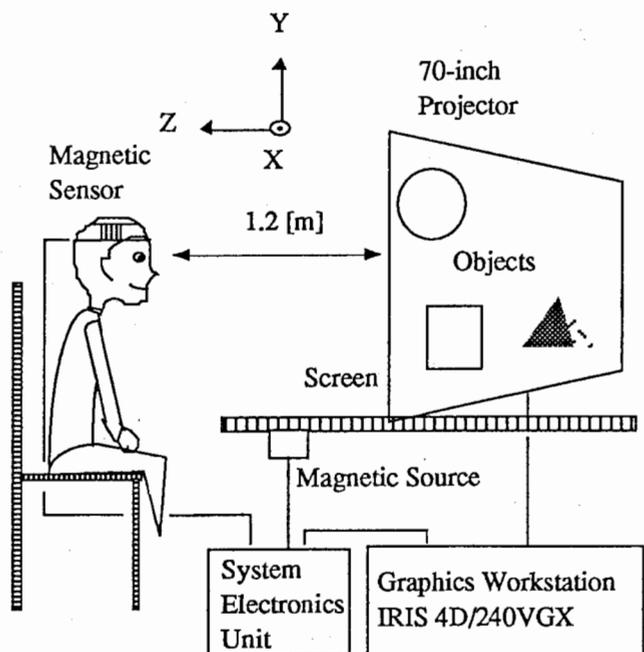


図1 運動視表示システム

System generating images with motion parallax

3次元世界（仮想空間）を考える。スクリーン及び仮想空間の各点は、3次元基準座標系（以下ワールド座標系）で記述する。

(2) ワールド座標系で観察者の視点を求める。視点は、厳密には、眼球の節点に相当するが、ここでは、簡単のため眼球中心を代替として用いた。この検出方法としては、先ず、ワールド座標系の原点に各座標軸の方向に時分割で順次磁界を発生する装置を設置し、この磁界を頭部につけたセンシングコイルで検出し、頭部の位置と方向を検出する[8]。ここで、あらかじめ眼球中心をセンシングコイルからの定数ベクトルとして求めておくと、2つのベクトルの和として眼球位置が求まる。

(3) 表示対象の3次元形状は対象ごとに設けられる対象座標系で記述する。従って、仮想空間内の表示対象はワールド座標系内にある対象座標系で記述される。

(4) 以上のようにして、ワールド座標系において、スクリーンの位置、表示対象の位置と方向、視点の位置が決定されたので、視点から表示対象を見たときにスクリーンに投影される画像を計算し表示する。

以上の処理の流れを図2にまとめて示す。

表1に装置の仕様を簡単に示す。グラフィックスワークステーションは、100万ポリゴン/秒の画像生成が可能なIRIS 4D/240 VGX（シリコングラフィクス社製）を用いた。表示装置は、70インチの背面投射型のプロジェクタ（エレクトロホーム社製）で、解像度は、1280×1024画素である。磁気センサ（マクダネルダグラス社製）は、サンプリング周波数60Hz、精度2.5mmである。なお、本システムでは、左右の視点を検出して、各画像を生成表示することで、両眼立体視を実現することも可能であるが、遅れ時間が約2倍になり、遅れの影響を把握しようとする時間的な精度が悪くなるため、ここでは単眼視とした。

表1 運動視表示システム仕様

表示画面	70インチ、背面投射型	
	解像度	1280×1024
	カラー表示	
	ステレオ表示可	
視点計測	精度	6 mm
	サンプリング周波数	60 Hz
	視点移動範囲	磁界発生部から 1.5 m 以内
画像生成	IRIS 4D/240 VGX (SGI製)	
	描画速度	1/60 sec (max)

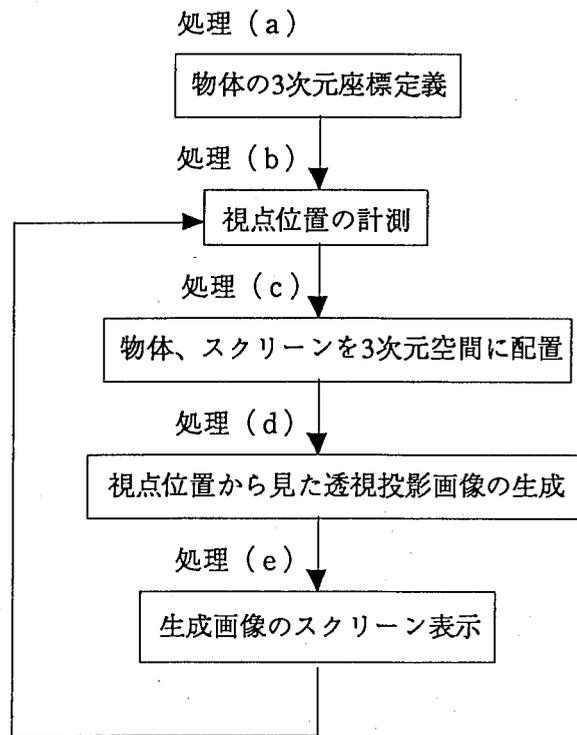


図2 運動視CGのフロー

Flow chart to generate images with motion parallax

### 3. 画像生成の並列処理

図3にCGを利用した場合の運動視表示におけるタイミングチャートを示す。図3において画像生成とは、物体の3次元形状、位置が既知として視点位置からディスプレイを通して見た透視投影画像を生成する時間を示す。運動視表示の遅れは、視点位置検出からその視点位置に対応した画像を表示するまでの表示遅れと視点移動時に画像変化の滑らかさを決定する画像の更新レートの2種類である。尚、図3のタイミングチャートにおいて、視点サンプリング1、2、3に対応する画像生成処理が重なっている部分が存在するが、並列処理、パイプライン処理を行うことを示す。並列処理、パイプライン処理の概要を図4に示す。グラフィックスで画像を生成表示する場合、モデリング変換、ビューイング変換、ディスプレイ処理を順に行なう必要があり、これらの処理をモジュール化する。(a)の並列パイプライン処理において、l、m、n、o、pは、処理のモジュールを示す。このモジュールのセットを5段用意することにより並列パイプライン処理が可能となる。(b)の並列処理は、例えば、画面を4分割、9分割して各部(A、B、C、等)について独立に一連の処理(l、m、n、o、p)を並列に行なう。従って、分割数が増すと、処理時間の短縮となる。

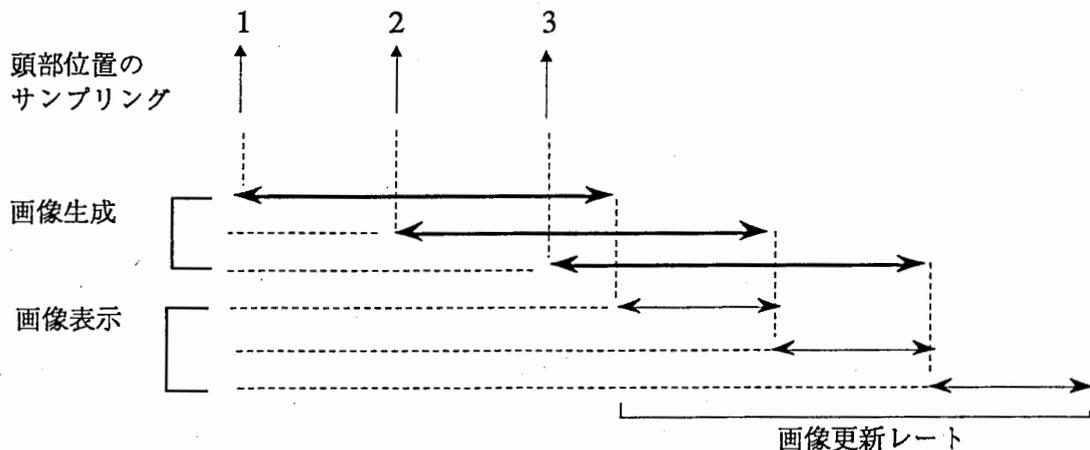
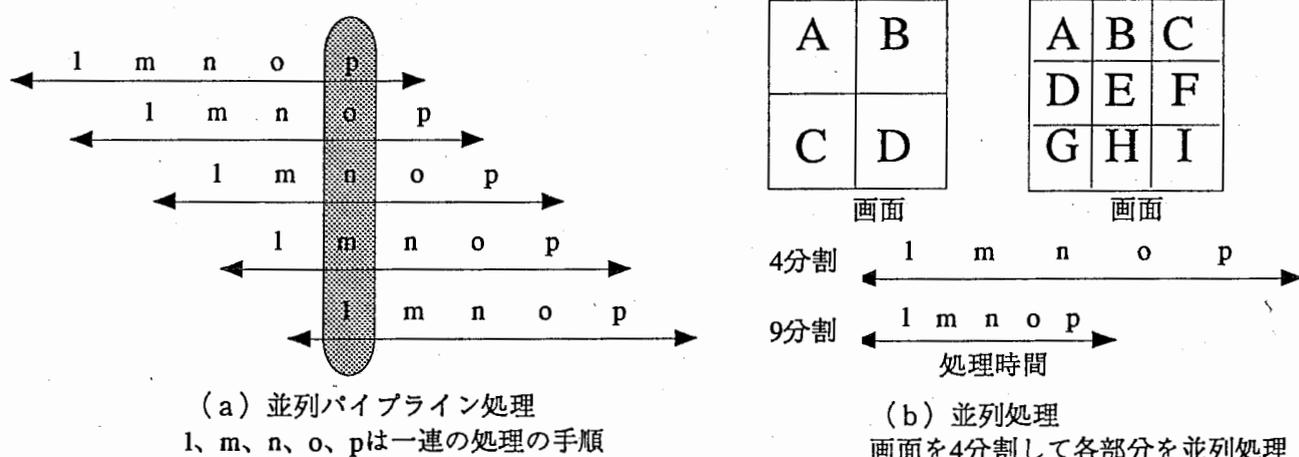


図3 運動視表示タイミングチャート  
Timing chart of images with motion parallax



(a) 並列パイプライン処理  
l、m、n、o、pは一連の処理の手順

(b) 並列処理  
画面を4分割して各部分を並列処理

図4 並列処理とパイプライン処理

#### 4.運動視における表示遅れの評価

##### 4.1 表示遅れによる影響 (主観評価実験)

視点が静止状態から移動を開始した場合に、表示画像が連動せず遅れて動く、あるいは、視点が移動状態から静止状態に変わったにもかかわらず表示画像がそれまで動いていた方向に進んでしまう場合のように、視点と表示画像の間に表示遅れがあると運動視は不自然に感じられやすい。不自然さの知覚度は、視点の移動方向、物体の種類の影響を受けることが視覚心理的な知見から推定できる。そこで、視点の移動方向と表示遅れの知覚特性の関係について調べた。

##### 4.1.1 実験方法

図3に示すように表示遅れとは、頭部位置を検出してから、画像表示までの時間を意味する。ここでは表示遅れの許容時間を求めることを目的としているので、表示遅れが知覚されやすい条件とした。すなわち、視点の移動方向としては、人間の視覚特性が、垂

直、水平方向の変化に敏感なことから[10]、画像変化がこの方向になるように視点を移動させる。実験は、図1のシステムを用いて、被験者に起立着席を繰り返させて視点を上下に移動させる場合と、着席状態で視点を左右方向に移動させて遅れを判断させる2種類を行なった。視点移動速度、範囲は遅れを知覚しやすいように被験者が調整した。表示対象の3次元モデルは、方向異方性のない部屋モデルでモノクロ表示し、大きさは、高さ3[m]、幅4[m]、視点位置から奥の壁までは10[m]である。遅れ時間は、1/30[sec]~10/30[sec]の間で、1/30[sec]間隔に変えて実験を行なった。なお、1/30[sec]は、システムの性能上、最小の画像生成時間である。ディスプレイまでの視距離は1.2[m]、被験者は4名、各遅れ時間2回づつランダムに提示して、遅れの評価は、(1)遅れを感じない、(2)遅れを少し感じる、(3)遅れを明らかに感じるの3段階とした。

#### 4.1.2 実験結果

図5は、視点を上下移動(□、○印)、及び、左右移動(+、×印)した場合の表示遅れを主観評価結果である。本図では、設定した遅れ時間に対して、(被験者数)×(実験回数)を総評価度数とし、それに対する検知限の度数(相対度数と呼ぶ)を□、+印で、また、許容限の相対度数を○、×印で示し、正規確率用紙にプロットした。例えば、遅れ時間120msecで遅れを全く感じない評価の度数が4回の場合、4回/(4名×2回)=0.5となる。視点を上下に移動させた場合と左右に移動させた場合の遅れ時間を比較すると、検知限、許容限ともに上下移動時のほうが短く、遅れに対して敏感な結果となった。

遅れをどの時点で知覚するか被験者に尋ねた結果、視点の移動方向転換時、停止直後、移動開始直後であった。これは、視点移動速度の変化が顕著な場合に、遅れが知覚されやすく、また、遅れがあっても視点と同じ方向に画像が動いている場合は知覚されにくいことを示している。

#### 4.1.3 考察

運動視における表示遅れの知覚特性は、頭部の移動

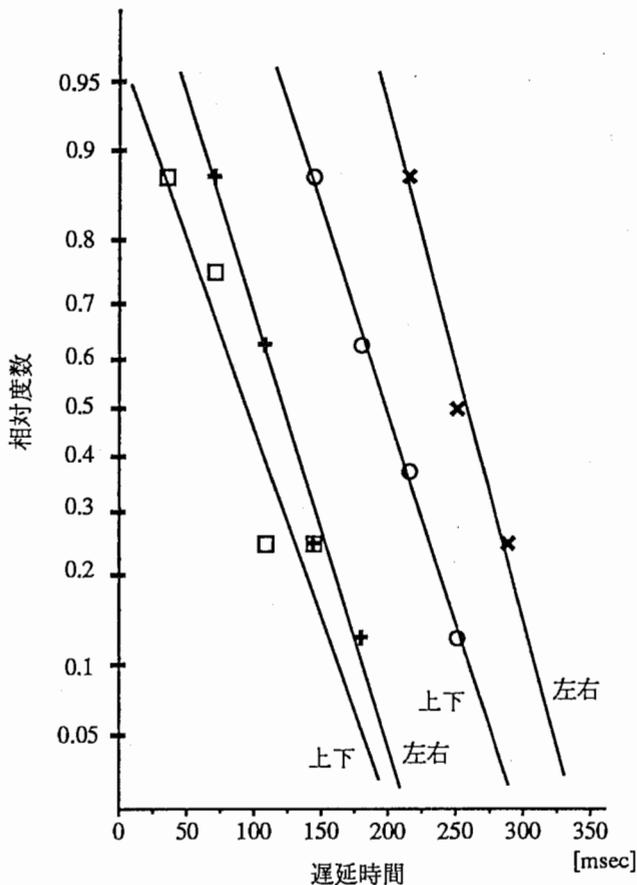


図5 運動視の応答遅れの知覚特性

Experimental result of motion parallax delay

方向によって異なる傾向が見られた。頭部を水平に移動させる場合と、垂直に移動させる場合では、表示遅れ時間が同じでも、垂直方向の方が遅れ時間が大きいと知覚される結果となっている。3.1.1で述べたように、被験者の頭部の移動速度は、上下と左右で大きな変化はないため、この理由としては、人間の視覚特性の影響が考えられる。従来、方向依存性のある視覚特性の例としては、一定の長さの直線を垂直と水平に提示した場合、垂直方向の方が水平方向よりも長く見える[9]、等速度で移動する物体を見る場合、垂直方向の方が水平方向よりも速く見える、などが知られている[10]。本実験の結果より、これら視覚特性が表示遅れの知覚にも影響する可能性があることを示唆している。この場合、運動視CGでは、上下の動きに対する遅れ知覚を考慮して設計する必要が生じてくる。

#### 4.2. タスク実験

##### 4.2.1. 実験方法

表示遅れの影響を定量的に把握するために、運動視を伴ったタスクを行ない、頭部の動きに対する表示遅れ時間をパラメータとしてその作業量を調べた。画像表示は、画像の更新レートを30[hz]固定とし、遅れ時間は、1/30[sec]~1[sec]の範囲で2/30[sec]間隔でランダムに設定した。タスク内容は、図6に示すような直方体のワイヤフレームモデル(H24×D24×W60 [cm])を画面中央に表示し(視点・物体間距離125cm)、側面の手前と奥の2辺が重なって見えるように視点を移動させるものである。例えば、直方体の左側面の場合、手前と奥の2辺が完全に重なるように視点位置を微調整し、

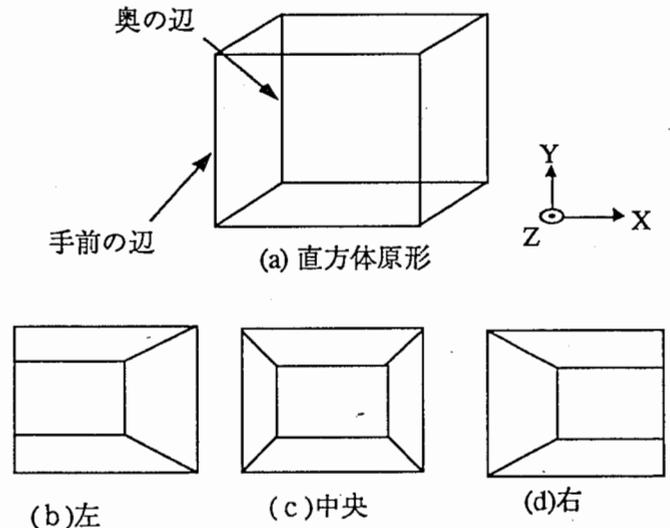


図6 直方体の見え方

完全に重なればシステムがピープ音で作業終了を知らせ、これを1回のタスクとする。次ぎに直方体の右側の側面に視点を移動し同様のタスクを行なう。直方体の左右両側面について交互にタスクを行ない、1分間に行なえる回数を作業量とした。実験は各被験者6名について3回行なった。

#### 4.2.2. 実験結果

図7に表示遅れ時間と作業量の関係を示す。横軸は遅れ時間、縦軸は、6名の被験者の各遅れ時間における1分間の作業量を示す。遅れ時間と作業量の関係を2次の回帰曲線で近似したグラフを示す。その結果、遅れ時間の増加とともに(a)のように指数関数的に作業量が低下するタイプと、(b)のように直線的に作業量が低下する2種類の傾向が見られた。

#### 4.2.3 考察

システムの応答時間とユーザ作業量の関係は熟練者に対しては応答時間の増加に伴い指数関数的に減少することが指摘されている[15]。応答時間が短くなれば、作業量は増すが非常に短くなれば作業量は飽和す

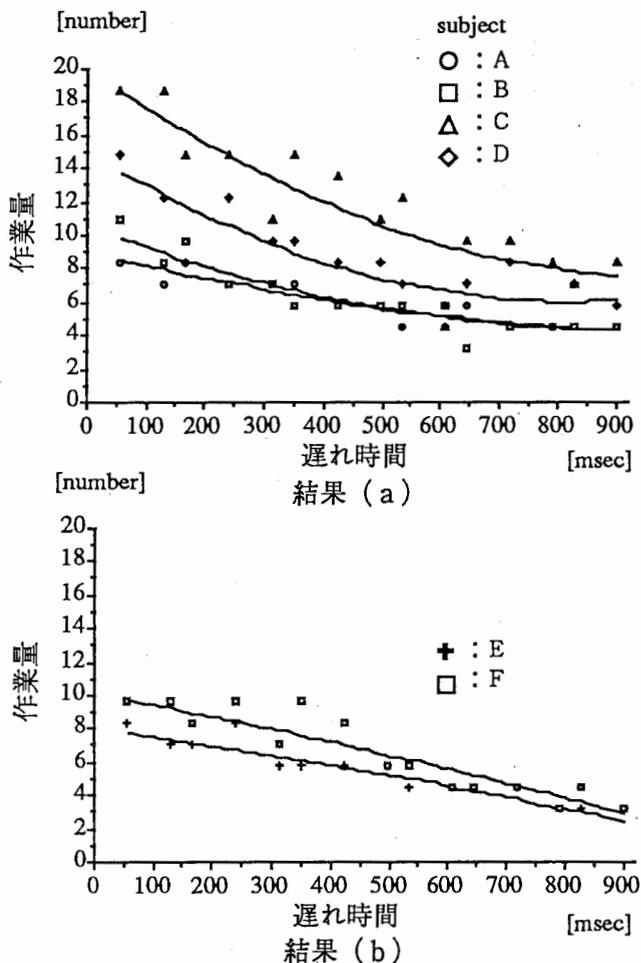


図7 タスク実験結果

ると考えらる。遅れの許容時間は、この作業量の飽和する時間にある。今回の実験ではこの作業量の飽和点は得られず、今後、遅れ時間2/30[sec]から4/30[sec]の間で作業量の変化を調べる予定である。

#### 4.3 画像更新レートの影響

運動視CGにおいて、単位時間当たりの画像生成回数が少ない場合、画像変化は不連続になり、フリッカが知覚され不自然になる。生成回数を一定としたとき、表示画像の変化量は、(1)物体の位置、(2)頭部移動速度によって決まる。図8に示すように、視点が移動するとディスプレイ上では物体の位置が変化して見える。object Aとobject Bのように、視点の移動量Lが同じでも、視点・物体間距離D1がD2よりも大きい場合、ディスプレイ上での画像の変化量L1はL2より大きくなる。そこで、人間が随意的に頭部を動かすことができる速度を考慮しつつ、運動視を実現するために必要な画像更新レートについて調べた。ここで、画像更新レートとは、図3に示す様に視点サンプリング周波数を意味する。

#### 4.3.1 実験方法

図9は、画像更新レートの評価実験構成図である。

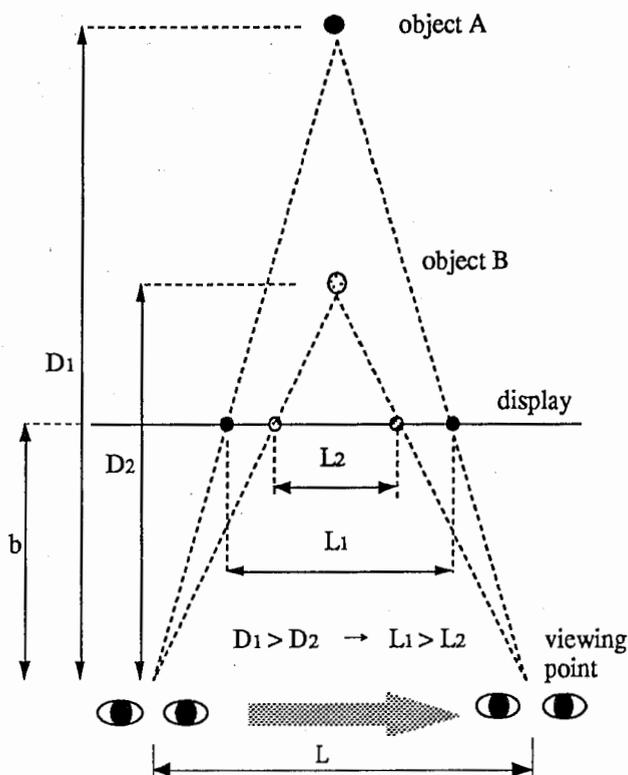


図8 視点・物体間距離と視点移動時の画像変化

Relationship between images with motion parallax and distance from viewpoint to object

被験者は、着席して頭部を左右に等速度で動かして表示物体を観察する。画像生成周期を変えて、表示物体が視点に連動して動く滑らかさを主観評価した。ここで、頭部を等速度で動かすために、画面下には等速に移動する指標を表示した。被験者は指標を周辺視で捉えこの動きに合わせて頭を動かす。表示される指標は、左右方向に等速度で30cmの振幅で振動する。また、指標速度は、日常生活において人間が随意的に頭部を移動させる場合の移動速度を考慮して、1、5、10、20、30[cm/sec]の5段階とした。なお、上記頭部の動き制御方法による頭部移動速度の誤差は、磁気センサで調べた結果、ほぼ10%であった。視点と表示物体間の距離は、120cm、200cm、500cmの3段階とした。モデルは、奥行き10cmの12面体の3次元サフェスモデルをグローシェーディングで表示した。画像更新レートは、システムの性能上、30[hz]を最高周波数として、15[hz]、10[hz]、7.5[hz]・・・と30[hz]の1/n倍とした。表示物体の動きの滑らかさを、○:滑らかである、△:少しちらつきを感じる、×:明らかにちらつきを感じる、の3段階で評価した。被験者は、男性3名、女性1名である。

#### 4.3.2 実験結果

頭部移動速度、視点と物体間の距離、および滑らか

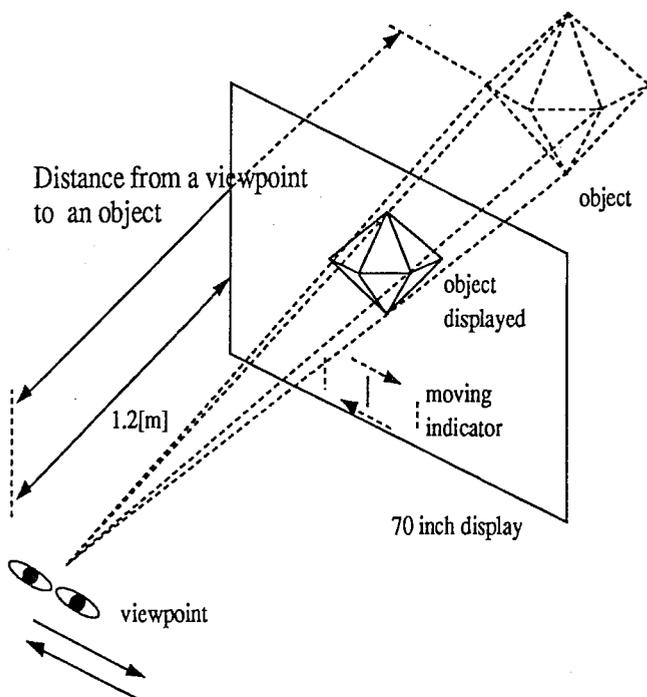


図9 コマ落とし実験構成図

Experimental configuration of field-repeated images with motion parallax

な動きを表示するために必要な画像更新レートの関係を図10に示す。被験者数が4名中3名が○の評価を与えた画像更新レートをプロットした。視点と物体間の距離を120cm、200cm、500cmと変えた場合について別々に示している。この図から視点移動速度が増加する場合、又は、視点・物体間距離が増加する場合、画像更新レートが増加していることが分かる。システムの性能上、画像更新レートの上限は、30[hz]であるから30[hz]を越える周波数については測定不可能である。視点・物体間距離が、200[cm]の場合には、頭部移動速度が20[cm/sec]以上において、500[cm]の場合には、頭部移動速度が10[cm/sec]以上において、30[hz]以上の画像更新レートが必要なことを示している。

#### 4.3.3 考察

運動視表示において、利用者に滑らかな動きを提供するために必要な画像更新レートを求める。前述のように、ディスプレイ上の画像変化は、視点・物体間距離が長い場合は位置の変化が大きく、距離が短くなると回転変化が大きくなる。しかし、それにもかかわらず、図10では、必要な更新レートが距離とともに増えていることから、運動視CGでは、距離が長い場合の画像位置変化に注意して画像更新レートを定めることが重要といえる。そこで、視点・物体間距離500cmの場合について、視点が水平に、且つ、画面に平行に移動する条件で、画像が不連続と知覚されないフレーム

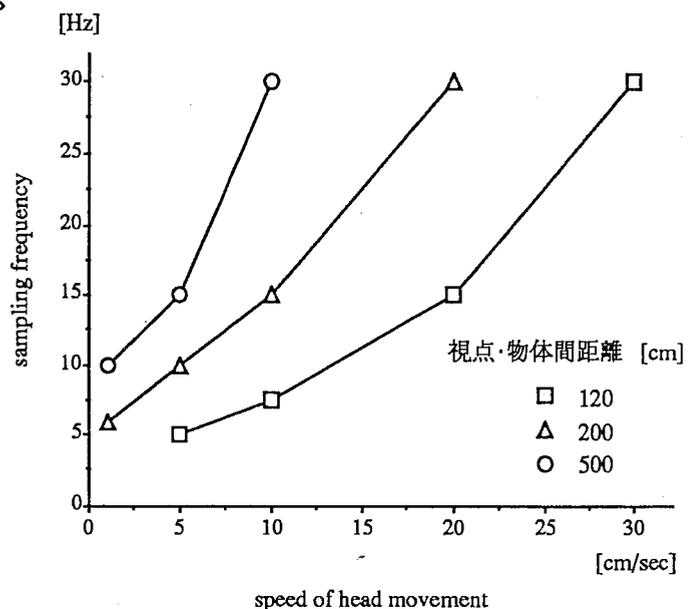


図10 コマ落とし実験結果

Experimental results of field-repeated moving images with motion parallax

表2 フレーム間画像変化量

視点・物体間距離 [cm]	D	120			
視点移動速度 [cm/sec]	v	5	10	20	30
視点サンプリング周波数 [Hz]	f	5	7.5	15	30
フレーム間画像変化量 [cm]	M	0.04	0.05	0.05	0.05
視点・物体間距離 [cm]	D	200			
視点移動速度 [cm/sec]	v	1	5	10	20
視点サンプリング周波数 [Hz]	f	6	10	15	30
フレーム間画像変化量 [cm]	M	0.07	0.20	0.27	0.27
視点・物体間距離 [cm]	D	500			
視点移動速度 [cm/sec]	v	1	5	10	-
視点サンプリング周波数 [Hz]	f	10	15	30	-
フレーム間画像変化量 [cm]	M	0.08	0.25	0.25	-

間画像変化量Mを求める。Mは次式で与えられる。ここで、視点と物体間の距離をD、視点とディスプレイの距離をb、視点の移動速度v、画像更新レートをfとする。

$$M = \frac{D - b}{D} \cdot v \cdot \frac{1}{f} \quad (1)$$

本実験のパラメータを(1)式に代入し、Mを求めた結果を表2に示す。視点・移動速度Vが1cm/secと遅い場合には、Mは0.08と小さくならないが、5cm/sec以上では0.25で飽和している。また、D=200cmの場合についても、同様に5~10cm/sec以上で飽和する傾向が見られた。この説明については、補償眼球運動と視力の関係から別途検討が必要であるが、表2においてM=0.25をv>10cm/secにも適用すると、人が頭部を速く動かす条件(V=30cm/sec)において、f=90Hzとなる。

以上、3.1、3.2項より運動視表示システムを設計する場合、表示遅れ100msec以下、画像更新レート90Hz以上が望ましい条件と言える。

## 5. 高速画像生成表示法

### 5.1 頂点数と処理時間

上記の表示遅れを検知されない時間で、できるだけ精細と感ぜられる画像を高速生成する表示法について述べる。一般に、CGで画像生成する場合、生成時間は物体を形成する頂点数により決まる。図1.1は、高速グラフィックワークステーションIRIS 4D/240VGXを用いて球と円柱の頂点数を変えてモデリングした物体を画像生成表示した場合の頂点数と画像生成表示時間

の関係を示した例である。実験では、球と円柱のモデルを一個づつ同時に表示し、表示モデルによる処理時間の差を調べるために、ワイヤフレームモデルで表示した場合と、サーフェスモデルでシェーディング表示した場合についてそれぞれ画像生成時間を測定した。ここで画像生成時間とは、視点位置、物体の3次元位置を一定として、図2のフローの処理(c)、(d)にかかる時間を示す。頂点数の操作は、球モデルの緯度、経度の分割数、及び、円柱モデルの円周、高さの分割数を変えることにより行った。同図より頂点数にほぼ比例して時間が増加することが分かる。

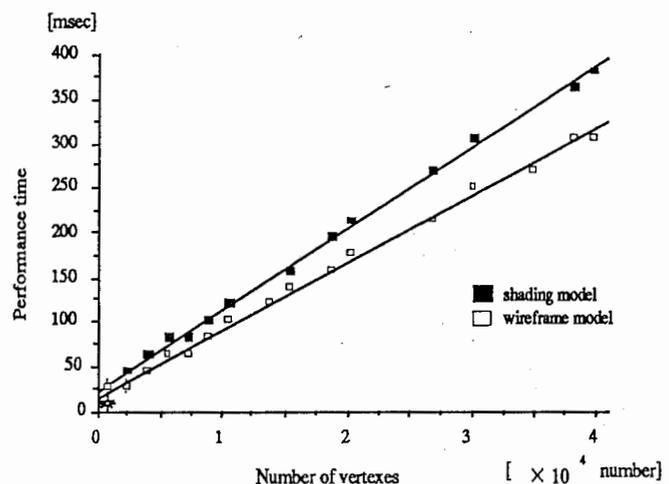


図1.1 表示モデルの頂点数と画像生成表示時間

Performance time to generate images by IRIS

### 5.2 視点を利用した頂点削除

透視投影変換により画像生成表示する場合、デスク

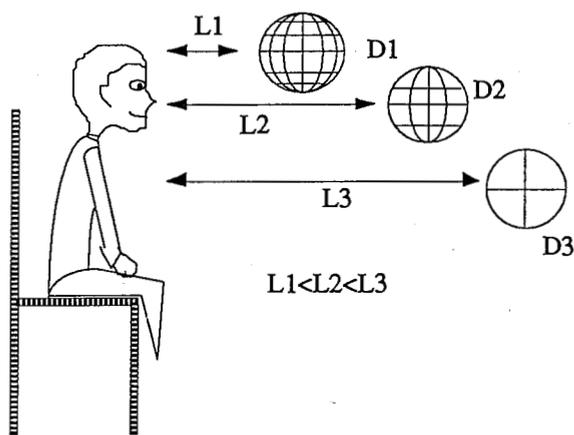


図 1 2 階層化モデル

Differently detailed model based on distance from viewpoint to object

レイに表示される物体の大きさは視点から物体の距離により決まる。例えば、視点から物体の距離が遠い場合、物体は小さく表示され、視点からの距離が近い場合は大きく表示される。従って、物体を1通りの頂点数でモデル化した場合、視点の近くにある物体は頂点数が粗になり画像の品質は悪くなる場合があり、一方、遠くにある物体はディスプレイ上で単位面積当たりの頂点数（以下、表示密度）が増加し、ディスプレイの解像度を越えることもあり得る。ディスプレイの解像度以上に細かくモデル化しても画質は向上せず頂点数が増加した分だけ画像生成時間が長くなるといった不合理が生ずる。

そこで、同一の表示物体で頂点数を変えたモデルを階層的に用意しておき、視点からの距離に応じてモデルを選択することにより、遠方の物体についてはディスプレイ上での表示密度の飽和を防ぎ、又、近くの物体については、画質を低下させずに画像生成の高速化を図る手法について検討する[2]。

### 5.2.1 視点を利用した階層化モデル画質評価実験

図 1 2 において L1、L2、L3 は視点と物体間の距離を示す。各距離に対して利用者が画質の劣化を感じない最小の頂点数 (minimum number of vertex の略以下 MNV) で定義された表示モデルを D1、D2、D3 とする。L1 < L2 < L3 に対して、頂点数は、D1 > D2 > D3 の関係がある。

### 5.2.2 実験方法

人間は角の知覚に優れているため、表示された物

体に角を知覚するか否かを主観評価することにより、物体表面の滑らかさを評価できることが知られている [11]。そこで、物体の表面を記述する頂点数を十分に滑らかに見える頂点数から段階的に減らし、初めて角を知覚したときの頂点数を上記 MNV として求めた。表示モデルとしては、表面の滑らかさを表現しやすく、また評価の際個人差が少ないと思われる球を用いた。球は地球儀方式で定義し緯度、経度の分割数を変えてモデルを階層化した。図 1 3 に地球儀方式で定義した球の階層化モデルの例を示す。分割数を少なくすると球面に角が知覚される。また、提示画像に遠近感を与えるために、部屋のなかに球を配置し、球、部屋ともにグローシェーディングのサフェスモデルで表示した。実験装置は図 1 のシステムを用いた。実験条件を表 3 に示す。

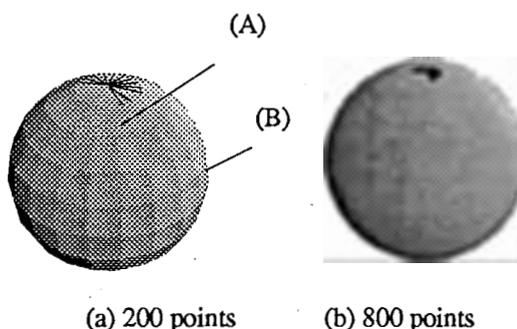


図 1 3 頂点数を変えた球のモデル  
Example of sphere model

表 3 画質評価実験条件

項目	条件
被験者数	6名
照明	暗室
視距離	1.2 m
表示モデル モデルの位置	球 半径 45 cm スクリーン中央奥 120 - 1000 cm

視距離とはスクリーンと視点間の距離

### 5.2.3 実験結果

実験結果を図 1 4 に示す。縦軸が球を定義している頂点数、横軸が視点と表示物体の距離を示す。グラフは全被験者の MNV の平均値と分散を示す。図 1 4 から視点・物体間距離に対する MNV の減少度を調べると、

視点・物体間距離が120[cm]のMNVに比べて1000[cm]のMNVは、頂点数が70%減少している。

以上の実験により、視点・物体間距離が長くなるに従って、画像の品質を落とさずに頂点数を減らせることが分かる。また、このことは計算時間が同じであればより多くの物体を表示できることを示している。

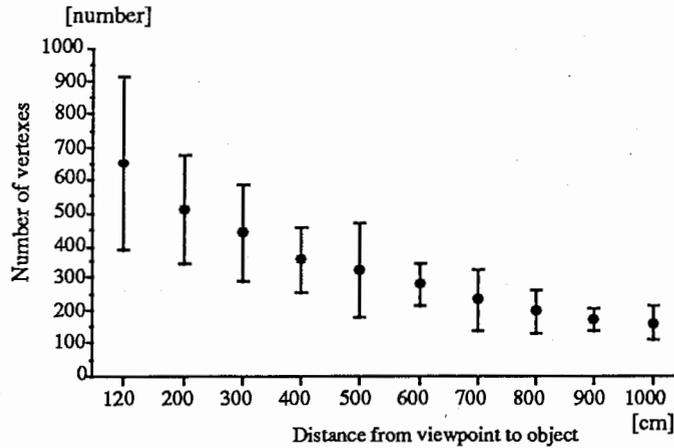


図 14 視点・物体間距離と表示モデルの必要最小頂点数

Minimum number of vertexes of sphere with smooth surface at each distance

#### 5.2.4 考察

本手法の適用にあたっては、階層化モデルの作成法、及び、視点・物体間距離と選択する階層の関係性を明らかにする必要がある。ここでは、球の階層化モデルについて述べる。

球面の滑らかさを表現するために必要な球の頂点数は、ディスプレイに表示される球の大きさとディスプレイの解像度により決まると考えられる。球面の滑らかさの劣化は、球を形成する隣接ポリゴンの境界線(図13(A))が知覚されるか、または、球全体の輪郭(図13(B))の滑らかさが失われることにより知覚される。前者については、頂点数を減らして大きなポリゴンを形成してもグーロシェディングを用い、ポリゴン内の輝度は線形補間することにより知覚されにくくできる[12]。後者については、ポリゴンが大きくなると輪郭の角が容易に知覚される。球の画質はこの後者により決まってしまうので、この輪郭を滑らかに見せる観点からモデルの作成法について検討した。

#### 5.2.5 球の画質と視点・物体間距離

球の輪郭線の一部を図15に示す。球の中心をO、緯度による一分割した角度を $\theta$ 、球面上の頂点をa、

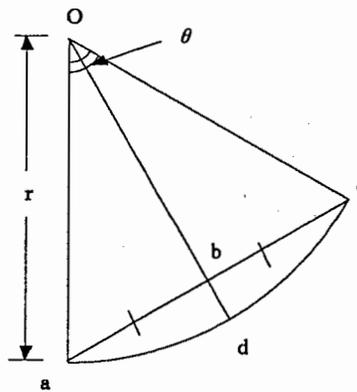


図 15 球の断面図

A partial cross section of contour of sphere

c、直線acの中点をbとし、直線obとの球面の交点をdとする。球の近似モデルの輪郭線は直線abcであり、真の球の輪郭線は曲線adcである。この直線abcと曲線adcが

ディスプレイの解像度上、近似さ

れて同様に表示されれば球輪郭は滑らかに見えると考えられる。そのためには、近似誤差が最大になる直線bdとディスプレイの解像度resとの関係は、

$$\text{直線bd} \leq \text{res}$$

を満たすのが望ましい。そこで、図14で求めた視点・物体間距離とMNVとの関係がこの式を満たしているかどうかについて調べた。直線bd、球の分割数、視点・物体球間距離の関係を求めると、式(2)が得られた。式の導出は付録に示す。

$$\overline{bd} = \frac{d_0}{d_1} \cdot r_1 \cdot \left(1 - \cos \frac{\pi}{2n}\right) \quad (2)$$

球の頂点数が与えられると球の分割数nが決まるため、図14で求められたMNVはnに変換可能であり、これを(2)式に代入すると直線bdが求まる。このようにして、図14の縦軸を直線bdに変換した結果を図16に示す。ディスプレイの解像度(図16のpixel size)は1mm

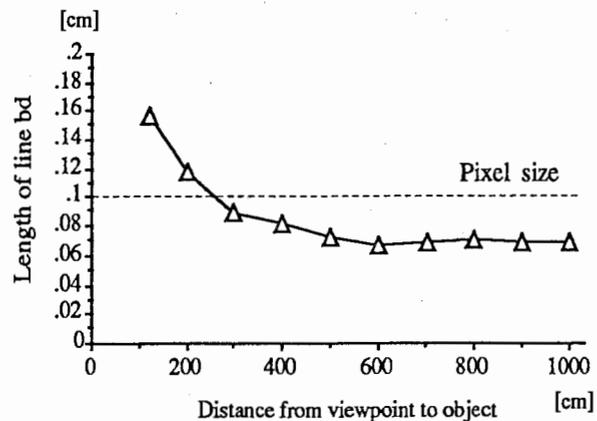


図 16 ディスプレイ表示された球の直線bdの長さ

Length of line bd at each distance

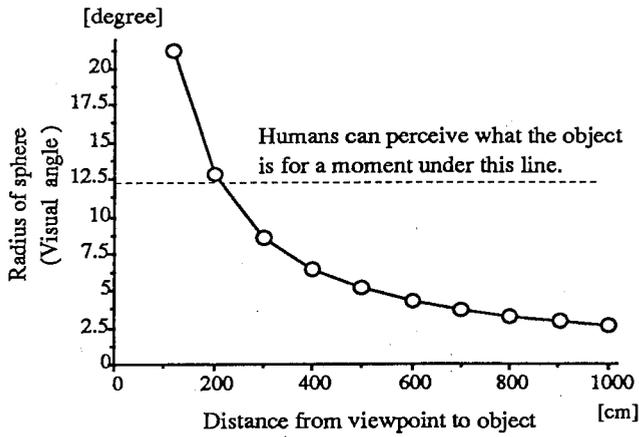


図 17 ディスプレイ表示された球の視野角  
Radius of sphere displayed in perspective

(70インチプロジェクタ、画素数1280×1024)であるから、視点・物体間距離300[cm]以上では、直線bd<resを満たしている。しかし、視点・物体間距離が200[cm]以下では直線bd>resとなり、直線bdがディスプレイの解像度より長くても球面は滑らかに知覚されている。

この理由について考察する。図17は視点・物体間距離と球をのぞむ視角の関係を示したものである。人間の有効視野を12°程度として[13]、球全体が有効視野に入る視点・物体間距離を求めると200 cm程度となる。この結果を図16と比較すると、直線bd≤resを必要とするのは球が有効視野内に入る場合、逆に、球が有効視野より大きければ直線bd>resでも球は滑らかに知覚されることが分かる。以上の結果から、球を少ない頂点数で生成するには、視点・物体間距離および有効視野とディスプレイ上の球の大きさの関係を考慮するのが望ましいと言える。

### 5.3 視線を利用した頂点削除

人間の視野内における弁別能力は、注視点が最も高く、注視点からの距離が遠くなるに従い低下する事が知られている[14]。視線検出装置で注視点を検出し、注視点近傍にはきめ細かくモデリングした物体を表示し、注視点以外の領域については、精細度を粗くモデリングした物体を表示することにより、画質の劣化を与えず頂点数を減らすことが可能となる[2]。CGで画像生成表示する場合、物体を形成する頂点数により処理時間が決まるため、頂点数が減少する分高速化が図れる。

#### 5.3.1 適応的選択

図 1 8 に注視点からの距離をパラメータとして、同

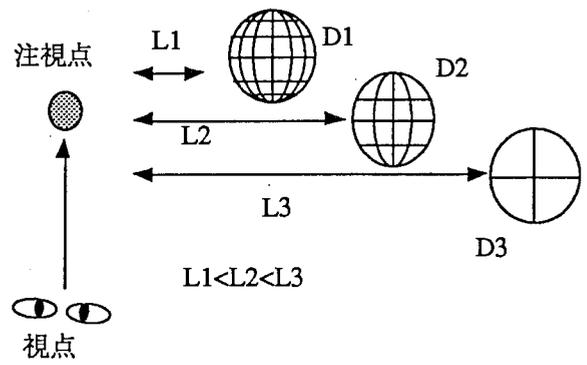


図-1 8 視線を利用した高速画像生成表示法

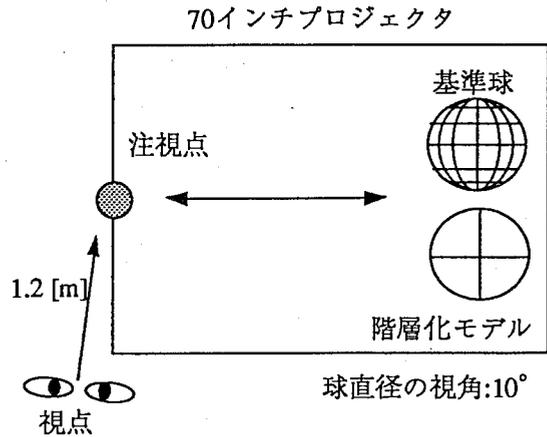


図 1 9 実験構成図

一物体のモデルのデータベースから、階層を選択し画像生成表示する方法の概略図を示す。図 1 8 において L1、L2、L3は注視点と物体間の距離を示す。各距離に対して利用者が画質の劣化を感じない最小の頂点数で定義された表示モデルをD1、D2、D3とする。L1=0としてD1が注視点にあるとすれば、L1、L2、L3の距離差の大きさによってはD2、D3の頂点数はD1に比べ減少すると考えられる。

### 5.4. 視線を利用した階層化モデル画質評価実験

#### 5.4.1 実験方法

物体を画質の劣化なく表示するために必要な最低の頂点数を注視点から物体までの距離を変えて調べ、視線を利用した階層化モデルの高速画像生成表示法の有効性を評価する。実験構成図を図 1 9 に示す。被験者にディスプレイ上の固定点を注視させ、注視点から離れた位置に高精細にモデリングした球(基準球)と階層化モデルの球を同時に提示し、階層化モデルの精細度を粗くしていき基準球と階層化モデルの球の差を検知させた(2重刺激法)。注視点から球を表示する距離を変えて2つの球の差が検知された場合の頂点数を調べた。画質の総合的な評価語の一つとして、滑らか

さが提案されており[15]、表示モデルは滑らかさを評しやすく、評価の個人差が少ない人工的な物体として球を提示した。球は地球儀方式で定義し緯度、経度の分割数を変えてモデルを階層化した。被験者は3名とした。また、提示画像は、背景を廊下とし、球、廊下ともにモノクロでグローシェーディングのサフェスモデルを表示した。

#### 5.4.2 実験結果

実験結果を図20に示す。横軸に注視点からの表示物体の視角距離、縦軸に2つの球の差が検知された場合の階層化モデルの頂点数を示す。物体が注視点にある場合に比べて、注視点からの距離が50°になると頂点数が90%減少しており、高速化が図れることを示唆している。

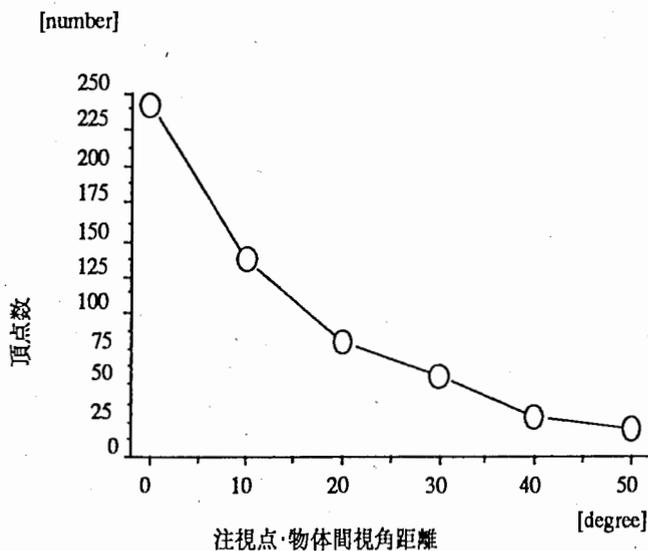


図20 注視点から外れた場合の必要頂点数

## 6. むすび

利用者の視点位置を計測し、その視点位置に対応したCG画像を生成表示する運動視表示システムを用いて、表示遅れの影響について検討し以下の結果を得た。

(1) 視点移動に対する画像変化の応答については、視点の移動方向により、表示遅れの知覚特性が異なり、視点の移動方向が水平方向時に比べて垂直方向時の方が、遅れに対して敏感となる知見が得られた。

(2) 視点移動中の画像変化の滑らかさを決める画像の更新レートについては、視点移動に伴う画像変化の滑らかさを表現するために必要な視点のサンプリング周波数と、視点の移動速度、視点・物体間距離の関係を示した。

(3) (1)、(2)より視点移動速度30cm/sec以下、視野角60度のディスプレイを用いる条件のとき、位相遅れ100msec以下、画像更新レート90Hz以上であれば表示遅れは検知されない。

高速画像生成表示法については、同一の物体に対して、頂点数を変えた複数のモデルを作成し、視点と物体間の距離、または、注視点と物体との偏角をパラメータとして、画質の劣化を生じない最小の頂点数のモデルを選択する手法を検討した。又、表示物体が、球面を持ったモデルについては視点・物体間距離、ディスプレイの解像度と画質劣化を与えないための球モデルの頂点数の関係を示した。

今後は、より複雑な表示物体の頂点数を変えた階層化モデルを作成する手法について研究を進めていく必要がある。

## 謝辞

日頃ご指導頂く(株)ATR通信システム研究所葉原耕平会長、山下紘一前社長、寺島信義社長、同所知能処理研究室岸野文郎室長に深謝します。また、実験を進めるにあたり被験者を快く引受て頂いた皆さんに感謝します。

## 参考文献

- [1] 伴野、岸野："臨場感通信会議におけるヒューマンインタフェース技術"、人工知能学会誌 6 No.3 pp.358-369 (1991)
- [2] 伴野、竹村、石橋、岸野："視線追跡型高速画像生成表示法の提案"、TV全大、pp91-92 (1990)
- [3] 畑田 豊彦："各種運動知覚現象"、O plus E No.68 pp127-134 (1985)
- [4] J.J.Gibson："The Ecological Approach to Visual Perception"、Houghton Mifflin (1950)
- [5] 星野、比留間、山田、福田："運動視差利得を変化させたときの視覚効果"、信学論 (D-II) j73-D-II No.2 pp.224-231 (1990.2)
- [6] 宮原 誠："動きの知覚特性"、テレビ誌、27 No.1 pp24-30 (1973)
- [7] 田所、三橋："被写体速度と視覚の関係"、NHK技研月報9 pp.422-426 (1968)
- [8] F.H.Rabb et al., "Magnetic position and Orientation tracking system"、IEEE Trans. Aerosp. & Electron. Syst., AES-15, No.5 (Sep.1979)
- [9] 盛永 四郎："知覚心理学"、明玄書房 (1969)
- [10] 田崎、大山、桶渡："視覚情報処理"、朝倉書店

- [11] D.Noton : "A Theory of Visual Pattern Perception", IEEE Trans. Systems science and Cybernetics, scc-6, No.4, pp.349-357 (1970)
- [12] H.Grouaud : "Continuous Shading of Curved Surfaces", IEEE Trans. Computers c-20 No.6, pp623-629 (1971)
- [13] 畑田 豊彦 : "自然視画像と視覚特性", O plus E No.74, pp121-130 (1986)
- [14] 畑田, 坂田, 日下: "画面サイズによる方向知覚誘導効果", テレビ誌, 33, 5, pp407-413 (1979)
- [15] 宮原 誠: "時空間相関の視知覚特性とテレビ画像の高品質かに重要な物理要因", テレビ誌, vol.40 No.1 pp46-53(Jan.1986)
- [16] 中島 正之: "CGによるヒューマンインターフェース", テレビ誌, vol.44 No.8 pp.977-980 (1990)
- [17] 伊藤正美 : "生体信号処理の基礎", pp.57, オーム社(1985)

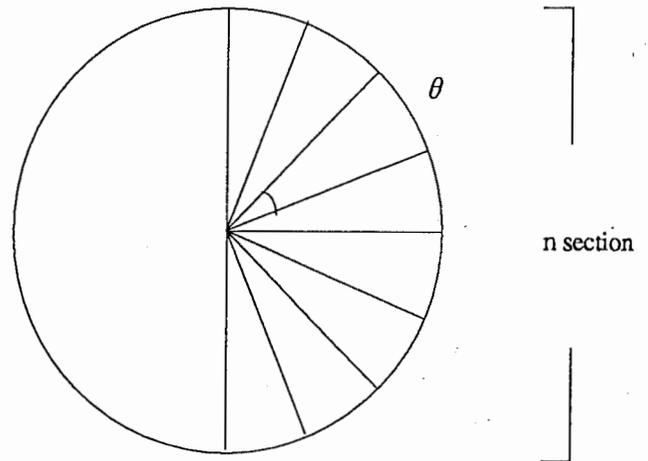


図 A・1 緯線による球の分割

Division of sphere by latitude

付録

図 1 5 における直線bd (球面誤差)、球の分割数、視点物体間距離の関係式 (2) を導く。図 1 5 より、(A・1) 式が求まる。

図A・1に示すように緯線により球をn等分するので $\theta$ は(A・2)式となる。視点、ディスプレイ、物体球の位置関係を図A・2に示す。ディスプレイに表示される球の半径を $r_0$ 、実際の物体球の半径を $r_1$ 、視点ディスプレイ間距離を $d_0$ 、視点・物体間距離を $d_1$ とすると物体球を視点から見てディスプレイに透視投影表示した場合の $r_0$ 、 $r_1$ 、 $d_0$ 、 $d_1$ の関係は、(A・3)式となる。 $r=r_0$ より(A・2)式、(A・3)式を(A・1)式に代入すると(2)式が得られる。

$$\overline{bd} = r - r \cdot \cos \frac{\theta}{2} \quad (A \cdot 1)$$

$$\theta = \frac{\pi}{n} \quad (A \cdot 2)$$

$$r_0 = \frac{d_0}{d_1} \cdot r_1 \quad (A \cdot 3)$$

$$\overline{bd} = \frac{d_0}{d_1} \cdot r_1 \cdot \left(1 - \cos \frac{\pi}{2n}\right) \quad (2)$$

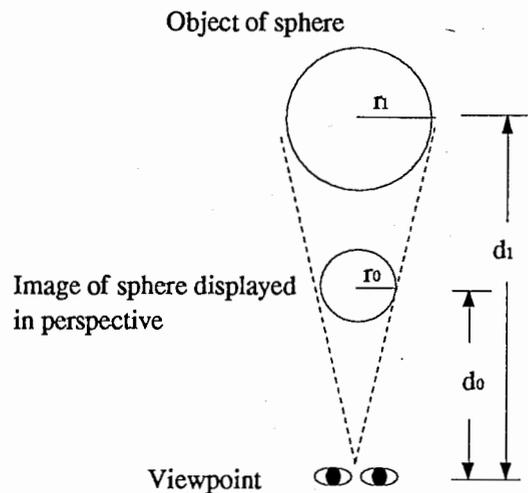


図 A・2 視点、ディスプレイ、物体球の位置

Position of viewpoint, display, object of sphere

## 実験用ソフトウェア

### [1] 時間計測プログラム

グラフィックスワークステーション(IRIS)を用いて実験を行う場合、画像生成処理等の処理時間を計測するためのプログラムである。UNIXのシステムクックロックを利用して1970.1.1からの通算秒が出力される。

ファイル名 : get\_time.h

プログラムリスト : リスト1

#### 仕様

出力: 時間 単位 sec  
分解能 1/100 sec  
1/1200 sec

備考 IRISタイマーの設定

分解能 1/100 secの場合

ftimer ↓(キー入力)

Fast clock is off. Fast itimers are off.

分解能 1/1200 secの場合

ftimer -f on ↓

ftimer

Fast clock is on processor 0. Fast itimer are off.

#### 使用方法

時間を計測したいプログラムにget\_time.hをincludeし、時間を知りたい箇所に、get\_double\_time()、または、get\_float\_time()を挿入する。使用例を以下に示す。

(例)

main()

```
{  
    t1=get_double_time();
```

処理(A)

```
    t2=get_double_time();
```

```
    t=t2-t1;
```

```
}
```

tに処理(A)にかかった時間が求まる。

### [2] 三次元座標データファイル生成プログラム

(1)磁気センサの三次元データを取り込み、sungraphを用いて表示可能な形式のファイルを生成する。

尚、詳細については、'90年SCC試作ソフトウェアの、"視線検出応用ソフトウェア"(1版90年10月28日)、機能操作説明書を参照のこと。

ファイル名: dt\_write.c

#### (2)xgraphを用いたグラフ表示

Xウィンドウをワークステーション上で使用している場合は、グラフ表示ソフトウェア"xgraph"を用いると、非常に見やすい時系列データ等のグラフが作成可能である。一例として、dt\_write.cで作成されたデータファイルを、xgraphで表示可能なファイル形式に変換し、さらに、時系列データから速度を求めて表示させた例を図Bに示す。実験条件は、図9に示す指標を、速度10cm/secで移動させ、磁気センサを持った手を指標に併せて連動させた場合の手の位置と移動速度を表している。

尚、位置座標の時系列データが得られた場合に、速度を求めるには次式を用いた[17]。ここで、 $v_i$ は、時刻*i*における移動速度、 $y_i$ は時刻*i*における位置座標、 $T$ はサンプリング周期を示す。

$$v_i = \frac{y_{i+2} + y_{i+1} - y_{i-1} - y_{i-2}}{6T} \quad (B)$$

```

/*****
** システムタイムを返す **
** 引数 : なし **
** 戻り値 : 正の小数值 **
*****/
#include <stdio.h>
#include <sys/time.h>

static double get_double_time();
static float get_float_time();

static double get_double_time()
{
    struct timeval Timeval;
    struct timezone Timezone;

    if (gettimeofday(&Timeval, &Timezone) == -1) {
        fprintf(stderr, "System call error 'gettimeofday' !!\n");
        exit(-1);
    }

    return((double)((long)Timeval.tv_sec +
                    (long)Timeval.tv_usec / (double)1000000.0));
}

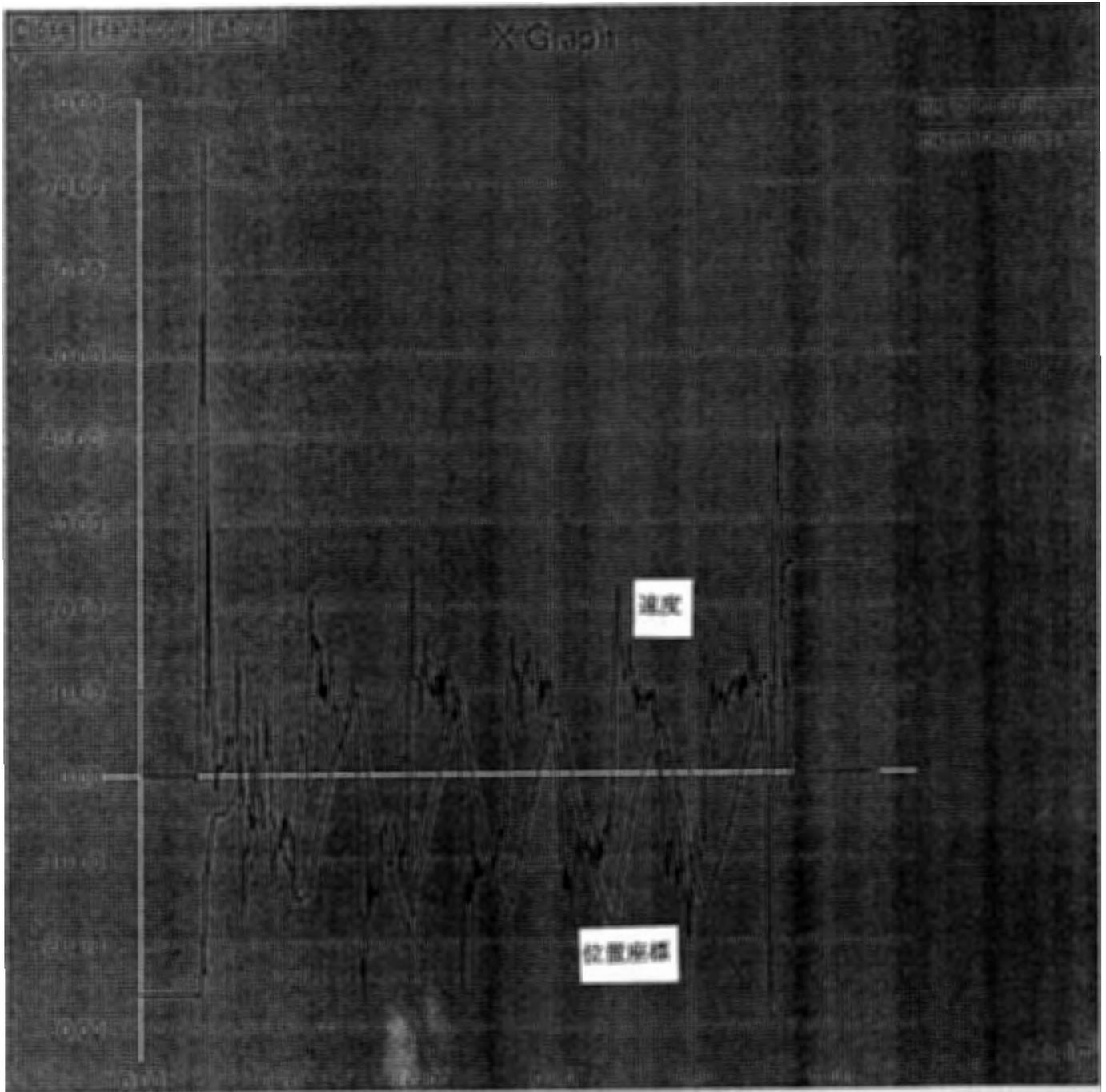
static float get_float_time()
{
    double double_time;
    long integer_time;

    double_time = get_double_time();
    integer_time = double_time / 10000;
    integer_time *= 10000;

    return((float)(double_time - integer_time));
}

```

リスト1 時間計測プログラムリスト



\*\*\*a: 位置座標    \*\*\*s: 速度

図B 手の移動軌跡  
(Xgraphp を用いて表示)