

## 没入型 VR における 動的被写界深度効果の実装と評価

吉柳俊佑† 蔡東生†

本研究では、ユーザの視線に応じてリアルタイムに変化する被写界深度効果を没入型 VR 上で再現し、VR システムの没入感の向上を図る。今回、動的被写界深度の具体的な実装手法と、リアルタイム性の検証を行った結果について報告する。

### Implementation and estimation of Depth of Field blur effect in immersive VR

SHUNSUKE KIRYU† DONGSHENG CAI†

We implements the depth of field blur effect depends on the sight line of user for the purpose of improvement of the immersion in virtual reality systems. This paper reports the method of implementation in detail and the result of examination for confirming whether depth of field blur effect is real-time or not.

† 筑波大学大学院システム情報工学研究科  
Tsukuba University, Graduate School of Systems and information Engineering

### 1. はじめに

現実と等価な現象空間を提供することで高いシミュレーション能力を有する VR(Virtual Reality)の領域では、より現実の環境へと近づけるために人間の奥行き知覚システムを再現する方法がある。人間の奥行き知覚は様々な奥行き知覚手掛かりの統合によって行われる[1]が、被写界深度効果も有効な奥行き知覚手掛かりの1つとして知られている[2]。

被写界深度とは焦点調節に起因する現象であり、人間の眼においては水晶体が鮮明な像を結像する凝視点まわりの距離の範囲のことをいう。被写界深度の範囲外に存在する物体はぼけて見えるが、その様子を図1に示す。画像(a)では人間が画像の近景を凝視していることを想定し、(b)では画像の遠景を凝視していることを想定している。(a'),(b')はそれぞれ(a),(b)の被写界深度効果によって生じるぼけの大きさを濃淡で表現しており、被写界深度の範囲を直観的に理解することができる。白い部分ほどぼけが大きく、黒い部分ほど鮮明であることを表している。

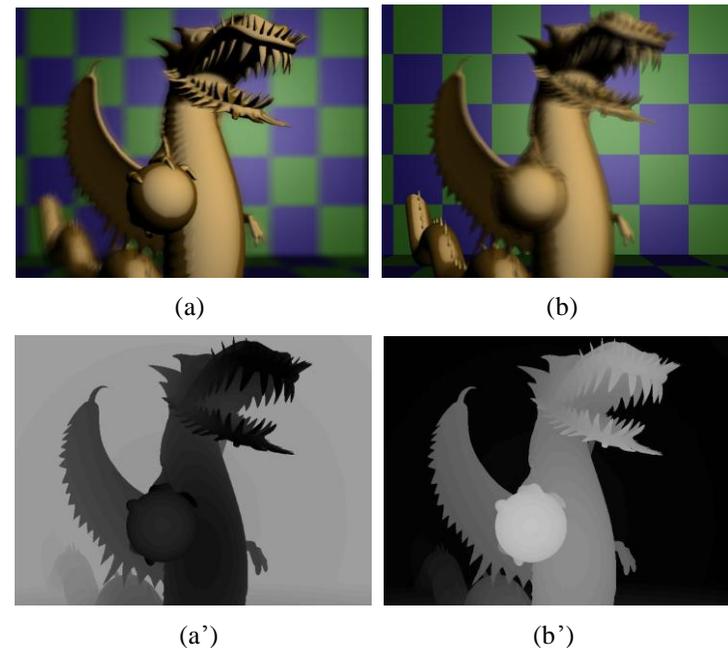


図1. 被写界深度効果

被写界深度効果は写真や映画、ゲームなどの画像・映像分野で広く用いられており、特定の対象に注目させることなどに効果的に使われる。VR の領域では一般的にぼけがない全焦点画像が提示されるが、被写界深度効果を加えることで現実感を高めることができる[3]。しかし、インタラクティブ性を欠き、任意の対象に注目することができなくなるといった問題がある。したがって、視線解析装置を用いることで、ユーザの視線に応じて変化するリアルタイムな被写界深度効果を適用し、VR の現実感の向上を図る。

視線解析装置を用いた動的な被写界深度効果を実現することで VR の質を高める研究として Hillaire らの研究[3]がある。被写界深度効果に加え、周辺ぼけやモーションブレイクなどの様々なリアルタイムの視覚効果が実装されているが、奥行き知覚手掛かりとして効果が高いとされる両眼視差や輻輳が考慮されておらず、現実感を高める余地は多く残されていると言える。奥行き知覚手がかりは、複数手掛かりの統合により、単独で用いる場合よりも効果が増減する場合があります[4]、また、より現実感の高い VR システムを構築する上においても、これらの奥行き知覚手がかりを導入する必要性は高いと筆者は考える。

本研究では、両眼視差および輻輳を考慮した没入型 VR システムにおいて、ユーザの視線に応じて動的に変化する被写界深度効果を生じさせることで、奥行き知覚精度の向上、延いては VR の現実感の向上を試みる。具体的には、視線解析装置により計測されたディスプレイ上の凝視位置を入力として、CoC(Circle of Confusion)をシミュレートするモデル[5]に基づいて没入型 VR 中の 3D オブジェクトに被写界深度効果を適用するポストプロセッシングを行い、計算結果の画像を 3 面の没入型 VR システムにリアルタイムに表示する。本稿では、上記技術の実装手法およびリアルタイム性の検証結果を報告する。

## 2. Virtual Reality

バーチャルリアリティ(Virtual Reality;VR)とは日本語には”人工現実感”, ”仮想現実感”などと訳され, 「現実そのものではないが現実のエッセンスと効力を有するもの」[6]という意味を持つ。VR は、人間が現前する現実空間から現象空間を得るメカニズムと同様のメカニズムを用いて現前しない空間の本質的情報を与えることで、現前するものと等価な現象空間を認識させることができる。現前する現実と等価な現象空間で自在に実験・検証を行うことができる強力なシミュレーション能力は、医療、教育、建築、軍事、建築、学術、航空宇宙、ロボティクス、コミュニケーション、アミューズメントといった非常に幅広い分野で応用の検討、あるいは実用化が行われている。

没入型 VR は、VR においても特に、ユーザの視界を立体映像で覆い、頭の動きに

応じて映像を自然に変化させる等の手法で人工空間に自分が入り込んだかのような高い現実感を得ることができる VR システムのことを指す。映像提示手法として、大型のスクリーンにプロジェクタで投影するタイプ(Immersive Projection Display; IPD)と、頭部にディスプレイを装着するタイプ(Head Mounted Display; HMD)がある。

前者の代表的システムとして、CAVE(Cave Automatic Virtual Environment)[7]が挙げられる。これは 1992 年にイリノイ大学が開発したもので、複数の矩形形状大型スクリーンを前面、両側面、場合によっては上面や下面にも配置し、各スクリーンにプロジェクタで立体映像を投影する。ユーザは体ごと立体映像に囲まれるため、非常に高い没入感が得られる。予め想定された視点位置以外で映像を見ると、スクリーンの境界部分で映像が歪むという欠点があるが、頭部位置の変化に合わせて映像を変化させることで解決できる。しかし、これも観察者が一人の場合に限る。2 人以上の観察者の場合においても、この歪みを低減させるスクリーンとして球状スクリーンなどが提案・開発されているが、視野角は CAVE に劣り、投影境界部分に歪み補正をかけなければならない。共通の欠点として、設備が大型のため広い設置スペースが必要なこと、導入のコストが高額になりやすいことがある。

後者の HMD を用いたシステムは、装置が小型であるため、広いスペースを必要とせずに没入型 VR を実現できるという大きな利点があり、持ち運びも容易である。しかし、視野角がそれほど広くなく没入感 CAVE 方式より劣ること、頭部移動にあわせた映像変化の際に発生する映像提示遅延の影響が大きいこと、解像度の面などで実用的な性能をもつ装置は高額になること、といった欠点がある。

本研究では、ユーザの視線計測に際して頭部に装着する方式の視線解析装置を用いるため、前者の方式を採用し、図 2 に示すような没入型 VR システムを構築した。



(a) (b)  
図 2. スクリーン投影方式の没入型 VR

後述の表 1 において、寸法・解像度等を表記する。スクリーンは前面、左側面、右側面の 3 面構成となっており、図 2 の(a)は没入型 VR システムの外観を撮影したものであり、(b)は実際に映像を投影した様子を撮影している。

### 3. Circle of Confusion

被写界深度効果によってぼけが生じる仕組みを紹介する。物体空間内における眼からの奥行き距離をここでは深度値と言うが、ある一点の深度値が凝視点の深度値と異なる物体空間内の一点は、結像面すなわち網膜に円状に散乱して写像される(図 2)。この円は CoC(Circle of Confusion)と呼ばれ、CoC が眼の最小分解能より大きいとき、ぼけが生じる。すなわち、CoC が最小分解能より小さくなるような物体空間上の範囲が被写界深度となる。CoC の直径  $D_{CoC}$  は以下の式で表される：

$$D_{CoC} = \left| D_{lense} \frac{f(z_{focus} - z)}{z(z_{focus} - f)} \right| \quad (1)$$

where  $\frac{1}{d_{screen}} + \frac{1}{z_{focus}} = \frac{1}{f}$

$D_{lense}$  はレンズ(眼における水晶体)の直径、 $f$  はレンズの焦点距離、 $z_{focus}$  は凝視点の深度値、 $z$  は写像される点の深度値、 $d_{screen}$  はレンズと結像面との距離を表す。凝視点から離れている点ほど、像平面上における CoC の直径は大きくなり、ぼけが大きくなる。

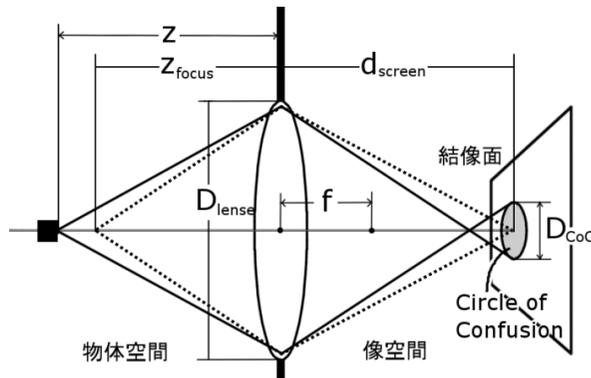


図 3. Circle of Confusion

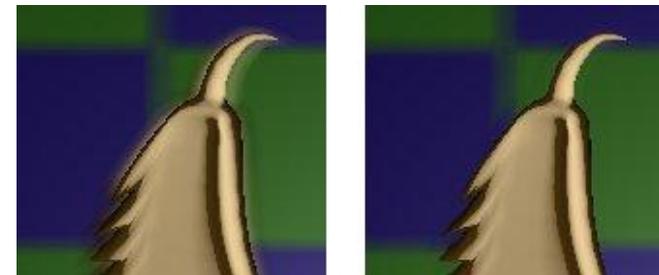
### 4. 先行研究

リアルタイム被写界深度効果の実現手法には、ポストプロセッシングによってぼかしを加える手法や前景と後景のみぼかしてから鮮明な中景と合成する手法[7]などがある。本研究では、Hillaire ら[3]の手法と同様に、典型的なレンズモデルを用いて CoC をシミュレートするポストプロセッシング手法を用いる。

Riguer ら[5]は、この手法の具体的な実装手法を述べるとともに、ポストプロセッシング手法が抱える色漏れという問題を解決しており、主に次の 2 ステップから成る。

- 1) CoC 直径を計算し、その結果および描画色・深度値の 3 つを出力する。
- 2) CoC 直径のフィルタカーネルによって描画色をサンプリングし、ぼけを計算する。このとき、カーネル中心とタップの深度値を比較することで各タップの重みを変更する。その後、ぼかした結果の画像を出力する。

色漏れとは、後景のぼかす際に前景の鮮明な色をサンプリングすることで前景の鮮明な物体の境界がぼけているように見える問題である。Riguer らの手法[5]では、ステップ 2 において鮮明なピクセルの重みを小さくすることで、色漏れの発生を抑えている。



(a)色漏れあり (b)色漏れなし

図 4. 色漏れ

#### 4.1 ステップ 1

ステップ 1 では、各ピクセルの深度値から、(1)式に基づいて CoC 直径を計算し、CoC 直径・深度値を 1 枚のテクスチャに出力する。さらに 3D オブジェクトに対して任意のライティングを施した描画色を別のテクスチャとして出力する。ステップ 2 において、CoC 直径と描画色はぼけの計算に、深度値は色漏れの回避にそれぞれ用いられる。

## 4.2 ステップ2

ステップ2では、ぼけの計算および出力をする。予め生成した直径1のフィルタカーネルにステップ1で算出した  $D_{CoC}$  を積算し、サンプリングするピクセルを決定する。サンプリングピクセルの深度値がカーネル中心の深度値より、小さいときにはそのピクセルの  $CoC$  直径を重みとし、大きいときには重みを1とすることで色漏れの問題を回避する。各ピクセルの描画色に重みを積算し、その総和を重みの和で除算したものをカーネル中心の最終的な描画色とする。同様の処理を画像中のすべてのピクセルに対して施し、一枚のテクスチャとして出力する。

## 5. 実装手法

本研究では、4章で述べた Riguer らの手法[5]を拡張することで没入型 VR における動的被写界深度効果を実現する。なお、ユーザの頭部位置・角度は現段階では、固定を想定している。実装手法は以下の4ステップで構成される：

- 1) 視線解析装置から得られた凝視位置データに事前計算を施す。
- 2) 立体視・没入型 VR 用投影変換およびライティング計算を行い、描画色・深度値を出力する。
- 3) 凝視位置データを基に  $CoC$  直径を計算し、その結果と描画色・深度値を出力する。
- 4)  $CoC$  直径のフィルタカーネルによって描画色をサンプリングし、ぼかし値を計算する。このとき、カーネル中心とタップの深度値を比較することで各タップの重みを変更する。その後、ぼかした結果の画像を出力する。

以上の処理を、前面、左右側面の両眼それぞれに対して行う。すなわち、計6枚の画像を一度の処理で計算し、バッファに出力する。

### 5.1 ステップ1

ステップ1では描画処理の事前準備として、以降のステップで入力に用いる視線データの処理を行う。凝視点深度値の計算に、視線解析装置から出力されるスクリーン上の凝視位置データを用いるが、注意しなければならないことがある。人間は一点を凝視しているときにおいても、実際には眼球が固視微動と呼ばれる高速な微小振動を行っており、最も周波数の高いトレモアという振動は20-100Hzである。したがって、凝視位置は一定でなく、サンプリング周波数がさらに高い視線解析装置による凝視位置データを直に凝視点深度値計算に用いると、高速にぼけが変化する画像が生成され

る。しかし、実際には固視微動の速度に比べて焦点調節の速度は非常に遅いため、好ましい現象ではない。そこで、凝視位置データに高周波成分を除去する適切なLPF(Low Pass Filter)を適用することで凝視点深度値の急激な変化を低減する。

LPFを適用した凝視位置データから凝視点深度値を求める(図5)。本研究では、現段階では、頭部位置および角度は固定であり、視線は常に前方のスクリーン中にあることを想定しているため、視点から凝視点までの距離はピタゴラスの定理より以下の式で求められる：

$$z_{conv} = \frac{d_{scr} - d_{eye}}{d_{eye} - (x_L - x_R)} \quad (2)$$

$z'_{conv}$  は視点から凝視点までの距離、 $z_{scr}$  は視点からスクリーンまでの距離、 $d_{eye}$  は両眼間距離、 $x_L, x_R$  はそれぞれ左右眼のスクリーン上における凝視点の  $x$  座標を表す。求めた  $z'_{conv}$  を定義域 0-1 の深度値に変換することで凝視点深度値  $z_{conv}$  が得られる。しかし、凝視点深度値  $z_{conv}$  は視線解析装置のキャリブレーション誤差の影響から実際の凝視点位置と若干異なる場合がある。したがって、ステップ3で深度値を決定する際の指標として用いるに留める。

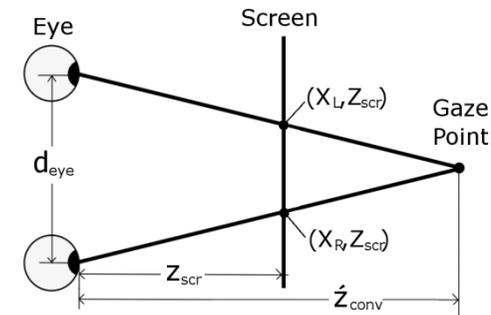


図5. 凝視点深度値の計算

### 5.2 ステップ2

ステップ2では、3D オブジェクトに対する各種行列変換・ライティングを行う。ほとんどのグラフィックライブラリに予め用意されている平行移動・回転・拡大縮小などの一般的な操作に加え、立体視を実現する投影変換および没入型 VR を実現する投影変換を行う。

立体視を実現するには、通常 1 つだけ用意する視点カメラを両眼用に 2 つ用意する必要がある。まず没入型 VR システムに対するユーザの頭部位置・角度を決定し、予め設定された両眼間距離だけ離して左右眼の位置を決定する。左右眼それぞれに対して、没入型 VR 用の投影変換[6]を行うことで没入型 VR 上での立体視が実現できる。本研究では、前面、左側面、右側面の 3 面のスクリーンから成る直方体形状の没入型 VR システムを構築する。

最後に Riguer ら[5]の手法と同様に、任意のライティング計算を行って描画色を決定し、描画色を 1 枚のテクスチャに、深度値を別のテクスチャとして出力する。ただし、グラフィックスライブラリ規定の深度値を用いた場合、ユーザ頭部角度の変化に対して常に一定であるため適切なぼけが得られない。したがって、本研究ではカメラから 3D オブジェクトの各頂点までの、カメラの視線に水平方向の距離を計算し、0-1 の値域に変換したものを深度値として用いる。

没入型 VR システムの床の中心を原点とする座標系において、カメラの視線方向ベクトルを $(E_x, E_y, E_z)$ 、3D オブジェクトの頂点を $(V_x, V_y, V_z)$ とすると、カメラの視線に水平方向の距離  $z'$  および 0-1 の値域に変換した  $z$  は以下の式で表される。

$$z' = \frac{|E_x V_x + E_y V_y + E_z V_z|}{\sqrt{E_x^2 + E_y^2 + E_z^2}} \quad (3)$$

$$z = \frac{z' - n}{f - n}$$

$n$  はカメラから near クリップ面までの距離、 $f$  はカメラから far クリップ面までの距離を表す。ステップ 3 において、描画色は用いず、深度値  $z$  は焦点の深度値を決定するのに用いられる。

### 5.3 ステップ 3

ステップ 3 では、CoC 直径の計算に用いる焦点の深度値を決定する。人間は一般的に凝視位置に焦点を合わせるため、凝視位置付近の深度値を焦点の深度値として用いる。ただし、固視微動や視線解析装置の誤差の影響を考慮して、最も鮮明な像が得られる中心窩に写像される領域の深度値のうち、ステップ 1 で凝視位置データから求めた深度値  $z_{conv}$  に最も近い値を焦点の深度値として採用する：

$$z_{focus} = \min_{x^2 + y^2 < r^2} |z_{conv} - z| \quad (4)$$

$z_{focus}$  は求める焦点の深度値、 $x, y$  はそれぞれ参照ピクセルの  $x, y$  座標、 $z$  は参照ピクセルの深度値、 $z_{conv}$  はステップ 1 で凝視位置データから求めた深度値、 $r$  は中心窩に写像されるディスプレイ上の領域の半径を示す。 $z_{focus}$  と各種パラメータを(1)式に代入して求めた  $D_{CoC}$  を、深度値が格納された入力テクスチャの別の成分に出力する。ただし、中心窩に写像される領域を全探索すると計算コストが非常に高くなりリアルタイム性が損なわれるため、サンプリング数および間隔は読み込む 3D オブジェクトによって変更する。

### 5.4 ステップ 4

ステップ 4 では、Riguer らの手法[5]と同様の手法(4.2 節参照)を用いてぼけを計算する。

## 6. 動作確認

本研究では、視線解析装置に SR Research 社の Eye-Link II を用いた。視線解析装置と立体視用の眼鏡を実際に装着した様子を図 6 に示す。今回は、サンプリング周波数 250Hz のスクリーン上の凝視点データのみを取り扱い、キャリブレーションプログラム等は予め用意されているものを用いた。提案手法を実装した後、動的被写界深度効果の動作を確認するために、表 1 のような環境において簡単な実験を行った。なお、ライティング手法には Phong Shading を用いた。

描画プログラムの実装は OpenGL+C++ を用いて行った。リアルタイム性を実現するために、ステップ 2-4 をプログラマブルシェーダである GLSL を用いて実装し、テクスチャの取り扱いに FBO(Frame Buffer Object)や MRT(Multiple Render Target)、3D オブジェクトの取り扱いに VBO(Vertex Buffer Object)等を実装し、描画処理の高速化を図った。

表 1. 実験環境

視線解析装置		Eye-Link II @ 250Hz
没入型 VR	スクリーン	90.5x69.3" (each for front and left, right)
	解像度	1024x768(XGA)
HostPC : CPU		Intel Pentium 4 @ 2.80GHz
DisplayPC : CPU		Intel Xeon 5675 @ 3.07GHz
DisplayPC : GPU		Quadro FX 4600
3次元立体視環境		BenQ MP772ST@120Hz & Nvidia 3D VISION Pro



図 6. 視線解析装置と 3 次元立体視眼鏡

### 6.1 リアルタイム性の検証

VR において現実感の高いシステムを構築するには、リアルタイム性が重要となることはよく知られている。そこで、本研究における動的被写界深度効果のリアルタイム性を検証するために fps(frame per second)の計測を行った。動画は、わずかに異なる複数枚の画像を高速に切り替えて提示することで作られるが、fps とは 1 秒当たりの画像の描画枚数の事を言い、fps が低いと映像を動画と認識するのが困難になる。fps が増加するに伴って現実感が増加し、60fps 以上は大きく向上しないとされている。本実験で用いた 3 次元立体視環境はフレームシーケンシャル方式と呼ばれるもので、左右の眼に提示する画像を 120Hz で切り替える方式である。すなわち、片眼につき 60Hz で認識されるため、ユーザは 120fps の動画を 60fps として感じる事ができる。したがって、本稿では便宜的に、ユーザが知覚する fps(実際に提示する fps の半分)を計測する。

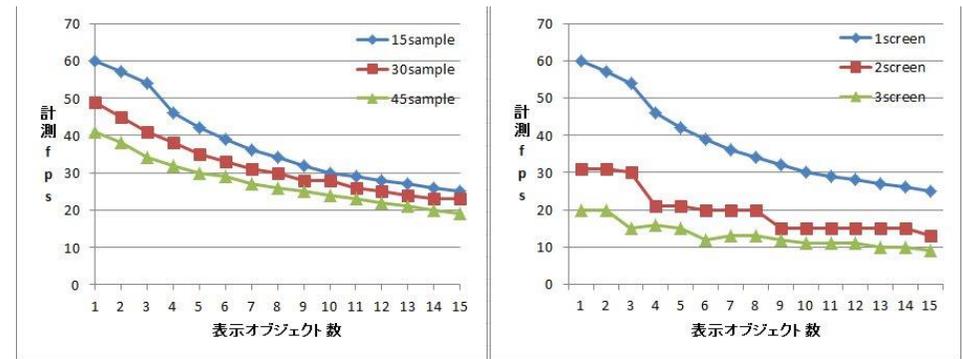
また、描画処理速度に大きく関わる要素は、3D オブジェクトの頂点数と、ぼけの計算で用いるフィルタカーネルのサンプリング数、および表示画面数であるため、ある 3D オブジェクトを 1 つずつ増加した場合の fps をいくつかのサンプル数間、表示画面数間で比較することでリアルタイム性の検証を行った。今回は、頂点数 48335、三角形面数 94920 の 3D オブジェクトを用いた。

### 6.2 実験結果

15 サンプルと 30 サンプル (表示画面数 1) で比較した場合のグラフを図 7(a)に、表示画面数 1-3 枚 (15 サンプル) で比較した場合のグラフを図 7(b)に示す。各グラフは横軸に表示オブジェクト数、縦軸に計測 fps を取っている。表示画面数 1 においてサンプル数 15 のとき 60fps が計測され、高いリアルタイム性を有していることが確認できる。提示画像を動画と認識できる下限 fps を 30fps とするが、サンプル数を増加させた場合においても、ある程度までは下限 fps を上回っていることが確認できる。一方、

表示画面数を増加させた場合、下限 fps を上回るのが困難であることがわかった。その原因として、画面表示の部分の処理は並列化されておらず、表示画面数に正比例して処理時間が増大していることが挙げられる。

この結果は、本研究の動的被写界深度効果を適用するコンテンツの指標として用いることができるが、この効果をさらに有効に適用するためには、fps とフィルタのサンプル数、オブジェクトの頂点数・面数、表示画面数 (あるいは視野角) が VR の現実感にどの程度の影響を与えているかを評価する必要がある。



(a) サンプル数間比較

(b) 表示画面数間比較

図 7. 表示オブジェクト数対計測 fps

## 7. まとめと今後の課題

本研究では、VR の現実感向上を目的として、3 次元立体視環境における視線解析装置を用いた動的被写界深度効果を Riguer らの手法[5]を拡張することによって実現し、その具体的な実装手法を示した。提案手法および描画処理高速化のための技術を OpenGL+C++ を用いて実装した後、fps の計測実験を行った。実験結果より、没入型 VR システムの表示面数を少なくすることで、ある程度のリアルタイム性を有することを確認した。

今後の課題として、リアルタイム性の向上と動的被写界深度効果の有効性の評価が挙げられる。今回、様々な技術を用いて描画の高速化を図ったが、3 面すべてに表示したとき、高いリアルタイム性を有しているとは言えなかった。その理由の 1 つとして、画面表示の並列化を行っていないことが挙げられる。この問題を解決することで幾らかのリアルタイム性の向上が見込めるだろう。

動的被写界深度効果の定量的評価として、没入型 VR システムで用いられる奥行き知覚精度の計測による没入感評価実験を検討している。被験者が、知覚した奥行き距離を回答することで、実際に提示した距離との差異を評価する実験である。動的被写界深度効果の有無によって生じる没入感について、また、被写界深度ぼけを強めることによる没入感の向上について、評価することによって動的被写界深度効果の有効性を明らかにしていきたい。

## 8. 参考文献

- [1] M. S. Landy and L. T. Maloney, E. B. Johnston, and M. J. Young: Measurement and modeling of depth cue combination: in defense of weak fusion, *Vision Res.* 35, pp.389–412, 1995.
- [2] P. Rokita: Generating depth-of-field effects in virtual reality applications, *IEEE Computer Graphics and Applications*, 16(2), pp.18-21, 1996.
- [3] S. Hillaire et al.: Using an Eye-Tracking System to Improve Camera Motions and Depth of Field Blur Effects in Virtual Environments, *Proc. IEEE Virtual Reality Conf. (VR 08)*, IEEE Press, pp. 47–50, 2008.
- [4] 一川 誠: 多様な情報源からの奥行き知覚過程”, *山口大学工学部研究報告* 48(2), pp.231-237, 1998
- [5] G. Riguer, N. Tatarchuk, and J. Isidoro: Real-Time Depth of Field Simulation ShaderX2: Shader Programming Tips and Tricks with DirectX 9, Wolfgang Engel, ed., Wordware, pp. 529–556, 2003.
- [6] 第 13 回「大学と科学」公開シンポジウム組織委員会: バーチャルリアリティ 人工現実感を人間のかかわりを考える, 株式会社クバプロ, 1999
- [7] Cruz-Neira, Carolina, Daniel J. Sandin, and Thomas A.DeFanti: Surround-Screen Projection-Based Virtual Reality: TheDesign and Implementation of the CAVE, *Computer Graphics, SIGGRAPH Annual Conference Proceedings*, 1993.