

# マーカー追跡に基づく拡張現実感システムと そのキャリブレーション

加藤博一<sup>\*1</sup>, Mark Billinghurst<sup>\*2</sup>, 浅野浩一<sup>\*1</sup>, 橋啓八郎<sup>\*1</sup>

An Augmented Reality System and its Calibration based on Marker Tracking

Hirokazu Kato<sup>\*1</sup>, Mark Billinghurst<sup>\*2</sup>, Koichi Asano<sup>\*1</sup> and Keihachiro Tachibana<sup>\*1</sup>

**Abstract** - We describe a fiducial marker tracking method for an augmented reality system. Square markers are used for fast precise registration and a camera model that includes a distortion function is also employed. We also describe a system calibration method. In order to display virtual objects on a head mounted display (HMD) screen the relationships between the HMD, human eyes and head mounted camera are very important. Camera calibration is also vital for accurate marker tracking. We describe a novel HMD calibration technique which does not require much effort from the user and provides very good usability. With good calibration our tracking method provides very accurate tracking and runs at 30 frames/sec on a SGI Octane computer.

**Keywords** : Augmented Reality, HMD Calibration, Marker tracking, 3D display

## 1. はじめに

人と人とのコミュニケーションや共同作業に、コンピュータはますます活用されるようになってきた。共同作業を円滑に効率よく遂行させるために、優れたユーザーインタフェースを持つ支援システムも多く開発されてきた。それらの発展型のものとして、3次元CSCW(Computer Supported Collaborative Work)システムが考えられる。現実に関心を持った状況での作業においては3次元実物体を用いた作業が可能であり、また、コミュニケーションにおいては空間的手がかりが活用できる。このような作業・コミュニケーション環境を取り込んだ3次元CSCWシステムはまだ実用化の段階に至っていないが、人工現実感技術はそのようなシステムの実用化に向けて欠かせない技術となるであろう。

人工現実感技術は、対面する人々が共同作業時に活用する様々な情報、例えば、物体操作、声、ジェスチャーなどを取り扱うことができるために、3次元CSCWシステムに適すると考えられる[1]。DIVE project [2], GreenSpace [3]などの多人数参加型人工現実感システムは、それらの環境の中で共同作業が直感的に行われることを示してきた。しかしそれらのほとんどはユーザーを仮想世界の中へ没入させるものであり、ユーザーを現実世界から遮断し、現実世界の道具の使用を許さないものであった。

Grudinが指摘した[4]ように、CSCWシステムがユーザーに対して従来からの作業形態の変更を強いる場合には、そのシステムは敬遠されやすい。これは従来型の作業

環境とCSCWシステムが提供する作業環境の間の不連続性が原因となる。石井らはCSCWシステムにおけるシームレスの重要性を述べている[5]が、没入型の人工現実感システムが現実世界と仮想世界の間に関心する大きな不連続性を作り出しているのは明らかである。

拡張現実感技術は仮想物体を現実世界の中に重畳させる技術であり、シームレスな3次元CSCWシステムにまさに適する技術として考えられる。Studierstube[6], Transvision[7], AR2 Hockey[8]などのシステムにおいては、ユーザーはシステムが生成する仮想物体と同時に実環境に存在する対話者・共同作業者と見ることができる。つまり、ユーザーは従来通りに現実世界での従来型コミュニケーション環境を維持した上で、人工現実感技術が作り出す新たなインタフェースを扱うことができる。対面での共同作業環境においては、没入型人工現実感システムより、現実世界とのインタラクティブ性が維持される拡張現実感システムの方がユーザビリティの面で優れているとの報告もある[9]。

このような背景の中で、我々は拡張現実感技術を用いた3次元CSCWシステムの開発を行っている。基本的なシステム構成としては、シースルー型のHMD(Head Mounted Display)を表示デバイスとし、仮想物体表示の基準座標となるマーカーの3次元位置をHMDに取り付けられた小型カメラによって得られる画像情報から検出するものである。各マーカーにその識別を可能とする独自のパターンを与えることで、複数の仮想物体の取り扱いを可能とする。図1に遠隔ユーザーのビデオ映像を映し出す仮想スクリーンと共有黒板を使用する例を示す。

しかし、このようなシステムを構築する上で、克服しな

\*1: 広島市立大学情報科学部

Faculty of Information Sciences, Hiroshima City University

\*2: Human Interface Technology Laboratory, University of Washington

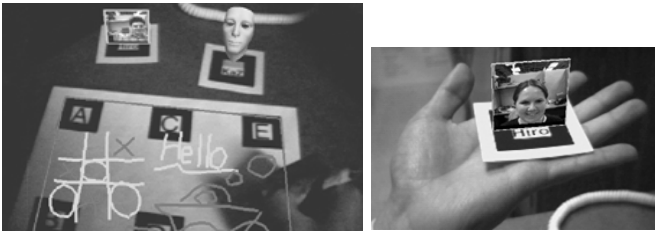


図 1 . 拡張現実感技術を用いたテレビ会議システム  
Figure 1. An example of teleconferencing system using AR technique.

なければならない問題がいくつか残されている . その中で今回我々が特に検討したのは ,

- 1 ) マーカー計測精度 : 画像処理によるマーカーの 3 次元位置検出精度 , およびその処理時間
- 2 ) 表示位置精度 : シースルー型の HMD によって実環境に重畳して映し出される仮想物体の表示位置精度

の 2 点である .

マーカー追跡に基づく拡張現実感システムは , これまでも多く報告されている . State らは , 2 台のカメラと磁気センサーを併用した方法を提案した [10] . 2 種類のセンサーを用いることでお互いの欠点を補う手法である . この手法では , マーカーの 3 次元位置検出に 2 台のカメラによるステレオ視を用いている . Neumann らは 1 台のカメラによる方法を提案した [11] . この中ではマーカー検出のロバスト性に関して多くの検討が行われているが , 複数マーカーの検出が必要とされている . Stricker らは正方形マーカーを用いた追跡手法を提案した [12] . この手法はマーカーの部分的な隠蔽にも対応可能な優れた手法であるが , そのために , また高速化のために 2 次元画像平面内のマーカー位置の線形予測結果に基づく手法を採用している . このため , 手でマーカーを動かす場合のような早い動きには対応できない恐れがある . また , トラッキングに先立つ初期捕捉処理に時間を要するという問題も抱えている . Rekimoto も正方形マーカーを使用した手法を提案している [13] . この方法では , 観測値に対し正方形としての制約からマーカー平面の法線ベクトルの最適化を行っている .

これらの多くはカメラで撮影したビデオ映像に仮想物体を重ねて表示する , いわゆるビデオシースルー型と呼ばれるもので , この場合 , 奥行き方向の計測誤差は比較的問題とならず , 映像の上での 2 次元的なずれのみが問題となる . これに対し我々が想定する , いわゆる光学シースルー型においては , ユーザーの両眼に視差を与えた仮想物体映像を提示することで , HMD スクリーンを透過して見える実世界の 3 次元空間中に 3 次元物体として仮想物体を表示するものである . この場合には , マーカーの 3 次元的な位置計測精度が一層重要となる . ビデオシースルー型と光学シースルー型の長所短所に関しては , Azuma

の文献 [14] にまとめられている . システム構築の際はこれらを考慮して , どちらを選択すべきかを考えるべきである . 我々が想定する 3 次元 CSCW システムでは実世界対象物の取り扱いの容易性を重要視しており , ビデオシースルー型の解像度の低さ , 明るさに対するダイナミックレンジの低さが極めて重要な問題点となり , 光学シースルー型を選んだ .

表示位置精度の問題における困難さは , HMD を装着したユーザーの両眼と HMD スクリーンという人間・機械系を対象にするところにある . 例えば , 視差を与えた仮想物体映像を異なる人間が観察した場合 , 彼らが認知する物体位置は異なるであろう . これは , ユーザーの両眼間距離・焦点距離の個人差 , および , HMD 装着時の HMD スクリーンと目の相対的位置・角度のばらつきによる . そこで , 実物体としてのマーカーを基準に仮想物体を表示し , それをユーザーが正しい位置に見るには , この人間・機械系に存在するパラメータを正確にキャリブレーションする必要がある . もちろん人間の含む系のために , そのキャリブレーションにおけるユーザーへの負荷も考慮されるべき点となる .

佐藤らは , 光学シースルー HMD における位置合わせ手法を提案した [15] . 磁気センサーから得られた情報を画像情報に基づき補正する手法であるが , 目とカメラ視点の距離と比較して観察対象が遠い状況を想定している . Klinker らは , 光学シースルー HMD のキャリブレーション手法を紹介した [16] . この手法ではキャリブレーションの間 , 頭部を固定する必要があり , ユーザーにとって快適な方法とはいえない .

我々は現在 , 3 次元 CSCW システムを構築中であるが , 本論文ではそのシステムの中で使用する正方形マーカーを用いた位置検出手法 , 両眼・HMD 系のキャリブレーション手法について紹介する . また , この方法に基づく仮想物体表示システムを用いて , 簡単な評価実験を行ったのでその結果についても報告する .

## 2 . マーカー位置検出法

### 2 . 1 システム構成と座標系

#### 2.1.1 システム構成

HMD には , Virtual i-O 製 iglasses を用いた . これは , NTSC ビデオ信号を入力とし , 263x234 の解像度を有する . カメラは関東電子工業製の CS9300 を使用し , 焦点距離 4 mm (水平画角約 60 度) のレンズを用いて実験した . 図 2 にカメラを装着した HMD の写真を示す . 画像処理・CG 生成を行うコンピュータには , SGI 製 Octane (R12000, 300MHz) を用い , 320x243 の解像度で画像処理を行った . マーカーには , 白色カード上に書かれた 80mm 四方の黒色正方形を用いた . その内部 40mm 四方は各マーカーを識別するため



図2. カメラを装着したHMD  
Figure 2. HMD with attached small camera.

のパターンとして利用する。

### 2.1.2 取り扱う座標系

図3にシステムが取り扱う座標系を示す。仮想物体はマーカー座標系の上で表現される。カメラ座標系は焦点位置を原点、画像平面に垂直な方向をZ軸、画像のx,y軸に平行な方向を各X,Y軸とする。マーカー座標系で表現された点 $(X_m, Y_m, Z_m)$ は、回転・平行移動でカメラ座標系に変換可能で、その座標系では $(X_c, Y_c, Z_c)$ と表記する。透視変換モデルによって投影される画像平面は理想スクリーン座標系と呼び、この座標系においては $(x_c, y_c)$ に写るものとする。また、拡張現実感システムにおいては広範囲における位置計測が要求され、そのために広角レンズが用いられる。我々が用いるレンズも広角のものであり、樽型歪みが発生する。そこで、理想スクリーン座標系から歪み関数によって変換された座標系を観測スクリーン座標系とし、実際にカメラから取り込まれたデータをこの座標系で表現し、その座標値を $(x_d, y_d)$ で表す。

HMDと目は理想的な透視変換モデルで表現でき、HMDの映像表示面をHMDスクリーン座標系と呼び、その上の座標値を $(x_s, y_s)$ で表す。このHMDスクリーン座標系に対し同一方向にX,Y軸を持ち、垂直な方向にZ軸を有する

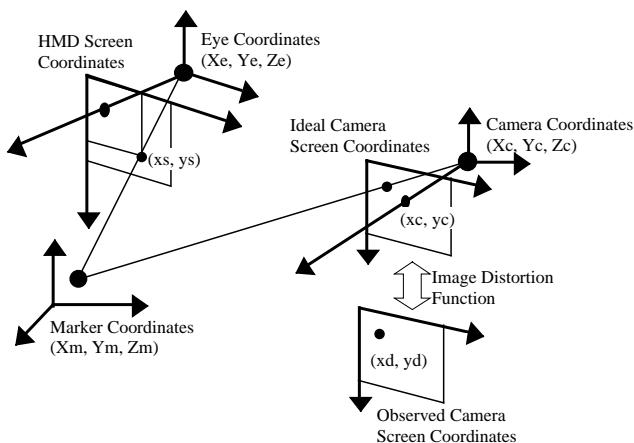


図3 システムが取り扱う座標系  
Figure 3. Camera and HMD model and the related coordinate systems.

目の焦点位置を原点とする座標系を目座標系と呼び、その座標値は $(X_e, Y_e, Z_e)$ で表す。この目座標系の設定においては、目の視線方向をZ軸に考えやすいがHMDスクリーン座標系との間に透視変換モデルを成立させるためには、そのスクリーンとの相対的な関係によって規定されるこの座標系設定が重要となる。また、これら座標系は左右両眼に対して考えなければならないが、ここでは説明の簡素化のために省略する。

### 2.1.3 処理の概要

コンピュータ内での処理は、大きく3つに分割できる。最初の処理は入力された画像から、マーカー領域を抽出し、そのマーカーの理想スクリーン座標系における4頂点座標値を求め、さらにパターン認識によりマーカーを識別するマーカー検出処理である。次の処理は、得られた4頂点座標値からマーカーの3次元位置情報、具体的には、マーカー座標系からカメラ座標系への座標変換行列を求める処理である。最後の処理は、得られた座標変換行列を用いて、HMDの左右のスクリーンの適切な位置に仮想物体を描画する処理である。以下の節では、それぞれの処理について説明する。

## 2.2 マーカー検出処理

### 2.2.1 前処理

入力された画像(320x243画素)に対して、固定閾値による2値化、連結領域ごとの面積・外接長方形計算を行う。これらの処理を1パスで行うアルゴリズムを用いることで高速化を図っている。

### 2.2.2 マーカー抽出

面積値によって巨大領域と微小領域を除外し、外接長方形情報から画像境界に接する連結領域も除外する。残された連結領域に対して輪郭線追跡を行い、輪郭線上の画素位置をすべて記憶する。輪郭線データに対して折れ線近似を行い、4本の線分によって十分な精度で近似できた連結領域をマーカー候補とする。このときの4つの折れ点の座標値を記憶しておく。

### 2.2.3 マーカー識別

図4にマーカーの例を示す。マーカーの中央部に識別用のパターンが描かれているが、このパターンを事前に登録したパターンとのテンプレートマッチングによって識別する。そのためには、まず画像の正規化が必要となる。マーカーは透視変換モデルによって理想スクリーン座標系に投影されるので、マーカー座標系 $X_m-Y_m$ 平面内の点 $(X_m, Y_m, 0)$ は、以下の式で理想スクリーン座標系上の点

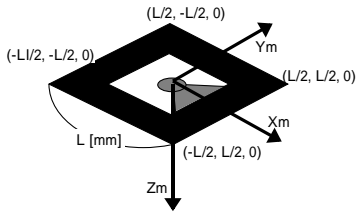


図 4 . マーカーの例

Figure 4. An example of markers.

$(x_c, y_c)$ に変換される .

$$\begin{bmatrix} hx_c \\ hy_c \\ h \end{bmatrix} = \begin{bmatrix} C_{11} & C_{12} & C_{13} \\ C_{21} & C_{22} & C_{23} \\ C_{31} & C_{32} & 1 \end{bmatrix} \begin{bmatrix} X_m \\ Y_m \\ 1 \end{bmatrix} \quad (式 1)$$

マーカーの大きさは既知なので、その情報と2.2.2で検出された4頂点の座標値を用いると、この式のすべてのCの値を求めることができる。従って、マーカー内部のパターンはこの式によって正規化できる。具体的には、マーカー内部のパターン領域を64x64に分割し、その領域に対応する画素値を入力画像から抜き出し、64x64画素のパターン画像を得る。これを16x16に縮小しテンプレートマッチングに使用する。マーカーの回転に対処するために、90度おきに回転したテンプレートを4枚作成しておき、それらと入力画像に対し以下の式で類似度を計算し、最大値をとるものをマーカーの種類および方向と見なす。

$$s^{(i)} = \frac{\sum_{i=1}^N (x_i - \bar{x}) \cdot (x_i^{(i)} - \bar{x}^{(i)})}{\sqrt{\sum_{i=1}^N (x_i - \bar{x})^2} \sqrt{\sum_{i=1}^N (x_i^{(i)} - \bar{x}^{(i)})^2}} \quad (式 2)$$

ここで $x_i$ は画像ベクトルのi番目の要素を示す。 $\bar{x}$ は要素の平均値で、Nは次元数、この場合は256となる。 $x^{(i)}$ はi番目のテンプレート画像を意味する。この式は明るさの正規化された2つの画像ベクトルの余弦を求めることになる。

#### 2.2.4 頂点位置検出

マーカーの各辺に対応する輪郭線データに最小2乗法で直線当てはめを行い、それら直線の交点を頂点座標値とする。この直線当てはめの際、以下の歪み関数による変換を行い、理想スクリーン座標系における頂点座標値を求める。

$$\begin{aligned} \frac{z_d}{z_c} &= \{1 - pz_c^2\} \\ z_c^2 &= (x_c - x_{c0})^2 + (y_c - y_{c0})^2, \quad z_d^2 = (x_d - x_{c0})^2 + (y_d - y_{c0})^2 \\ x_d &= \frac{z_d}{z_c} (x_c - x_{c0}) + x_{c0}, \quad y_d = \frac{z_d}{z_c} (y_c - y_{c0}) + y_{c0} \end{aligned} \quad (式 3)$$

ここで $(x_c, y_c)$ は理想スクリーン座標系での座標値 $(x_d, y_d)$ は観測スクリーン座標系での座標値である。また、 $p$ は歪み率 $(x_{c0}, y_{c0})$ は歪み中心座標値で、この3パラメータはカメラキャリブレーションによって求めておく。ここでの計算では、(式3)の逆変換が必要となるが、その計算は初期値を $(x_d, y_d)$ としたニュートン法の4回の繰り返しで十分な精度が得られることを確認している。観測された正方形マーカー画像では樽型歪みの影響でその辺は曲線となるが、この変換によって辺は直線として取り扱うことができる。

本来は、2.2.2の処理からこの変換式を利用し、理想スクリーン座標系ですべての処理を行ったほうが精度向上が期待できるが、計算時間との関係でそれは行っていない。

#### 2.2.5 後処理

処理を繰り返す中でマーカーのパターン識別に失敗する可能性が考えられる。そこで過去のマーカーの検出位置とその大きさを記憶しておき、検出されたマーカーの位置・大きさが記憶されている値に類似している場合には、同一マーカーと見なす。これにより、マーカーのパターン識別誤りを抑制する。

### 2.3 マーカーの3次元位置推定

これはマーカー座標系からカメラ座標系への変換行列 $T_{cm}$ を推定する問題である。この変換行列は、回転移動成分 $R_{3 \times 3}$ と平行移動成分 $T_{3 \times 1}$ からなる。理想スクリーン座標系とカメラ座標系の間は透視変換モデルで変換でき、その具体的な変換行列は事前にキャリブレーションにより求めておく。これらの座標系の関係は(式4.5)のように表現できる。(式1)はマーカー座標系の $X_m$ - $Y_m$ 平面内座標値と理想スクリーン座標系の関係式であるが、以下の式はマーカー座標系の3次元座標値を取り扱うものである。

$$\begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix} = T_{cm} \begin{bmatrix} X_m \\ Y_m \\ Z_m \\ 1 \end{bmatrix} = \begin{bmatrix} R_{3 \times 3} & T_{3 \times 1} \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_m \\ Y_m \\ Z_m \\ 1 \end{bmatrix} = \begin{bmatrix} R_{11} & R_{12} & R_{13} & T_1 \\ R_{21} & R_{22} & R_{23} & T_2 \\ R_{31} & R_{32} & R_{33} & T_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_m \\ Y_m \\ Z_m \\ 1 \end{bmatrix} \quad (式 4)$$

$$\begin{bmatrix} hx_c \\ hy_c \\ h \\ 1 \end{bmatrix} = P \begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix} = \begin{bmatrix} P_{11} & P_{12} & P_{13} & 0 \\ 0 & P_{22} & P_{23} & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix} \quad (式 5)$$

マーカー座標系におけるマーカーの4頂点の座標値は図4に示すように設定されており、これらに対応する理想スクリーン座標系における座標値もこれまでの処理で得

られている．これら情報から (式4)における  $T_{cm}$  を求めるが，その手順は，

- 1) 回転移動成分  $R_{3x3}$  の推定
- 2) 平行移動成分  $T_{3x1}$  の推定
- 3) 変換行列  $T_{cm}$  の修正

となる．

### 2.3.1 回転移動成分 $R_{3x3}$ の推定

理想スクリーン座標系におけるマーカーの頂点位置から向かい合う2辺の直線の方程式が得られる．それを(式6)のように表し (式5)の  $x_c, y_c$  をこれに代入することで (式7) が得られる．

$$a_1x + b_1y + c_1 = 0, \quad a_2x + b_2y + c_2 = 0 \quad (式6)$$

$$\begin{aligned} a_1P_{11}X_c + (a_1P_{12} + b_1P_{22})Y_c + (a_1P_{13} + b_1P_{23} + c_1)Z_c &= 0 \\ a_2P_{11}X_c + (a_2P_{12} + b_2P_{22})Y_c + (a_2P_{13} + b_2P_{23} + c_2)Z_c &= 0 \end{aligned} \quad (式7)$$

この式はカメラ座標系によって表現される3次元空間中の平面の方程式であるが，3次元空間中のマーカーの辺がこの平面内に存在することを意味する．マーカーの向かい合う2辺は平行なのでその方向ベクトルは一致し，(式7)の2平面の面内方向となる．つまり，(式7)の2平面の各法線ベクトルの外積として計算されるベクトルが，平行2辺のカメラ座標系における方向ベクトルとなる．この計算を2組の平行2辺に対して行うことで，マーカーの隣り合う2辺の方向ベクトル  $U_1, U_2$  を求めることができる．理想的には，この2つのベクトルは直交するが，計測誤差により実際には直交しない．そこで，図5に示すように計測された2つの方向ベクトルに対し，それらを含む平面内で直交する2つの単位ベクトル  $V_1, V_2$  を計算し，これを  $U_1, U_2$  の代わりに用いる．またこの  $V_1, V_2$  の外積を計算することでマーカー平面に垂直な方向の単位ベクトル  $V_3$  も得られる．このとき  $\{V_1, V_2, V_3\}$  が，マーカー座標系からカメラ座標系への回転変換成分  $R_{3x3}$  となる．ただし，2.2.3で行ったテンプレートマッチングの結果から2つ方向ベクトルとマーカー座標系のX軸Y軸の対応関係および正逆方向を判定しておく必要がある．マーカー座標系のZ軸はマーカー平面下向きに設定しているので，カメラ座標系原点から理想スクリーン座標系でのマーカー対角線交点を向く方向ベクトルとの内積が正になる方向に設定する．

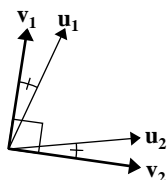


図5．方向ベクトルの修正

Figure 5. Compensation of direction vector.

### 2.3.2 平行移動成分 $T_{3x1}$ の推定

(式4)(式5)を結合し，マーカー4頂点のマーカー座標系での座標値，理想スクリーン座標系での座標値を代入すると， $T_1, T_2, T_3$  に関する1次方程式が8個得られる．行列  $P, R$  が既知なので，これらの式から  $T_1, T_2, T_3$  を計算できる．

### 2.3.4 変換行列 $T_{cm}$ の修正

以上の計算で  $T_{cm}$  が計算されるが，回転行列の計算においてしばしば大きな誤差を伴う．そこで，再度，画像情報を利用し回転行列の修正を行う．(式4)においては回転行列は9個のパラメータで表現されているが，これを3つの回転角 ( $a$ :  $Z_m$  軸の傾斜方向， $b$ :  $Z_m$  軸の傾斜角度， $c$ :  $X_m$ - $Y_m$  平面の  $Z_m$  軸周りの回転角) で表現する．これはZYZオイラー角表現を修正したものである．通常のオイラー角表現ではベクトルの微小変動が回転角に大きな変化を及ぼす場合があるが，この表現ではそのような影響が小さい．(式8)に回転行列を回転角で表した式を示す．

$$R = \begin{bmatrix} \cos a \cos b \cos(c-a) - \sin a \sin(c-a) & -\cos a \cos b \sin(c-a) - \sin a \cos(c-a) & \cos a \sin b \\ \sin a \cos b \cos(c-a) + \cos a \sin(c-a) & -\sin a \cos b \sin(c-a) + \cos a \cos(c-a) & \sin a \sin b \\ -\sin b \cos(c-a) & \sin b \sin(c-a) & \cos b \end{bmatrix} \quad (式8)$$

この式より(式9)が導出でき，回転行列  $R_{3x3}$  から各回転角を求めることができる．

$$\begin{aligned} \cos b &= R_{33} \\ \cos a &= R_{13} / \sin b \\ \sin a &= R_{23} / \sin b \\ \sin c &= (R_{32}R_{13} - R_{31}R_{23}) / (R_{13}^2 + R_{23}^2) \\ \cos c &= -(R_{13}R_{31} + R_{23}R_{32}) / (R_{13}^2 + R_{23}^2) \end{aligned} \quad (式9)$$

そこで，これまでに求めた  $T_{cm}$  を用いて，マーカー4頂点の座標値を(式4,5)に代入することで，その理想スクリーン座標値を計算できる．この計算値と実際に画像処理によって求められた値の誤差の2乗和が少なくなるように  $a, b, c$  の値を修正する．具体的には，山登り法を用いて10回の繰り返し処理により新たな回転成分  $R_{3x3}$  を求める．さらに2.3.2の処理を再適用し平行移動成分  $T_{3x1}$  も更新する．

この処理の理論的な収束性は証明していないが，初期値が適切に与えられること，回転ベクトルの微小変化が3つの回転角  $abc$  に対しても微小変化しか与えないことから，妥当なものと考え，実験的に問題がないことを確認した．

## 2.4 仮想物体表示

仮想物体の表示に関しては，OpenGLグラフィックスラ

イブラリーを用いている[17]。このライブラリーはCG生成のために、透視変換計算を実行する射影行列と平行回転移動を取り扱うためのモデルビュー行列をユーザーが自由に設定できる。そこで、図3に示すカメラ座標系からHMDスクリーン座標系への変換行列を事前にキャリブレーションによって求めておき、この変換行列を射影行列に代入しておけば、逐次観測によって得られるマーカー座標系からカメラ座標系への変換行列  $T_{cm}$  をモデルビュー行列に指定するだけで、マーカー座標形状で表現されたCGモデルを、実環境にそれが存在する場合にHMDスクリーンに投影される場所と同位置に表示することができる。つまり、ユーザーはマーカー座標系を基準に表現された仮想物体を実世界のマーカーに関連してあるかのように見ることができる。

### 3. キャリブレーション原理

#### 3.1 カメラキャリブレーション

カメラキャリブレーションは、これまでも数多く発表されている[18,19 など]が、本論文では下記に示す簡便な方法を用いている。この方法は観測スクリーン座標系と理想スクリーン座標系間の歪み関数(式3)の3パラメータ  $p, x_{co}, y_{co}$  を求める部分と、カメラ座標系から理想スクリーン座標系への透視変換行列  $P$  を求める部分からなる。

##### 3.1.1 歪み関数のキャリブレーション

図6(a)に歪み関数キャリブレーション用パターンを入力画像を示す。(式3)で表される歪み関数では一般に直線が曲線に変換されるが、歪み中心を通過する直線に関しては変換後も直線となる。そこで、キャリブレーション用パターンを上下左右に動かし、パターン内の2直線が直線として写る位置を見つけ、そのときの2直線の交点座標値を歪み中心座標値  $(x_{co}, y_{co})$  とする。この作業では、入

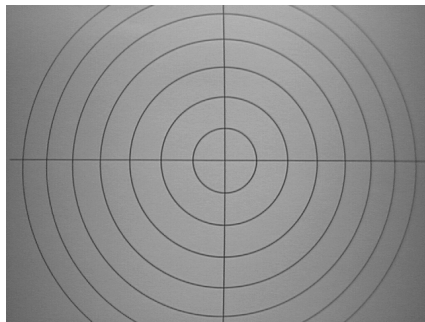
力画像にコンピュータが生成した2直線を重ね合わせ、目視により直線の評価を行った。

また、キャリブレーション用パターンは半径が比例する同心円を描いているが、樽型歪みが起きると外の円ほど半径が小さく写る。図6(a)においてもその様子が確認できる。そこで、コンピュータ内でも(式3)の歪み関数に適切なパラメータを代入し入力画像と同様なパターンを合成し重ね合わせ、両者が一致するように目視でパラメータを調節し、そのときの歪み率をカメラの歪み率として採用する。この作業では、平面をカメラ光軸に対して垂直に置く必要があり、またカメラの歪み率とともに同心円の半径の比例定数も未知数として取り扱う必要があり、時間と注意深さの要求される処理であった。また、目視による作業であるために作業技能の影響も受ける。今後、既存のカメラキャリブレーション手法を参考に改良を行う必要があると思われる。

##### 3.1.2 透視変換行列のキャリブレーション

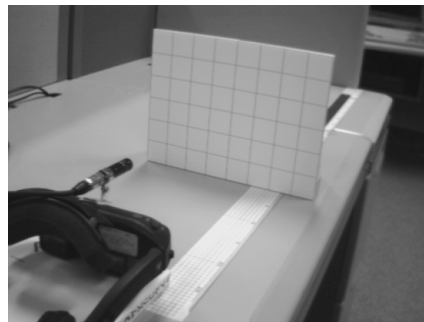
ここでは、図6(b)に示す格子板を使用する。この格子をカメラで撮影し、キーボード操作で各格子線に対して直線フィッティングを行う。実際には、格子が直線として写るのは理想スクリーン座標系なので、上で求めた歪み関数を利用し曲線フィッティングを行うことになる。カメラに対し奥行き方向に格子板を移動させながら、この操作を数回繰り返す。

すべての格子線に対してフィッティングを行った後、理想スクリーン座標系における各格子線の交点位置  $(x_i, y_i)$  およびその点に対応する格子点がわかる。適当に3次元座標系を設定することで格子点を  $(X_i, Y_i, Z_i)$  の形式で表現する。観測されるすべての格子点に対してその3次元座標値とスクリーン座標系での座標値を(式10)に代入することで、パラメータ  $C_{ij}$  に関する一次方程式が多数得られ、その値を計算できる。この行列  $C$  は(式11)に示すように透視変換成分と回転・平行移動成分に分割することが



(a) 歪み補正用パターン入力画像

(a) A pattern for compensation of image distortion.



(b) 透視変換キャリブレーション用格子板

(b) A grid board for calibration of

perspective transformation.

Figure 6. Camera calibration

でき、 $\mathbf{R}$ の回転行列としての制約を用いることで、行列 $\mathbf{P}$ を計算できる。これが求めるべき透視変換行列になる。

$$\begin{bmatrix} hx_c \\ hy_c \\ h \\ 1 \end{bmatrix} = \begin{bmatrix} C_{11} & C_{12} & C_{13} & C_{14} \\ C_{21} & C_{22} & C_{23} & C_{24} \\ C_{31} & C_{32} & C_{33} & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_f \\ Y_f \\ Z_f \\ 1 \end{bmatrix} \quad (式 10)$$

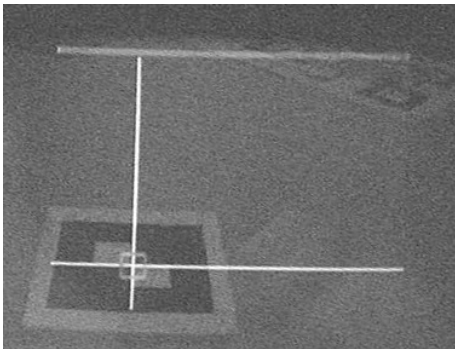
$$\begin{bmatrix} C_{11} & C_{12} & C_{13} & C_{14} \\ C_{21} & C_{22} & C_{23} & C_{24} \\ C_{31} & C_{32} & C_{33} & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} = k \begin{bmatrix} P_{11} & P_{12} & P_{13} & 0 \\ 0 & P_{22} & P_{23} & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} R_{11} & R_{12} & R_{13} & T_x \\ R_{21} & R_{22} & R_{23} & T_y \\ R_{31} & R_{32} & R_{33} & T_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (式 11)$$

### 3.2 HMD キャリブレーション

カメラ座標系からHMDスクリーン座標系への変換行列を求めることが、HMDキャリブレーションである。2.1.2で述べたように、HMDスクリーンと目の焦点位置を原点とする3次元座標系の関係は完全な透視変換モデルで考えることができる。また、この目座標系とカメラ座標系は、回転・平行移動で変換できる。つまりこれらの関係は(式10)と全く同じ式で表現できる。つまり、カメラ座標系で表現される3次元座標値とそれに対応するHMDスクリーン座標値の組が多数あれば、変換行列は計算可能である。



(a) キャリブレーション中のユーザー  
(a) A user in calibration procedure.



(b) ユーザーの視点からの眺め  
(b) An image from user viewpoint.

図7. HMDキャリブレーションの様子  
Figure 7. HMD calibration.

HMDスクリーンの適当な位置にヘアカーソルを表示する。HMDを装着したユーザーは手にマーカーを持ち、その中心がヘアカーソルに一致して見えるように動かす。うまく一致したところで、マウスクリックを用いてコンピュータにその情報を知らせる。図7(a)は作業時のユーザーを示す。(b)はユーザーが見る眺めを示す。ただし(b)は実際の作業中の画像ではなく、説明のために模擬して撮影したものである。コンピュータではマーカー追跡プログラムを動かしておき、ユーザーがマウスクリックした時点の $T_{cm}$ を記録しておく。この変換行列の中の平行移動成分 $T_{x1}$ が、マーカーのカメラ座標系での位置になっている。つまり、ヘアカーソルの座標値とこのマーカー位置の組を複数求めることで、カメラ座標系からHMDスクリーン座標系への変換行列を求めることができる。

従来HMDのキャリブレーションは、頭部の固定などユーザーに苦痛を与えるものであったが、この手法は頭部固定の必要もなく、数回試行すれば、短時間で高精度なキャリブレーションを遂行できる。

## 4. 精度評価実験

### 4.1 マーカー計測精度の評価

マーカーの計測精度を評価するために、一辺が80mmのマーカーを奥行き方向に動かし、その計測値を記録した。カメラ前方約150mmの地点に基準点を置き、そこから100mmごと6カ所で計測した。マーカーは画像のほぼ中央に写るように設置した。画像位置情報については基準点からの相対距離を求めた。一地点について連続した100回の計測値の平均と標準偏差を算出した。マーカーに奥行き方向0,30,45,60,85度の傾斜角を与え、計測を繰り返した。なお、カメラに対して傾斜角0度の場合、最もカメラに近い地点においては、画像中のマーカー領域は面積比で約25%であった。傾斜角85度の場合には、500mm以降は画像中のマーカー領域が微少となり計測不能であった。図8に奥行き計測誤差、図9に傾斜角計測結果を示す。カメラから遠ざかるほど誤差が大きくなるのがわかる。特に500mm以降の位置誤差の増加が目立つ。傾斜角に関しては0度の場合の計測値の誤差が大きいのがわかる。また、各計測地点における計測値の標準偏差は最大のもので、奥行き計測誤差に関しては3.5mm、傾斜角に関しては1.3度であった。

本手法は3次元CSCWシステムを目指すものであり、ユーザーの手の届く範囲にマーカーを置くことを想定している。このことから、カメラ前方150mm～550mmのほとんどの場合で、位置誤差に関して10mm以内、角度誤差に関して3度以内という精度は満足できるものと考えられる。さらに遠くの位置にマーカーを置く必要がある場合には、より大きなマーカーを用いる必要がある。また、画



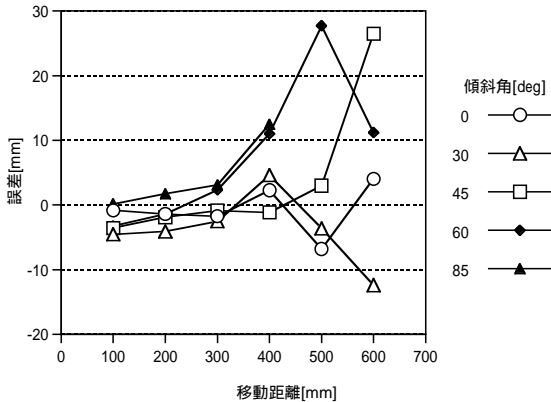


図 8 . 奥行き計測誤差

Figure 8. Errors of position estimation.

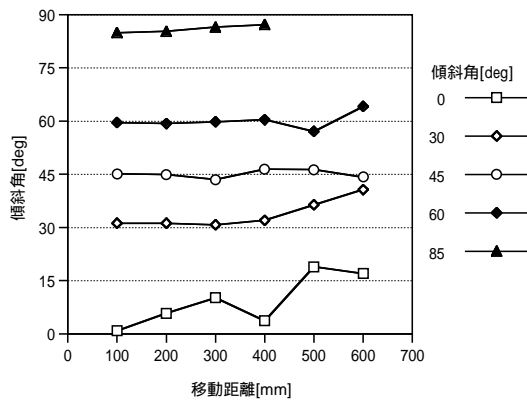


図 9 . 傾斜角計測誤差

Figure 9. Errors of rotation estimation.

像平面に平行な方向への移動に対する位置計測誤差は奥行き方向のそれに比べ小さく、カメラ前方約500mmの地点で3mm以内であった。計測値の標準偏差が大きな場合には、表示した仮想物体が振動しユーザーに不快感を与えるが、この実験結果ではそれが小さいことがわかる。

本手法では画像の歪み補正を行っているが、その効果を客観的に調べるためには画像周辺部における精度評価が本来不可欠である。しかし今回の実験では設備面の制約から画像周辺部における計測精度評価が行われていないので、今後さらに厳密で広範囲にわたる精度評価をすべきと考えられる。ただし、基準点におけるマーカーの4頂点は比較的画像の周辺部に写るものであり、このことを考慮すると画像周辺部における精度が著しく悪化することはないと思われる。実際に本手法の精度が満足できるものか否かは、ユーザーがどの程度の誤差を感じるかによって判断されるべきである。それに関しては次節で述べる。

#### 4. 2 表示精度の評価

80mm 四方の黒色正方形マーカーの書かれた 120mm 四方の厚紙を用い、この黒色正方形部分と完全に一致する

ように青色の正方形を表示するプログラムを作成して評価した。青色正方形の表示には2.4で述べた方法を用いた。ユーザーは実際の黒色の正方形と合成された青色正方形を同時に見ることができ、両者の3次元空間内でのずれの程度を口頭で述べてもらい記録した。ユーザーには定規を渡しておき、それとの比較によってずれの程度を調べてもらった。ずれには、平行移動のみではなく回転成分も含まれるが、マーカー座標系原点中心に関する回転成分は、マーカー計測誤差に起因し、HMDキャリブレーションには関与しないので、平行移動量のみを数値として記録した。その他のユーザーの報告なども同時に記録し後半にまとめている。

課題は、

- (1)マーカーを手に持った場合  
(目・マーカー間距離約300mm)
- (2)机においた場合  
(目・マーカー間距離約400mm)
- (3)手の届かない位置にマーカーをおいた場合  
(目・マーカー間距離約800mm)

の3つで行った。実験手順は、

- a) 標準パラメータによる評価、その後、休憩
- b) キャリブレーション実施、続けて評価
- c) HMDを取り外し、休憩後、再装着・評価

とした。標準パラメータとは事前に第3者のキャリブレーションによって得られたパラメータである。キャリブレーションには、HMDスクリーン上の5点で各2回ずつ計測を行い、片眼につき10点のデータでキャリブレーションを行った。テストユーザーは表1に示す6名である。キャリブレーション方法に関しては、実演を交えながら事前に説明した。表2にテストユーザーからの報告をまとめた。Aのユーザーは左目の視力が弱く、マーカーをほとんど認識できずキャリブレーションに失敗した。他のユーザーは2～3分でキャリブレーションを終えた。表からわかるように、第3者のキャリブレーションパラメータでは、仮想物体の位置合わせに大きな誤差が発生しているが、キャリブレーション後には誤差がほとんど観測されなくなった。HMD再装着後の誤差はわずかに増加傾向が観測された。キャリブレーション実施後の位置合わせ精度の改善には、ユーザーは満足のいく感想

表 1 . テストユーザー

Table 1. Test user

	視力(右,左)	使用経験
A	0.5, 0.2	なし
B	1.5, 1.5	1回
C	1.2, 1.2	なし
D	0.5, 0.5	なし
E	0.5, 0.5	なし
F	1.5, 1.5	1回



表 2 . ユーザーの評価結果  
Table 1. Results of user evaluation

	所要時間 (分)	実験	課題 1 (mm)	課題 2 (mm)	課題 3 (mm)
A	5	(a)	5	5	20
		(b)	左目の視力不足で キャリブレーション失敗		
		(c)			
B	3	(a)	20	20	20
		(b)	0	0	5
		(c)	3	3	6
C	2	(a)	20	30	35
		(b)	0	5	5
		(c)	0	5	5
D	2	(a)	2	2	2
		(b)	2	2	2
		(c)	0	2	2
E	2	(a)	5	5	10
		(b)	0	0	0
		(c)	0	1	2
F	2	(a)	10	10	10
		(b)	0	0	0
		(c)	1	2	3

を述べた。本研究の目的からすると、ユーザーの主観評価が重視されるべきではあるが、表 2 の結果を素直に受け入れることはできない。ユーザーは実際よりも好意的な報告をしている可能性がある。その理由としては、

- 1) 仮想物体には視差情報として奥行き感を与えるが、本来HMDが有するスクリーンの結像位置は2~3mの位置になるために、数十cmの距離にあるマーカーに焦点を合わせることで、仮想物体は常にぼけて見える。つまり、多少のズレは、この映像のぼけが吸収すると思われる。
- 2) ユーザーはこのような技術に親しんでなく、結果が予想を超えていた。

などが考えられる。特に1の要因は課題として残り、結像位置の変更可能なHMDを用いると結果が悪くなる恐れがある。

また、マーカー追跡の精度に比べて、ユーザーの主観評価がよりよい結果になっているのは非常に興味深い。特に目・マーカー間距離が800mmの場合においては、マーカーの計測値には大きな誤差が含まれていると考えられる。しかし、ユーザーは大きなズレを報告していない。ユーザー自体の視覚の奥行き方向に対する精度が良くないことも原因と考えられるが、本HMDのキャリブレーション方法自体にも原因が考えられる。2.1で示したように、仮想物体の表示にはカメラ座標系におけるマーカーの3次元位置計測、およびカメラ座標系からHMDスクリーン座標系への座標変換の2段階の処理が必要なので、本来はそれぞれの部分における高精度化が要求され、全体としての精度は各部分の精度を上回ることは考えにくい。しかし、本手法においてはHMDキャリブレーション自体にマーカーの3次元位置計測結果を用いている。このことは、マーカーの3次元位置計測に系統的な誤差が含まれ

る場合においては、その誤差を含めてキャリブレーションが行われることを意味している。つまり、マーカーの3次元位置計測結果に誤差が含まれていても、HMDスクリーン座標系においては正しい位置に仮想物体を表示し得る場合もある。この点の厳密な検討はモデルとして用いた式の数学的な解析が必要となるであろう。

このような処理系の特性は、今回のようなマーカー座標系と頭部に固定された座標系のみを用いるシステムに対しては有利ではあるが、物理空間内の静止点におかれた世界座標系を必要とするシステムにおいてはカメラ座標系での正確な3次元位置が重要となり問題となるであろう。そのような場合には、2台をカメラを用いたステレオ視による高精度な3次元位置計測が要求されると思われる。

また評価実験後のユーザーへのグループインタビューによって指摘された主な問題としては、

#### 1) キャリブレーションに関して：

- ・HMDの透過率が低くマーカーが見にくい。
- ・キャリブレーション中にうまく処理が行われているかどうか不安。
- ・手がふるえる。片目を閉じるのが辛い。
- ・両眼で20回という計測が苦痛。すべてで4回以内または無操作がよい。
- ・何回か試行すると、より短時間で精度の良いキャリブレーションができると思う。

#### 2) 拡張現実感システムに関して：

- ・仮想物体の表示時間遅れが非常に気になる。
- ・遠くに置いたマーカー上の仮想物体に関しては位置ずれより振動が気になる。
- ・HMDの視野が狭い。暗い。

などが報告された。キャリブレーションに関しては苦痛という報告があるが、キャリブレーション時間が2~3分という結果は、頭部固定の従来方式に比べると悪くないと思われる。

### 4.3 処理速度

このマーカー追跡手法は、SGI社Octane(R12000,300MHz)の上で30フレーム/秒の速度で動作した。その状態でCPUは約50%の待機時間を有しており、仮想物体表示を行った場合でも、データ量に依存はするがこの速度は維持できると思われる。ただし、HMDへの画像伝送のためにモニター映像信号をNTSCにダウンコンバートし、HMD側ではフィールド信号を左右のHMDに振り分けるという動作が行われており、ユーザーの報告にもあるように表示に時間遅れが生じている。遅れ時間の厳密な計測はしていないが、我々のシステム構成においては、0.1~0.3秒程度の遅れ時間が生じている。こ

の問題にソフトウェアで対応するには、何らかの予測機構を組み込む必要があると思われる。

## 5. まとめ

本論文では 拡張現実感システムのためのマーカー追跡手法とキャリブレーション手法について述べた。正方形マーカーの頂点情報を用いた位置姿勢推定法を提案し、それが十分に高速かつ高精度であることを確認した。また、光学シースルーHMDのキャリブレーション手法に対しても、頭部固定の必要がない容易で簡便な手法を提案した。ユーザーテストの結果からも、このキャリブレーション手法により、3次元仮想物体が現実世界の適切な位置に表示されることが確かめられた。今後は3次元CSCWシステムの構築に向け、表示の遅れ時間に関する対策、マーカーの部分隠蔽問題などに取り組んでいく方針である。

## 参考文献

- [1] A. Wexelblat, "The Reality of Cooperation: Virtual Reality and CSCW", Virtual Reality: Applications and Explorations. Edited by A. Wexelblat. Boston, Academic Publishers, 1993.
- [2] C. Carlson, and O. Hagsand, "DIVE - A Platform for Multi-User Virtual Environments", Computers and Graphics, Nov/Dec 1993, Vol. 17(6), pp. 663-669.
- [3] J. Mandeville, J. Davidson, D. Campbell, A. Dahl, P. Schwartz, and T. Furness, "A Shared Virtual Environment for Architectural Design Review", CVE '96 Workshop Proceedings, 19-20th September 1996, Nottingham, Great Britain.
- [4] J. Grudin, "Why CSCW applications fail: Problems in the design and evaluation of organizational interfaces", Proceedings of CSCW '88, Portland, Oregon, 1988, New York: ACM Press, pp. 85-93.
- [5] H. Ishii, M. Kobayashi, K. Arita, "Iterative Design of Seamless Collaboration Media", Communications of the ACM, Vol 37, No. 8, August 1994, pp. 83-97.
- [6] D. Schmalsteig, A. Fuhrmann, Z. Szalavari, M. Gervautz, "Studierstube - An Environment for Collaboration in Augmented Reality", CVE '96 Workshop Proceedings, 19-20th September 1996, Nottingham, Great Britain.
- [7] J. Rekimoto, "Transvision: A Hand-held Augmented Reality System for Collaborative Design", Proceeding of Virtual Systems and Multimedia '96 (VSMM '96), Gifu, Japan, 18-20 Sept., 1996.
- [8] T. Ohshima, K. Sato, H. Yamamoto, H. Tamura, "AR2Hockey: A case study of collaborative augmented reality", Proceedings of VRAIS'98, pp.268-295 1998.
- [9] M. Billinghurst, S. Weghorst, T. Furness, "Shared Space: An Augmented Reality Approach for Computer Supported Cooperative Work", Virtual Reality Vol. 3(1), 1998, pp. 25-36.
- [10] A. State, G. Hirota, D. T. Chen, W. F. Garrett, M. A. Livingston, "Superior Augmented Reality Registration by Integrating Landmark Tracking and magnetic Tracking", Proceedings of SIGGRAPH96, pp.429-446, 1996.
- [11] U. Neumann, S. You, Y. Cho, J. Lee, J. Park, "Augmented Reality Tracking in Natural Environments", Mixed Reality - Merging Real and Virtual Worlds (Ed. by Y. Ohta and H. Tamura), Ohmsha and Springer-Verlag, pp.101-130, 1999.
- [12] D. Stricker, G. Klinker, D. Reiners, "A Fast and Robust Line-based Optical Tracker for Augmented Reality Application", Proc. of 1st International Workshop on Augmented Reality, Nov.1, 1998.
- [13] J. Rekimoto, "Matrix: A Realtime Object Identification and Registration Method for Augmented Reality", Proceedings of Asia Pacific Computer Human Interaction 1998 (APCHI'98), Japan, Jul. 15-17, 1998.
- [14] R. Azuma: "A Survey of Augmented Reality", Presence: Teleoperations and Environments, 6, 4, pp.355-385, 1997
- [15] 佐藤, 山本, 田村, "カメラと3次元センサの組み合わせによる現実空間と仮想空間の位置合わせ手法", 日本バーチャルリアリティ学会論文誌, Vol.4, No.1, pp.295-301, 1999.
- [16] G. Klinker, D. Stricker, D. Reiners, "Augmented Reality: A Balancing Act Between High Quality and Real-Time Constraints", Mixed Reality - Merging Real and Virtual Worlds (Ed. by Y. Ohta and H. Tamura), Ohmsha and Springer-Verlag, pp.325-346, 1999.
- [17] J. Neider, T. Davis, M. Woo, "OpenGL Programming Guide(日本語版)", 星雲社, 1993.
- [18] R. Y. Tsai, "A versatile Camera Calibration Technique for High-Accuracy 3D Machine Vision Metrology Using Off-the-Shelf TV Cameras and Lenses", IEEE Journal of Robotics and Automation, Vol. RA-3, No. 4, pp.323-344, 1987.
- [19] J. Weng, P. Cohen and M. Herniou, "Camera calibration with distortion models and accuracy evaluation", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 14, No. 10, pp. 965-980. 1992.

(1999年7月30日受付)