

2009年度 卒業論文

複眼カメラによる恒常視線映像生成

2010年2月15日

大阪大学 基礎工学部
システム科学科 生物工学コース
大城研究室
森口裕樹

主査: _____ 日付: _____

副査: _____ 日付: _____

概要

全方位カメラの開発により広範囲の映像を撮影することができるようになった。全方位カメラの撮影範囲の広さを活かした応用研究も数多くなされている。応用研究には、市街地映像マップに見られるように全方位カメラを移動させながら撮影を行う事例が多い。しかし、カメラを移動させながらの撮影ではカメラの位置・姿勢を推定することが困難である。また、時間的に連続した画像データを大量に扱うため、データ量が大きくなり 実時間処理を行うことは難しい。処理量の問題を解決するためには、画像情報を用いずに、簡易にカメラの位置・姿勢を推定する手法が有効である。

本研究では、映像の撮影を行いながらカメラの姿勢を同時に計測できるデバイスを製作した。デバイスは4台の無線カメラを持つ。カメラは正四面体の各頂点にレンズが外向きになるように配置し、正四面体を透明な球殻で覆った。デバイスは球形をしており無動力で回転しながら進行する。また、デバイスの中心部には自身の姿勢を測定するために3軸加速度センサを取り付けた。加速度センサにより重力ベクトルを検出することで、回転した角度と回転軸を求め、姿勢を推定することができた。最後に、撮影した映像を求めたカメラの位置・姿勢を用いて補正した。また、補正した結果としてカメラの位置・姿勢に関わらず視線を一定に保った映像を提示した。

Abstract

Omnidirectional images can be obtained easily by omnidirectional camera. Various resarches,such as free viewpoint images,urban picture map have been carried out,using omnidiretional cameras.

Omnidirectional cameras are used as a video recorder in some applicational researches. However, to measure the position of the camera is difficult while the camera moves. Besides, because of huge size of the video data, it is also difficult to calculate the position of the camera in real-time. To solve this problem, estimation of the position of cameras without video data is efficient.

In this research,a device which can take video and the position simultaneously was made . The device has four cameras which can communicate without wire and acrylic sphero shell. The device is sphere, so the device can moves without power. The cameras were arraged on the four verteices of regular tetrahedron. An accelerometer was set at the center of the device in order to measure the orientation. The pose was calculated from the angle and the axis of the rotation. In result, constant view video was produced by correcting the video from the pose of the device

目次

第1章	はじめに	1
第2章	カメラによる撮影と映像の生成	2
2.1	広範囲な映像の撮影	2
2.2	映像の生成	3
2.3	まとめ	6
第3章	複眼カメラによる恒常視線映像生成	7
3.1	回転するカメラと画像の傾き	7
3.2	傾きの補正	8
3.3	画像の回転方向	11
第4章	実装	12
4.1	システム構成	12
4.2	デバイス概要	12
4.3	カメラ配置	15
4.4	デバイス座標系・カメラ座標系変換行列	17
4.5	画像の表示	18
第5章	結果	22
第6章	おわりに	24
	謝辞	25
	参考文献	26

図 目 次

2.1	全方位カメラ	2
2.2	全方位型マルチカメラシステム	3
2.3	全方位映像	4
2.4	任意視点映像	5
2.5	恒常視線映像	6
3.1	画像の傾き	7
3.2	傾きの大きさ	8
3.3	座標系	9
3.4	加速度センサー座標系	9
3.5	傾き算出	10
3.6	外積と回転方向	11
4.1	システム構成	12
4.2	デバイス外観	13
4.3	無線カメラ	13
4.4	加速度センサー	13
4.5	デバイス見取り図	15
4.6	デバイス断面図	15
4.7	デバイス断面と正四面体断面	16
4.8	頂点設定	16
4.9	ポリゴンの配置	18
4.10	デバイスの回転とポリゴン	19
4.11	デバイス初期姿勢	20
4.12	視体積	21
5.1	撮影する風景	22
5.2	補正画像	23

表 目 次

4.1 装置の仕様	14
---------------------	----

第1章 はじめに

現在、計測技術、画像処理技術、コンピュータの演算処理能力などの向上に伴い、多様な撮影方法および映像の生成方法が研究されている。

まず、広角レンズや魚眼レンズといった特殊なレンズを用いることにより広範囲の画像を撮影することが可能になった。また、全方位カメラの開発によりカメラの周囲 360deg の映像の撮影が可能になった [1]。全方位画像の広い撮影範囲を活用して、全方位カメラを用いた応用研究が数多く行われている。例えば、ステレオ計測や任意視点映像生成に全方位画像を利用することで広範囲のモデリング、レンダリングを行うことが可能になった。

しかし、全方位画像を用いたとしても、撮影した画像をもとにモデリングやレンダリングを行う場合、カメラの位置・姿勢の情報が不可欠となる [2][3]。カメラの位置・姿勢情報はマーカーや対応点を利用して算出できる。画像を元に位置・姿勢を算出しない場合は、あらかじめ決まった位置にカメラを固定しておく手法がある。しかしながら、多数のカメラを用いる場合はカメラの配置に労力を要し、現実的でない。カメラの位置合わせは各カメラに写り込んだ画像の対応点の探索により行われることが多い。一方、カメラの位置情報を利用しない全方位画像の応用手法に市街地映像マップがある [4][5]。市街地映像マップでは、全方位カメラを搭載した車で市街地を走行し、撮影した画像の張り合わせを行う。実データを地図に取り入れることで、建造物の外観や風景を提示できる。市街地映像マップの場合は三次元モデリングを行う際ほどカメラの位置・姿勢を正確に求めなくてもよい。基本的には画像中の特徴点を対応させて画像を張り合わせていくことで、シームレスな画像が生成される。

また、画像中の特徴点を利用してカメラの位置・姿勢を推定しようとする研究も行われている [7]。森らの研究では、カメラを自由に動かしながら撮影を行い、特徴点の移動量やベクトルからカメラの位置・姿勢を推定することができる。また、推定したカメラの位置・姿勢を元に画像を補正することで、視線を一定に保った、安定した動画を生成している [8]。

従来の手法では、静止画像や動画を元にカメラの位置・姿勢を求めたり、特徴点の対応付けの処理を行ったりすることで映像の生成を行う [9]-[12]。しかし、これらの処理には時間と労力を要する。特に、動画のような、時間的に連続的な画像を大量に扱う場合は、データ量も膨大になるため画像ベースの処理を行うのは現実的でない。画像の張り合わせを行う場合も画像ベースでなくカメラの位置・姿勢をもとに行うことで処理量を減らすことが得策である。

本研究では、全方位の映像とカメラの位置・姿勢を画像ベースでない手法で推定するデバイスを製作する。また、推定した位置・姿勢をもとに撮影された映像の補正を行いカメラの位置・姿勢に関わらず視線を安定させて動画を生成することを目標とする。これを恒常視線映像と呼ぶ。恒常視線映像とは、カメラが傾いたり回転しても、カメラの姿勢に基づいた補正により、常に上下左右方向を保った状態で表示される映像のこととする。

第2章 カメラによる撮影と映像の生成

カメラの撮影技術の進歩により広範囲の映像を撮影することが可能になった。例えば、レンズやミラーを用いて、広範囲の空間を一度に撮影することができる。また、多数のカメラを同時に用いることで全方位の映像を生成する技術も開発されている。一方、映像の提示手法に関しても、画像処理技術や計測技術の発展に伴い様々な手法が提案されている。

2章では、広範囲の映像を撮影する技術と、映像の生成手法について述べる。

2.1 広範囲な映像の撮影

2.1.1 広角レンズ

広範囲な映像を撮影する方法の一つに、広角レンズや魚眼レンズを用いる方法がある。広角レンズは光を屈折させることで広い角度から光を集め受光素子に送ることができる。広角レンズは構造は一般的なレンズと同じであるが、一般のレンズよりも焦点距離が短く、画角を大きくとることができる [13][14]。

2.1.2 全方位カメラ

全方位カメラを用いることでカメラを移動・回転させることなく一度の撮影でカメラの水平方向の周囲 360deg の画像を取得することができる。全方位カメラにはミラーが搭載されている。ミラーは水平方向について周囲 360deg から入ってくる光を受光素子に反射する (図 2.1)。

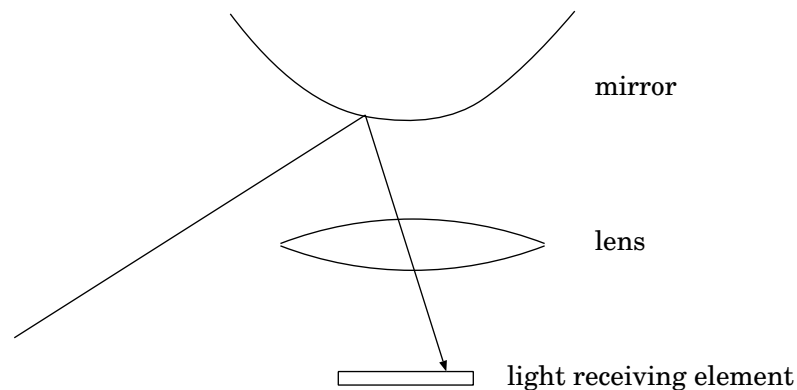


図 2.1: 全方位カメラ

2.1.3 全方位型マルチカメラシステム

全方位型マルチカメラシステムは複数のカメラの画像をつなぎ合わせることで全方位の画像を取得するシステムである。全方位型マルチカメラシステムでは用いるカメラの台数を増やすことで一度に撮影できる範囲を広げることができる。また、全方位型マルチカメラシステムではミラーをレンズの前面に配置する必要がないため、ミラーによる死角ができないという利点がある。

Point Grey Research 社の Ladybug では 6 台のカメラを同時に扱うことで、デバイスの周囲の全天球の映像を取得することができる (図 2.2)。

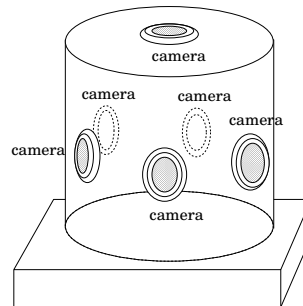


図 2.2: 全方位型マルチカメラシステム

2.2 映像の生成

2.2.1 全方位映像

全方位映像は全方位カメラによって撮影することができる。360 度パノラマ写真は全方位映像の一つである。また、全方位カメラや全方位マルチカメラシステムの応用が進み、市街地映像マップの生成等に用いられるようになった。市街地映像マップでは地図と市街地の写真などの実データを統合することで、従来の地図と違い、道路や建物の図だけでなく建造物の外観や風景を閲覧することが可能である。例えば、三次元空間に配置したポリゴンに市街地の写真をテクスチャとして張り付けることで市街地映像マップが生成できる [6]。

全方位映像生成法の研究

名古屋大学の佐藤らは、車載の全方位カメラを用いて市街地の同経路を多数回走行し撮影を行った。撮影を行うとともに GPS で全方位カメラの位置を特定し、さらに画像間の対応をとり、画像をつなぎ合わせることで正確な地図を生成することに成功している [4]。

google 社の開発した google street view では世界各地の街並みをユーザの任意の地点から閲覧することができる 2.3。google street view では佐藤らの研究のようなシームレスな地図は構成されていないが、数十メートルごとの地点における全方位映像を提示している。google street view ではマルチカメラシステムを搭載した自動車では撮影を行い、撮影した画像を特徴点を元につなぎ合わせることで半球状のパノラマ画像を生成させ、半球状の映像を地図上に配している。

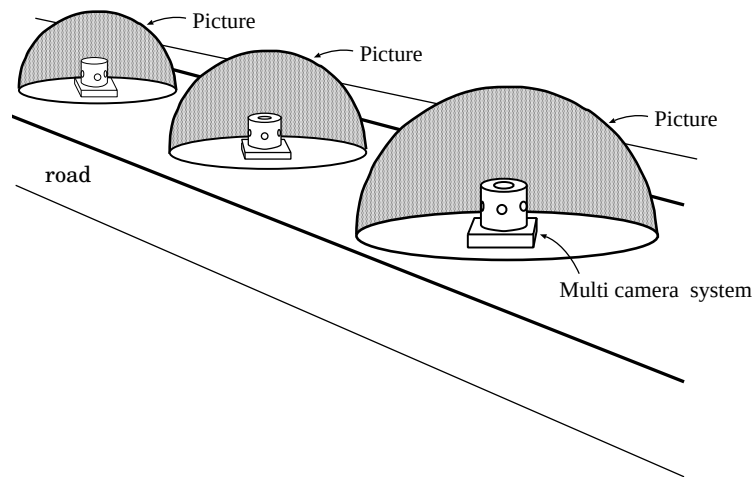


図 2.3: 全方位映像

2.2.2 任意視点映像

任意視点映像とはユーザが自由に指定した視点からの画像を、有限枚の画像から生成する技術である。近年では任意視点映像技術の研究が行われ、将来的にはサッカーなど、スポーツ中継への応用が期待される。図 2.4 では競技場の周囲に配置した複数のカメラにより任意視点映像を生成する様子を表している。任意視点映像ではユーザーが任意に選んだ視点から試合の様子を見ることができる [15][16]。

任意視点映像生成の研究

名古屋大学の谷本らは自由視点テレビの開発を行っている [17]。自由視点テレビでは光線空間法を用いて任意視点映像を生成している。光線空間法では得られた複数枚の画像をもとに、光線空間を補間する。光線空間とは実空間内のすべての点に各点から発する光の方向・強度を記録した空間である。光線空間の補間は位置・姿勢が既知の複数のカメラによって撮影された画像から行われる。補間した光線空間を切り出すことで任意視点からの画像を生成できる [18][19]。また、全方位カメラを用いた任意視点映像生成の研究も数多くなされている [20]-[22]。奈良先端大学の石川らは複数台の全方位カメラで撮影した各画像の対応点をもとにして任意視点映像を生成することに成功している [23]。

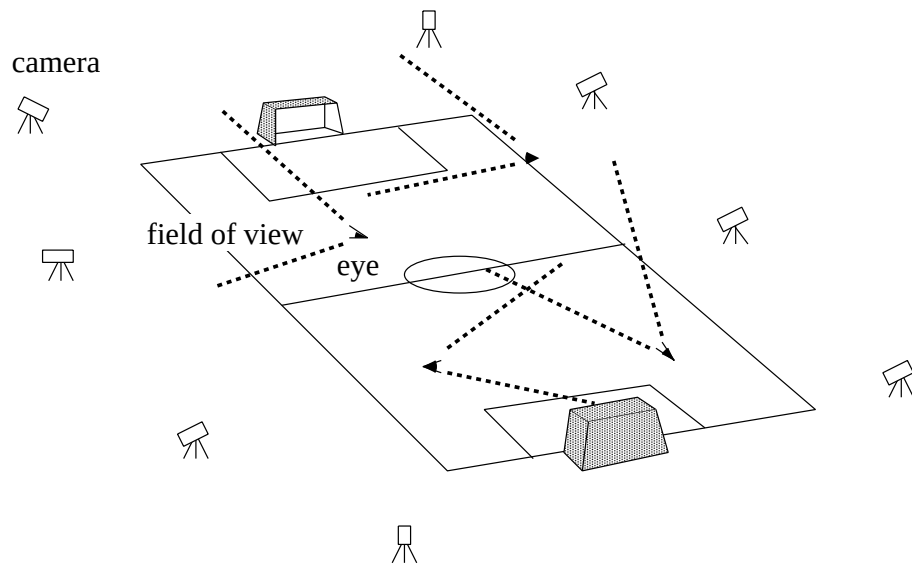


図 2.4: 任意視点映像

2.2.3 恒常視線映像

恒常視線映像とは、カメラが移動・回転しても、カメラの姿勢に基づいた補正により、常に上下左右方向を保った状態で表示される映像である (図 2.5)。

簡易な恒常視線映像を生成することができる身近なデバイスにデジタルカメラがある。デジタルカメラには手ブレ補正機能が搭載されており、手の振動によるカメラの移動を補正する。

恒常視線映像生成の研究

恒常視線映像生成手法の一つに ViewPLUS 社の MotionSPHERE がある。MotionSPHERE は取得した全方位画像の補正を行うシステムである。MotionSPHERE はカメラを不規則に回転・移動させながら撮影した映像に対して補正を行い、カメラが回転しているにも関わらず映像は回転せず常に一定の方向を保って表示させることができる [8]。MotionSPHERE では撮影した画像から特徴点を見つけだし、特徴点の変位を計算することにより画像の補正を行っている。

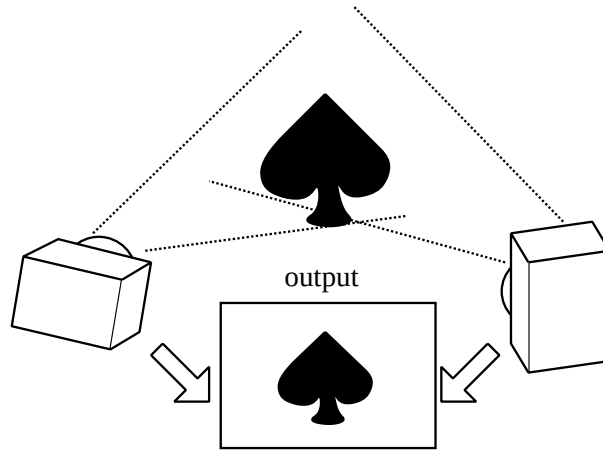


図 2.5: 恒常視線映像

2.3 まとめ

多様な映像の生成手法が研究されているが、いずれの手法でもカメラの位置・姿勢を求めたり、画像中の対応点を求めることになる。また、カメラを移動させながら撮影を行い、カメラの位置姿勢をもとに映像を再構成する。従来の研究ではあらかじめ位置や姿勢を計算したうえでカメラを配置するか、撮影した画像から抽出した特徴点を利用することで位置・姿勢を求めている。位置姿勢を求めてカメラを配置する方法は、自由視点テレビのような、多数のカメラを用いるような大掛かりなシステムでは現実的ではない。また、MotionSPHEREのような、撮影した画像からカメラの位置姿勢を求める手法は特徴点が少ない空間や、うまく対応点を利用できない場合に使用できない。google street view などの市街地映像マップにおいても計算量や情報量の大きさが障害となり実時間での処理が困難となる場合がある。

本研究では、加速度センサで検出した重力ベクトルを元に、カメラの位置・姿勢を推定する手法を提案する。また推定したカメラの姿勢により、撮影した映像の補正を行い恒常視線映像を生成するデバイスを作製する。デバイスは球形をしており、自転しながら撮影を行うことができる。また、複数のカメラを搭載し、全方位の撮影ができるものとする。

第3章 複眼カメラによる恒常視線映像生成

3章では恒常視線映像生成のための手法を提案し、各処理を解説する。

提案手法ではカメラは透明な球殻に内蔵されており、回転しながら撮影を行う。また、球殻には加速度センサーが搭載されており、重力ベクトルを検出できる。さらに、得られた重力ベクトルからカメラの位置姿勢を求めることができる。

球体のデバイスは転がして扱うため、カメラは回転しながら撮影を行う。本章では加速度センサーにより取得した重力ベクトルを用いて画像の回転を補正する方法を示す。

3.1 回転するカメラと画像の傾き

カメラが回転すると撮影される画像も同様に傾く。図 3.1 に正立しているカメラと画像、傾いたカメラと画像を示す。

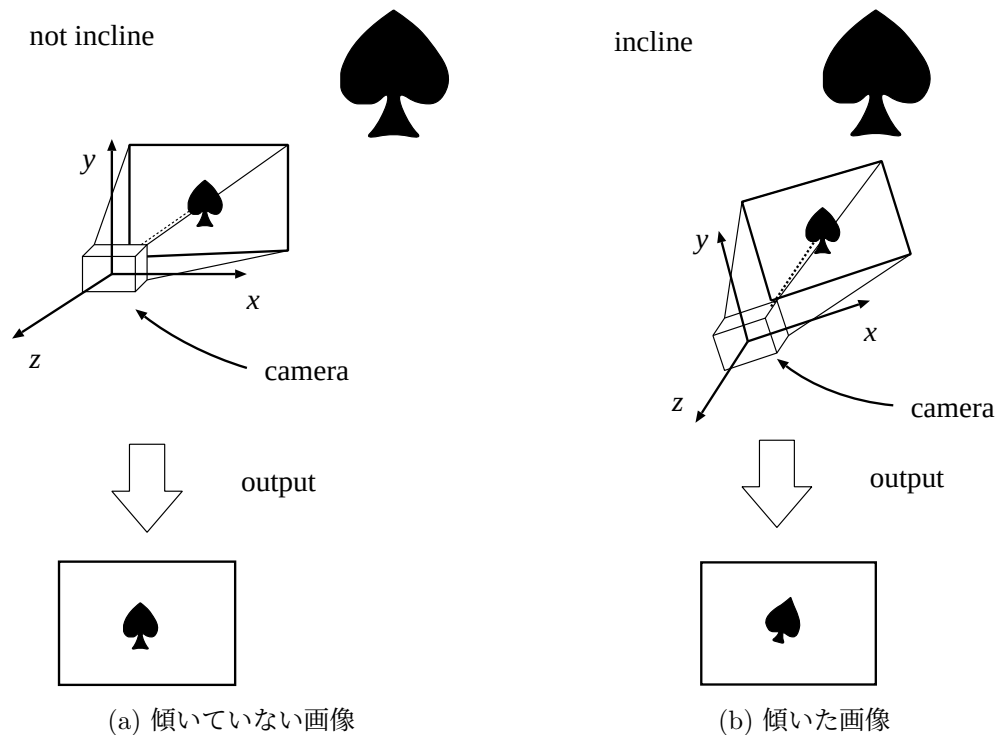


図 3.1: 画像の傾き

カメラの回転には任意の軸についての回転が考えられる。ここで、本研究で扱う画像の傾きの定義を述べておく。まず、図 3.2 のようにカメラ上に座標系を定義する。本システムで補正する回転はカメラの光軸を中心とした回転である。 x, y の二軸については、補正を行わない。 x, y 軸を中心とする

回転をあたえてしまうとカメラの光軸方向と画像の光軸方向がずれ、視線方向と表示する画像の間に矛盾が生じてしまうためである。図 3.2 に画像の傾き θ を示す。 θ はカメラ座標系における y 軸をワールド座標系上のベクトルに変換したときにワールド座標系での y 成分が正、かつ x 成分が 0 となるようにカメラ座標系を z 軸について回転させたときの回転角である。

world coordinate system

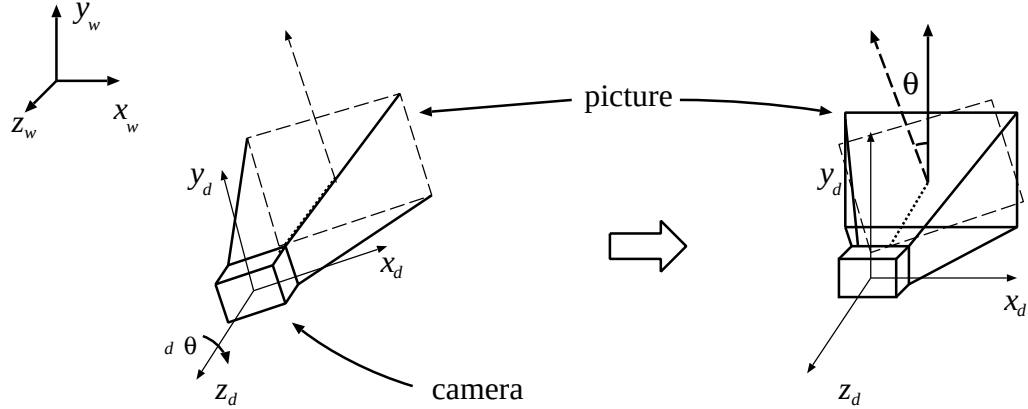


図 3.2: 傾きの大きさ

3.2 傾きの補正

前節で述べた画像の傾きを補正するための手法を述べる。本研究ではワールド座標系・デバイス座標系・カメラ座標系の三つの座標系を定義する。

3.2.1 座標系

図 3.3 に三つの座標系の関係を示す。ワールド座標系は実空間内に y 軸方向を重力ベクトルに平行になるように定義した右手系の座標系である。デバイス座標系はデバイスの中心を原点とする左手系の座標系である。三軸はデバイスの中心に設置した加速度センサーの座標系と一致している (図 3.4)。カメラ座標系はカメラの光軸方向が z 軸、上方向が y 軸、右方向が x 軸となるように定義する。カメラ座標系は 4 台のカメラそれぞれに付随している。

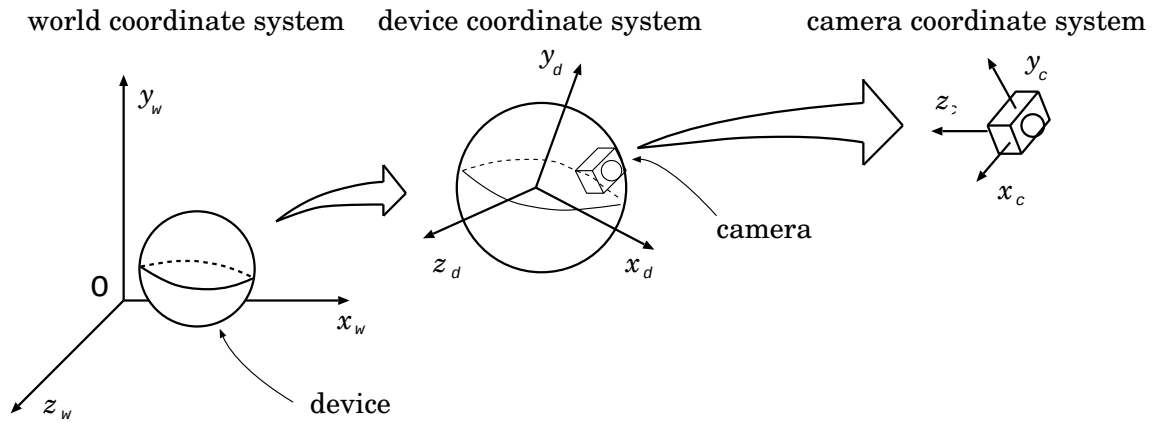


図 3.3: 座標系

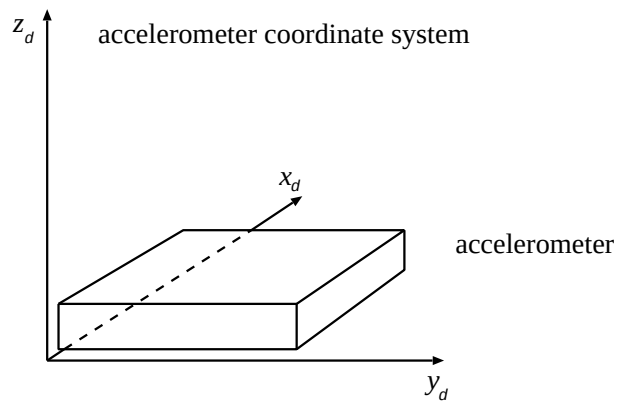


図 3.4: 加速度センサー座標系

3.2.2 回転角の算出

加速度センサによって取得された重力ベクトルはデバイス座標系における表示になっている。まず、デバイス座標系での重力ベクトルをカメラ座標系におけるベクトルに変換する。変換した重力ベクトルを $\vec{g}_c$ とし、さらに $\vec{g}_c$ のカメラ座標系の x - y 平面に対する射影を、 $\vec{g}'_c$ とする。 $\vec{g}'_c$ とカメラ座標系の y 軸、すなわち $\vec{y}_c$ の間の角 θ を求める (図 3.5)。

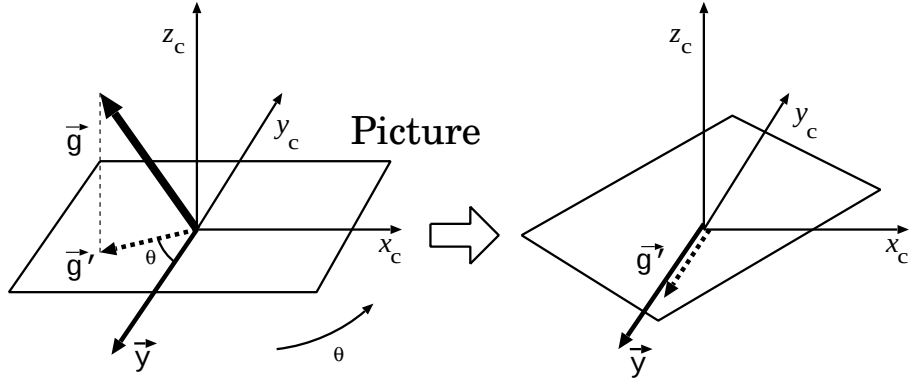


図 3.5: 傾き算出

求めた θ が画像の回転角の大きさである。画像を θ だけ回転させると画像が補正できる。 θ は以下の式で求める。

デバイス座標系からカメラ座標系への変換行列を R_{dc} とするとカメラ座標系における重力ベクトル $\vec{g}_c$ は次式で求まる。

$$\begin{aligned}\vec{g}_c &= \begin{pmatrix} g_{cx} \\ g_{cy} \\ g_{cz} \end{pmatrix} \\ &= R_{dc} \begin{pmatrix} g_x \\ g_y \\ g_z \end{pmatrix}\end{aligned}\quad (3.1)$$

さらに、求めたベクトルをカメラ座標系の xy 平面へ射影すると

$$\vec{g}'_c = \begin{pmatrix} g_{cx} \\ g_{cy} \\ 0 \end{pmatrix}\quad (3.2)$$

となる。ここで、

$$\vec{y}_c = \begin{pmatrix} 0 \\ -1 \\ 0 \end{pmatrix}\quad (3.3)$$

とすると、傾きの大きさ θ は次式で求められる。

$$\begin{aligned}\theta &= \arccos \frac{\vec{g}'_c \cdot \vec{y}_c}{|\vec{g}'_c| |\vec{y}_c|} \\ &= \arccos \frac{-g_{cy}}{|\vec{g}'_c|}\end{aligned}\quad (3.4)$$

求めた傾きの大きさ θ だけ画像をカメラ座標系の z 軸について回転させることで画像を補正する。

3.3 画像の回転方向

前節の式により回転角の大きさを求めることはできるが、回転の方向は決まらない。回転の方向は $\vec{g}'_c$ と $\vec{y}_c$ の外積から求めることができる。 $\vec{g}'_c$ と $\vec{y}_c$ の外積は次式のようにになる。

$$\begin{aligned}\vec{g}'_c \times \vec{y}_c &= \begin{pmatrix} g_{cx} \\ g_{cy} \\ 0 \end{pmatrix} \times \begin{pmatrix} 0 \\ -1 \\ 0 \end{pmatrix} \\ &= \begin{pmatrix} 0 \\ 0 \\ -g_{cx} \end{pmatrix}\end{aligned}\tag{3.5}$$

求めた外積は z 成分にのみ値を持つ。z 成分が正なら画像は z 軸に関して正方向の回転、z 成分が負なら負方向の回転を与え、画像を補正する (図 3.6)。

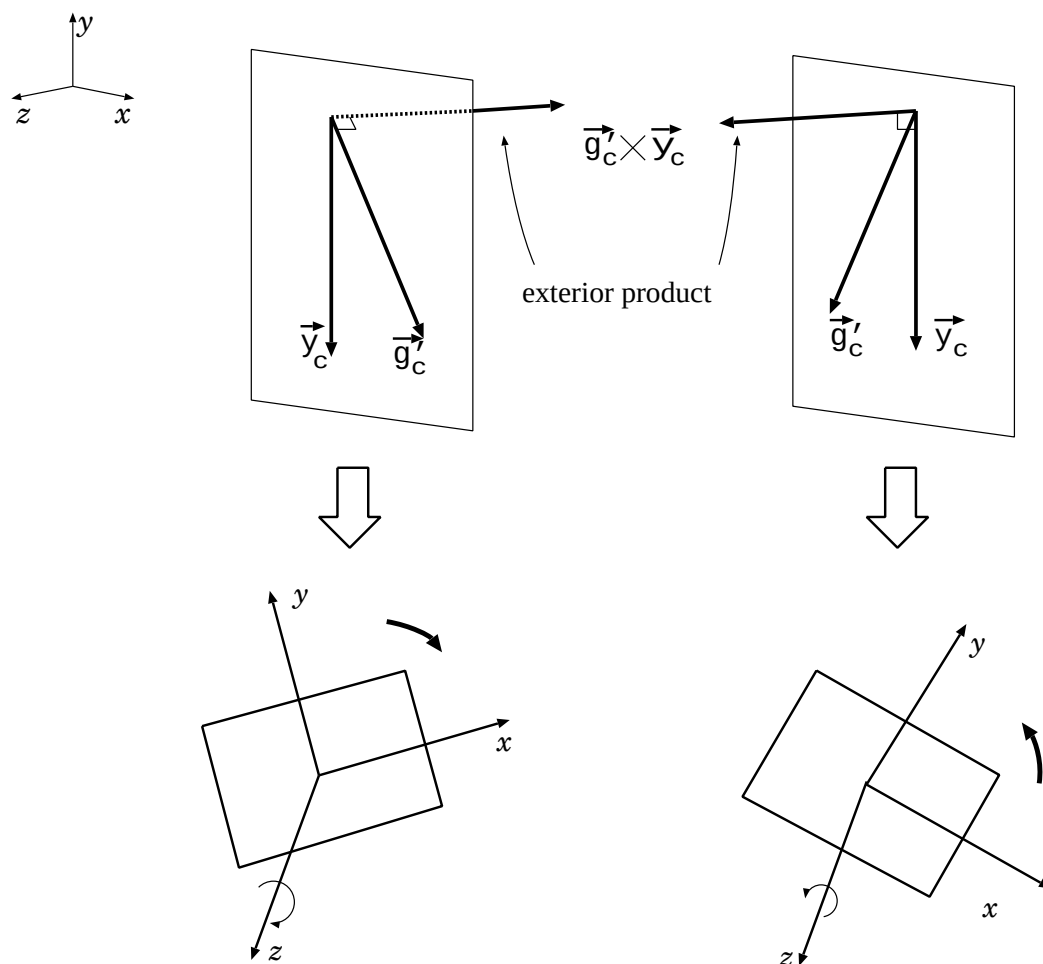


図 3.6: 外積と回転方向

第4章 実装

4章では、前章で述べた理論をもとに本研究で製作したデバイスについて説明する。主に、デバイスの設計とデバイスを構成する部品の配置、装置の仕様、実験で用いたパラメータについて述べていく。

4.1 システム構成

本システムでは4台のカメラを移動させながら周囲の環境を撮影する。また、この時、加速度センサーにより重力ベクトルの変化を記録する。記録した重力ベクトルの変化からカメラの位置・姿勢を計算し、撮影画像の補正をすることで恒常視線映像を生成する。システム構成を図4.1に示す。4台のカメラと受信機の間は無線で通信を行う。加速度センサー・PC間の通信は送受信機を介し、Bluetoothで通信を行う。

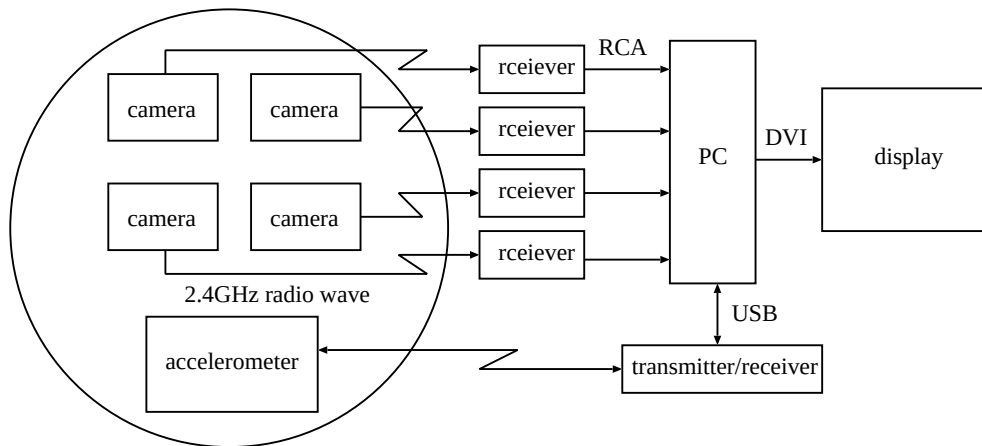


図 4.1: システム構成

4.2 デバイス概要

図4.2にデバイスの外観を示す。本デバイスでは直径170mmの亚克力製の球殻の内部に、各カメラを正四面体の頂点を構成するように配置する。またデバイスの中心には重力ベクトルの成分を検出するために加速度センサーを取り付ける。

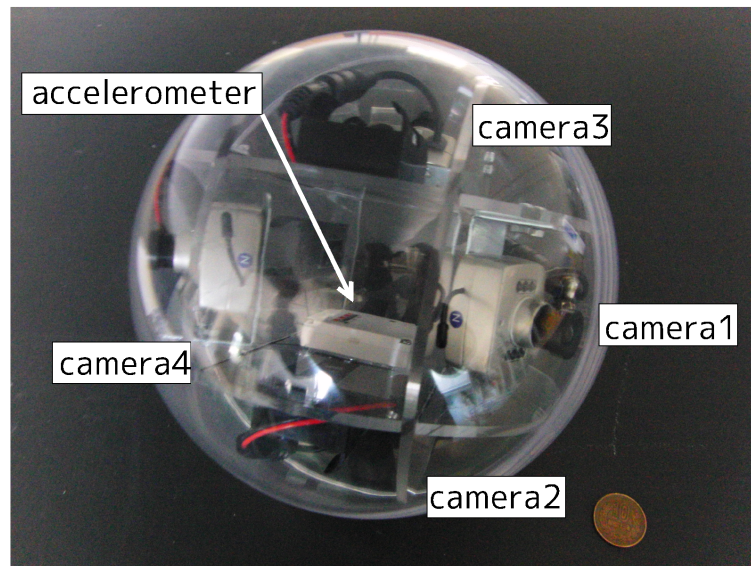


図 4.2: デバイス外観

デバイスを構成する装置は4台の無線カメラ (図 4.3) と加速度センサー (図 4.4) である。

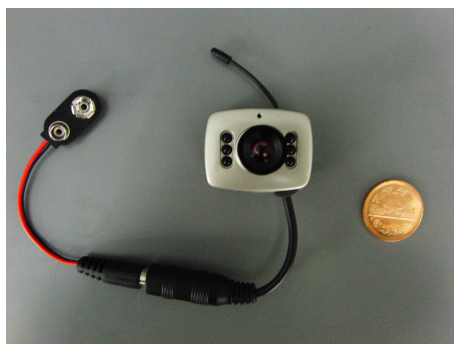


図 4.3: 無線カメラ

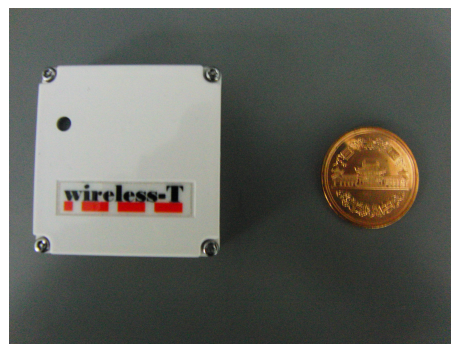


図 4.4: 加速度センサー

実装に用いた装置の仕様を表 4.1 に示す。

表 4.1: 装置の仕様

PC	OS CPU Memory Capture Board Graphic Board	Microsoft, WindowsXP professional Edition Intel, Xeon, 3.2Ghz×2(2CPU) 3GB I-O DATA, GV-VCP3R/PCI3 ATI technologies, Radeon X1900
bluetooth アダプタ	製品名	プリンストンテクノロジー, PTM-UBT5
加速度センサ	製品名 内蔵センサ 最小計測幅 計測範囲	ワイヤレステクノロジー, 小型無線加速度センサ 日立金属, H48C 86.24mm/s ² -29.4×10 ³ mm/s ² ～+29.4×10 ³ mm/s ²
無線カメラ	撮像素子 水平画角 画素数 無線周波数	1/3 カラー CMOS 62deg 31 万画素 2.414GHz 2.432GHz 2.450GHz 2.468GHz

4.3 カメラ配置

本デバイスにはアクリル製の球殻内に4台の無線カメラと加速度センサーを搭載している。図 4.5 にデバイスの見取り図、図 4.6 にカメラの設置角を示す。

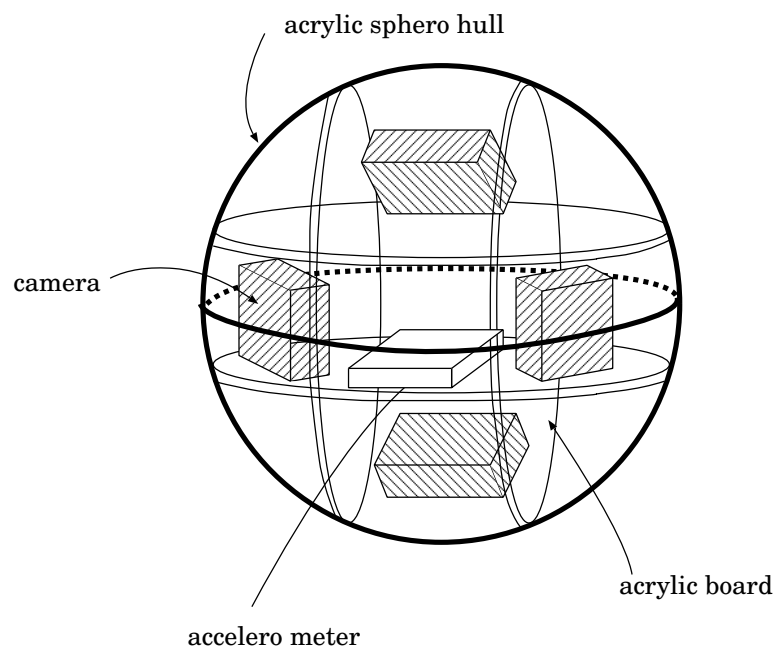


図 4.5: デバイス見取り図

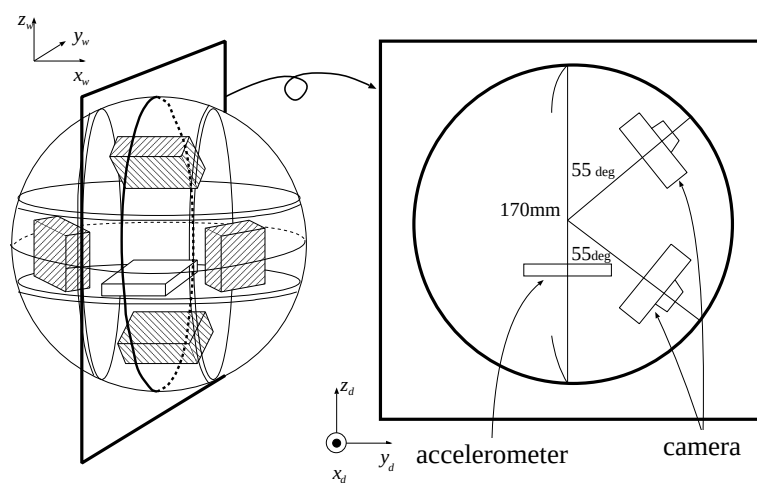


図 4.6: デバイス断面図

カメラの設置角は光軸が正四面体の重心と頂点を結ぶ直線に一致するように設値した。角度の大きさは以下の手順で算出した。

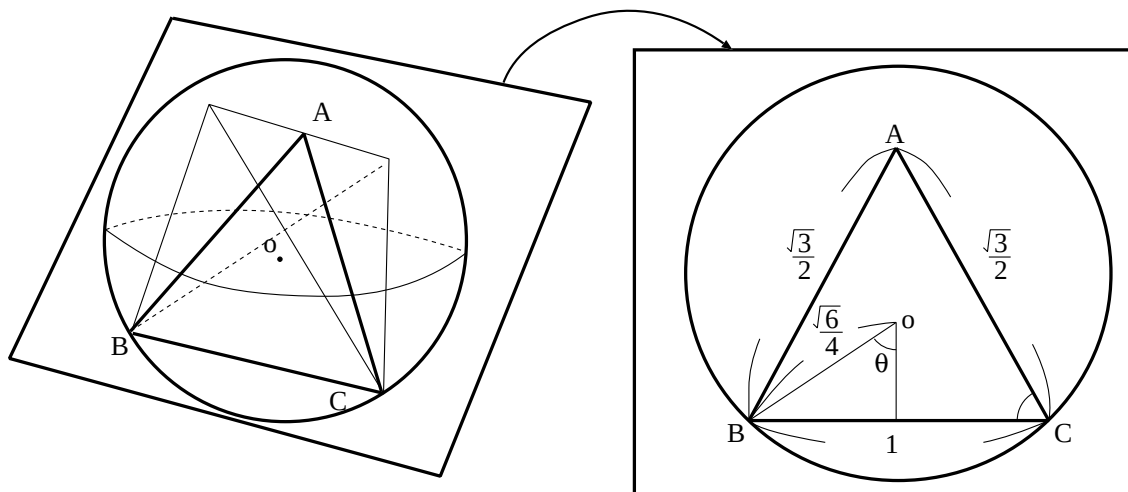


図 4.7: デバイス断面と正四面体断面

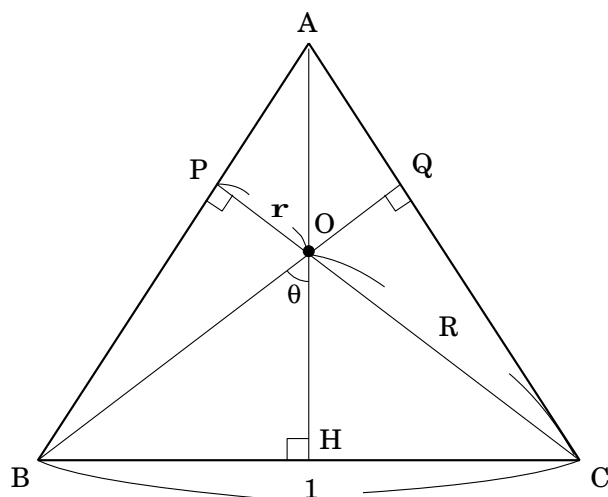


図 4.8: 頂点設定

図 4.7、4.8 に示すように、正四面体の重心を O、AO の延長と BC の交点を H、CO の延長と AB の交点を P、BO の延長と AC の交点を Q とする。ここで、 $OC=OB=R$ 、 $OP=OQ=r$ 、 $BC=1$ とすると

$$AB = \frac{\sqrt{3}}{2}, \quad BP = \frac{2}{3} \times AB = \frac{\sqrt{3}}{3}$$

である。また、

$$CP = \sqrt{1^2 - \left(\frac{\sqrt{3}}{3}\right)^2} = \frac{\sqrt{6}}{3} \quad (4.1)$$

と求まる。以上から r 、 R について次の連立方程式が成立する。

$$\begin{cases} R + r = \frac{\sqrt{6}}{3} \\ r^2 + \left(\frac{\sqrt{3}}{3}\right)^2 = R^2 \end{cases} \quad (4.2)$$

これを解くと

$$r = \frac{\sqrt{6}}{12}, R = \frac{\sqrt{6}}{4}$$

と求まる。また、

$$RH = \sqrt{\left(\frac{\sqrt{6}}{4}\right)^2 - \left(\frac{1}{2}\right)^2} = \frac{\sqrt{8}}{8}$$

である。よって求める角 θ は、

$$\theta = \arccos \frac{\frac{\sqrt{8}}{8}}{\frac{\sqrt{6}}{4}} \doteq 55\text{deg}$$

と求まる。

4.4 デバイス座標系・カメラ座標系変換行列

デバイス座標系からカメラ座標系への変換行列を以下に示す。変換行列は前節に示した加速度センサーおよび無線カメラの配置関係から求めた。

デバイス座標系から各カメラへの変換行列 $R_1 \sim R_4$ は

$$R_1 = \begin{pmatrix} 0 & 0 & -1 \\ \sin\theta & \cos\theta & 0 \\ -\cos\theta & \sin\theta & 0 \end{pmatrix} \quad (4.3)$$

$$R_2 = \begin{pmatrix} 0 & -1 & 0 \\ -\sin\theta & 0 & \cos\theta \\ \cos\theta & 0 & \sin\theta \end{pmatrix} \quad (4.4)$$

$$R_3 = \begin{pmatrix} 0 & 1 & 0 \\ -\sin\theta & 0 & -\cos\theta \\ \cos\theta & 0 & -\sin\theta \end{pmatrix} \quad (4.5)$$

$$R_4 = \begin{pmatrix} 0 & 0 & 1 \\ \sin\theta & -\cos\theta & 0 \\ -\cos\theta & -\sin\theta & 0 \end{pmatrix} \quad (4.6)$$

である。本システムでは前節で算出した通り、 $\theta=55\text{deg}$ とした。

4.5 画像の表示

4.5.1 ポリゴンの配置

画像の表示は三次元空間内に配置した矩形のポリゴンにテクスチャを貼り付けることで行う。ポリゴンの配置はカメラの配置に対応させている。すなわち、正四面体の頂点の位置を中心とし、正四面体の重心と頂点を結ぶ直線を法線方向とする矩形のポリゴンを配置する (図 4.9)。

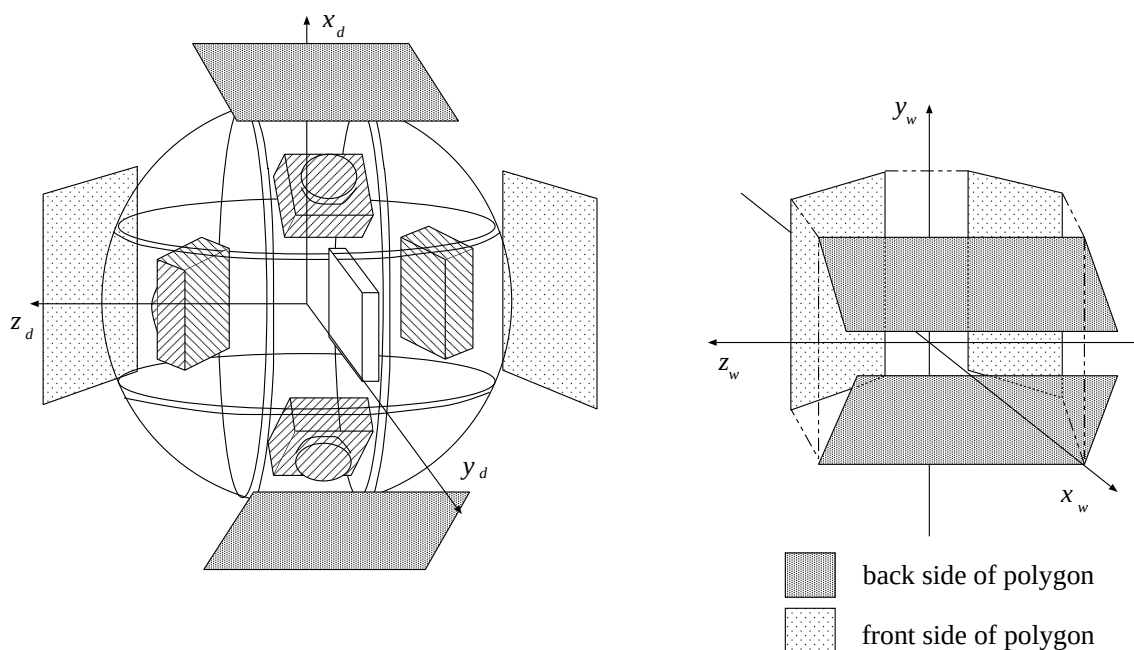


図 4.9: ポリゴンの配置

4枚の長方形は全ておもて面が原点に正対している。テクスチャは長方形のおもて面に貼る。図 4.9 の原点を視点とすることで、球体の中心から周囲を眺める映像が生成できる。ポリゴンはカメラ座標系における重力ベクトルの変化量をもとに回転する。ポリゴンが回転することで画像の傾きが補正される。

また、ポリゴンはカメラの位置に対応させて配置するので、デバイスの回転に合わせてポリゴンの配置も変化させていかなければならない。本システムでは、ワールド座標系におけるデバイスの回転軸と回転角の大きさを求め、正四面体状に配置した4枚のポリゴン全体に求めた回転を与えることで変換を行う(図4.10)。

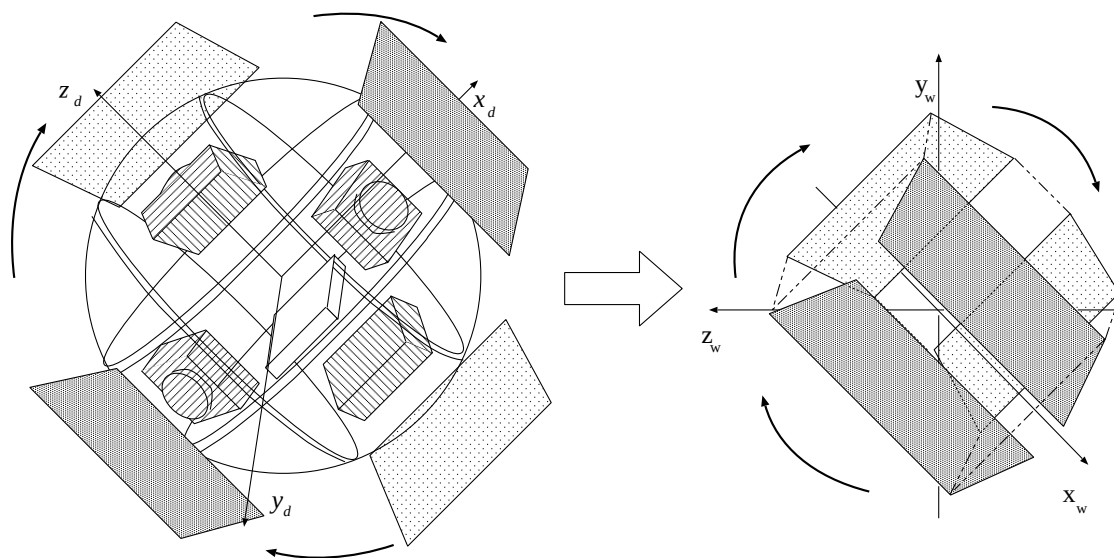


図 4.10: デバイスの回転とポリゴン

回転角と回転軸の算出方法を以下に示す。デバイス座標系の初期姿勢とワールド座標系の関係は図 4.11 のようになる。

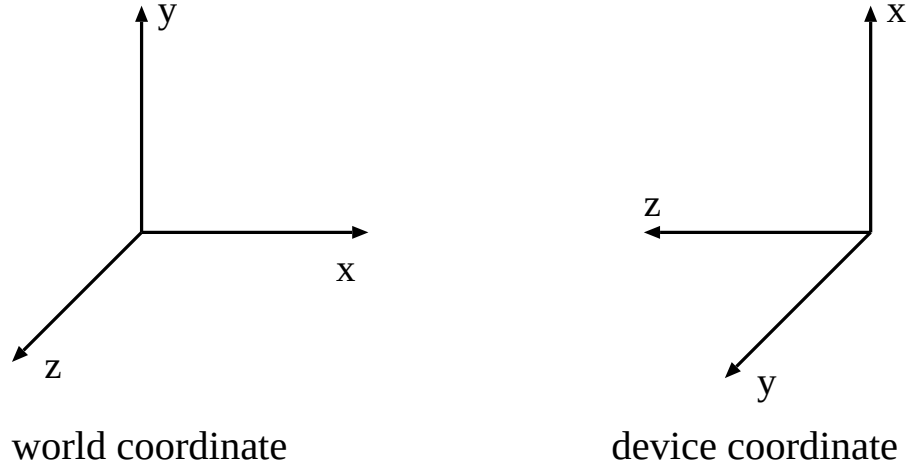


図 4.11: デバイス初期姿勢

デバイスの回転軸、回転角の大きさは初期状態での重力ベクトル $\vec{g}_{d0}$ と各姿勢における重力ベクトル $\vec{g}_d$ を比較することで求めることができる。以下に算出の手順を示す。

まず、回転角の大きさ θ は

$$\theta = \arccos \frac{\vec{g}_d \cdot \vec{g}_{d0}}{|\vec{g}_d| |\vec{g}_{d0}|} \quad (4.7)$$

簡単のため $\vec{g}_{d0}$ を次のようにおく。

$$\vec{g}_{d0} = \begin{pmatrix} -1 \\ 0 \\ 0 \end{pmatrix} \quad (4.8)$$

θ は次式で求められるようになる。

$$\theta = \arccos \frac{-g_{dx}}{|\vec{g}_d|} \quad (4.9)$$

回転軸は次式で求まる。

$$\begin{aligned} \vec{g}_{d0} \times \vec{g}_d &= \begin{pmatrix} -1 \\ 0 \\ 0 \end{pmatrix} \times \begin{pmatrix} g_{dx} \\ g_{dy} \\ g_{dz} \end{pmatrix} \\ &= \begin{pmatrix} 0 \\ g_{dz} \\ -g_{dy} \end{pmatrix} \end{aligned} \quad (4.10)$$

ただし、求めた回転軸はデバイス座標系におけるものである。長方形はワールド座標系に配置しているので、求めた回転軸をワールド座標系上のベクトルに変換しなければならない。デバイス座標系からワールド座標系への変換行列 R_{dw} は図 4.11 から、

$$R_{dw} = \begin{pmatrix} 0 & 0 & -1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \quad (4.11)$$

デバイス座標系からワールド座標系への変換行列を用いてワールド座標系での回転軸は次のように表記できる。

$$R_{dw} \begin{pmatrix} 0 \\ g_{dz} \\ -g_{dy} \end{pmatrix} = \begin{pmatrix} g_{dy} \\ 0 \\ g_{dz} \end{pmatrix} \quad (4.12)$$

求めた回転軸を中心に4枚のポリゴン全体を回転させることで、ポリゴンはワールド座標系におけるカメラの位置と対応した位置に表示される。

4.5.2 恒常視線映像

恒常視線映像では常に一定方向に視線を保って映像を提示する。図 4.12 にポリゴンと提示する映像の関係を示す。図に示す四角錐 O-ABCD が恒常視線映像としてディスプレイに提示される範囲で

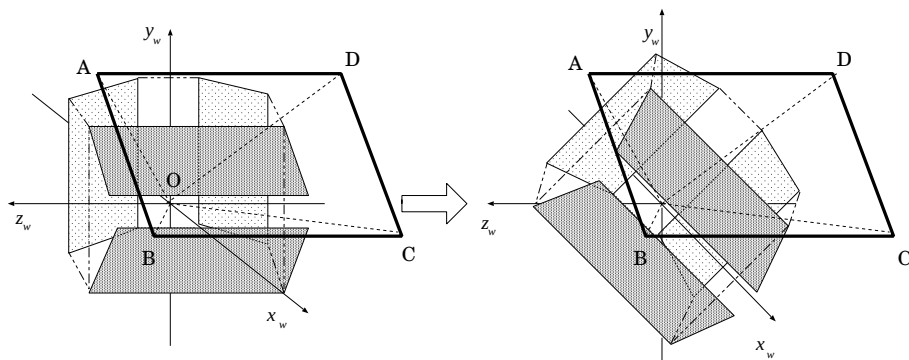


図 4.12: 視体積

ある。四角錐 O-ABCD はポリゴンの移動に関わらず、一定の空間を占める。左は回転前のポリゴンの位置、右は回転後のポリゴンの位置を示す。

第5章 結果

補正の効果を確かめるため、図 5.1 に示すように文字列を壁に貼り付け、デバイスを転がしながら文字列を撮影した。

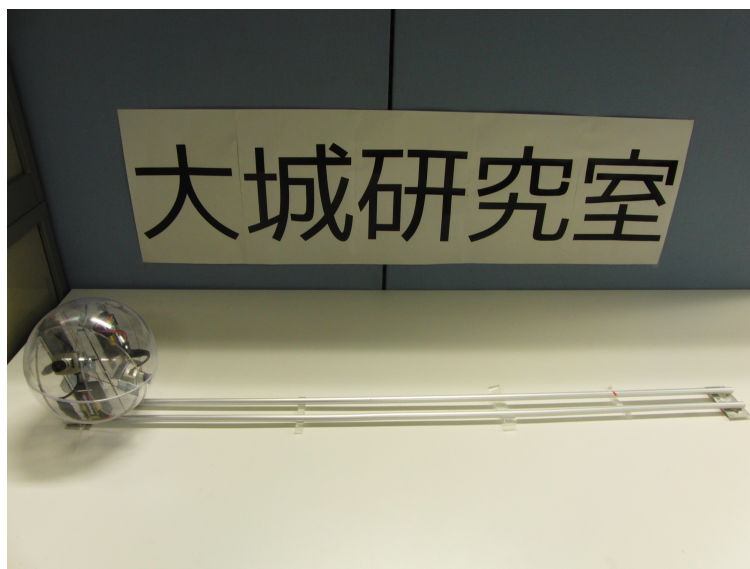
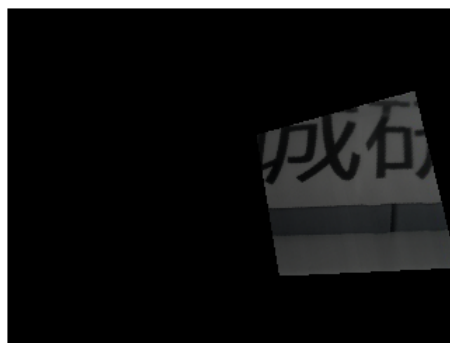


図 5.1: 撮影する風景

図 5.2 に傾きを補正した画像の組を示す。画像の提示領域が傾いているのに対し、写り込んだ文字列は上下方向が正しく表示されている。



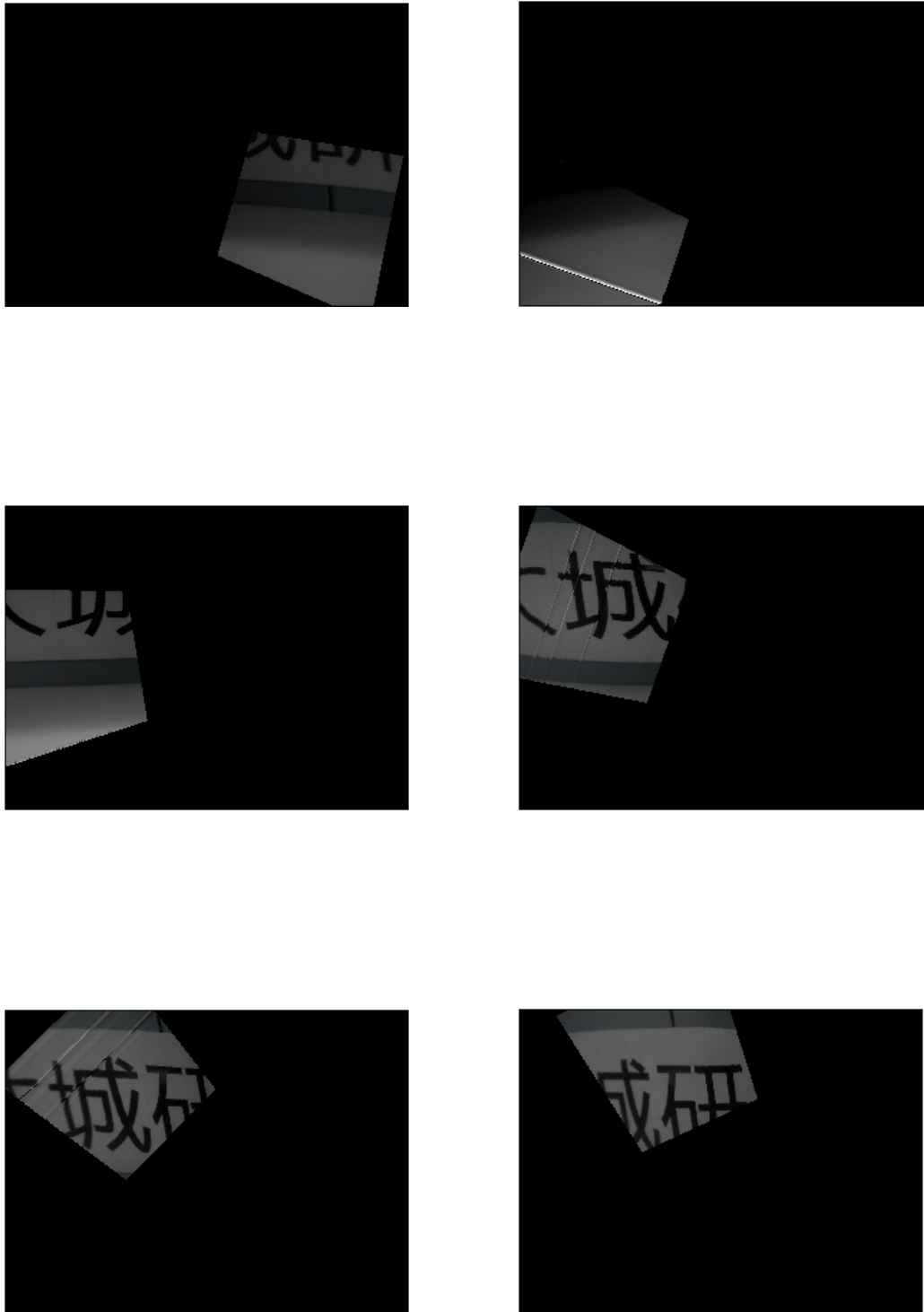


図 5.2: 補正画像

第6章 おわりに

本研究では、まず、無動力で自由に移動・回転ができる撮影装置を製作した。自由な移動・回転を可能にするために、デバイスには無線通信可能なカメラ、加速度センサを搭載した。無拘束な撮影装置を用いることで、撮影装置を回転させながら撮影を行うことができた。加速度センサにより重力ベクトルを検出し、重力ベクトルの回転角および回転軸を求めることでデバイスの姿勢を算出した。次に、算出したデバイスの姿勢をもとに取得した画像の補正を行った。画像の補正を行うことにより、上下方向を正確に提示し、恒常視線映像を生成した。

謝 辞

本研究に取り組むに当たり、まず、進捗報告や日頃から研究の相談をする中でアイデア面、技術面において厚いご指導を賜りました大阪大学大学院基礎工学研究科 大城理 教授に心から感謝いたします。論文をまとめるにあたり 助言を頂いた大阪大学大学院生命機能研究科 田中宏喜 助教に御礼申し上げます。また、デバイスを作成する上でご助力を頂きました、大阪大学大学院基礎工学研究科 荒木勉 教授ならびに坂口良幸 技官に篤く御礼申し上げます。大阪大学大学院基礎工学研究科 井村誠孝 准教授には本研究を進める上で担当指導教員として、私の拙い論文に対して、細やかな指導を頂きました。また、4月以来 OpenGL 等々、技術面・知識面において多大なご指導をいただきました。大阪大学大学院基礎工学研究科 黒田嘉宏 助教には日々の研究生活の中で、談話などの機会を作っていただき、ご指導をいただきました。臨床医工学融合研究教育センタ 鍵山善之 特任助教には画像の処理に関してご指導をいただきました。大城研究室のスタッフの方々に心から感謝いたします。 関西医科大学からの研究生である吉田健志氏、また、芦田洋敏氏、金森恒志氏、福塚大介氏、山田和広氏、奥村陽介氏、重枝慧氏、末竹哲也氏、田中翔太氏、徳井隆博氏、濱田友貴氏、吉本俊輔氏には研究だけでなく、日常生活においても相談に乗っていただくこともありました。また、先輩方のおかげで、1年間楽しい研究生活を送れましたことを深く感謝します。最後に、日々の研究生活を共に過ごし、夜遅くまで研究室に残って助け合い励ましあってきた4回生の近藤良氏、繁田悠氏、清水祐司氏、増田拓氏、利原康寛氏に感謝します。

参考文献

- [1] 山澤一誠: “全方位視覚センサ HyperOmni Vision に関する研究 -移動ロボットのナビゲーションのために-”, 大阪大学 博士論文, 1997
- [2] 徐剛: “写真から作る 3 次元 CG イメージ・ベースド・モデリング&レンダリング” 近代科学社, 東京, 2005
- [3] 岸野文郎, 佐藤隆夫, 横矢直和, 相澤清晴, 有川正俊: “画像と空間の情報処理” 岩波書店, 東京, 2000
- [4] 佐藤准嗣, 高橋友和, 井手一郎, 村瀬洋: “車載全方位カメラと G P S を用いた市街地映像マップの構築” 電子情報通信学会大会講演論文集, pp.193, 2005
- [5] 王金戈, 小野晋太郎, 池内克史: “時系列高さ画像とそれを利用した市街地実写画像と住宅地図との位置合わせ” 画像の認識・理解シンポジウム, 2008
- [6] 佐藤亮, 小野晋太郎, 永塚遼, 川崎洋: “車載全方位ビデオ映像を用いたイメージベースレンダリングによるドライビングシミュレータの提案” Technical report of IEICE.PRMU, pp.1-6, 2009
- [7] 山口浩司, 広田光一, 廣瀬通孝: “実写画像からのカメラ位置・姿勢情報の抽出に基づく空間の構築” 電子情報通信学会技術研究報告.MVE, マルチメディア・仮想環境基礎 pp.55-60, 2001
- [8] Hiroki Mori, Dairoku Sekiguchi, Shigesumi Kuwashima, Masahiko Ianami, Fmitoshi Matsuno: “Motion SPHERE” ACM SIGGRAPH 2005 Emerging technologies, ArticleNo.15, 2005
- [9] 池田聖, 佐藤智和, 横矢直和: “全方位型マルチカメラシステムのキャリブレーションによる高解像度な全天球パノラマ動画の生成” 日本バーチャルリアリティ学会研究報告, Vol.8, No.2, pp.31-36, 2003
- [10] 大江統子, 佐藤智和, 横矢直和: “幾何学的位置合わせのための自然特徴点ランドマークデータベースを用いたカメラ位置・姿勢推定” 日本バーチャルリアリティ学会論文誌, Vol.10, No.3, 2005
- [11] 佐藤智和, 池田聖, 横矢直和: “複数動画からの全方位型マルチカメラシステムの位置・姿勢パラメータの推定” 電子情報通信学会論文誌 D-II Vol.J8-D-II No.2, pp.347-357, 2005
- [12] 薄充孝, 中川知香, 佐藤智和, 横矢直和: “ランドマークデータベースに基づく静止画像からのカメラ位置・姿勢推定” 日本バーチャルリアリティ学会論文誌 Vol.2, No.3, pp.161-170, 2008
- [13] 桑島幹: “よくわかる最新レンズの基本と仕組み” 秀和システム, 東京, 2005
- [14] 光学のすすめ編集委員会: “光学のすすめ” オプトロニクス社, 東京, 1997

- [15] 糟谷望, 北原格, 亀田能成, 大田友一: “自由視点映像技術を用いたサッカーシーンの選手視点映像” 電子情報通信学会総合大会, 2009
- [16] 稲本奈穂, 斎藤英雄: “多視点スポーツ映像からの自由視点映像合成と提示” 画像の認識・理解シンポジウム (MIRU2004), 2004
- [17] 谷本正幸: “自由視点映像技術” 映像情報メディア学会誌, Vol.60, No.1, pp.29-34, 2006
- [18] Marc Levoy, Pat Hanrahan: “Light Field Rendering” International Conference on Computer Graphics and Interactive Techniques Proceedings of the 23rd annual conference on Computer graphics and interactive techniques table of contents, pp.31-42, 1996
- [19] Masayuki Tanimoto: “Overview of free viewpoint television” Signal Processing Image Communication 21, pp.454-461, 2006
- [20] 冨手要, 山澤一誠, 横矢直和: “複数の全方位画像を用いた広範囲な任意視点画像の生成” 電気情報通信学会 信学技報, pp.65-70, 2002
- [21] 高橋桂太, 苗村健: “合焦判定に基づくマルチカメラ画像からの全焦点自由視点画像合成” 電子情報通信学会和文論文誌, Vol.J88-D-II, No.3, pp.573-584, 2005
- [22] 矢口悟志, 斎藤英雄: “未校正多視点カメラからの自由視点映像生成” 電気情報通信学会技術研究報告, 100(633), pp.39-46, 2001
- [23] 石川智也, 山澤一誠, 横矢直和: “View Morphing と Visual hull を用いた自由視点映像生成 ” 電子情報通信学会 技術研究報告, PRMU2004-136, pp.13-18, 2004