

平成19年度

筑波大学第三学群情報学類

卒業研究論文

題目 ハンドジェスチャによる範囲選択手法

主専攻 知能情報メディア主専攻

著者 淵 一馬

指導教員 田中二郎 高橋伸 三末和男 志築文太郎

要 旨

リモコンなどの機器を必要とせず家具や家電を操作できるインタフェースとして、ハンドジェスチャに関する研究が数多くされてきた。しかし、従来のハンドジェスチャに関する研究では上下左右の方向を指し示したり、手の平のある位置を判別するといった点や方向に着目したものが提案されているが、それらの情報を複合的に扱い、離れた範囲を選択しようとするものはない。

本研究ではハンドジェスチャを用いて範囲を選択する手法を提案し、その手法を用いて撮影するシステムを実装した。これにより離れた範囲を選択し撮影することが可能になった。また、そのシステムにおける精度を検証した。

目次

第1章 序論	1
1.1 背景	1
1.2 ハンドジェスチャによる範囲選択の有効性	1
1.3 言葉の定義	2
1.4 本研究の目的	2
1.5 本論文の構成	3
第2章 関連研究	4
2.1 範囲を扱う研究	4
2.1.1 ディスプレイ上でのマルチタッチによる範囲選択	4
2.1.2 生活空間における範囲選択	4
2.1.3 撮影におけるインタフェース	5
2.2 提案	5
第3章 範囲選択手法	6
3.1 範囲選択に適したハンドジェスチャの考察	6
3.2 範囲選択手法を用いた範囲選択	7
3.3 範囲の求め方	8
第4章 撮影システム「HipHopHand」	10
4.1 HipHopHand	10
4.1.1 撮影の流れ	11
4.1.2 撮影の様子	11
4.2 システムの実装	12
4.2.1 システムの構成	13
4.2.2 開発環境	14
4.2.3 両手を叩いたことの認識	15
4.2.4 顔と両手領域の抽出	15
4.2.5 キャリブレーション	17
4.2.6 三次元座標の取得	18
4.2.7 衝突点の計算	21
4.2.8 角度計算	22

4.2.9	PTZ の決定	23
第 5 章	精度評価	24
5.1	ステレオカメラにおける誤差	24
5.1.1	計測方法	24
5.1.2	計測結果	24
5.2	重心を扱うことにおける誤差	27
5.2.1	計測方法	27
5.2.2	計測結果	28
5.3	範囲選択における精度	31
5.3.1	評価の方針	31
5.3.2	実験の開始の手順	31
5.3.3	評価実験 1	32
5.3.4	評価実験 2	33
5.3.5	実験結果	33
5.4	考察	37
第 6 章	結論	38
	謝辞	39
	参考文献	40

図目次

3.1	様々なジェスチャ	6
3.2	範囲選択手法を用いて範囲を選択している例	7
3.3	選択したい範囲における注目点	8
3.4	選択したい範囲の広さ	8
4.1	カメラの CM におけるワンシーン	10
4.2	撮影の様子	11
4.3	処理の流れ	12
4.4	システムの概観	13
4.5	システムの構成	13
4.6	カメラの配置	14
4.7	顔と両手領域抽出の様子	16
4.8	キャリブレーションの様子	17
4.9	左右のカメラによるマッチング	18
4.10	三角測量	19
4.11	極小ベクトルを足していく様子	21
4.12	部屋における座標系の向き	22
4.13	ズームレベルの決定	23
5.1	x 座標の誤差	24
5.2	y 座標の誤差	24
5.3	z 座標の誤差	25
5.4	距離の誤差	25
5.5	取得したい座標の位置	27
5.6	計測した顔と両手座標の角度	27
5.7	実験の様子	32
5.8	選択する目標物が近いとき	32
5.9	選択する目標物が遠い時	32
5.10	評価実験 2 で選択するもの	33
5.11	範囲の広さを選択する改善方法	37

表 目 次

4.1	三角測量のパラメータ	19
4.2	角度計算のパラメータ	20
5.1	ステレオカメラからの距離でまとめた結果	25
5.2	座標系ごとに固定してまとめた結果	26
5.3	距離による誤差	28
5.4	右手を上にして座標を取得したときの結果	29
5.5	右手を下にして座標を取得したときの結果	29
5.6	顔の座標を取得したときの結果	30
5.7	評価実験の結果	33
5.8	条件付きでの評価実験の結果	34
5.9	評価実験 1:撮影対象が近いときの結果	34
5.10	評価実験 1:撮影対象が遠いときの結果	34
5.11	評価実験 2:角度ごとにまとめた結果	35
5.12	評価実験 2:被験者と角度ごとに失敗をまとめた結果	35
5.13	評価実験 2:被験者ごとにまとめた結果	36

第1章 序論

1.1 背景

現在、コンピュータは日常生活においてどこにでも見かけることができるほど遍在している。今後、このコンピューティング環境は日常生活のあらゆるところで、ユーザから目に見えない形でユーザを支援する「ユビキタスコンピューティング」といわれる環境になると予想される。将来、そのユビキタス環境で望まれるインタフェースは、できるだけユーザにとって負担がかからず、かつ直感的に操作できることが求められている。そのようなインタフェースとして、ハンドジェスチャを用いたインタフェースが挙げられる。

ハンドジェスチャはユビキタス環境における重要なインタフェースである。なぜならばインタフェースの基本が日常生活において行う身体的動作となるため、ユーザにとって直観的で扱いやすく、自然に扱うことができる。また、ユーザはリモコンなどの操作デバイスが必要としないので、操作したいときにその場で操作を行うことができる。そのような理由でハンドジェスチャを用いた様々なインタフェースが研究されてきた。

ハンドジェスチャを用いた研究として、指先で空中に文字を書きコンピュータに入力するものや、掌を動かしてフローメニューを扱い文字入力するものがある [1] [2]。これらの研究はユーザの手の位置を取得することにより操作が行えるインタフェースである。他にもユーザの手の示す方向により操作するインタフェースがある。例えば、上下左右の四つの方向を指で指すことにより TV 画面を操作するものや、指で指す動作で物体を指定するといったものがある [3] [4]。

以上のように、ハンドジェスチャを利用した研究は数多く研究されている。しかし、ハンドジェスチャに関する研究では手の平のある位置を利用したり上下左右の方向を指し示すといった位置や方向に着目したものが提案されているが、それらの情報を複合的に扱い、空間における範囲を選択しようとするアプローチは少ない。

1.2 ハンドジェスチャによる範囲選択の有効性

ハンドジェスチャにより範囲を選択できるようになることで、有効だと思われる利用シーンがいくつか挙げられる。

最初に撮影をするケースを挙げる。日常の中で「この状況を撮影したい」と思ったら、まず撮影機器を準備しなくてはならない。しかし、撮影機器を準備するのは手間であったり、撮影機器を準備している間に、撮影したい状況が終わってしまうこともあるかもしれない。も

し撮影機器を準備することなく、思いついたときすぐにハンドジェスチャにより撮影できるなら、ユーザは撮影機器を準備する手間をかけず、ユーザの「撮影したい」という日常生活の中では突発的に起こるだろう欲求を満たすことができる。

また、最近普及してきた公共の場における大画面での操作に使用することも考えられる。公共の場にある大画面は、大画面を操作するために使う操作デバイスを設置するのが難しいため、利用するには素手によるハンドジェスチャが有効である。しかし、大画面操作ではその画面の大きさのため、直接触って操作するには、画面の大きさが大きくなればなるほど難しいという問題がある。そこで、離れた場所から範囲を選択する手法は、ユーザがどこにいても操作することを可能にする点で有効である。

他にも、将来ロボット技術が発達し一般的にもっと普及したならば、ロボットへの適用も考えられる。ガラスのコップを床に落として割ったときにガラスが散らばった範囲を指定することで、自動で掃除してくれる掃除機ロボットにガラスが飛散した範囲を掃除してもらえらるだろう。

また、複合現実感での利用も考えられる。複合現実感とは、今まで Virtual Reality などのようなコンピュータ上でしか表現できなかった仮想世界と私たちが暮らしている実世界を継ぎ目なく統合する技術の総称である。今まで平面的な画面の中で表現されていた仮想世界の情報を現実世界と統合することから、複合現実感におけるアプリケーションは実世界の距離情報を新たに活用するケースも想定され、三次元空間上に表示されたデータを離れた場所から範囲選択をし、複数選択することも考えられる。

1.3 言葉の定義

範囲という言葉は多様な用途で使われる。例えば「携帯の基地局から 5km の範囲」といえば三次元的な物理空間における範囲であり、「PC のディスプレイにおける範囲」といえば画像空間内の二次元的な平面における範囲である。さらに「1 から 9 までの範囲」といえば数字における範囲を指し、「想定範囲」といえば起こった事象そのものを指す。範囲という言葉は様々な用途で使われるが、本研究における範囲を「物理空間における部分空間」と定義する。また、範囲選択手法とは「範囲選択をする際に行うジェスチャ」のことを指す。

1.4 本研究の目的

本研究ではユーザがハンドジェスチャで範囲を選択できる範囲選択手法を提案する。この手法はユーザの選択している範囲における注目点とその広さを、顔と両手の三次元座標を用いて取得することで認識する。これによりユーザは自分から離れている範囲も選択することができる。

また、本研究では範囲選択手法によって選択された範囲を撮影するシステムを実装する。この撮影システムはユーザの指定している範囲を三次元座標により割り出し、その範囲を撮影する。

1.5 本論文の構成

本章では研究の背景と目的について述べた。第2章では関連する既存研究に関して述べ、本研究における範囲選択手法の既存研究との違いを述べる。第3章では本研究で提案する範囲選択手法に関して説明する。第4章では実装した撮影システムに関して、第5章では実装したシステムの精度実験を行った結果を記述する。最後に、第6章で結論を述べる。

第2章 関連研究

この章では範囲選択をする既存研究に関していくつか問題点をあげたのち、本研究の提案を述べる。

2.1 範囲を扱う研究

ジェスチャにより範囲を選択している研究は、その選択したい範囲の空間やジェスチャに合わせて様々なものがある。

2.1.1 ディスプレイ上でのマルチタッチによる範囲選択

タッチスクリーンのような、直接画面を触ることで操作できるデバイスが日常生活に普及してきている。その中でも複数の点を認識できるマルチタッチインタフェースが、近年注目を浴びている。

マルチタッチインタフェースの研究に、ハンドジェスチャを意識したテーブルトップディスプレイでのインタラクションを研究しているものがあり、提案するジェスチャの中に範囲を選択するものがある [5]。指を折り曲げ、両手でコーナーを作ることで囲んだ範囲を選択できる。また、マルチタッチインタフェースに圧力センサを加えた研究がある [6]。この研究では五本の指で画面を押し付けることで、指で囲まれている範囲にあるものを選択するといった操作ができる。この範囲は画面内における範囲を手によって押しつけた圧力で検知している。

マルチタッチインタフェースはマウスなどでの操作と違い、複数の操作を同時に行えるという利点があり、例えば3D 物体を見る視点を移動させながら、拡大縮小するといったことができる。このインタフェースで選択される範囲は生活空間ではなくディスプレイ内の画像であるが、ハンドジェスチャで囲うという点で本研究と関連がある。ただ、ディスプレイ上でのマルチタッチによる範囲選択は、範囲を選択する対象が二次元であるため、三次元的な空間を範囲選択するためのものではない。そのため、これらのジェスチャは生活空間における範囲選択にはあまり適したものではない。

2.1.2 生活空間における範囲選択

生活空間など、実世界で範囲を指定するジェスチャをしているものとして ComeCam [7] という研究がある。この研究の焦点はネットワークカメラによりコミュニケーションを行うこ

とであるが、コミュニケーションを行う際にマーカーを用いて空間上にマスクを表示したり、指定した場所へカメラをズームさせることができる。マスクを表示したいときはマーカーを手を持ち、カメラに向かって自分の周りに弧を描くことで、マスクを表示したい自分の位置とマスクを表示させる大きさを指定できる。ズーミングのときは手に持っているマーカーをカメラに向かって前後に動かすことで、自分のいる位置にカメラをズームさせることができる。この研究は生活空間における範囲を認識しているが、ユーザから離れた範囲を認識するものではない。

2.1.3 撮影におけるインタフェース

撮影するとき、我々は目から見える景色を保存するという動作を行っている。その動作は目から見える景色の一部分を囲いこみ、その囲い込んだ範囲を撮影するという行為である。目から見える景色の一部を囲い込むという動作が範囲選択をしている動作であることから本研究と関連がある。

HULGER の作った「A.PIX」というデジタルカメラ [8] は、カメラを変形させフレームによって四角い枠を作ることで、その四角い枠内を撮影する。枠の大きさを変化させて四角いフレームを作り、しっかりと写真の構図を考えた上で写真を撮ることが撮影する上で重要なことであると主張している。他にも、両手によりハンドフレームを作り撮影するものがある [9]。この研究は両手の人差し指と親指を互いに合わせ、ハンドフレームを作ることにより目から見えるハンドフレームの枠内を撮影する。

これらの研究は視界の中にある選択したい範囲を、四角い枠を作ることで選択し、生活空間における離れた範囲を選択しているようにも思える。しかし、これらの動作は目から見える景色を二次元的に保存するだけのものであり、選択された範囲を認識して外部のシステムがその情報を利用できるものではない。

2.2 提案

既存研究における範囲選択は、選択する範囲が自分の手の届く範囲に限られた。また、離れた範囲をハンドジェスチャで指定し、画像として写すことができて、三次元的に範囲を認識するものではない。選択された範囲を映像として取ってくるだけでなく、三次元情報を用いて範囲を認識できるならば、ユーザの周囲にあるシステムが選択された範囲を識別して利用することができるようになる。

よって本研究では離れた範囲を、特殊な器具を必要とせずに、ハンドジェスチャを用いて選択することができる範囲選択手法を提案する。そしてその範囲選択手法により、ユーザが選択した範囲を三次元情報を用いて認識して撮影するシステムを実装する。

第3章 範囲選択手法

本章では本研究で提案する範囲選択手法に適したハンドジェスチャを考察した後、範囲選択手法を用いた範囲選択を説明し、その選択された範囲の求め方について述べる。

3.1 範囲選択に適したハンドジェスチャの考察

空間上の範囲を選択するハンドジェスチャは複数考えられる。本研究では目から見える景色を手で囲むことで範囲を選択することを基本とし、ハンドジェスチャとして以下の手法を考察した。

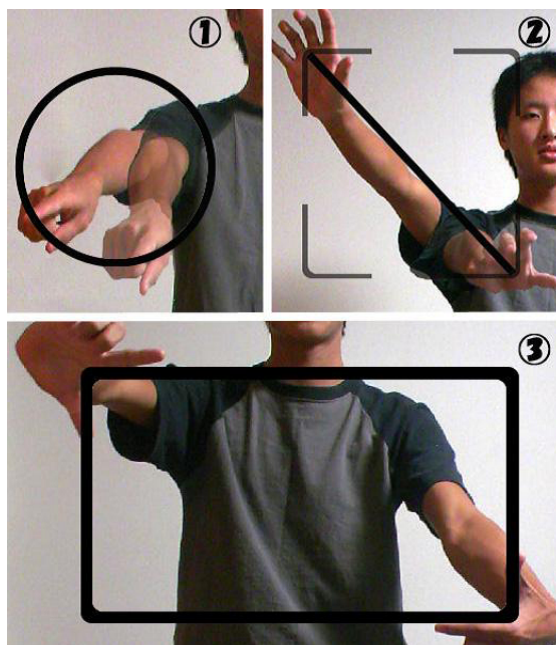


図 3.1: 様々なジェスチャ

1. 「軌跡型」 2. 「対角型」 3. 「両手対角型」

3つの手法「軌跡型」「対角型」「両手対角型」についてそれぞれの特徴を述べる。

図 3.1 の 1、軌跡型は人差指で軌跡を描くことで範囲を描く。一本の腕で直感的に範囲を指定でき、指先の軌跡により好きな形で範囲を指定することができる。図 3.1 の 2、対角型は選

択したい範囲を四角形として捉え、対象角を片手で結ぶことにより範囲選択をする。容易に範囲が指定できるので単純な操作に向いている。図 3.1 の 3、両手対角型は両手で四角形を作ることにより範囲を指定する。素早く範囲を指定でき、範囲内にある指定したいものが動くときも対応しやすい。

「軌跡型」は日常生活での「なぞる」という動作を基本とし、「対角型」は片手での、「両手対角型」は両手での「囲む」という動作を基本とする。

「なぞる」という行為は基本的に平面的な空間で行う動作であるため、本研究の目的である三次元で範囲を選択することからはずれる。また、片腕を動かしての範囲選択は両手を固定する場合と比べ、人によって誤差が大きくなると推測した。今回は正確に範囲選択できることが重要であると考え、「両手対角型」を採用した。

3.2 範囲選択手法を用いた範囲選択

範囲を指定するためのインタフェースとして、本研究では図 3.1 の 3、「両手対角型」を使った範囲選択手法を提案する。ユーザは指定したい場所の目から見える景色を、両手で囲むことにより範囲を指定する。具体的には人差し指と親指を 90 度に広げ、目の前に両手で四角形を作り、体を指定したい範囲の方向に向ける。そうすると両手により作られた四角形を伸ばしていった先にあるものを範囲として認識する。図 3.2 は PC の周囲を範囲選択しようとしているものである。

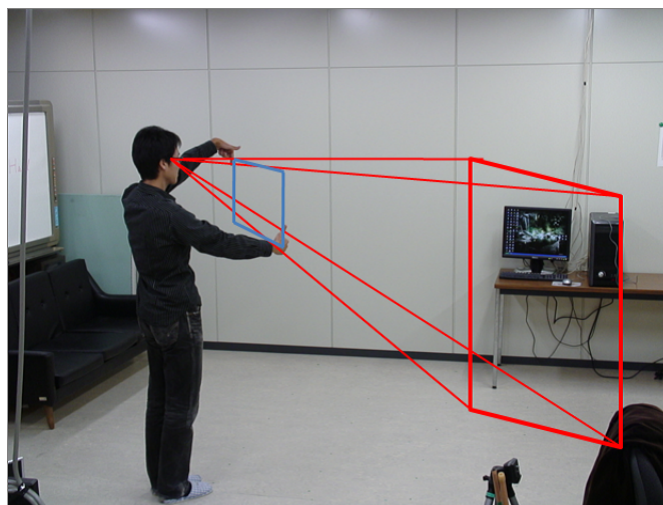


図 3.2: 範囲選択手法を用いて範囲を選択している例

3.3 範囲の求め方

ユーザの指定したい範囲を計算するときの重要な情報はユーザが「1. どこに注目しているか」「2. どこまでの広さを指定したいのか」の2点であると考えた。よって両手で作られた枠の中心を取ってくることで1を、範囲の両端となる右手と左手の座標を扱うことで2の条件を満たし、それらの座標を元に範囲を計算する。両手で作られた枠の中心へユーザの視点からベクトルを伸ばす。そのベクトルを伸ばしていったぶつかった先がユーザの注目している物体がある座標である。同様にユーザの視点から右手と左手にそれぞれベクトルを伸ばす。それらをさらに伸ばしていったぶつかった先の座標を囲んだ範囲がユーザの指定している範囲の広さである。

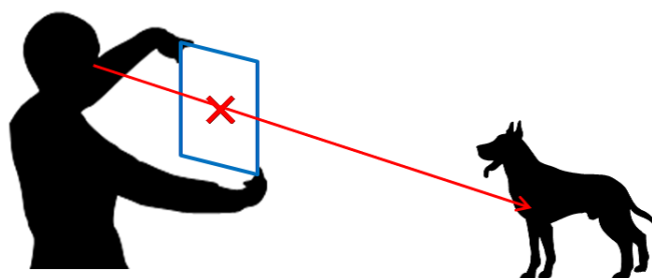


図 3.3: 選択したい範囲における注目点

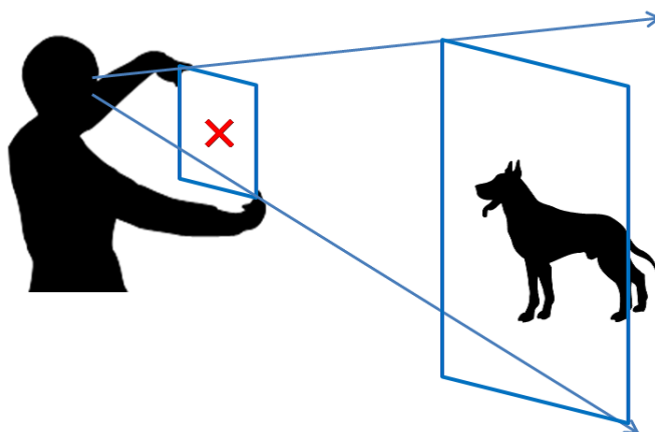


図 3.4: 選択したい範囲の広さ

今回、撮影するときに見える景色はユーザにとって二次元である。このことを考えれば、ユーザの指定したい範囲を計算するときはユーザの注目している座標からユーザの視点までの距離を取り、そして、両手からベクトルを伸ばし物体にぶつかった座標よりも、先ほど取った距離と等距離にあるベクトル上の座標を扱うほうがよいかもしれない。しかし、本研究ではユーザの選択している範囲を認識するのに、ユーザの視界における二次元平面の広さで

はなく、生活空間上の範囲の広さを認識することに焦点を置いたため、このような手法をとらなかった。この両手対角型による範囲選択手法を使うインターフェースでは顔と両手の座標、枠の中心と両手へ顔から伸ばしたベクトルがぶつかった衝突点の計 6 点が計算の元となるが、それらの使い方は選択された範囲を扱う用途によって変化すると考えられる。今回の撮影システムの計算方法は、のちに 4.2.9「PTZ の決定」で述べる。

第4章 撮影システム「HipHopHand」

本研究では範囲選択手法を用いた撮影システムとして「HipHopHand」を実装した。この章では HipHopHand について説明する。

4.1 HipHopHand

HipHopHand はユーザが直感的なインターフェースにより撮影ができるシステムの名称であり、名前の由来は、本研究で扱うハンドジェスチャ「両手対角型」の様子がヒップホップにおける手振りと似ていると考えたためである。

今回提案した範囲選択手法は、よく使われているジェスチャである。この手法は写真や絵の構図を考える際に使われる。この範囲選択手法がカメラの CM¹で使われていることを考慮しても、一般的に撮影するためのジェスチャとして認知されていることがわかる。よってこのジェスチャは撮影しようとする状況において、ユーザにとって自然なものであり、インターフェースとして直感的に扱うことができると考えられる。



図 4.1: カメラの CM におけるワンシーン

¹ソニー製品情報 | α <アルファ> <http://www.sony.jp/products/Consumer/dslr/>

4.1.1 撮影の流れ

撮影の流れは次のようになる。撮影の開始の合図は両手を叩くことにした。

1. ユーザは範囲選択手法を行う合図として両手を叩く
2. ユーザは指定したい範囲を向き、視界の中の指定したい範囲を両手で囲む
3. 指定した範囲が撮影用カメラにより撮影される

4.1.2 撮影の様子

図 4.2 は上の画像から「犬、PC、それら全体」という順番で撮影した画像である。ユーザの囲んだ範囲がしっかりと撮影されていることがわかる。

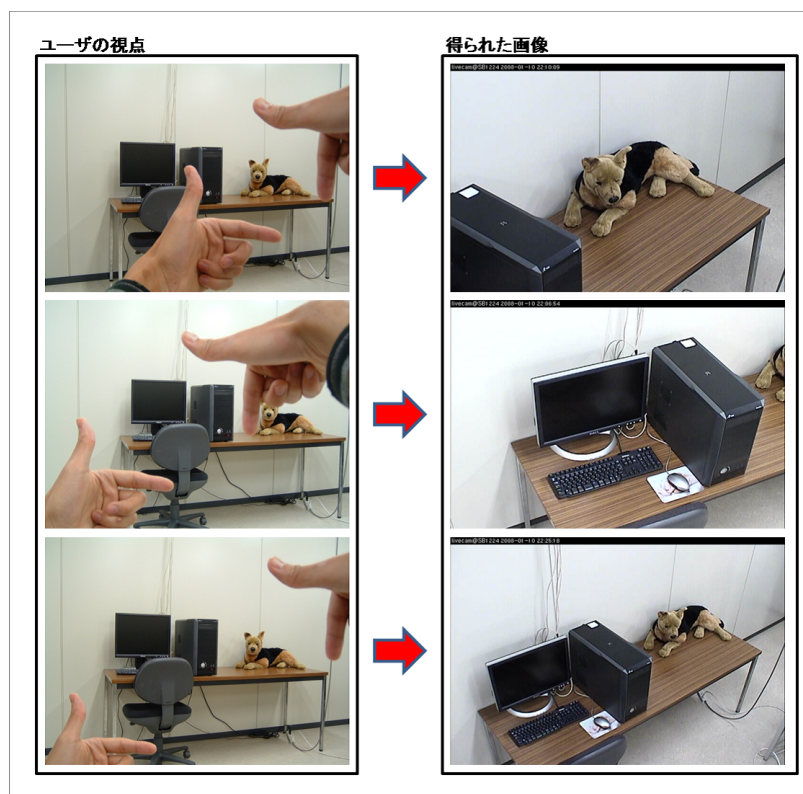


図 4.2: 撮影の様子

範囲選択手法により撮影した画像は、実際にユーザが見ている視点とは違う、撮影用カメラからの視点となっている。本研究は範囲を選択することに主眼を置いているためこの問題を扱わなかった。自由視点画像を生成できる環境を構築すれば、撮影システムとしてさらにユーザが満足できるものとなるだろう。

4.2 システムの実装

範囲選択手法により指定された範囲を認識し撮影するためには、ユーザの顔と両手がどこにあるのか、三次元で認識しなくてはならない。本研究ではそのためのシステムを次のように実装した。

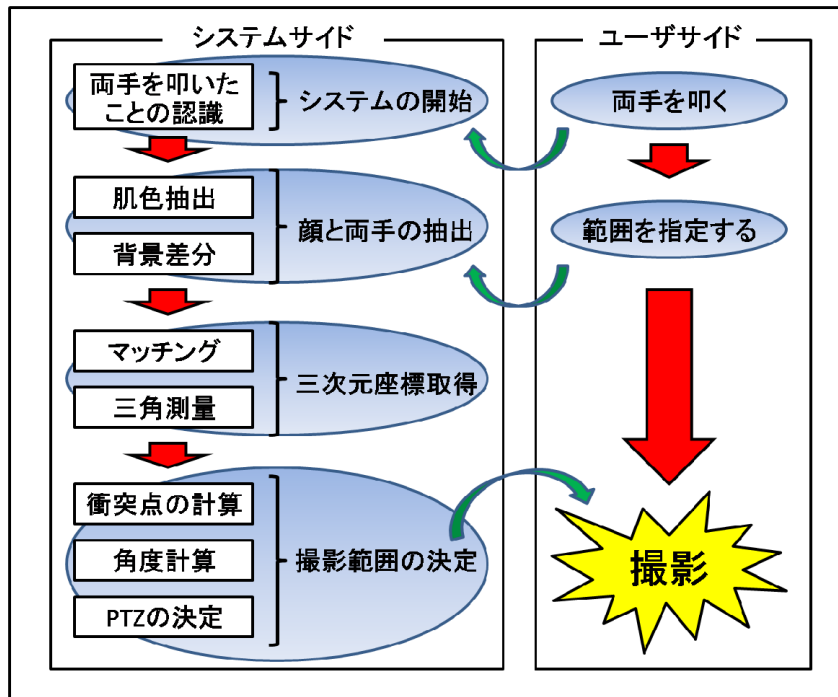


図 4.3: 処理の流れ

この撮影システムは図 4.3 のような流れで処理をする。まずシステムはユーザが両手を叩いたことを合図として処理を開始する。その後、ユーザが伸ばした両手と顔の位置を検出するために「肌色抽出、背景差分」によって顔と両手を画像から抜き出す。得られた顔と両手の画像における重心を用いて、ステレオカメラ上におけるマッチングを行い、マッチングで対応した点同士を使い三角測量により三次元座標を取得する。得られた両手と顔の三次元座標から「ユーザがどこの範囲を指定したいのか」を取得した後、その範囲にカメラを向け、撮影を行う。

4.2.1 システムの構成

図 4.4 はシステムのイメージを、図 4.5 はシステムの実際の様子を示したものである。部屋の天井にステレオカメラとして用いるカメラを2台置く。その他にパン、チルト、ズーム (PTZ) のできる撮影用のカメラを1台配置する。手を鳴らしたことを認識するマイクは座標取得用カメラに内蔵されている。それぞれのカメラは、処理するための計算機とつながっている。

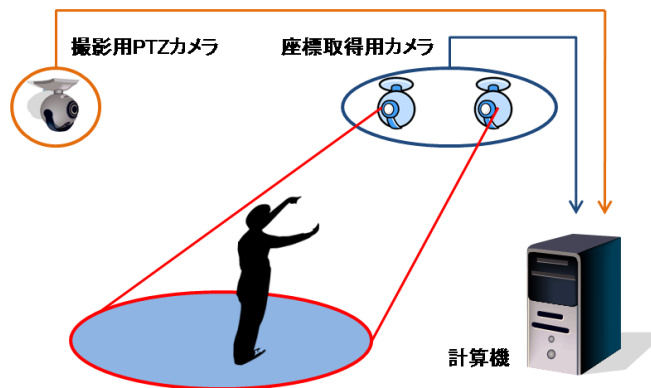


図 4.4: システムの概観

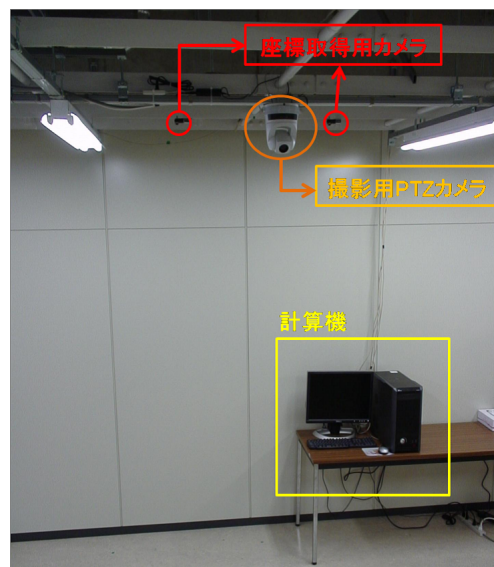


図 4.5: システムの構成

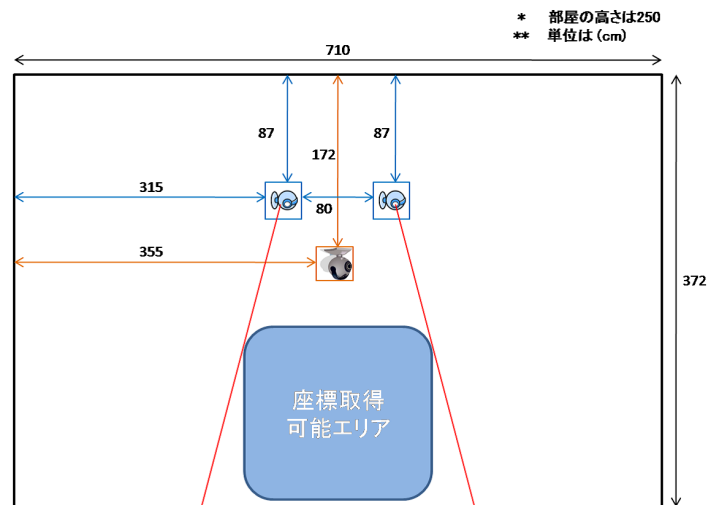


図 4.6: カメラの配置

図 4.6 は上から見た部屋の配置を示している。部屋の大きさは床が $710\text{cm} \times 372\text{cm}$ 、天井までの高さは 250cm であり、撮影用のカメラは天井の中央に取り付けている。座標取得可能エリアとは、ステレオカメラにより顔と両手の座標を取得できる範囲であり、領域は身長 180cm の人が $0.9\text{m} \times 0.6\text{m}$ の範囲内に立てば撮影できた。撮影用 PTZ カメラはその大きさから、部屋の高さより 10cm 低い高さとなる。

4.2.2 開発環境

OS は WindowsXP で VisualStudio を用いて C++により実装を行った。またライブラリとして DirectShow による画像取得とカメラ画像における輝度などのパラメータが設定できる EWCLIB [10] を利用した。

また、顔と両手の座標を取得するのに使用したステレオカメラは Creative 社の Live Cam! Notebook Ultra を、範囲選択を表示するのに使用した撮影用 PTZ カメラは AXIS 社の AXIS 214 ネットワークカメラを用いた。前者のカメラは画像の pixel 数を 640×480 、フレームレート 10fps で使用し、後者は画像の pixel 数を 704×480 で使用した。PC の CPU は Pentium3.2GHz、メモリは 1GB である。

4.2.3 両手を叩いたことの認識

両手を叩いたときに鳴る音は破裂音であり、音声をデジタルデータとして取り込んだ際には通常の音と比べ波形が短時間にとても大きくなる。それを利用してある一定値以上、波形が強くなった場合に手拍子と判断した。

この両手を叩いたことの認識を研究室で試用したところ、友人と普通に話したり何らかの作業をしていても問題はなかったが、稀に机から何か物を落としたときなどに誤認識していた。今回実装した手法では、単純に何か大きい音を出せば誤認識してしまうが、本研究ではシステム開始の合図は本質ではないので、稀に大きい音によって誤認識が起きてしまうのに問題はないとした。

4.2.4 顔と両手領域の抽出

部屋の天井に取り付けられたカメラの画像から顔と両手の3つの領域を抽出する。人間の顔と手を抽出するには肌色分布における色の閾値を使う方法がある。肌色領域の閾値に関してはさまざまな方法を試行した結果、今回は YUV 空間における閾値を扱う。以下に RGB から YUV 空間への変換式と閾値を示す [11]。次に書かれている $Y'U'V'$ は疑似的な YUV 空間となっており、Y に 16、U と V には 128 を加えることで一般的な YUV 空間に変換できる。

$$\begin{aligned} Y' &= (256 \times R + 504 \times G + 98 \times B) / 1000 \\ U' &= (-148 \times R - 291 \times G + 439 \times B) / 1000 \\ V' &= (439 \times R - 368 \times G - 071 \times B) / 1000 \end{aligned} \quad (4.1)$$

$$48 < Y' < 224 \quad -34 < U' < -3 \quad 3 < V' < 127 \quad (4.2)$$

単純に色による識別をするだけでは周囲にある木の机や段ボールも肌色と認識してしまうため、背景差分法によりそれらを除去した。背景差分法は元となる画像からの $Y'U'V'$ の差分の合計がある閾値以上であれば背景ではないと扱う。今回の実装では閾値は 40 とし、背景差分で比較するのに使う画像はシステムが起動する際に撮影する。

図 4.7 は顔と両手領域を抽出している様子である。左上の画像が元となる画像で、右上はその元となる画像を式 4.2 の閾値で肌色抽出した画像である。段ボールと机が閾値の範囲内に入ってしまったため、人物の左上、右下のほうで顔と両手の領域と一緒に抽出してしまっているのがわかる。また、左下は左上の元となる画像とシステム起動時に撮影した画像の差分をとった画像で、顔と両手だけではなく体全体と影を抽出してしまっている。

肌色抽出、背景差分の単体での処理では顔と両手の領域を抽出できない。そこで肌色抽出と背景差分の両方を利用することでこの問題を解決した。肌色抽出と背景差分で取得した画像の白い領域を合わせることで右下の画像を得ることができる。結果としてできた画像にはいくらかノイズがのるが、顔と両手はある一定以上の大きさを持つので、ある一定値以下の大きさのものを取り去ることで除去できる。

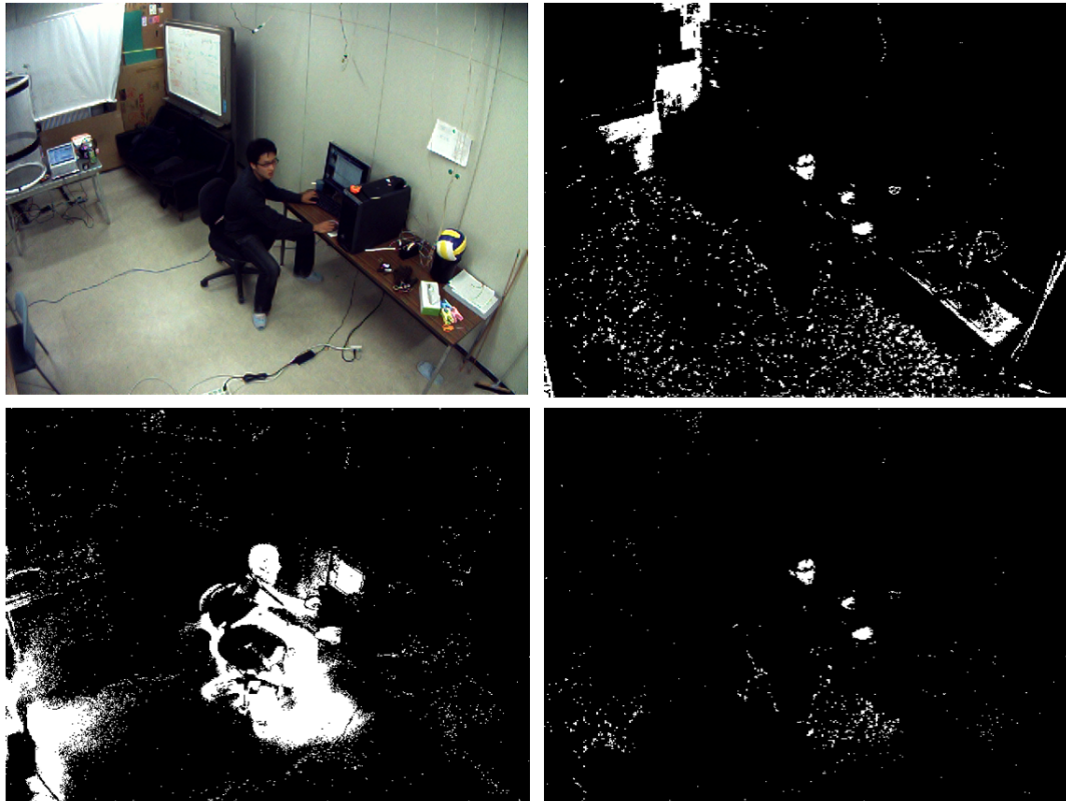


図 4.7: 顔と両手領域抽出の様子
 左上:元画像 右上:肌色抽出 左下:背景差分 右下:結果画像

肌色抽出と背景差分によってできた画像から3つの最大領域を抜き出すことで、顔と両手の領域を取り出す。探索のアルゴリズムは次のようになる。

画像の横の座標を x 、縦の座標を y 、横の長さを $width$ 、縦の長さを $hight$ とする。抽出された画像は肌色抽出と背景差分により黒と白の画像に分けられていて、白色領域が肌色であると認識する。まず画像中の pixel 座標 (x,y) を $(0,0)$ から $(width,hight)$ まで、1pixel ずつ探索していく。そして白色の pixel にぶつかったらその pixel の上下左右を探索し、上下左右も白色ならばさらにその pixel の上下左右を探索し、ということを再帰的に続けていく。こうすることで固まった白色の領域を認識する。白色を発見した際、「その pixel は発見済みである」という情報を格納することで、重複して数えることを避ける。

今回の実装では人間が一人で写っているケースのみを考えているため3つの最大領域を取得することで顔と両手の領域を認識する。そして眼鏡や指輪をしていて肌色領域が離れている場合に対応するため上下左右は数 pixel 離れているところまで探索している。

4.2.5 キャリブレーション

カメラには内部パラメータと外部パラメータがある。内部パラメータとは、レンズの曲り具合や焦点距離など、カメラ自体の性能におけるパラメータである。一方、外部パラメータとはカメラの姿勢や向き、位置など、カメラの配置におけるパラメータのことである。

カメラの内部的なキャリブレーションは Zhang の手法 [12] による OpenCV でのキャリブレーションも試みたが、カメラのドライバに付属されていたもののほうが精度が高かったため、ドライバに付属されているものによってキャリブレーションを行った。外部パラメータはカメラの配置をできる限り並行にあわせることで行った。計算で必要となる焦点距離は Zhang の手法を使い取得した。ここで得られた焦点距離の単位は (pixel) であり、カメラから得る画像のピクセル数を変更するたびに新しく取得しなくてはならない。

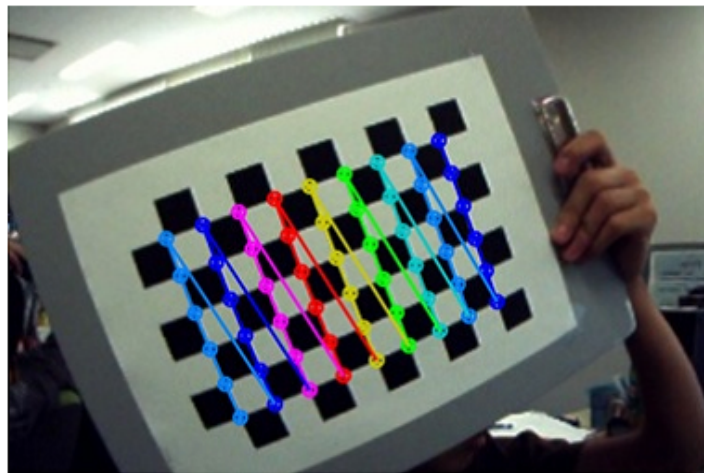


図 4.8: キャリブレーションの様子

Zhang の手法は既存のパラメータを 3 枚以上撮影することでカメラパラメータを推定できるキャリブレーションのための手法であり、そのためのソフトウェアが公開されている [13]。既存のパラメータは長さが既知である白黒の四角形が並んでいる画像であり、この元となる画像のテンプレートがソフトウェアに付属されている。このテンプレートを印刷し、平らな板に貼ってその画像を撮影することで簡単にキャリブレーションができる。今回は左右のカメラ各 10 枚ずつ撮影してキャリブレーションを行った。

4.2.6 三次元座標の取得

三次元座標を取得するには、まず最初にマッチングを行わなくてはならない。ステレオカメラにおけるマッチングとは、ステレオカメラで使う2台のカメラそれぞれの画像中における対応点を見つけることで、今回の場合は顔と両手がそれぞれどのように2つの画像で対応しているか調べる必要がある。

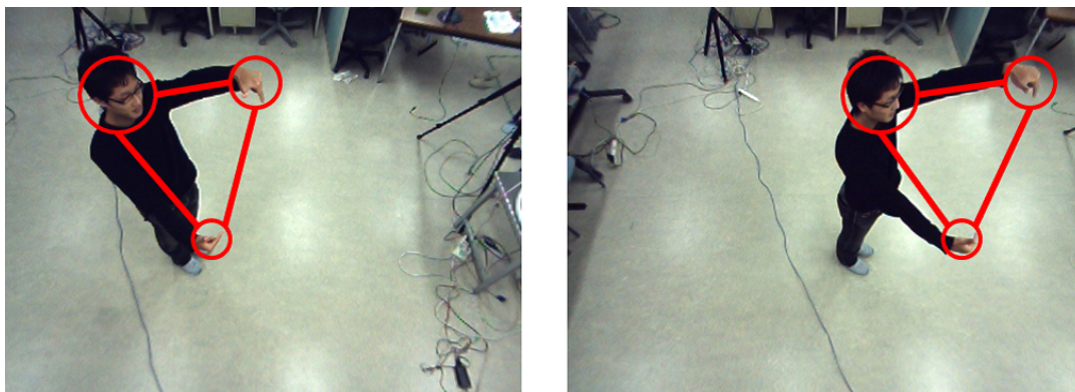


図 4.9: 左右のカメラによるマッチング

図 4.9 はステレオカメラから撮影された画像で、左が左側のカメラから、右が右側のカメラから撮影されたものである。天井に取り付けたカメラは上下逆に取り付けたものを上下反転させているので、通常ステレオカメラと左右反転の対応関係になる。これは計算の際に原点を対象にカメラ画像の横軸のみ反転させることで処理した。

2つのカメラは平行に並べて置かれているので、一方のカメラの画像ともう一方のカメラの画像は縦にはずれず、顔と両手それぞれがほぼ等距離だけ横にずれる。つまり、一方の画像における顔と両手の位置関係がもう一方のカメラでも横に平行にずれたまま同様である。ゆえにそれぞれの位置関係が等しいことを利用してマッチングを調べた。扱っている位置情報は肌色領域の重心である。

三次元座標の取得には三角測量を用いる [14]。図 4.10 は座標系の y 軸の正方向から見たときの図である。カメラ L の座標が原点となり、三次元座標を取得したい物体が P にあるとする。 P の座標はカメラ画像上の座標から次のように取得できる。

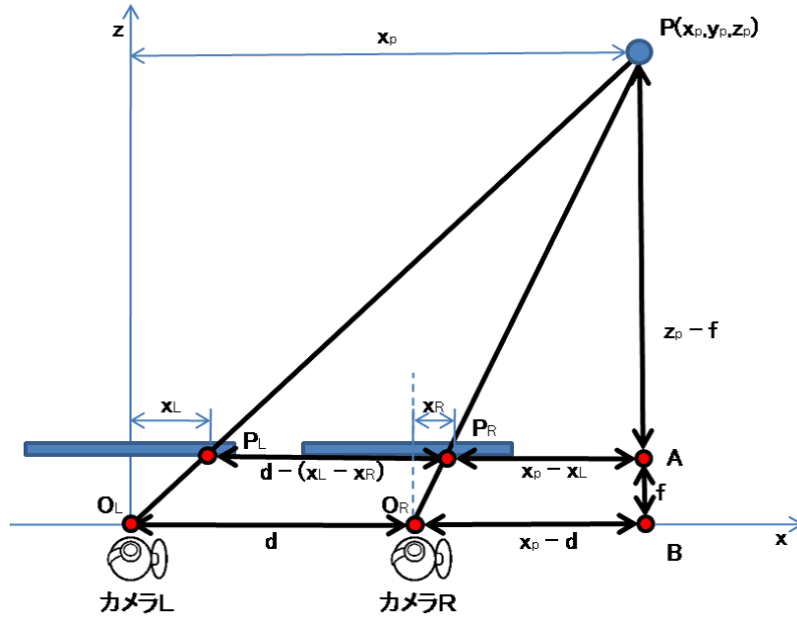


図 4.10: 三角測量

d	: カメラ間の距離
f	: カメラの焦点距離
X_L, Y_L	: カメラ L 画像上の x, y 座標
X_R, Y_R	: カメラ R 画像上の x, y 座標
P	: 三次元座標を測定したい物体

表 4.1: パラメータ

相似となっている三角形 PO_LO_R , 三角形 PP_LP_R の底辺と高さ方向の長さには比例関係があるので次式を得る。

$$Z_p = d \times \frac{f}{(X_L - X_R)} \quad (4.3)$$

同様に三角形 PO_RB , 三角形 PP_RA の底辺に着目すると次式が得られる。

$$X_p = d \times \frac{X_L}{(X_L - X_R)} \quad (4.4)$$

Y_p に関しても同様に行うと、結果として次を得る。

$$Y_p = d \times \frac{Y_R}{(X_L - X_R)} = d \times \frac{Y_L}{(X_L - X_R)} \quad (4.5)$$

ステレオカメラにより顔、両手の座標を取得し計算された衝突点の座標は、ステレオカメラが原点となった座標である。これを撮影用カメラを原点とした座標系に変換するためには、取得された座標を並行移動させ、 x, y, z 軸を中心として回転させることを行う。実際には、ステレオカメラから撮影用カメラへの座標変換は $x = 40, y = 69, z = 65(\text{cm})$ の値で並行移動させ、 x 軸に -35 度、 z 軸に 90 度回転させた。計算には以下の行列にステレオカメラから得た座標を代入して取得する。

x, y, z : 変換前の座標
 x', y', z' : 変換後の座標
 θ : 回転の角度
 $x1, y1, z2$: 移動量

表 4.2: パラメータ

並行移動

$$\begin{bmatrix} x' & y' & z' & 1 \end{bmatrix} = \begin{bmatrix} x & y & z & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ x1 & y1 & z1 & 1 \end{bmatrix} \quad (4.6)$$

x 軸中心に回転

$$\begin{bmatrix} x' & y' & z' \end{bmatrix} = \begin{bmatrix} x & y & z \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{bmatrix} \quad (4.7)$$

y 軸中心に回転

$$\begin{bmatrix} x' & y' & z' \end{bmatrix} = \begin{bmatrix} x & y & z \end{bmatrix} \begin{bmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{bmatrix} \quad (4.8)$$

z 軸中心に回転

$$\begin{bmatrix} x' & y' & z' \end{bmatrix} = \begin{bmatrix} x & y & z \end{bmatrix} \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (4.9)$$

ステレオカメラを原点とした顔、両手の三次元座標取得の計算方法はこのように行った。しかし、実際には肌色認識による肌色領域で顔や両手以外を画像として取得したり、それによるマッチングの誤差が起こることがあるため、これらのノイズを除去しなくてはならない。その方法は次のように行った。

ユーザがシステム開始の合図として両手を叩いたら、範囲を選択するまでの時間として0.5秒待つ。その後画像として12フレームの画像を取得して、それぞれのフレームでの顔、両手の三次元座標を取得する。その12フレームで取得した三次元座標の中に、部屋の範囲の外に座標を認識しているものを除去する。その後、顔と両手それぞれの12フレーム中の平均を出す。そしてその平均から30cm以内にあるものを、正しく座標を取得できたフレームとして扱い、それらの画像から得た座標の平均を顔と両手の三次元座標として扱った。

4.2.7 衝突点の計算

得られた顔と両手の座標からベクトルを作り、何らかの物体に当たるまでの衝突点を計算する。

両手の座標の中点へ顔の座標からベクトルを作り、ベクトルの長さを小さくする。その小さくしたベクトルを両手の中点の座標に繰り返し足していき、その座標が何らかの物体に当たったら、その座標が衝突点であると判断する。この場合、顔から手の長さはおよそ1mだとしたとき、割る値が10であれば最大10cmの誤差が出る。しかし10cmの誤差で得られる座標は目標物の内部になるため、撮影というケースでは気にしなくてよい誤差であると推測した。物体のマッピングは天井、壁、床のみを入力して実装した。

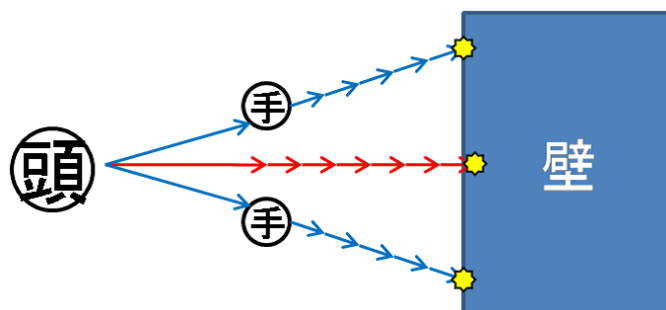


図 4.11: 極小ベクトルを足していく様子

4.2.8 角度計算

PTZ の値を決定するために、4.2.7 のように計算した合計三点へのカメラからの角度を計算する。x,y,z の座標の向きは図 4.12 のようになる。それぞれの座標は矢印の向いている方向に正であり、x 座標は図 4.12 の写真からみて奥の壁に垂直に向かう方向、y 座標は左に向かって、z 座標は床に垂直な方向が正となる。計算は $\cos^{-1}$ によって次式により求めた。水平方向は床と天井に平行な方向、垂直方向は床と天井に垂直な方向を示し、それぞれ撮影用カメラからのパン、チルトに対応している。

$$s = \cos^{-1} \frac{x}{\sqrt{x^2 + y^2}} \quad (4.10)$$

$$t = \cos^{-1} \frac{\sqrt{x^2 + y^2}}{\sqrt{x^2 + y^2 + z^2}} \quad (4.11)$$

$$\text{水平方向の角度} = \left(\frac{s}{\pi} \right) \times 180 \quad \text{垂直方向の角度} = \left(\frac{t}{\pi} \right) \times 180 \quad (4.12)$$

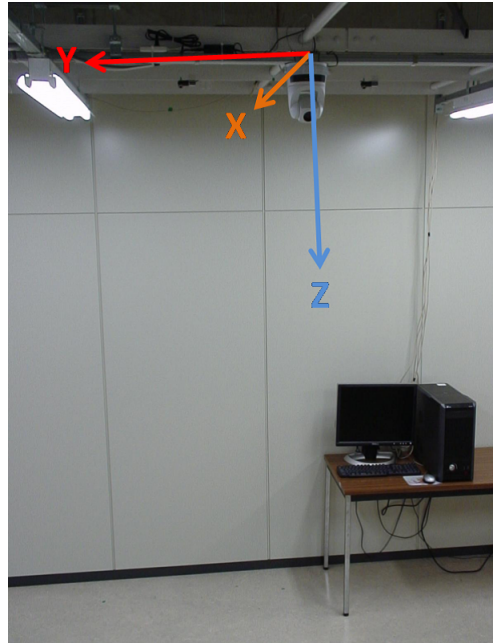


図 4.12: 部屋における座標系の向き

4.2.9 PTZ の決定

得られた衝突点3点へのカメラからの角度よりパン、チルト、ズームするレベルをそれぞれ計算する。このレベルとは、撮影用カメラがパン、チルト、ズームする量を決定する値のことである。

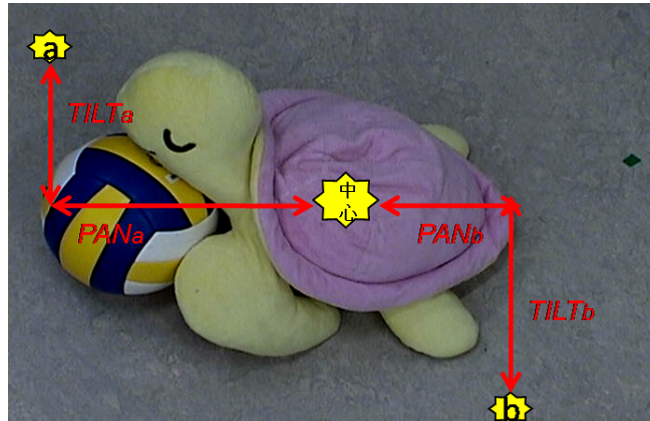


図 4.13: ズームレベルの決定

カメラから見て横方向の角度をパン、縦方向への角度をチルトする。ズームレベルは、中心にカメラを向けた際に、右手と左手から伸ばしたベクトルの衝突点が含まれるようにズームする。今回は二つの衝突点の両方が含まれるようにし、画像の端にぴったり衝突点がくるようにズームする大きさを定める。

図 4.13 は撮影用カメラをパン、チルト、ズームして撮影した画像である。点 a,b は右手と左手から伸ばしたベクトルの衝突した点であり、中心は右手と左手の中点から伸ばした衝突点である。右手と左手の中点を伸ばしたベクトルの衝突点に向けてカメラを動かすので、そこがカメラで撮影した画像の中心となり、中心からの点 a,b への水平方向の角度を PAN_a, PAN_b 、垂直方向への角度を $TILT_a, TILT_b$ とする。

両方の衝突点が入るようにするカメラの画角を求めるには、一番広い角度差に合わせればよい。撮影用カメラで撮影した画像の縦×横の pixel 数は 704×480 なので、パンよりもチルトのほうを大きめにとらなくてははいけない。具体的にはチルトである縦方向の角度を $704/480$ 倍して、 $PAN_a, PAN_b, TILT_a, TILT_b$ の中から最大となる角度を探す。使用する AXIS の画角が最大 48 度、最小 2.7 度でありズームする値が「1～9999」の範囲で比例することから

$$\text{求めたい Zoom レベル} = \frac{48 - \text{最大だった角度}}{48 - 2.7} \times 9999 \quad (4.13)$$

で求めることができる。尚、角度が 48 度以上ならズームレベルを「1」に、2.7 度以下なら「9999」にした。撮影用カメラへのパン、チルト、ズームの命令は以下の URL にアクセスすることで行う。

• [http://\(サーバ名\)/AXIS-cgi/ptz.cgi?=\(値\)&tilt=\(値\)&zoom=\(値\)](http://(サーバ名)/AXIS-cgi/ptz.cgi?=(値)&tilt=(値)&zoom=(値))

第5章 精度評価

実装した撮影システム「HipHopHand」における精度の評価を行った。どの程度の範囲選択ができるかの評価実験を行う前に、まず、実装した撮影システムがどの程度の精度で座標を取得できるか検証する。座標を取得する上で誤差が出る要因として「ステレオカメラにおける誤差」と「肌色の重心を座標として扱うことによる誤差」が考えられ、それぞれの要因がどの程度の誤差をもつか調べる。その後、実際にどの程度の範囲選択ができるかの評価実験を行う。

5.1 ステレオカメラにおける誤差

5.1.1 計測方法

ステレオカメラがどの程度の誤差を持つか調べるのに、ここでは既知の座標でLEDを光らせ、そのLEDの座標をステレオカメラで取得した。計測した範囲はx座標(-60~0)、y座標(-45~45)、z座標(30~210)で、60×90×180cmの直方体の形となっている。その直方体の範囲をそれぞれx,y,zの座標系で30cmごとに区切った場所にて計10回ずつ三次元座標を取得し、その平均を扱った。ここで取得した座標はライブカメラを原点にした座標である。

5.1.2 計測結果

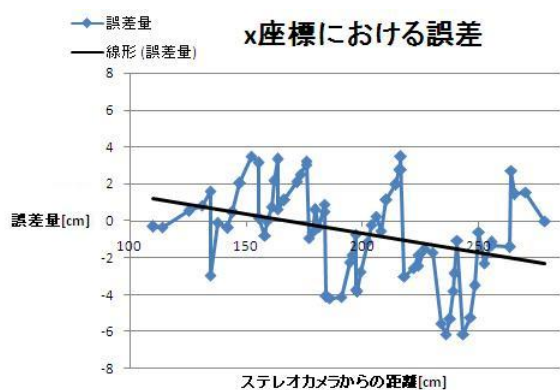


図 5.1: x 座標の誤差

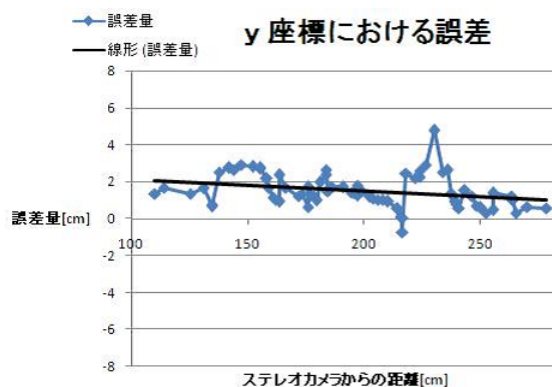


図 5.2: y 座標の誤差

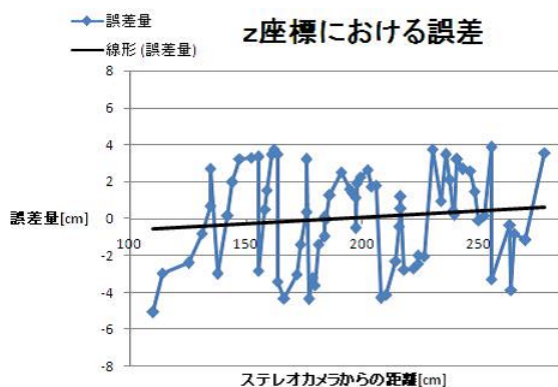


図 5.3: z 座標の誤差

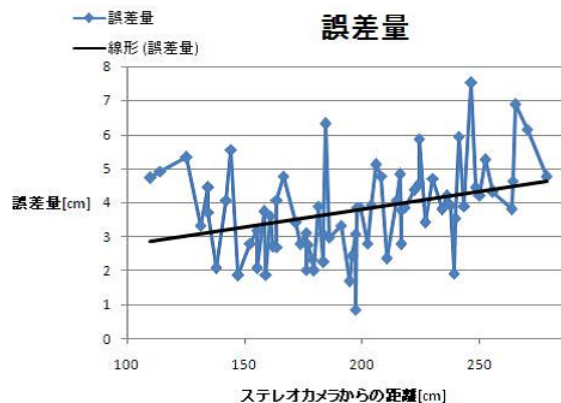


図 5.4: 距離の誤差

表 5.1: ステレオカメラからの距離でまとめた結果

	誤差 (x)	誤差 (y)	誤差 (z)	距離の誤差
平均	-0.6	1.5	0.1	3.8
絶対値の平均	2.0	1.5	2.2	
最大	3.5	4.8	3.9	7.5
最小	-6.1	-0.7	-5.1	0.9
分散	6.2	0.8	6.4	1.7

最初にステレオカメラからの距離でまとめた結果を示す。図 5.1-5.3 は x,y,z 座標の誤差を、図 5.4 は得られた x,y,z 座標より求めた距離を、それぞれステレオカメラからの距離によってまとめたものである。縦軸がそれぞれの値における既知座標からの誤差であり、横軸がステレオカメラから計測点までの距離である。表 5.1 は精度実験で得られた値であり単位は [cm] である。誤差 (x)(y)(z) は x、y、z、それぞれの座標系の誤差の平均である。全体としての距離の平均誤差は 3.8cm となり最大で 7.5cm、最小で 0.9cm であった。

y 座標の誤差は x,z 座標と比べ若干少ない。この y 座標は回転移動する前はステレオカメラからみて横における座標である。x,z 座標における分散が大きいことから、回転移動ではなく、ステレオカメラから見て縦と奥行きの座標の取得が横の座標取得と比べ誤差が大きくなっていることがわかる。また単純な平均では y 座標の誤差が大きかった。正の方向へ一定でずれていること、また、分散が少ないことから、ステレオカメラから撮影用カメラへの原点の並行移動の際、約 1.5cm ほどずれて計算していると思われる。

図 5.4 において、カメラからの距離が離れていくと誤差が大きくなっている。カメラからの距離が遠くなるほど誤差が大きくなっている要因は、三次元座標の計算方法がステレオカメラによる三角測量なので、ステレオカメラからの奥行きが遠くなるほど誤差が大きくなることが考えられる。カメラからの距離が近いときに若干誤差がばらついている要因として、計測時における LED の配置の誤差がカメラから近いゆえに画像中での誤差が多くなってしまったこ

と、また、LED も大きさを持つことが考えられたが、特にそういった傾向はみられなかった。

表 5.2: 座標系ごとに固定してまとめた結果

固定点	誤差 (x)	誤差 (y)	誤差 (z)	絶対値 (x)	絶対値 (y)	絶対値 (z)	距離	分散
x=0	1.09	1.76	0.10	1.49	1.76	2.52	3.77	0.04
x=-30	-0.70	1.22	-0.21	1.89	1.28	1.88	3.43	0.03
x=-60	-2.24	1.48	0.29	2.71	1.48	2.08	4.11	0.11
y=45	-0.75	1.94	-1.63	1.31	1.94	2.43	3.68	0.06
y=15	-1.84	1.88	1.51	2.85	1.88	1.84	4.29	0.11
y=-15	-1.37	1.24	1.69	2.11	1.24	2.06	3.63	0.11
y=-45	1.50	0.88	-1.32	1.86	0.96	2.30	3.50	0.05
z=210	-2.08	1.85	-2.15	2.20	1.85	2.75	4.70	0.06
z=180	-1.93	1.74	-0.65	2.30	1.74	2.29	4.17	0.18
z=150	-0.76	1.39	-0.48	2.15	1.39	2.45	3.86	0.07
z=120	-0.01	1.34	0.37	1.73	1.34	1.51	2.96	0.09
z=90	0.57	1.21	0.66	2.03	1.21	1.35	3.05	0.09
z=60	0.50	1.38	2.60	1.77	1.50	2.60	3.89	0.13
z=30	-2.05	1.74	-1.73	2.18	1.74	2.33	4.26	0.07

次にそれぞれの座標を固定した場合の誤差をまとめたものをみる。表 5.2 は一定の値ごとに固定して誤差の平均を取ったのをまとめたものである。固定点 x=0 とは x 座標の値が 0 でまとめた場合を示す。絶対値 (x)(y)(z) は誤差の絶対値を取り平均したもので、距離は距離の誤差量の平均を取っている。分散は距離の分散を示す。z 座標は値が低ければ低いほど天井に近いので、値が低いほどステレオカメラに近い。

ここで取った値は単純な平均なので一概には言えないが、z 座標の値はステレオカメラからの距離に比例するので、図 5.4 にある近似曲線と似たような増減を示している。x 座標、y 座標の値を固定した際には特に有意な結果は見当たらなかった。

ステレオカメラからの距離でまとめたときと同様に、どの点に固定しても全体的に y 座標はわずかに正の方向に傾いているので、やはり撮影用カメラへの原点の並行移動で誤差が生まれているものと思われる。

カメラの外部パラメータを手動で平行にした場合、カメラが傾くなどして x,y,z 座標のいずれかの座標系が一定にずれたりする。しかし、図 5.1、図 5.3 の線形近似は傾いているが、分散が大きいため、そのような傾向があることをはっきり示せなかった。全体的に誤差が分散する要因の一つとして、ステレオカメラのレンズによる歪みを内部パラメータで修正が十分に満足するものでなかったことが考えられる。

5.2 重心を扱うことにおける誤差

顔と両手の座標を取ってくる時、肌色領域の重心を扱うことが誤差の要因として考えられる。この誤差がどの程度であるか知るため、「0,45,90,135,180 度」のそれぞれの角度における顔、両手の三次元座標を取得し、既知座標との誤差を求めた。

5.2.1 計測方法

最初に LED で座標をとり、その後同じ点から顔、両手の三次元座標をとる。LED により取得された三次元座標を既知座標とし、その後に顔と両手の三次元座標を取り誤差を計ること、ステレオカメラによる誤差の要因を少なくし、ステレオマッチングで重心を扱うことによる誤差を取得した。今回の手法で取得したい点の位置を図 5.5 の赤い点とし、既知座標をその点に合わせて計測する。今回の計測では右手が上の場合と、下の場合とで同じ手を計測している。これは左手の誤差を考える場合は、右手と左手は線対象であり、左右を回転すればいいので問題はないとした。

計測する点はステレオカメラの距離が違う 3 点で、「顔、右手が上のとき、下のとき」をそれぞれ「0,45,90,135,180 °」に向けて 10 回ずつ取り、その平均を取得した座標として扱う。

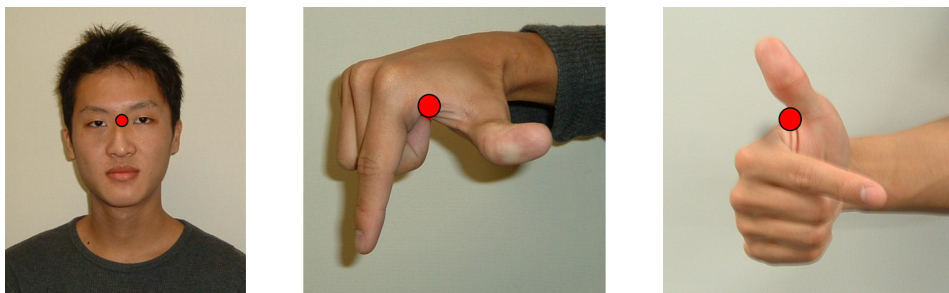


図 5.5: 取得したい座標の位置

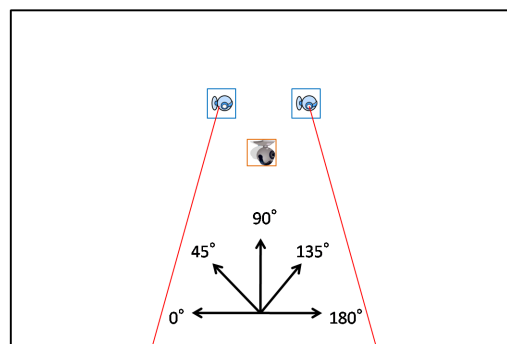


図 5.6: 計測した顔と両手座標の角度

5.2.2 計測結果

計測した結果を示す。表 5.4-5.6 はそれぞれ右手が上にあるとき、右手が下にあるときと顔をそれぞれの角度、距離で計測した結果である。距離 1 の座標は $x=40.3, y=-41.6, z=146.3$ で、ステレオカメラからの距離は 157.3、距離 2 の座標は $x=42.0, y=15.0, z=191.8$ で距離は 196.9、距離 3 の座標は $x=43.0, y=69.7, z=226.1$ で、距離は 240.5 の点を示す。単位は [cm] である。ステレオカメラを原点として x 、 y 、 z の座標系の向きは図 4.10 のようになっている。

顔や両手の角度による距離計測では誤差が見られたものの、規則性はみられなかった。理由として、ステレオカメラに使っているそれぞれのカメラにおいて、肌色領域の認識による重心の取得にずれが、要因として大きく、角度による要因がそれとくらべ小さかったためと考えられる。重心の取得にズレが生じた理由として、カメラの距離が 80cm と比較的広いためにカメラに写る肌色領域の重心と、実際の顔や両手の重心とにズレが生じていると考えられる。おそらく、左側のカメラは顔や両手の左側が多く写ったため左側に、右側のカメラは右側が多く写ったため右側に重心がずれている。ゆえに、ステレオカメラから見て奥側に距離計測の誤差が起こる。 z 座標に注目すると、全体的にカメラより奥に認識している。これをまとめたのが表 5.3 にある。

今回の計測結果が奥にずれていることより、補正することで精度が向上すると考えた。座標取得での距離 1 は、ステレオカメラからの距離を要因とした誤差をなるべく大きく示すため、カメラからの距離が近い点を扱った。実際に撮影する際にはあまり使わない距離なので、距離 2 と距離 3 に焦点を置いて考える。誤差は、三角測量の特性から指数関数的に奥に座標を取得することが予想されるが、撮影で座標を取得する奥行きの範囲は約 60cm であるため線形近似で扱えると仮定した。そのため、今回は式 5.1 によって補正する。式 5.1 では取得した z 座標を、距離 2 と距離 3 が z 座標においてどの程度奥にいつているかの平均をとったもので割ることで補正している。 z がステレオカメラにより取得した座標であり、 z' が補正された座標である。

$$z' = \frac{z}{\left(\frac{191.8 + 46.8}{191.8} + \frac{226.1 + 61.3}{226.1} \right) \times \frac{1}{2}} \quad (5.1)$$

表 5.3: 距離による誤差

	誤差 (x)	誤差 (y)	誤差 (z)	絶対値 (x)	絶対値 (y)	絶対値 (z)	距離の誤差
距離 1	2.6	-4.7	15.0	10.4	7.3	24.0	28.7
距離 2	7.6	4.0	46.8	76	4.1	46.9	48.0
距離 3	9.9	17.3	61.3	9.6	17.6	61.3	64.7

表 5.4: 右手を上にして座標を取得したときの結果

	角度	誤差 (x)	誤差 (y)	誤差 (z)	絶対値 (x)	絶対値 (y)	絶対値 (z)	距離の誤差
距離 1	0 °	-2.9	-2.6	2.5	2.9	2.6	2.5	4.6
	45 °	-3.9	1.3	-7.8	3.9	1.3	7.8	8.8
	90 °	5.2	-6.7	26.3	5.2	6.7	26.3	27.7
	135 °	0.4	0.1	4.5	0.4	0.1	4.5	4.5
	180 °	25.1	-13.5	-11.4	25.1	13.5	11.4	30.7
距離 2	0 °	8.8	1.4	44.6	8.8	1.4	44.6	45.5
	45 °	5.3	3.0	46.9	5.3	3.0	46.9	47.3
	90 °	6.5	6.0	48.6	6.5	6.0	48.6	49.4
	135 °	6.9	7.1	49.9	6.9	7.1	49.9	50.9
	180 °	8.8	5.3	48.6	8.8	5.3	48.6	49.7
距離 3	0 °	8.4	15.2	54.7	8.4	15.2	54.7	57.4
	45 °	7.4	18.7	58.0	7.4	18.7	58.0	61.4
	90 °	8.6	18.9	58.4	8.6	18.9	58.4	61.9
	135 °	11.4	19.6	58.9	11.4	19.6	58.9	63.1
	180 °	12.1	19.3	58.7	12.1	19.3	58.7	63.0
全体	平均	7.2	6.2	36.1	8.1	9.2	38.7	41.7

表 5.5: 右手を下にして座標を取得したときの結果

	角度	誤差 (x)	誤差 (y)	誤差 (z)	絶対値 (x)	絶対値 (y)	絶対値 (z)	距離の誤差
距離 1	0 °	-17.2	3.4	3.3	17.2	3.4	3.3	17.8
	45 °	-8.4	6.8	-23.0	8.4	6.8	23.0	25.4
	90 °	-8.2	7.8	-24.6	8.2	7.8	24.6	27.1
	135 °	11.9	-2.9	45.3	11.9	2.9	45.3	46.9
	180 °	12.6	-10.5	42.6	12.6	10.5	42.6	45.7
距離 2	0 °	5.3	-0.1	48.7	5.3	0.1	48.7	49.0
	45 °	5.0	2.3	51.4	5.0	2.3	51.4	51.7
	90 °	2.8	2.6	42.8	2.8	2.6	42.8	43.0
	135 °	8.7	4.3	52.5	8.7	4.3	52.5	53.4
	180 °	8.0	3.5	47.8	8.0	3.5	47.8	48.6
距離 3	0 °	8.4	14.8	54.7	8.4	14.8	54.7	57.3
	45 °	7.3	16.6	60.9	7.3	16.6	60.9	63.6
	90 °	9.6	18.9	62.4	9.6	18.9	62.4	66.0
	135 °	10.9	17.2	63.6	10.9	17.2	63.6	66.8
	180 °	15.9	17.5	67.3	15.9	17.5	67.3	71.3
全体	平均	4.8	6.8	39.7	9.4	8.6	46.1	48.9

表 5.6: 顔の座標を取得したときの結果

	角度	誤差 (x)	誤差 (y)	誤差 (z)	絶対値 (x)	絶対値 (y)	絶対値 (z)	距離の誤差
距離 1	0 °	-17.5	-19.4	28.7	17.5	19.4	28.7	38.8
	45 °	1.1	-3.9	26.1	1.1	3.9	26.1	26.5
	90 °	16.8	-13.8	29.2	16.8	13.8	29.2	36.4
	135 °	10.8	-6.9	44.1	10.8	6.9	44.1	45.9
	180 °	13.3	-10.4	39.8	13.3	10.4	39.8	43.2
距離 2	0 °	4.8	1.7	45 °	4.8	1.7	45 °	45.3
	45 °	4.0	6.7	50.7	4.0	6.7	50.7	51.3
	90 °	9.4	11.5	54.7	9.4	11.5	54.7	56.7
	135 °	15.7	1.0	28.1	15.7	1.0	28.1	32.2
	180 °	13.9	4.4	43.0	13.9	4.4	43.0	45.4
距離 3	0 °	4.9	11.0	67.6	4.9	11.0	67.6	68.7
	45 °	5.6	16.7	69.5	5.6	16.7	69.5	71.7
	90 °	10.9	21.4	68.4	10.9	21.4	68.4	72.5
	135 °	14.9	22.1	58.7	14.9	22.1	58.7	64.5
	180 °	12.7	11.9	58.3	9.6	10.1	43.6	60.9
全体	平均	8.1	3.6	47.5	10.2	10.7	46.5	50.7

5.3 範囲選択における精度

5.3.1 評価の方針

このシステムにおいてどの程度の精度で範囲を選択できるかの評価実験を行った。範囲選択の評価として「評価実験 1:同一の場所で範囲の大きさを指定する」「評価実験 2:方向性の違う範囲を指定する」の二つの試行により評価を行う。評価実験 1 では 3 つのぬいぐるみを 50cm 間隔で並べて「1 個のみ選択」と「2 個の選択」、そして「3 個全ての選択」を行い精度を検証する。これは足元にある場合と 2m 離れた場所にある場合の 2 回に分けて行う。評価実験 2 では 45 度ごとに 5 つの撮影対象を右から、もしくは左から撮影してもらい、0~180 度方向への範囲選択ができるかを調べる。今回、実際に協力していただいた被験者は 4 名で、22-24 歳の、情報科学か情報工学を学んでいる学生である。

5.3.2 実験の開始の手順

被験者は条件として「長袖を着ていること」「明らかに太い縁をした眼鏡をかけていないこと」を満たすこととする。実験開始前に、被験者には「都合が悪くなったらいつでも実験を中断できること」、また「被験者実験における情報を悪用することがないこと」を伝える。その後、被験者の来ている服のなかに肌色を誤認識させるものがないか調べるため、顔と両手を認識したら 2 値画像として表示するプログラムで確認する。以上のことに問題がなければ、被験者へ実験の説明を始める。

開始前に被験者へ実験の説明をする。まず最初に、「このシステムは範囲を選択し、選択した範囲を撮影するシステムであること」を説明し、実際にどのようなジェスチャでどのように撮影するかを実演する。そして今回、連続して撮影できない状況に陥ることがあるため、以下のことも説明した。

1. 顔は垂直に保たないと光が当たらなくなり認識しなくなる
2. 指定している範囲の認識は天井にあるカメラによって行っているため、カメラからみて手や顔が重なると認識しなくなる
3. 顔がどこにあるかは、肌色領域の大きさにより判断される

説明を終えたら、実際に被験者が満足するまで練習してもらい、その後、指定されたものを順番に囲んでもらう。選択するときは図 5.7 のように床に貼られた緑のテープで囲まれた枠内に立ち選択をする。そして被験者が作業内容を理解できるように目の前にディスプレイを置き、撮影されている画面を映した。

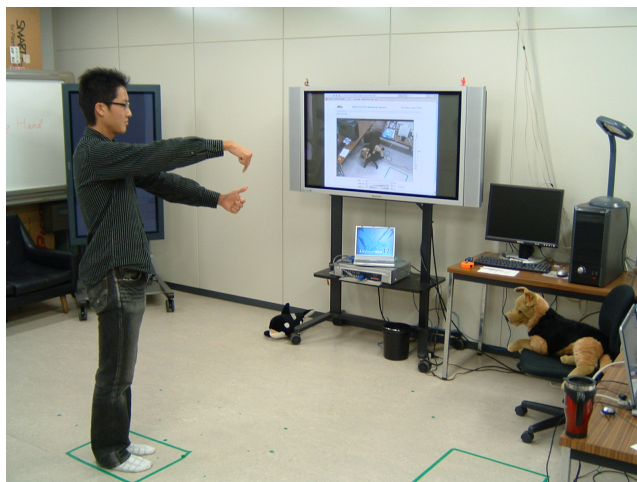


図 5.7: 実験の様子

5.3.3 評価実験 1

ある程度接近した関係にあるものをどれくらい選択できるかを調べるための実験を行った。具体的にはぬいぐるみを 50cm ごとに三つ置く。そしてそれらを三種類の方法「一つだけ、右と真ん中もしくは左と真ん中の二つ、三つすべて」で選択をする。試行した回数は一つだけの選択を 9 回、二つだけを 4 回、三つすべてを 3 回である。近い場合はぬいぐるみをすぐ足もとに置く。遠い場合の選択は 210cm 離れたソファの上に置いた。習熟による誤差を少なくするために 1 人目と 3 人目の被験者は近い場合から、2 人目と 4 人目の被験者は遠い場合から実験を開始した。

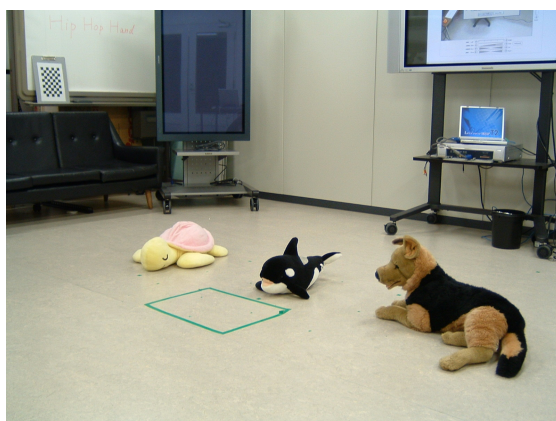


図 5.8: 選択する目標物が近いとき



図 5.9: 選択する目標物が遠い時

5.3.4 評価実験2

方向性の変化による範囲選択の精度を調べるための評価実験を行った。角度の設定は図5.6と同様であり、選択する対象物は0～180度の方向にそれぞれ45度ずつの計5つ置いた。ただ正面に関しては撮影用カメラの設置方向の都合上、カメラを向けることができないので、70度の方向に対象物を置いた。習熟による誤差を少なくするために1人目と3人目の被験者は図5.11にある番号1から5へ、2人目と4人目の被験者は5から1へ選択してもらった。試行はそれぞれを一つずつ選択して5回で一周だとして4周、計20回の選択を行う。

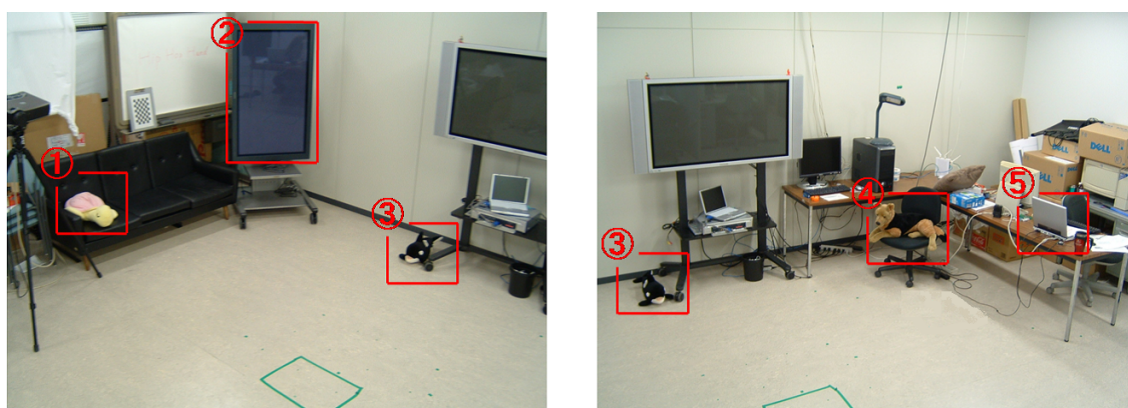


図 5.10: 評価実験2で選択するもの

1. カメのぬいぐるみ 2. 大型ディスプレイ 3. シャチのぬいぐるみ 4. 犬のぬいぐるみ 5. PC

5.3.5 実験結果

表 5.7: 評価実験の結果

試行回数	成功数	失敗数	成功率
363	162	201	0.45

評価実験の結果を述べる。今回のシステムは撮影のシステムとして作成したため、選択した範囲は撮影用カメラの画像に写ったものとする。評価実験の結果を表5.7に示す。試行回数は、うまくいく場合は全体で4人×52回の計208回で終わるが、実際には撮影対象が写るまで試行を繰り返したので結果として363回の試行を行った。

今回の実験ではズームの調整が上手くいかず、ほとんどの試行においてズームの値が「1」で、カメラの最大画角で撮影していた。これは衝突点が予想より広くとられていたためであると考えられる。そこで、一旦ズームでの囲い込みは考慮からはずし、成功したケースをパン、チルトが上手くいったときと仮定した場合で評価する。この成功したケースというのは撮影した画像の中心が目標物に一番近い場合である。評価実験1の一個のみの選択において

は、他のぬいぐるみより選択対象に中心に近い時に成功とする。評価実験 1 の複数個の選択においては、選択対象の端にあるぬいぐるみの中心の間にあれば成功とする。評価実験 2 は選択対象の中心にほかの対象物より近い場合に成功とする。

表 5.8: 条件付きでの評価実験の結果

試行回数	成功数	失敗数	成功率
214	162	52	0.76

結果として全体 363 回の試行のうち、55 %の確率で試行がうまくいかなかった。その失敗した回数のうち、74% が顔と両手から顔を判別することによる失敗であった。そこで、肌色領域のうちどれが顔であるかの認識がうまくいった場合の条件付き確率を調べたところ、表 5.8 のようになった。全体の試行回数は 214 回となり、試行が成功する確率は 76% と、比較的良好な結果を示した。これにより顔と両手の肌色領域から顔を認識することでシステムが改善されることがわかる。以下、条件付き確率と書かれている場合の条件は、顔と両手から顔の認識が出来たとした場合の確率を示す。

表 5.9: 評価実験 1:撮影対象が近いときの結果

選択数	成功	失敗 A	失敗 B	失敗 C	試行回数	成功率	条件付き確率
一個	26	10	16	3	55	0.47	0.67
二個	11	5	17	2	35	0.31	0.61
三個	8	4	9	0	21	0.38	0.67
全体	45	19	42	5	111	0.41	0.65

表 5.10: 評価実験 1:撮影対象が遠いときの結果

選択数	成功	失敗 A	失敗 B	失敗 C	試行回数	成功率	条件付き確率
一個	25	11	30	0	66	0.38	0.69
二個	12	4	7	0	23	0.52	0.75
三個	7	5	3	1	16	0.44	0.54
全体	44	20	40	1	105	0.42	0.68

表 5.9、5.10 は評価実験 1 を撮影対象の距離別にまとめた結果である。失敗 A はカメラが撮影した画像の中に撮影対象が含まれるが「撮影対象の一部が画像からはみ出す」「画像の中心から他の撮影対象の方が近い」ときの失敗である。失敗 B は顔と両手の認識に失敗したとき、失敗 C はズームの値が大きすぎることにによる失敗である。

近い場合と遠い場合の試行の成功率をみたところ、特に差は見られなかった。近い場合は床の四角い枠から 10cm、遠い場合は 200cm ほど離れている。距離が違うことにより試行の成功率に違いが出ると予測したが、今回は違いを示す結果がでなかった。理由として、

床に張った四角いテープの枠内から撮影すると、被験者の手から撮影対象までは近い場合は約 100cm、遠い場合でも約 150cm であった。これは撮影するときに、近い場合は身長分だけ遠くなり、遠い場合は手の長さだけ近くなったからである。これにより、近い場合と遠い場合の差分が少なくなったからであると思われる。

撮影対象の個数による結果では特に大きな差はなかったが、遠いところにある三個の撮影対象を選択する際に成功率が少し下がった。これは、ある撮影者が三つまとめて選択する際に、範囲選択をする時の上側にある手を若干上げる癖があったため、撮影対象が画像の下の方の端で切れることが 3 回連続で起こったためである。

表 5.11: 評価実験 2: 角度ごとにまとめた結果

角度	成功	失敗 A	失敗 B	失敗 C	試行回数	成功率	条件付き確率
0	14	2	10	0	26	0.54	0.88
45	16	0	6	0	22	0.73	1.00
70	16	1	22	0	39	0.41	0.94
135	13	3	24	0	40	0.33	0.81
180	15	1	4	0	20	0.75	0.94
全体	74	7	66	0	147	0.50	0.91

表 5.12: 評価実験 2: 被験者と角度ごとに失敗をまとめた結果

角度	0	45	90	135	180	失敗率
一人目	0	2	2	0	0	0.17
二人目	1	4	0	0	1	0.23
三人目	5	0	10	22	2	0.66
四人目	4	0	10	2	1	0.46
合計	10	6	22	24	4	0.45

角度ごとにまとめた結果を表 5.11 を示す。この表からわかるように、135 度のときに範囲を選択する精度が落ちた。理由として三人目の被験者が左手を上、右手を下にしていたので、撮影したときにカメラから見て両手が重なっていたためである。これにより誤認識が起きていたのだが、被験者はその理由に気づかず連続して試行に失敗してしまった。135 度方向における撮影では 24 回の失敗があるが、うち 22 回がこの被験者によるものである。表 5.12 は被験者と角度ごとに失敗数をまとめたもので、被験者によっては、うまくいかない角度があることがわかる。うまくいかない角度を選択するときは連続して失敗する傾向が強く、三人目の被験者は最大で 11 回連続で試行に失敗していた。135 度以外の角度では、70 度の撮影のときに失敗することが多かった。70 度の撮影のときは、被験者はカメラに向かって両手を突き出す形になる。このためカメラから見て画像中の両手の肌色領域が大きくなるので、比較的成功する確率が低くでたと考えられる。

条件付きの確率の場合は対象間の距離が広いため、比較的良好な結果が得られ、顔や両手の誤認識と重なりがなければ0～180度の方向を選択するのに問題はなかった。

表 5.13: 評価実験 2:被験者ごとにまとめた結果

被験者	評価 1(近)	評価 1 (遠)	評価 2	全体の成功率	条件付き確率
一人目	1.00	0.38	0.87	0.72	0.85
二人目	0.40	0.52	0.73	0.54	0.75
三人目	0.35	0.41	0.46	0.33	0.77
四人目	0.21	0.35	0.61	0.35	0.67
全体	0.46	0.42	0.50	0.45	0.76

被験者ごとにまとめたものを表 5.13 に示す。評価 1(近) は評価実験 1 の撮影対象が近い場合を、評価 1 (遠) は撮影対象が遠い場合、評価 2 は評価実験 2 のことを示し、それぞれの実験における成功率を表示している。今回の実験で被験者によって成功率に違いがあることがわかった。一人目の被験者は評価実験 1 の撮影対象が近い場合の試行において、全ての試行を成功させていた。一人目の被験者は条件を仮定しなくとも、条件付きの四人目の被験者より試行が成功した確率が高かった。三人目の被験者と四人目の被験が上手くいかなかった理由として、上に来ていた服が長袖の T シャツであり、試行を繰り返すたびに裾が肩の方にまわられていったことが挙げられる。

実験終了後にはアンケートを取り、以下のことを質問した。

1. 一つ目、二つ目の実験に関して何か気付いた点や思ったことはないか。
2. 今回の実験で行ったハンドジェスチャはやりやすかったか。また、もっと違うジェスチャで選択したくはなかったか。
3. 素手による撮影は、既存の撮影機器を使うのと比べてどうだったか。

1 つ目の質問ではカメラの位置を気にしながら、両手で範囲を選択するのは、両手の位置の調整が難しいという意見があった。また、遠くの範囲を選択するのは難しく感じた被験者がいた。二つ目の質問では特にこれといった意見はなかった。ただ始まりの合図として両手を叩く以外の方法が欲しいという意見があった。今回は全てうまくいっても一人当たり 52 回両手を叩かなくてはいけない。そして両手を叩いた音の認識率が悪く、認識率は約 67 %だった。次に被験者実験を行う際には、もっと簡単な合図を考える必要がある。三つ目の質問では 4 人中 3 人の被験者がカメラを用いずに撮影できることに利点を感じていた。

5.4 考察

問題点として、今回の評価実験ではズームの調整が上手くいかないこと、三つの肌色領域からの顔の認識率が悪いことがわかった。

撮影というケースにおいて、今回扱った計算手法では注目点の取得によるパン、チルトは上手くいっても、範囲の広さの取得によるズームがうまくいかないということがわかった。広さの指定が上手くいかない理由として、距離が遠くなることに比例して選択している範囲の誤差も大きくなっていくことが挙げられる。

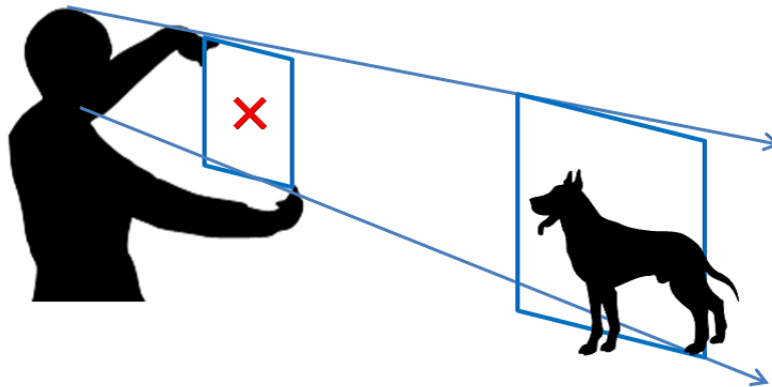


図 5.11: 範囲の広さを選択する改善方法

改善する方法を図 5.11 に示す。人間の視界は一つであり、計算するのには視点が一つでやるのがよいだろうという過程で今回の計算方法を採用した。しかし、三次元座標の取得が上手くいかなかったか、評価実験では範囲を広めに選択していた。他の要因として、予想より人間の視界は広いものかもしれないことが考えられる。ゆえに図 5.11 のように視点を広げることで指定する範囲の広さを指定する方法を挙げる。頭部の上と下から両手の上下へそれぞれベクトルを引き計算することで範囲を計算する。この計算方法ならば、範囲を選択する際に囲む両手の距離を顔の長さより小さくすることで、指定した範囲が遠くなった際にも選択する範囲を発散させることなく選択できる。

肌色領域からの顔の判別方法が今回の評価実験でうまくいかないことがわかった。特に評価実験で 3、4 人目となった被験者は、前髪が 1、2 人目となった被験者より長かったこと、上に着ていたのが長袖の T シャツで範囲選択のたびにめくれていったことを考えても肌色領域の大きさと判別するのは精度のよくない手法であった。改善する方法として、ユーザが手を叩いた瞬間の画像を利用することが考えられる。システムは手を叩いた音を検知したら、最初に画像を取得する。この画像では、手を叩いているのでユーザの両手はくっついている。基本的にユーザが顔より上で手を叩くことはない想定されるので、その画像から取得する 2 点の座標のうち高いほうにあるものを顔座標として扱うことで、精度があがると考える。

第6章 結論

本研究では離れている場所の範囲を選択するジェスチャとして「範囲選択手法」を提案した。この範囲選択手法により、ユーザはデバイスを身につけることなく、自分から離れている範囲を直感的に選択することができる。そしてこの範囲選択手法を用いた撮影システムとして「HipHopHand」を実装した。この撮影システムはユーザが両手を叩くことを開始の合図とし、ユーザは提案した範囲選択手法により撮影したい範囲を選択することで撮影を行うことができる。

その後、本研究では精度実験を行ってこのシステムの精度を検証した。いくつか精度的な誤差が大きい一面もあったが、この実験で誤差の要因が判別された。この要因を修正することで精度は大きく改善されることが予想される。

結果として本研究で提案した範囲選択手法、ならびに HipHopHand の実装・評価を通して、「離れた範囲を選択する」という新しい行動をできるようにした。この成果は、離れた範囲を選択するという新たな利用方法の可能性を広げ、ユビキタス時代におけるハンドジェスチャインタフェースの発展に貢献するものである。

謝辞

本論文を執筆するにあたり指導教員の田中先生、並びに高橋先生、志築先生、三末先生には大変貴重なご意見、アドバイスをいただきましたこと、誠に感謝しております。忙しい中ご指導いただきましたこと、この場を借りてお礼を申し上げます。ありがとうございました。

また、IPLAB のメンバー、特にユビキタスチームの皆様には日頃からの研究に関する話し合い、役に立つアドバイスをいただきまして、本当にありがとうございました。

最後に、私を育ててくれた両親、また、相談に乗ってくれた友人、差し入れを持ってきてくれた友人に感謝します。ありがとうございました。

参考文献

- [1] 保呂毅, 稲葉雅幸. 複数カメラを用いた手書き文字認識システム. In *Workshop on Interactive Systems and Software*, 2006.
- [2] 中村卓, 高橋伸, 田中二郎. Hands-popie:両手の動きを利用した日本語入力手法. In *Workshop on Interactive Systems and Software*, 2006.
- [3] Sören Lenman, Lars Bretzner, and Björn Thuresson. Using marking menus to develop command sets for computer vision based hand gesture interfaces. In *NordiCHI '02: Proceedings of the second Nordic conference on Human-computer interaction*, pp. 239–242, New York, NY, USA, 2002. ACM.
- [4] Yunde Jia, Shanqing Li, and Yang Liu. Tracking pointing gesture in 3d space for wearable visual interfaces. In *HCM '07: Proceedings of the international workshop on Human-centered multimedia*, pp. 23–30, New York, NY, USA, 2007. ACM.
- [5] Mike Wu and Ravin Balakrishnan. Multi-finger and whole hand gestural interaction techniques for multi-user tabletop displays. In *UIST '03: Proceedings of the 16th annual ACM symposium on User interface software and technology*, pp. 193–202, New York, NY, USA, 2003. ACM.
- [6] Joern Loviscach. Two-finger input with a standard touch screen. In *UIST '07: Proceedings of the 20th annual ACM symposium on User interface software and technology*, pp. 169–172, New York, NY, USA, 2007. ACM.
- [7] J. Higuchi, S. Takahashi, and J. Tanaka. Comecam: A communication support system between both ends of the live camera connection. In *CollabTech2007: Proceedings of the Third International Conference on Collaboration Technologies*, pp. 33–38, New York, NY, USA, July 12-13 2007. ACM.
- [8] BEN RAEMER. HULGER ‘ A.PIX ’ DIGITAL CAMERA. http://www.hulgerisation.com/?page_id=51.
- [9] 刀祢太輔, 梶本裕之. ハンドフレームを利用した直感的カメラインタフェースの研究. 日本バーチャルリアリティ学会第12回大会論文集, 9月2007.
- [10] Easy Web Camera LIBrary. http://www.geocities.jp/in_subaru/ewclib/index.html.

- [11] 中川弘隆. 高次局所自己相関特徴による高速画像認識モジュールの開発と自律走行型ロボットへの応用. 2002 年度北陸先端科学技術大学院大学知識科学研究科知識社会システム学専攻学位論文 (修士), 2002.
- [12] Zhengyou Zhang. A flexible new technique for camera calibration. *IEEE Trans. Pattern Anal. Mach. Intell.*, Vol. 22, No. 11, pp. 1330–1334, 2000.
- [13] GML C++ Camera Calibration Toolbox. <http://graphics.cs.msu.ru/index.html.en>.
- [14] 安居院猛, 長尾智晴. C 言語による画像処理入門. 昭晃堂, 2001.