

## 3次元姿勢情報を利用したジェスチャ・インタフェース

山内康晋、土井美和子

(株) 東芝 研究開発センター

〒210-8582 神奈川県川崎市小向東芝町1

TEL. 044-549-2286 FAX. 044-520-1308

{yasunobu.yamauchi, miwako.doi}@toshiba.co.jp

マウスに代表される我々を取り巻く計算機とのインタフェースは、いまだ2次元の域を出ていない。我々は、新しい3次元ジェスチャ入力インタフェースを目指して、距離画像より3次元姿勢情報を検出し、それによるインタフェースを構築している。距離画像の抽出には、開発している入力デバイス：モーションプロセッサを用いた。本論文では、取得した距離画像から求められる、画像モーメント、ならびに撮像物体表面の法線ベクトルを用いて、撮像物体の3次元的な姿勢情報を誤差5度以内にリアルタイムで検出できることを示す。人間の手を使った動作入力インタフェースに応用したアプリケーションを紹介する。手の傾きや回転といった3次元姿勢情報を利用することで、より直感的な情報入力インタフェースが実現できた。

## 3D posture detection for gesture interface

Yasunobu Yamauchi, Miwako Doi

Corporate Research and Development Center, Toshiba Corp.

1, Komukai Toshiba-cho, Saiwai-ku, Kawasaki, Kanagawa 210-8582, Japan

TEL. 044-549-2286, FAX. 044-520-1308

{yasunobu.yamauchi, miwako.doi}@toshiba.co.jp

Conventional human interface devices such as a mouse handle only 2D information. Our purpose is to build a 3D interface by a range image in real-time. A new kinetic and 3D image input device *Motion Processor* can detect the distance from the camera to the object surface in real-time. This paper shows an object's 3D posture detection mechanism using image moments and normal vectors of the target object's surface detected by the *Motion Processor*, and also shows the detection error of posture angle is within 5 degrees through an experiment. The detected 3D posture information is applied to human gesture interface: 3D object manipulation and a virtual ping-pong. These application shows that 3D gesture interface is essential to natural and intuitive communication between human and computers.

## 1. はじめに

近年、コンピュータは処理性能の飛躍的な向上、低価格化などを背景に、オフィスや家庭で広く利用されるようになってきた。また、認識精度の向上に伴う音声認識システムの実装などで、より自然なコンピュータとのコミュニケーションが実現されつつある。さらに、CCD カメラ等に代表される撮像デバイスの実装は、自然画像や動画といったイメージ情報を使ったインタラクションをよりユーザに身近なものとしてきた。しかし、未だ多くの人々にとって、コンピュータは必ずしも直感的に扱えるものになったとは言えない。

我々、人間同士のコミュニケーションでは、音声だけでなく、身振りや手振りといったビジュアルな情報が、コミュニケーション意図の伝達に重要な役割を果たしていると考えられている。このようなビジュアル情報を利用することで、直感的に、自然に、コンピュータを操作できる機構が切望されている。

従来の画像処理の多くは、2次元画像を対象に、ジェスチャなどの認識に関して様々な研究が行われてきている[1,2]。Snake法[7]を適用して撮像画像から手などの対象物を抽出するには、多大な時間と計算量を必要するという問題があった[3-5,17]。コンピュータの進化によりリアルタイムに処理できるようになったとしても、抽出されるのは、2次元での重心、傾き、形状、動きである。3次元操作インタフェース構築には、3次元空間の中で物体の重心、傾き、形状、動きが必要である。3次元画像取得方法としては、カメラ2台を使うステレオ法やレンジファインダがあるが、どちらも装置が大きく、さらに撮像自身がリアルタイムでないという問題がある。

我々は、リアルタイムに対象画像の切り出しが行えるデバイス「モーションプロセッサ」[8]により、3次元操作インタフェースの構築を行っている。モーションプロセッサは、光の反射の原理を用いて、背景画像を取り除くことで、対象物の画像のみを取り出す新規画像処理技術を採用した画像情報入力デバイスである。反射光量は、物体とカメラの距離によって変化する

ので、物体の3次元的な形状情報も取得することができる。

本論文では、モーションプロセッサで切り出された物体の3次元形状情報を基に、3次元空間内での撮像物体の動きや、姿勢検出手法を示す。さらに、人の手を対象に検出された3次元位置、姿勢情報を用いることで、仮想空間内の物体操作やインタラクションが行えることを示す。

以下、2節では対象切り出しを行うモーションプロセッサの概要を説明する。3節では、距離画像を対象に特徴量を算出することで、画像内回転姿勢、奥行き方向の傾き姿勢といった3次元姿勢情報の検出手法について説明し、4節では、手の姿勢情報を利用した仮想物体操作、およびインタラクションを行うアプリケーションを紹介する。

## 2. モーションプロセッサの概要

図1は我々が開発してきたモーションプロセッサと手を対象に撮像された画像である。中央のレンズの周囲に、近赤外光を照射するLEDを配してあり、レンズの奥に受光部のカスタム・センサチップがある。撮像画像を見ると、手の部分のみが検出されており、背景にある物体が撮像されていないことがわかる。これは、モーションプロセッサの動作原理に基づいている。

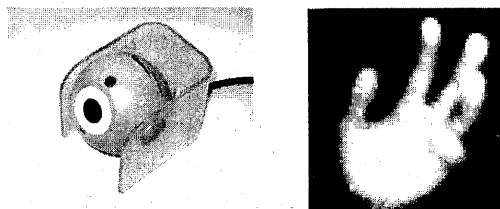


図1 モーションプロセッサと撮像画像

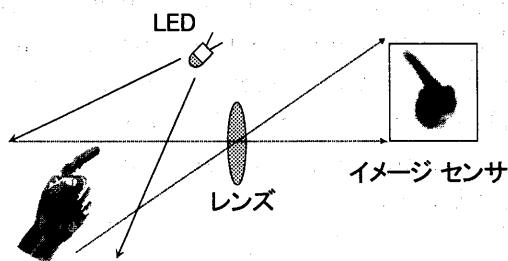


図2 動作原理

図2にモーションプロセッサの動作原理を示す。LED から発光された近赤外光が物体で反射され、センサ部分でその反射光を受光するようになってい

る。つまり、背景からの物体切り出しが可能なのは、反射光は距離の2乗に反比例するので、背景からの反射光は微弱であり、モーションプロセッサでは検出されないためである。また、画像中に見られる濃淡値は、モーションプロセッサから撮像物体までの距離に対応しており、画素値が濃い部分ほど、モーションプロセッサとの距離が近いことを示している。前述のとおり、反射光は距離の2乗に反比例して減衰するため、反射光の光強度を  $I$ 、物体までの距離を  $L$  とすると、式(1)のような関係式で表すことができる ( $k$  は定数)。

$$I = \frac{k}{L^2} \quad (1)$$

このように反射光強度と距離との間に相関関係があるため、反射光強度から、対象物の奥行き距離情報を得ることが出来るのである。厳密には、反射光の強度は、対象物体の表面属性(反射率といった反射特性や色)などに依存するが、手のように全体を通して均一とみなせる物体であれば、反射光の強度をそのまま奥行き情報に対応させることができる。また、撮像系に起因する撮像画素値の非線形性に関しては、補正モデルを仮定することで、奥行き距離値、およびに3次元空間内、同一平面上の濃度分布に関して、補正を行い適正な距離画像を取得することができる[10]。

表1 試作機の性能

解像度	64x64 pixels
奥行き深度	128 階調
動作レート	毎秒 25-50 frames
撮像距離	25-100 cm
画角	対角 70° (奥行 40cm で 40x40 cm <sup>2</sup> )
本体サイズ	W 75×H 80×D 95 [mm]
対応システム	PC/AT 互換機
対応 OS	Microsoft Windows95/98
接続方法	専用ボードで PCI 接続

試作したモーションプロセッサの性能を表1に示す。

以上のように、モーションプロセッサは、背景部分の除去と同時に、対象物の奥行き情報を距離画像としてリアルタイムに取得することの可能なデバイスであり、手や身体を使ったジェスチャ入力に適したデバイスである。

また、モーションプロセッサの特性を生かしたアプリケーションの作成を行う上で、高速な画像処理ライブラリは必須であり、我々は、SDK(Software Development Kit)を並行して開発してきている[9,11]。

### 3. 距離画像を用いた3次元姿勢情報の検出

我々は今まで、モーションプロセッサを利用し、人間の手を使った動作を入力とするマルチメディア・アプリケーションを作成してきた[9]。そこでは、形状パターンといった特徴量、あるいは、対象物体の重心位置やその変化成分を認識することで、アプリケーションを操作していた。その結果、ユーザの意図やテンポに応じたインタフェースの作成が可能となった。

ところが、人間の手の持つ情報量を考えた場合、単なる重心位置や形状パターンといった画像レベルの特徴量だけでなく、傾きや回転といった姿勢情報も重要な特徴量の一つである。特に、アナログな情報量の入力場面や、微妙なニュアンスをジェスチャで入力したい場合、形状パターンといったシンボルレベルの情報や、画像内重心位置といった撮像エリアなどの制約を受ける情報量で扱うよりは、手首の“ひねり”といった姿勢情報を用いる方が直感的で適している。

モーションプロセッサを用いたインタフェースを考えた場合、手のひねり量に応じた特徴量を検出できればよいことになる。そこで、我々は対象物体を構成する面の傾き姿勢に着目することとした。2節で述べたようにモーションプロセッサを用いれば、奥行き方向の距離情報を取得することが出来るため、対象物体表面の傾き具合を3次元的に検出することが可能である。

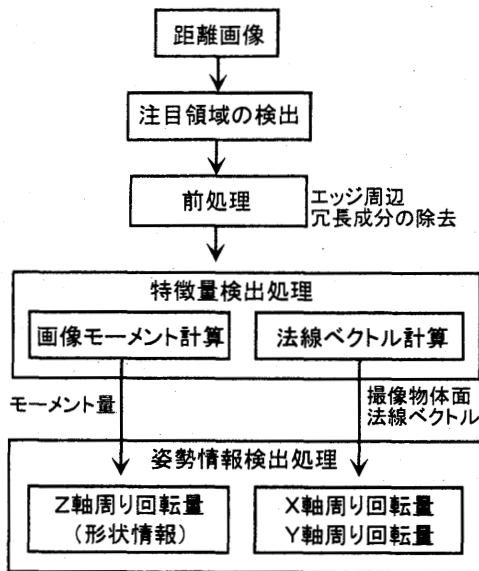


図3 3次元姿勢情報検出処理の流れ

る。図3は本システムにおける3次元姿勢情報検出処理の流れを表したものである。以下では、各ステップを順に説明していく。

### 3. 1 注目領域の検出と前処理

画像中から特定の領域を検出する場合、従来は色相情報や時間差分情報などを用いていた[6]。しかし、これらの手法では、撮像環境に強く依存したり、静止物体を検出できないなど、安定性に関する問題があった。モーションプロセッサを用いれば、距離情報から背景画像を除去することが可能であるだけでなく、上半身の前にある手など距離的に比較的近くにある複数物体を区別することも可能である[12]。ここで検出した複数の画像領域を、それぞれ注目領域 (Region of Interest : ROI) として設定することで、以後の認識処理をそのROI領域の画像のみに限定して行うことが出来るようになっている。これらの領域検出処理により、両手を区別すると同時に、それぞれの手の姿勢情報を検出するといったことが可能である。ROIによる領域切り出し画面の例を図4に示す。左手が四角形領域(ROI)に設定されていることが分かる。

前処理部分では、大きく分けて、ノイズ成分の除去と対象物体輪郭周辺に存在する冗長画素

の除去を行っている。モーションプロセッサは近赤外光の照射、および反射光の受光というアクティブセンサの特性上、撮像物体側面の距離成分も観測されてしまう。撮像物体表面の傾き具合を計測する上で、このような成分は計測誤差となる可能性があるため、輪郭線を構成するピクセルは冗長なピクセルとして除去を行っている。

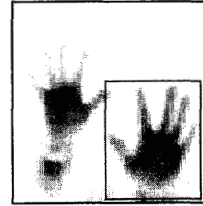


図4 ROIによる手領域の検出

### 3. 2 奥行き軸に対する回転姿勢の検出

モーションプロセッサ撮像画像における撮像物体の画像内回転量を検出することは、3次元空間中における撮像物体の奥行き方向、つまりZ軸周りの回転姿勢を検出していることになる。画像内回転量の検出には、撮像物体の平行移動、大きさに依存しない特徴量である画像モーメントを用いることにする。画像モーメントは、画像中の画素分布から撮像物体の大まかな姿勢検出に用いられる一般的な手法であり、撮像物体の細かな形状の差も吸収できるため、手のような撮像対象に対しても、適用できるものである[13,14]。

画像モーメントは、以下のように求めることができる。 $I(x, y)$ を画素 $(x, y)$ の画素値として、2次までの画像モーメント量 $M$ を以下に示す。

$$M_{00} = \sum_x \sum_y I(x, y) \quad (2) \quad M_{11} = \sum_x \sum_y xyI(x, y) \quad (3)$$

$$M_{10} = \sum_x \sum_y xI(x, y) \quad (4) \quad M_{20} = \sum_x \sum_y x^2 I(x, y) \quad (5)$$

$$M_{01} = \sum_x \sum_y yI(x, y) \quad (6) \quad M_{02} = \sum_x \sum_y y^2 I(x, y) \quad (7)$$

これらのモーメント量から画像重心 $(x_c, y_c)$ 、Z軸周り回転量 $\theta_z$ 、対象物体を最も近似する四辺形物体の長軸 $L_1$ 、短軸 $L_2$ は以下のように求められる。ここで、画像重心は撮像物体の位置、回転量は姿勢、 $L_1, L_2$ は中心軸に対する画像分布の縦横比を表している。

$$x_c = \frac{M_{10}}{M_{00}} \quad (8) \quad y_c = \frac{M_{01}}{M_{00}} \quad (9)$$

$$\theta_z = \frac{\arctan(b, (a-c))}{2} \quad (10)$$

$$L_1 = \sqrt{6(a+c+\sqrt{b^2+(a-c)^2})} \quad (11)$$

$$L_2 = \sqrt{6(a+c-\sqrt{b^2+(a-c)^2})} \quad (12)$$

ただし、

$$a = \frac{M_{20}}{M_{00}} - x_c^2 \quad (13) \quad b = 2\left(\frac{M_{11}}{M_{00}} - x_c y_c\right) \quad (14)$$

$$c = \frac{M_{02}}{M_{00}} - y_c^2 \quad (15)$$

Z 軸周りの回転姿勢量の概念図を図5に、図6に手の撮像画像から、画像モーメントを用いて手の傾き姿勢を検出した様子を示す。図6(b)では、長軸の長さ  $L_1$ 、短軸の長さ  $L_2$  の四辺形オブジェクトを回転角  $\theta_z$  だけ回転させており、手の回転姿勢が検出されていることがわかる。

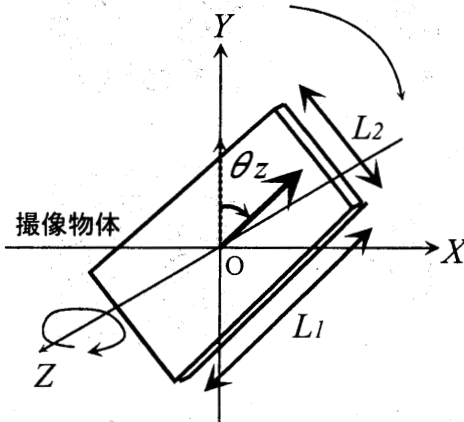


図5 Z 軸周り回転姿勢量の概念図

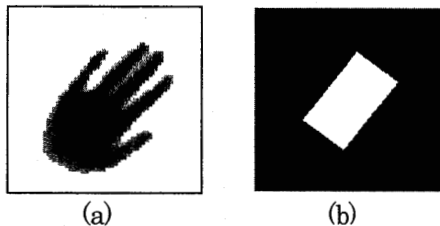


図6 Z 軸周り回転姿勢量の検出結果  
(a)撮像画像 (b)最近似四辺形オブジェクト

### 3. 3 奥行き方向傾き姿勢量の検出

1枚の画像を用いた3次元的な姿勢情報の

検出手法は、大きく2つに分けられる。一つ目は、あらかじめ検出したい姿勢の対象物体を撮像し、その撮像画像をテンプレート画像として登録しておくことで、撮像画像とのマッチングを取る手法である[4,17]。手の奥行き方向傾き姿勢量の検出を考えた場合、カメラに対する手の傾き姿勢を変えて撮像した画像をテンプレートとして複数登録しておき、撮像画像とのパターンマッチングを行うことで、手の姿勢を検出することができる。また、データグローブ等のセンサでテンプレート画像における姿勢量を計測しておけば、手の姿勢量も同時に取得することができる。しかし、この手法では、どの程度のテンプレート数を用意すれば精度よく姿勢検出が行えるか見極めが難しい上に、テンプレート数が増えると、検出までの適合度の計算に多大な時間がかかってしまう。

一方、撮像物体を表現する3次元モデルを定義し、撮像画像に当てはめることで、モデルのパラメータを検出するという手法も存在する[15,16]。手を対象にした傾き姿勢の検出では、指先、指のつけね、指の関節といった特徴点の角度パラメータなどを検出することになる。モーションプロセッサを使えば、各特徴点における奥行き距離値が取得できるため、手のひらを構成する3点を抽出し、姿勢を検出することができる。しかし、手を大きく回転させたり、指が少し曲がっていたりした場合、撮像されない領域（手前の指が奥の指を遮ってしまう現象など）が存在するという問題が起きてしまい、対応点の探索が困難になってしまう。

そこで、我々は、撮像物体表面の勾配情報から生成される法線ベクトルの分布を用いることで、傾き姿勢の検出を行うことにする。まず、モーションプロセッサで撮像した距離画像を三次元空間上にマッピングを行う。画素  $(i,j)$  の画素値を  $\lambda_{i,j}$  とした場合、各画素を  $(i,j, \lambda_{i,j})$  で表現される3次元空間中の1頂点に対応づけることにする。

次に、隣りあう画素、つまり隣り合う頂点で構成される各四辺形パッチの法線ベクトルを求める。まず、隣りあう3次元頂点情報からパッチを構成する各頂点の法線ベクトルを算出した

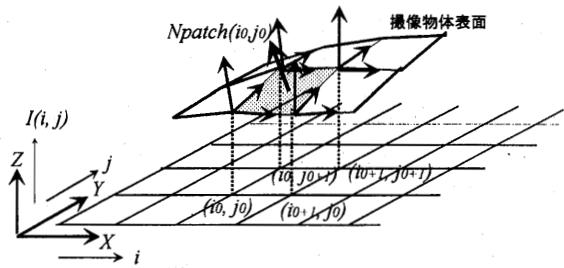


図7 距離画像の3次元表現と法線ベクトル

後、それらを平均化することで、四辺形パッチの法線ベクトルを算出する (図7参照)。

そして、撮像物体を構成する全四辺形パッチにおいて、上記で求めた法線ベクトルを平均化することで、撮像物体表面を代表する法線ベクトル成分を求める。2節で述べたモーションプロセッサの対象物体撮像原理より、各頂点の  $z$  値が0の四辺形パッチは背景とみなせるため、この段階で除外できる。 $N_{patch(i,j)}$ を四辺形パッチ  $patch(i,j)$ における法線ベクトル、撮像物体を構成する全パッチ数を  $C_{patch}$ すると、撮像物体の表面を構成する法線ベクトル  $N_{surface}$  は以下のように求めることができる。

$$N_{surface} = \left( \sum_{allPatch(i,j) \in \text{撮像物体}} N_{patch(i,j)} \right) / C_{patch} \quad (16)$$

最後に、撮像物体表面を代表する法線ベクトルから、X軸、およびY軸周りの姿勢回転量を求める。法線ベクトル  $N_{surface}$  のXZ平面への投影ベクトルを  $N_{surfaceXZ}$ 、YZ平面への投影ベクトルを  $N_{surfaceYZ}$ とすると、X軸周りの姿勢回転量  $\theta_x$  およびY軸周りの姿勢回転量  $\theta_y$  は以下のように求めることができる。

$$\theta_x = \arccos\left(\frac{N_{surfaceYZ} \cdot U_y}{|N_{surfaceYZ}|}\right) \quad (17)$$

$$\theta_y = \arccos\left(\frac{N_{surfaceXZ} \cdot U_x}{|N_{surfaceXZ}|}\right) \quad (18)$$

ただし、 $U_x = (1,0,0)$ 、 $U_y = (0,1,0)$

奥行き方向傾き姿勢量の概念図を図8に示す。図9は、手の撮像画像を対象に、法線ベクトルを用いた奥行き方向傾き姿勢を検出した様子を示している。図9(b)(d)は、撮像画像から検出さ

れた奥行き方向傾き姿勢量であるX軸/Y軸周りの回転量を、平板オブジェクトの回転処理に適用し、3次元表示したものである。手首周りの“ひねり”、“かざし”といった姿勢が検出できていることがわかる。

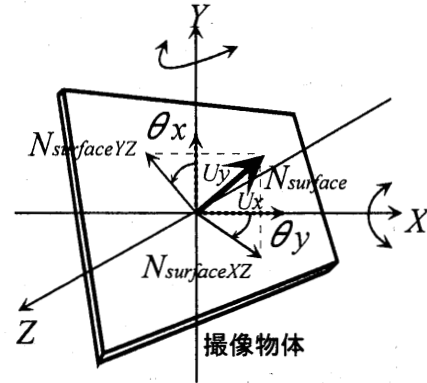


図8 奥行き方向傾き姿勢量の概念図

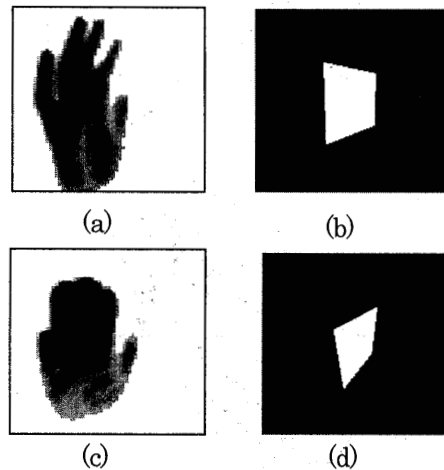


図9 奥行き方向傾き姿勢量の検出結果  
(a)撮像パターン1 (b)3次元表示結果1  
(c)撮像パターン2 (d)3次元表示結果2

### 3. 4 評価実験

3. 3節で述べた手法の実効性を評価するため、奥行き方向の傾き角度の検出実験を行う。ここでは、反射率の様な薄い板を参照物体として、板の断面が影響を与えない0度から80度までの範囲で、5度刻みにY軸周りに回転させ、その奥行き方向傾き角度の検出実験を行った。実験結果を図10に示す。横軸が板の傾き角度、縦軸が検出角度を表している。実験では、

修正を加えていない撮像画像と、2節で述べた画素値と距離値との間に存在する2乗減衰特性を補正し、線形特性に変換した補正画像とに対して行った。なお、各頂点z値に関しては、モーションプロセッサの画像解像度(x,y成分)である64階調と等しくなるよう、0~63の範囲に線形マッピングを行っている。実験の結果、撮像画像に対して補正処理を行うことで、誤差5度以内で検出でき、リアルタイムにインタフェースとして利用できることが確認できた。また、距離値の補正を行わないと、傾き角45度あたりで、最大10度程度大きめの値として検出される傾向にあることも分かる。

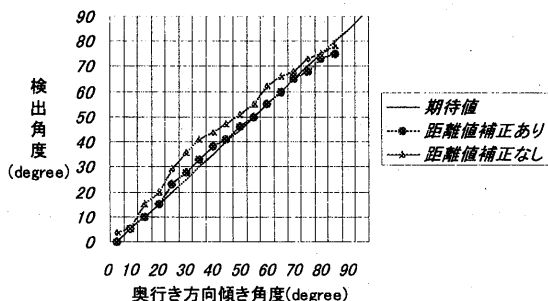


図10 奥行き方向傾き姿勢の検出実験結果

#### 4. 3次元姿勢情報のインタフェース応用

ここでは、モーションプロセッサで撮像した人間の手から、3節で述べた手法で3次元姿勢情報を検出し、インタフェースとして3次元アプリケーションに応用した例を紹介する。

##### 4.1 3次元物体操作

図11は、手の奥行き方向傾き姿勢量に応じて、3次元オブジェクトを回転させて眺めることの出来るアプリケーションの操作画面である。X軸、およびY軸周りの回転量として検出された手の傾き量を、3次元オブジェクトのX軸、Y軸周りの姿勢回転量にマッピングすることで、任意の角度の姿勢制御が可能となっている。ここでは、さらに臨場感を増すために、モーションプロセッサで撮像した手を3次元モデル化し[18]、オブジェクト操作画面にインポーズする

ことで、マウスやキーボードでなく、自分の手のひねり具合で、3次元オブジェクトを操作し、眺めている感覚が得られる。これは、バーチャルモールなどで、ショーケースに展示された商品サンプルを自分の手に取った感覚でいろんな角度から眺める、といった用途が考えられる。

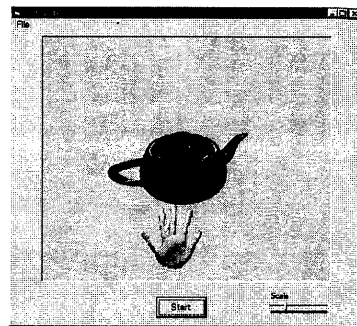


図11 物体操作画面

##### 4.2 バーチャル・ピンポン

図12は仮想空間内に設定された3次元コート内で、人の手と仮想ボールとのインタラクションを行い、ピンポンゲームを行うアプリケーション画面である。人の手は、仮想ラケットとして、仮想空間内にワイヤフレーム表示されている。モーションプロセッサ撮像画像内の手の重心位置から仮想ラケット位置を決定し、モーションプロセッサに対する手の奥行き方向傾き姿勢から仮想ラケット面の角度を決定している。仮想ボールは、仮想ラケット面との衝突が検出された場合、仮想ラケット面に対して正反射するようになっており、ユーザの手の傾け具合が打ち返したボールの軌道に反映するようになっている。手の姿勢がラケットの姿勢として3次元表示されるため、あたかも、自分の手を

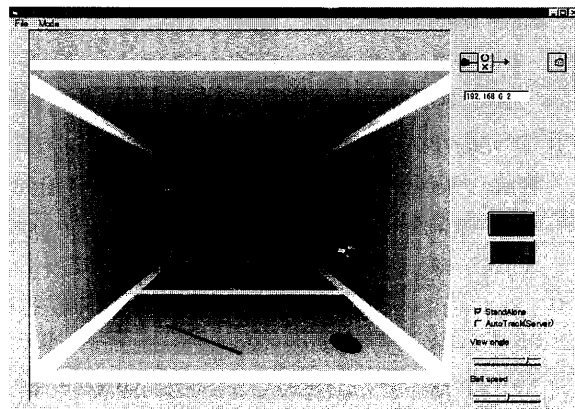


図12 バーチャル・ピンポン操作画面

使ったような感覚でボール操作が実現できる。また、このゲームはネットワークに対応しており、ネットワーク上で離れた相手とのピンポンゲームも可能であり、相手コート手前の仮想空間上に対戦相手の手画像を投影するなど、より臨場感の高いゲーム・シーンが構成できるようになっている。

## 5. おわりに

本論文では、モーションプロセッサによって撮像された距離画像から、画像モーメントによりZ軸周りの回転姿勢量、およびに撮像物体表面の距離勾配から求められる法線ベクトルより、奥行き方向の傾き姿勢量を算出することで、撮像物体の3次元姿勢情報の検出が行えることを示し、奥行き方向の姿勢検出実験からその有効性を確かめた。さらに、この手法を手の3次元姿勢情報の検出に適用することで、特殊な姿勢検出デバイスを装着することなく、非接触で仮想物体とのインタラクション操作が可能なアプリケーションを紹介した。今後は、上半身や両手といったセグメント単位での姿勢検出を通して、身体を使ったジェスチャ入力の可能なインタフェースの構築に取り組んでいきたい。

## 参考文献

- [1] V.I. Pavlovic, R. Sharma, and T.S. Huang, "Visual Interpretation of Hand Gestures for Human-Computer Interaction: A Review", *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol.19, no.7, pp.677-695, 1997.
- [2] 黒川隆夫, "ノンバーバルインターフェース", オーム社, 1994.
- [3] R. Cipolla, Y. Okamoto and Y. Kuno, "Qualitative visual interpretation of 3D hand gestures using motion parallax", *Proceedings of MVA '92*, pp.477-482, 1992.
- [4] T.J. Darell and A.P. Pentland, "Space-time gestures", *Proceedings of IJCAI'93*, 1993.
- [5] Miyake et al., "A gesture controlled human interface using an artificial retina chip", *Conference Proceedings of LEOS'96*, Vol.1, pp.292-293, 1996.
- [6] 岡隆一, 西村拓一, 向井利朗, "しぐさで伝える", *信学会誌*, Vol.82, pp.332-339, 1999.
- [7] M. Kass, A. Witkn and D. Terzopoulos, "Snakes: Active Contour Models", *International Journal of Computer Vision*, pp.321-331, 1988.
- [8] S. Numazaki et al., "A Kinetic and 3D Image Input Device", *CHI '98*, pp.237-238, 1998.
- [9] 三原功雄, 森下明, 梅木直子, 沼崎俊一, 山内康晋, 土井美和子, "ハンドアクションを用いた直感的な情報入力装置", *ヒューマン・インタフェース・シンポジウム*, pp.301-304, 1998.
- [10] 沼崎俊一, 森下明, 梅木直子, 土井美和子, "反射画像を用いた3次元位置検出手法に関する検討", *Interaction '99*, pp.177-184, 1999.
- [11] 山内康晋, 三原功雄, 梅木直子, 森下明, 土井美和子, "モーションプロセッサ用ソフトウェア開発キット", *情報処理学会 第58回全国大会*, No.4, pp.155-156, 1999.
- [12] 三原功雄, 山内康晋, 土井美和子, "モーションプロセッサを用いたビジョン型インタフェースの構築", *WISS '99*, pp.33-42, 1999.
- [13] W.T. Freeman et al., "Computer Vision for Interactive Computer Graphics", *IEEE Computer Graphics and Applications*, May/June 1998.
- [14] B.K.P. Horn, "Robot Vision", MIT press, Cambridge, Mass., 1986.
- [15] J.M. Rehg, and T. Kanade, "Visual Tracking of High DOF Articulated Structures: An Application to Human Hand Tracking", *Computer Vision-ECCV'94*, LNCS, Vol.801, pp.35-46, 1994.
- [16] 岩井儀雄, 八木康史, 谷内田正彦, "単眼動画像からの手の3次元運動と位置の推定", *信学論(D-II)*, Vol.J80-D-II, No.1, pp.44-55, 1997.
- [17] 渡辺孝彦, 李 七雨, 谷内田正彦, "インタラクティブシステム構築のための動画像からの実時間ジェスチャ認識—仮想指揮システムへの応用", *信学論(D-II)*, Vol.J80-DII, No.6, pp.1571-1580, 1997.
- [18] 山内康晋, 三原功雄, 土井美和子, "奥行き距離情報を用いた3次元可視化インタフェース", *情報処理学会 第59回全国大会*, No.特1, pp.71-72, 1999.