

ロボットシミュレータ：ORSにおける 4 J-1 空間ポインティングの実現と動作教示への応用

田中毅 金子俊一 本多庸悟

東京農工大学大学院工学研究科

1. まえがき 本研究の目的は、視覚センサーにより3次元の位置・姿勢を容易に計算機に入力する手法（以下空間ポインティングと呼ぶ）を検討することにある。3色の電球を取り付けたグローブを用い、カラーカメラで撮像した画像から手の位置・姿勢を求める手法について検討する。本研究は、当研究室で開発しているORS（Open Robot Simulator）[1]においてロボットの位置・姿勢を入力するインタフェースを作成するための基礎となるものである。

2. 空間ポインティングの実現 全体の処理の大まかな流れを図1に示す。大きく画像処理と位置・姿勢計算に分けられる。

2.1 画像処理 取り込んだ画像を走査し、電球の像を見つける。この重心が画像における電球の座標 $P_i(X_i, Y_i)$ ($i=1, 2, 3$) である。

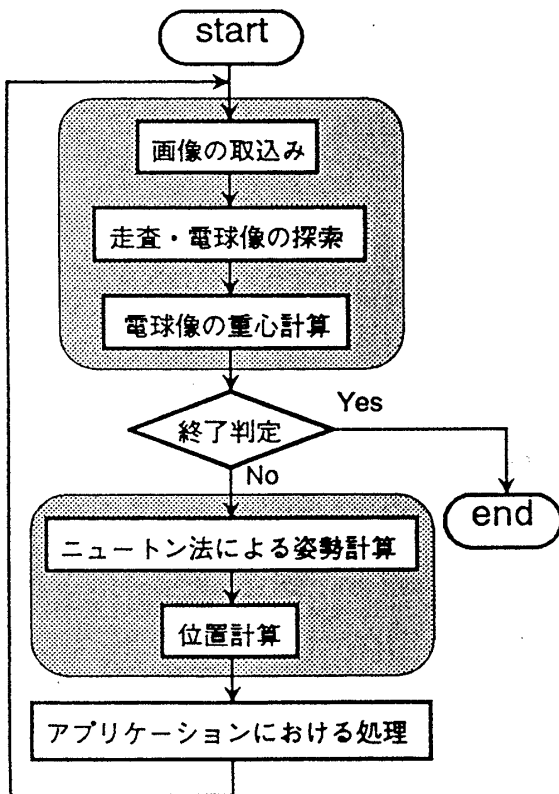


図1 空間ポインティング処理の流れ

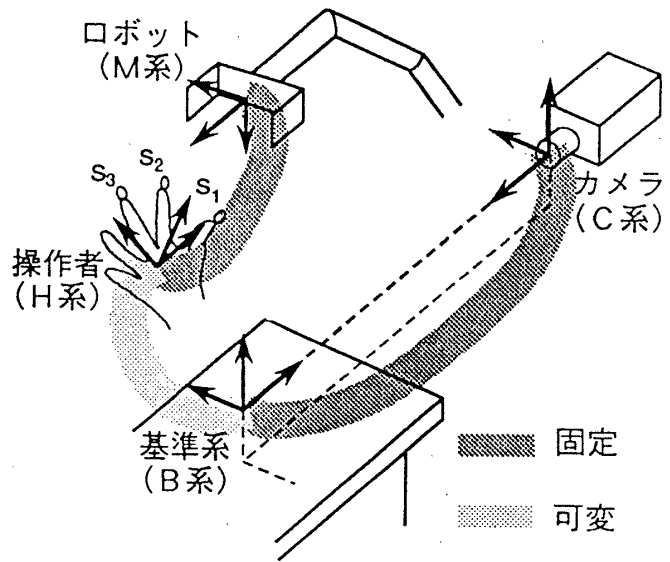


図2 座標系の設定

2.2 位置・姿勢の算出 座標系の設定を図2に示す。まず初期設定を行い、H系における各電球の座標 $s_i = (x_{Hi}, y_{Hi}, z_{Hi}, 1)^T$ (不変) を求める。これは、「最初の手の位置がB系の原点に一致する」という仮定と、H系の原点と電球の距離 d (既知) $= x_{Hi}^2 + y_{Hi}^2 + z_{Hi}^2$ であることから求められる。手の動きによるH系の変化量が手の位置・姿勢 $(x, y, z, \phi, \theta, \psi)$ となる。 $P'_i = (X'_i, Y'_i, 0, w)^T$ とす

ると、 s_i と P_i の関係は次式で表せる。

$$P'_i = T_P T_{CB} T_{BH} s_i \quad (1)$$

$$X_i = X'_i/w, \quad Y_i = Y'_i/w \quad (2)$$

ここで T_P 、 T_{CB} 、 T_{BH} は 4×4 行列であり、それぞれ透視投影、B系からC系への変換、H系からB系への変換（ロール・ピッチ・ヨー回転、並進）を表す。そして、 P_i が正投影像であると近似して、姿勢パラメータだけの連立方程式 $f_i(\phi, \theta, \psi) = 0$ を導出し、ニュートン法（式(3)）によって姿勢を求める。

$$\Omega^{(v+1)} = \Omega^{(v)} - J^{-1} f(\Omega^{(v)}) \quad (3)$$

$$\Omega = (\phi, \theta, \psi)^T, \quad f = (f_1, f_2, f_3)^T, \quad J = \left\{ \frac{\partial f_i}{\partial \Omega_j} \right\}$$

f と J の要素の例を式(4)と式(5)に示す。さらに求めた Ω を式(2)に代入して位置を求める。

2.3 実験 空間ポインティングの結果を図3のような3次元カーソルで表示した。正投影近似のため、わずかに誤差が生ずるが、操作者のイメージどおりにカーソルを動かすことができた。反応も速く、画像取込みから結果表示までの平均時間は約50msecであった。

3. 動作教示への応用 図2のH系からM系への変換は固定とする。ただしM系はロボットのさまざまな部分に対応する。例えばロボットアームの場合にはエンドエフェクタ（以下、EE）や関節、リンク等に置くことができ、操作者の手とEEが一致する、または手でEEを対向してつかむといった形も考えられる。また、初期設定の応用による把握動作の教示も可能である。すなわち、手の位置を固定するかわりに、指を動かすことによって、ロボットアームの開閉を指示することが可能となる。

$$f_i = \Delta Y_{ij} \left\{ \Delta x_{ij} (C\phi C\theta) + \Delta y_{ij} (C\phi S\theta S\psi - S\phi C\psi) + \Delta z_{ij} (C\phi S\theta C\psi + S\phi S\psi) \right\} - \Delta X_{ij} \left\{ \Delta x_{ij} (S\phi C\theta) + \Delta y_{ij} (S\phi S\theta S\psi + C\phi C\psi) + \Delta z_{ij} (S\phi S\theta C\psi - C\phi S\psi) \right\} \quad (4)$$

$$J_{i\phi} = \Delta Y_{ij} \left\{ \Delta x_{ij} (-S\phi C\theta) + \Delta y_{ij} (-S\phi S\theta S\psi - C\phi C\psi) + \Delta z_{ij} (-S\phi S\theta C\psi + C\phi S\psi) \right\} - \Delta X_{ij} \left\{ \Delta x_{ij} (C\phi C\theta) + \Delta y_{ij} (C\phi S\theta S\psi - S\phi C\psi) + \Delta z_{ij} (C\phi S\theta C\psi + S\phi S\psi) \right\} \quad (5)$$

$$C\phi = \cos\phi, \quad S\phi = \sin\phi$$

$$\Delta X_{ij} = X_i - X_j, \quad \Delta x_{ij} = x_{Hi} - x_{Hj}, \quad (i,j) = (1,2), (2,3), (3,1)$$

4. むすび 3次元の位置・姿勢を求めることができ、速度的にも十分な値が得られた。しかし正投影近似による誤差が生ずるので、今後は透視投影の場合も検討していく予定である。また、手の動きの限界を越えた位置・姿勢を指示する手法についても検討している。

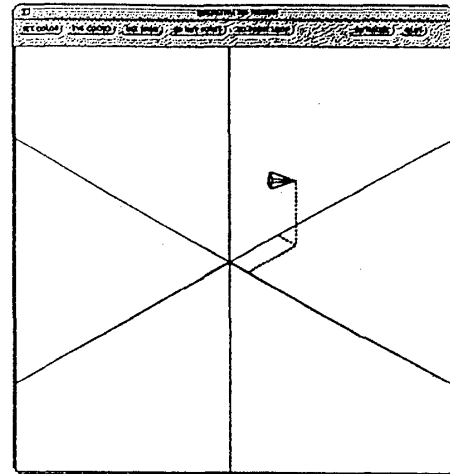


図3 3次元カーソル

参考文献

[1]金子俊一,長山清孝,本多庸悟:開放指向ロボットシミュレータ:ORS,情報処理学会第42回(平成3年前期)全国大会予稿(1991).
 [2]辺見一男,井上和夫:ディスプレイ上の像を把握操作できる3次元インタフェース—マーカー追跡による方法—,電子情報通信学会論文誌D-II, Vol. J74-D-II, No. 9, pp. 1264-1272, (1991).