

指さし動作認識による移動ロボットの操作

Robot Mobilization based on Finger Gesture Recognition

餅川 穂¹ 山下 晃弘¹

Minoru Mochikawa¹, Akihiro Yamashita¹

¹ 独立行政法人国立高等専門学校機構 東京工業高等専門学校

¹ National Institute of Technology, Tokyo College

Abstract: Recently, consumer robots are expected to support people's lives. The robots have to be able to interact intuitively with human when they help people, like carrying items. The interface that uses gestures is attracting attention in recent years. Because it can be an intuitive instruction to humans and we do not need to mount specialized equipment such as a laser pointer when we use it.

In this study, we construct a system of robotic mobilization to the indicated location based on finger gesture recognition using the skeletal tracking capabilities of Kinect (Microsoft Corporation). We use Mindstorms EV3 (LEGO Company) as a mobile robot and mount a Kinect on the robot. In this paper, we report the system and the experimental results on the recognition accuracy of the position pointing.

1. はじめに

現在、民生用ロボットは福祉や家事をさせるなど人の生活を支援するものとして注目されている。ロボットに作業をさせるときには、人が直接ロボットに指示したい場合も想定される。ジェスチャを使ったインタフェースは、人にとって直感的な指示ができ、特殊な器具の装着も必要ないので、近年注目されている。ジェスチャ認識を利用して機械を操作するための基礎研究[1][2][3]が数多く行われている。

これらの研究では、Microsoft が販売する Kinect を用いて人のジェスチャを認識する機能を実装しているが、Kinect の設置場所はロボット自体ではなく、机や専用の台などの固定された場所に設置して実装している例が多かった。つまり、人間のジェスチャを Kinect で認識し、遠隔地にあるロボットを操作する、という利用を想定している。一方で、ロボットと人間とのコミュニケーションを考えた場合、ロボット自体が直接人間のジェスチャを認識し、その結果に応じて適応的に振る舞う技術も重要である。

そこで本研究では、ロボットに Kinect を搭載する形で実装し、ロボットが人間のジェスチャを認識して適応的に行動を制御するシステムの実現を目的とする。今回実装したロボットは、小型のものを使用し、Kinect に角度を付け搭載することで、全身が映るように工夫をした。

本稿では、Kinect の骨格トラッキングシステムを使用して人間の関節の座標を取得し、ジェスチャ認

識を用いてロボットへの移動目標地点を指示するシステムの構築について述べるとともに、指さし位置推定における推定誤差の調査結果と考察について報告する。

2. システムの構築

2.1 システムの概要

被験者の骨格情報を Kinect で取得し、PC で指さし位置の計算を行った。今後は取得した指さし位置の推定値に対してロボットを移動する制御機能を実装する予定であるが、本稿では指さし位置の推定機能に焦点を絞り、その精度と誤差について考察する。システムの構成を図 1 に示す。

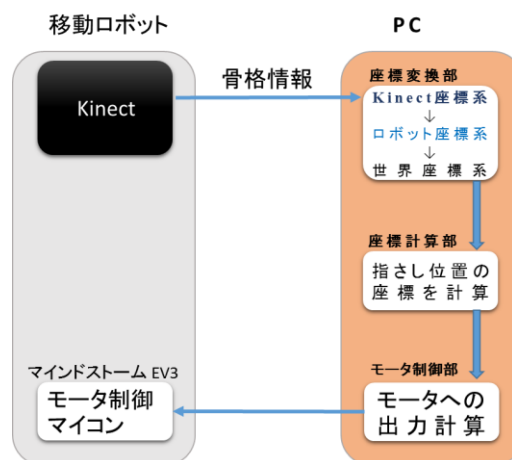


図 1 システムの構成

2.2 Kinect

本研究では Kinect の骨格トラッキングシステムを用いて、「頭、右手、右肘、右肩」の座標を取得した。本研究で用いた Kinect の仕様を表 1 に示す。

表 1 Kinect の仕様

センサー部	色および深度センサーを備えたレンズ
	ボイスマイク
	センサー調整を行う為のチルトモーター
視野角	水平視野：57 度
	垂直視野：43 度
	深度センサーの範囲：1.2m～3.5m
データストリーミング	320×240 解像度：16bit 深度：30 フレーム/秒
	640×480 解像度：32bit 深度：30 フレーム/秒
	16KHz の 16bit オーディオ
骨格トラッキングシステム	2 人のアクティブプレイヤーを含む 6 人までの人体を認識
	アクティブプレイヤー 1 人に対し 20 の関節をトラッキング
チルト角	上下 27 度

2.3 Mindstorms EV3

Mindstorms EV3 は LEGO 社とマサチューセッツ工科大学の共同開発により 2013 年に発売されたブロックロボットシステムである。Mindstorms のモータを 4 つ使用し、Kinect が搭載した形でも走行できるように実装した。曲がる時は左右のモータのデューティー比を変えることで実現している。作成したロボットを図 2 に、Mindstorms EV3 の仕様を表 2 に示す。

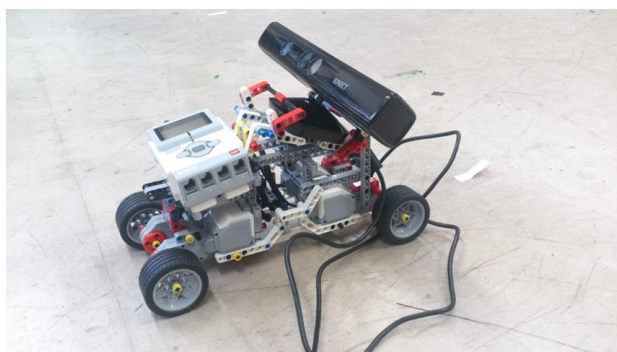


図 2 作成したロボット

表 2 Mindstorms EV3 の仕様

OS	Linux
CPU	ARM9
メモリ	64MB
ストレージ	フラッシュメモリ (16MB)

2.4 座標系の定義

ロボットにジェスチャ認識用のカメラが角度をつけて搭載されているため、得られる関節の座標を変換する必要がある。本研究では座標系を 3 つ定義し、変換を行ってから位置推定を行った。図 3 に座標系の説明の図を示す。

(1) Kinect 座標系

設置された Kinect を基準とした座標系でカメラを原点とする。

(2) ロボット座標系

ロボットを基準とした座標系でカメラから地面に向けた垂線と地面の交点を原点とする。この座標系は地面と水平となる。

(3) 世界座標系

人間を原点とした座標系とする。地面に対して垂直な方向を Y 座標とし、人間が向いている方向を Z 座標、Z 座標および Y 座標に直交する方向を X 座標と定義する。

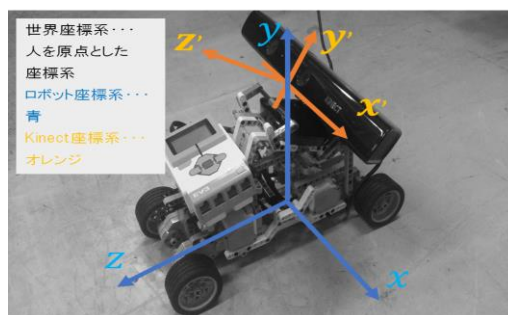


図 3 座標系

2.5 座標系の変換

(1) Kinect 座標系からロボット座標系への変換

Kinect 座標系からロボット座標系への変換は次に示す回転行列による回転と原点を地面まで平行移動することで得る。

$$\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \theta & -\sin \theta \\ 0 & \sin \theta & \cos \theta \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix}$$

図 4 Kinect 座標系からロボット座標系への変換

(2) ロボット座標系から世界座標系への変換

Y 軸中心で回転し、原点を人間のところまで水平移動させることで得る。

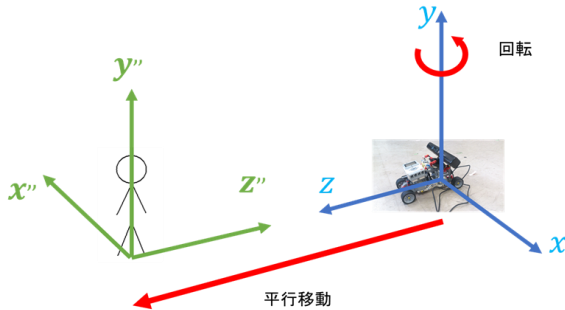


図 5 ロボット座標系から世界座標系への変換

2.5 指さし位置の推定

長谷川らの研究[1]で、指さしジェスチャによる推定位置を求める手法が提案されている。本研究とは Kinect の設置方法などシステム構成は異なるが、指さし位置の推定位置を求める式は同様の式を利用した。Kinect と地面のなす角を θ とする。人が示した位置の 3 次元は

$$\frac{x - x_{head}}{x_{hand} - x_{head}} = \frac{y - y_{head}}{y_{hand} - y_{head}} = \frac{z - z_{head}}{z_{hand} - z_{head}} \quad \text{式 1}$$

により求める。式 1 は頭と右手の座標から得たベクトルである。式 1 を変形し、指さし位置を式 2 と式 3 のように得る。

$$x = x_{head} + \left(\frac{h_{camera} - y_{head}}{y_{hand} - y_{head}} \right) (x_{hand} - x_{head}) \quad \text{式 2}$$

$$z = z_{head} + \left(\frac{h_{camera} - y_{head}}{y_{hand} - y_{head}} \right) (z_{hand} - z_{head}) \quad \text{式 3}$$

同様にして、右肘と右手、右肩と右手の座標も用いて指さし位置の推定位置を求めた。右肘と右手の座標を用いて求める人が示した位置の 3 次元は

$$\frac{x - x_{elbow}}{x_{hand} - x_{elbow}} = \frac{y - y_{elbow}}{y_{hand} - y_{elbow}} = \frac{z - z_{elbow}}{z_{hand} - z_{elbow}} \quad \text{式 4}$$

により求める。式 4 を同様に変形し、式 5 と式 6 はそれぞれ以下のようなになる。

$$x = x_{elbow} + \left(\frac{h_{camera} - y_{elbow}}{y_{hand} - y_{elbow}} \right) (x_{hand} - x_{elbow}) \quad \text{式 5}$$

$$z = z_{elbow} + \left(\frac{h_{camera} - y_{elbow}}{y_{hand} - y_{elbow}} \right) (z_{hand} - z_{elbow}) \quad \text{式 6}$$

右肩と右手の座標を用いて求める人が示した位置の 3 次元は

$$\frac{x - x_{shoulder}}{x_{hand} - x_{shoulder}} = \frac{y - y_{shoulder}}{y_{hand} - y_{shoulder}} = \frac{z - z_{shoulder}}{z_{hand} - z_{shoulder}} \quad \text{式 7}$$

式 7 を同様に変形し、式 8 と式 9 はそれぞれ以下のようなになる。

$$x = x_{shoulder} + \left(\frac{h_{camera} - y_{shoulder}}{y_{hand} - y_{shoulder}} \right) (x_{hand} - x_{shoulder}) \quad \text{式 8}$$

$$z = z_{shoulder} + \left(\frac{h_{camera} - y_{shoulder}}{y_{hand} - y_{shoulder}} \right) (z_{hand} - z_{shoulder}) \quad \text{式 9}$$

3. 実験

3.1 実験設定

2.5 節で示した式 2 と式 3 を利用して、右利きの 10 人に対して被験者の前方の床面に設置した 8 個のマーカーの位置に対する指さし動作の計測精度を求める実験を行った。人を原点としたとき、カメラは地面から高さ 0.13m、地面に対して 56° 、マーカーの間隔を 1m で配置した。指さしを必ず右手で 10 回行って精度を検証した。また、指さし位置の誤差が人の癖かアルゴリズム判断するためにレーザポインタを用いた実験も行った。図 6 にマーカーの配置を示す。図 7 に実験の様子を示す。

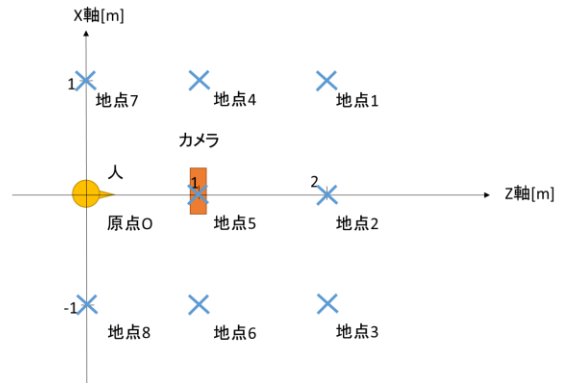


図 6 マーカーの配置



図 7 実験の様子

3.2 指さし位置の推定誤差

結果を図 8, 図 9, 図 10 に示す. 点の中心は推定位置の平均値で点の大きさは分散の大きさに比例するように作図した. 右肩—右手のベクトルが 1 番に誤差が小さく, 2 番目が頭—右手のベクトル, 3 番目が右肘—右手のベクトルとなった. 不偏分散は頭—右手のベクトルが 1 番に小さく, 2 番目は右肩—右手のベクトル, 3 番目が右肘—右手のベクトルとなった.

今回の実験データはノイズ除去を行ったデータで検証をした. ノイズ除去は X 軸, Y 軸, Z 軸のメディアンからのユークリッド距離を求め, その標準偏差を求めた. ユークリッド距離のメディアンを中心にして, 標準偏差の 2 倍を半径にした球から外のデータをノイズとした. ノイズ除去を行わない場合では誤差が 1m を超える地点もあった. 実際にロボットを移動させる時には, ノイズ除去を行い位置推定する必要があるといえる.

被験者それぞれ 10 回ずつ測定し, 得られた結果から, 前述の方法でノイズと判断したデータを除去し, まず, それらの平均値を計算する. その平均値と真値との誤差を計算し, 誤差について被験者 10 人分の平均値を計算した. その結果を表 3 に示す.

また, 被験者それぞれ 10 回ずつ測定して得られた結果の平均値について, 被験者 10 人分のデータの不偏分散を計算した結果が表 4 である. 表 4 が示す値は, 異なる被験者が指さし動作を行った場合にどれだけ個人差が生じるかを定量的に測定した値と捉えることができる.

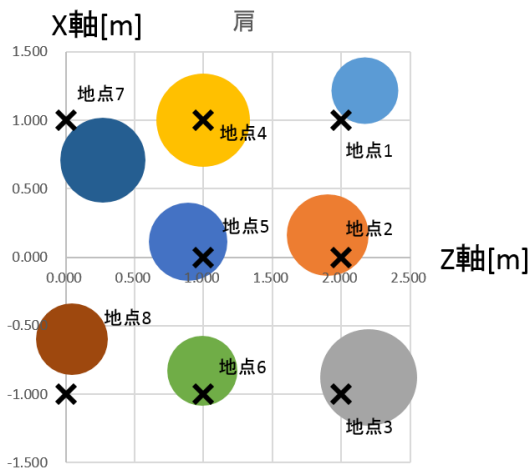


図 8 指さし位置の誤差 (肩—一手)

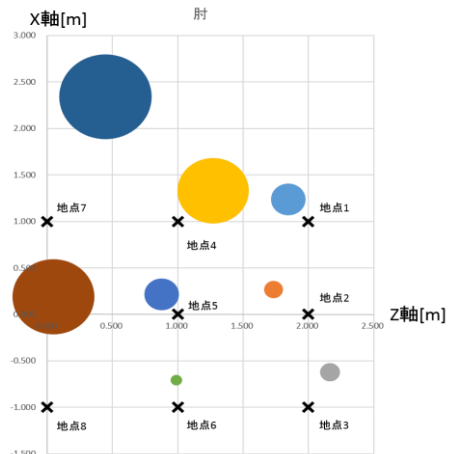


図 9 指さし位置の誤差 (右肘—一手)

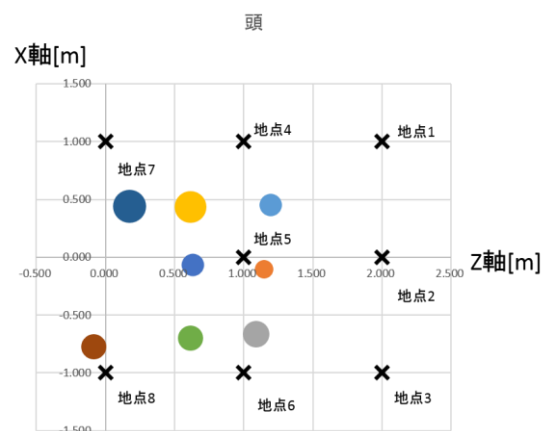


図 10 指さし位置の誤差 (頭—一手)

表 3 誤差の平均値

誤差の平均	肩		肘		頭		理論値	
	Z	X	Z	X	Z	X	Z	X
地点1	0.310	0.322	0.558	0.487	0.802	0.553	2	1
地点2	0.290	0.212	0.416	0.281	0.855	0.105	2	0
地点3	0.389	0.155	0.744	0.376	0.910	0.335	2	-1
地点4	0.164	0.191	0.522	0.627	0.387	0.567	1	1
地点5	0.140	0.131	0.197	0.228	0.368	0.068	1	0
地点6	0.110	0.171	0.220	0.292	0.382	0.300	1	-1
地点7	0.275	0.285	0.486	1.502	0.177	0.564	0	1
地点8	0.068	0.228	0.405	0.995	0.081	0.226	0	-1
平均	0.218	0.212	0.443	0.598	0.495	0.340		

表 4 指さし位置の不偏分散

不偏分散	肩		肘		頭	
	Z	X	Z	X	Z	X
地点1	0.236	0.206	0.829	0.431	0.009	0.006
地点2	0.181	0.030	0.704	0.027	0.021	0.003
地点3	0.254	0.031	1.110	0.077	0.008	0.006
地点4	0.043	0.068	0.857	0.965	0.004	0.030
地点5	0.026	0.012	0.151	0.033	0.005	0.002
地点6	0.022	0.014	0.191	0.065	0.004	0.003
地点7	0.067	0.256	0.220	4.897	0.004	0.008
地点8	0.006	0.575	0.724	10.089	0.001	0.002
平均	0.104	0.149	0.598	2.073	0.007	0.007

3.3 考察

実験結果は Z 軸の不偏分散も誤差の平均も人の左

側の点が右側の点よりも小さくなった。X 軸は右側が小さくなった。真値からの距離は左側にある点のほうが誤差大きくなった。右手で指さしを行ったので、左側が指し辛かったのが原因だと考えられる。

頭一手のベクトルは真横の地点以外は推定位置が真値よりも手前になり、肩一手のベクトルよりも誤差が大きくなった。手の地面に対する角度よりも頭一手の延長線の角度が大きくなったためである。しかし、不偏分散が最も小さくなっているため、補正を行うことで正確な指さし位置を得られる可能性が高いといえる。手の地面に対する角度と頭一手の地面に対する角度の誤差を減らすために頭の Y 軸座標を下げたり、得られた推定値に係数を乗算したりする方法が考えられる。

レーザポインタを用いた実験は指さしで行った実験よりも誤差が大きくなった。レーザポインタで当てている位置は手首の角度で大きく変わり、必ずしも肩一手の延長線を当てられていたとは限らない。なので、持ち方の癖で指さし位置に誤差ができたといえる。

4. まとめ

本研究では室内での指さし位置を認識するシステムを開発した。今後は、ロボット指さし位置の精度を高める。指さし位置の精度を高める方法として推定位置の補正や検出した座標のノイズを除去する機能を検討し、実装する。また、別の座標を用いたり、カメラと人との位置関係を変えて実験を行ってみたいとして別の指さし位置の推定手法も検討する。

参考文献

- [1] 長谷川 勉：“ジェスチャによる移動ロボットへの動作目標の指示と誤差修正”，第 24 回日本ロボット学会学術講演会，2011
- [2] 渡邊茂：“NUI を用いた移動ロボットプラットフォームの制作”，千葉職業能力開発短期大学校紀要 第 18 号，2013
- [3] 矢田幸大：“指さし対象認識による個人用知的移動体の直感的な操作”，情報処理学会第 75 回全国大会