

モーションキャプチャを用いた回転翼飛行ロボットの リアルタイム姿勢計測

佐々木卓哉¹, 本田泰²

¹ 室蘭工業大学 情報電子工学系専攻

² 室蘭工業大学 しくみ情報系

概要

本研究では、モーションキャプチャを用いて回転翼飛行ロボットの姿勢を計測する。従来の回転翼飛行ロボットの姿勢計測には加速度センサとジャイロセンサを用いていたが、それらのセンサは回転翼飛行ロボットに直接搭載しなければならないため、機体の振動によるノイズを含む。よって、回転翼飛行ロボットの外部にあり、かつ姿勢計測を可能とするシステムであるモーションキャプチャを利用する。

Real-time measurements of motion of a quad rotor flying robot by a motion-capture system

Takuya Sasaki¹, Yasushi Honda²

¹ Division of Information and Electronic Engineering, Muroran Institute of Technology

² College of Information and Systems, Muroran Institute of Technology

Abstract

We develop a system which measure the motion of a quad rotor flying robot by a motion-capture system. So far, accelerometer and gyroscope were used for measurements of the motion. However, they include much noise because the quad rotor vibrates the body of flying robot including the sensors. Therefore, as an external device, we use the motion-capture system for measurements of motion.

1 はじめに

ロボットとは人間の代わりに作業をする装置のことを指す [1]。人間の代わりに作業をするためには、外部環境に応じて行動を決定する反応行動をとる能力が必要である。また、外部環境が変化した際の適応行動も必要である。これらの自律行動に対する構成論的研究として飛行ロボットの反応行動に着目する [2][3]。

飛行ロボットの自律行動とは、人間の操作を必要とせずに飛行することを指す。それに必要な環境情報はセンサあるいは感覚器官を用いて取得しなけれ

ばならない。従来の研究では、飛行ロボットの感覚器官として、加速度センサとジャイロセンサを搭載し姿勢を計測していた。

人間の場合は自身の身体に備った感覚器官からの情報に頼らざるを得ない。しかし、ロボットは外部の感覚器官（センサ）で取得した環境情報を無線通信などを通じて自律行動に利用できる可能性がある。そこで、飛行ロボットの姿勢を外部から計測する手段（感覚器官）としてモーションキャプチャを用いる。本研究は、モーションキャプチャを用いて飛行ロボットの姿勢をリアルタイムに計測することを目標として行った。そのシステムを飛行ロボット

の姿勢制御に用いることも視野に入れている。リアルタイム性は、姿勢制御に応用した際に必要となる。

モーションキャプチャは専用のカメラを必要とするので、本研究では室内での姿勢計測および姿勢制御を想定する。本研究での姿勢制御が実現すれば、その飛行データを屋外で飛行する際の教師データとして教師付き学習に用いることができる。

2 回転翼飛行ロボット

本研究で対象とする回転翼飛行ロボット（以下、飛行ロボットと呼ぶ）を図1に示す[3]。実験装置に固定してある。図1のように x 軸, y 軸, z 軸をとり、各軸回りに飛行ロボットが回転した角度をそれぞれ ϕ , θ , ψ とする。また、図1に示すように、4つのモーターをそれぞれ $m1$, $m2$, $m3$, $m4$ とする。現在は x 軸回りの回転運動のみを扱っているため、 $m2$ と $m4$ は取り付けしていない。

飛行ロボットの姿勢計測には、加速度センサおよびジャイロセンサを用いている。加速度センサを1個、ジャイロセンサを6個搭載してある。加速度センサでは角度を計測し、ジャイロセンサでは角速度を計測する。

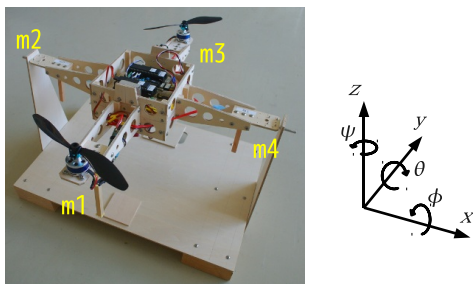


図1: 回転翼飛行ロボット

3 モーションキャプチャによる姿勢計測方法

モーションキャプチャ（以下、MCと呼ぶ）は、カメラから赤外線を照射し、反射された点の空間座標 (x 座標, y 座標, z 座標) を取得するシステムである。仕組みの概略を図2に示す。赤外線の反射には専用のマーカーを用いる。マーカーの表面には、赤外線を反射しやすい塗料が塗られている。MCを動作させるために Cortex というソフトウェアを用いる。

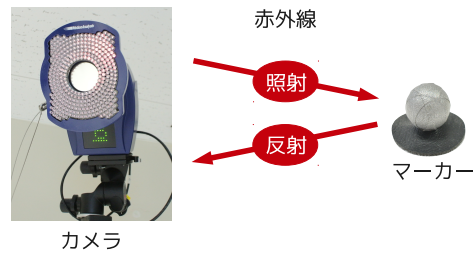


図2: モーションキャプチャの仕組み

MCの座標データによる飛行ロボットの姿勢計算方法を記述する。真上から見た飛行ロボットの写真を図3に示す。飛行ロボットに6つのマーカーを配置し、それぞれに図中の名前を付けた。 $m1 \sim m4$ はモーターの名前に合わせている。 $m1$, $m3$ の座標データをもとに飛行ロボットの傾きを求める。座標データを得るプログラムの開発には、Cortexの機能である Software Developers Kit（以下、SDKと呼ぶ）を用いている。

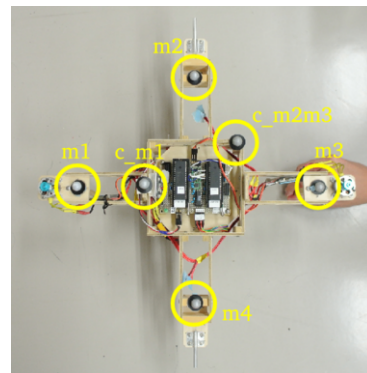


図3: 真上から見た飛行ロボットとマーカーの位置

次に、具体的な姿勢計測方法について記述する。図4は飛行ロボットを横から見た状態である。飛行ロボットの機体を長方形で表している。図4の ϕ を飛行ロボットの傾きと定義する。マーカー $m1$ の座標を (x_1, y_1, z_1) 、マーカー $m3$ の座標を (x_3, y_3, z_3) 、マーカー間の距離の半分の長さを r 、マーカーが地面と平行なときから上昇した高さを h とする。 r , h を (1), (2) 式で求め、それらを (3) 式に代入することで傾き ϕ が求められる。また、 ϕ を時間微分することで角速度が求められるので、角速度 ω_x は (4) 式で求められる。

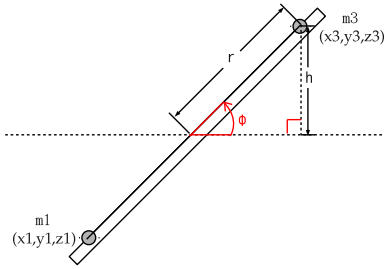


図 4: 真横から見た飛行ロボットの概略

$$r = \frac{\sqrt{(x_1 - x_3)^2 + (y_1 - y_3)^2 + (z_1 - z_3)^2}}{2} \quad (1)$$

$$h = \frac{(z_3 - z_1)}{2} \quad (2)$$

$$\phi = \sin^{-1} \left(\frac{h}{r} \right) \quad (3)$$

$$\omega_x = \phi'(t) \quad (4)$$

以上を踏まえて、MC を用いた飛行ロボットの姿勢計測システムを開発した。開発した姿勢計測システムの外観の概略を図 5 に示す。水色の Cortex-PC は Cortex を動作させる PC である。緑色の SDK-PC は MC を用いて飛行ロボットの姿勢を計測する PC である。2 つの PC は、8 つのカメラと接続するために LAN ケーブルで同一ローカルネットワークに接続している。黄色の Robot-PC は飛行ロボットの操作およびセンサ値の取得を行う PC である。

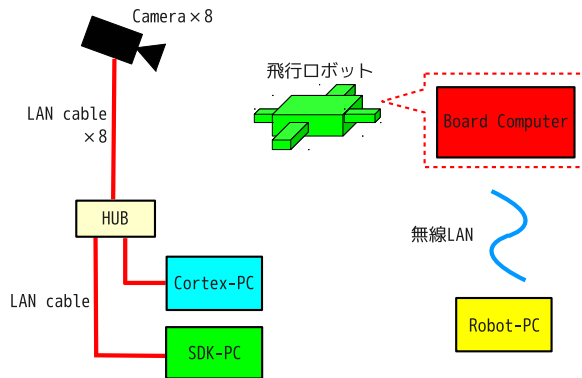


図 5: 姿勢計測システムの概略

4 MC によるリアルタイム姿勢計測

4.1 角度と角速度の計測

MC による飛行ロボットの姿勢計測システムを用いて、飛行ロボットの姿勢計測実験を行った。飛行

ロボットが傾いた角度と、そのときの角速度を飛行ロボットの姿勢データとする。MC によって計測した姿勢データと、飛行ロボットに搭載した各センサによって計測した姿勢データを比較した。センサの姿勢データとの違いとリアルタイムに姿勢計測できているかどうかを確認した。センサの値に含まれるノイズを確認するため、次の 2 通りの方法で飛行ロボットを運動させて実験した。実験方法 1 の結果を 4.2 小節に、実験方法 2 の結果を 4.3 小節にそれぞれ示す。

実験方法 1 手で上下に動かす

実験方法 2 プロペラを回転させ片側に動かす

4.2 MC による姿勢計測の確認

飛行ロボットをモーターを回転させず手で動かしたときに飛行ロボットが傾いた角度を図 6 に、角速度を図 7 にそれぞれ示す。横軸は時間 (s) であり、縦軸は図 6 では角度 ϕ (deg)、図 7 では角速度 ω_x (rad/s) である。図 1 の x 軸回りの角度と角速度をグラフにしている。ジャイロセンサは x 軸回りの角速度を計測している 1 個のみを用いている。4.3 小節の図 8、図 9 でも同様の軸と単位を取る。赤色のグラフは MC で計測した姿勢データ、緑色のグラフは各センサで計測した姿勢データ (図 6 では加速度センサ、図 7 ではジャイロセンサのデータ)、青色のグラフは加速度センサで計測した角度を移動平均化した値を示している。MC で飛行ロボットの姿勢計測ができていると判断できる。

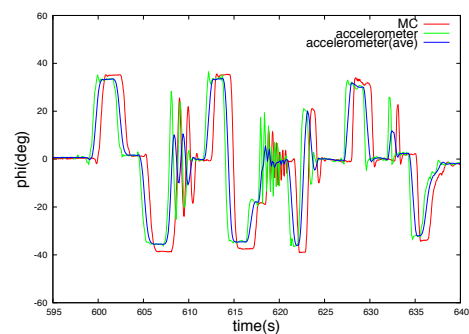


図 6: 手で動かしたときの姿勢データ比較 (角度)
赤: MC による角度, 緑: 加速度センサによる角度, 青: 加速度センサによる角度 (移動平均化)

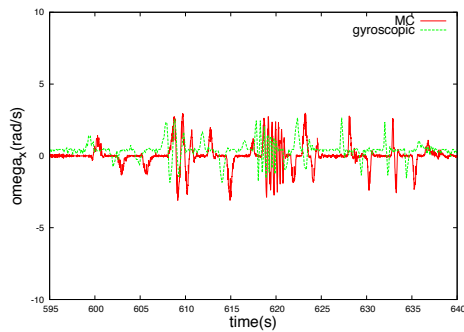


図 7: 手で動かしたときの姿勢データ比較 (角速度)
赤: MC による角速度, 緑: ジャイロセンサによる角速度

4.3 モーターで動かした実験結果

図 8, 図 9 は実験方法 2 のときの姿勢計測であり, 角度を計測した結果である. 図 8 を見ると, 加速度センサの角度 (緑色のグラフ) にノイズが含まれていることがわかる. これは, モーター駆動による飛行ロボットの機体の振動を検出してしまったためであると考えられる. そのため, 飛行ロボットの姿勢制御には移動平均化した値を用いているが, MC は飛行ロボットの外部にあるため, その影響を受けない.

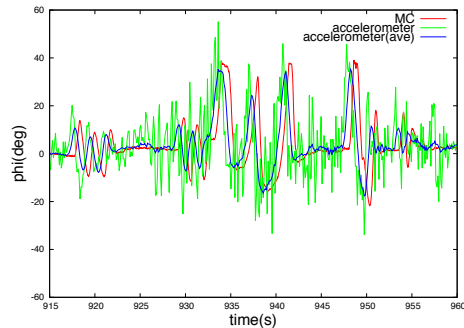


図 8: モーター駆動時の姿勢データ比較 (角度)
赤: MC による角度, 緑: 加速度センサによる角度, 青: 加速度センサによる角度 (移動平均化)

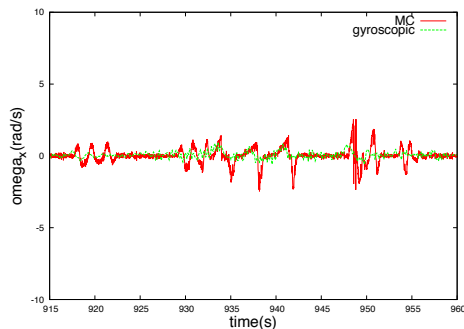


図 9: モーター駆動時の姿勢データ比較 (角速度)
赤: MC による角速度, 緑: ジャイロセンサによる角速度

4.4 MC による姿勢計測の時間遅れ

MC によってリアルタイムに計測した姿勢データを飛行ロボットの姿勢制御に利用することを検討する. 4.2 小節, 4.3 小節の図を見ると, MC による姿勢計測には時間遅れがあることが確認できる. MC による姿勢データは各センサの姿勢データに対して最短約 30 ミリ秒の時間遅れを持つことがわかった. 加速度センサの値はノイズを含むので (図 8 参照) 移動平均化した値を姿勢制御に用いている. この移動平均化処理には約 270 ミリ秒かかることがわかっている [3]. よって, MC の時間遅れを正確に算出したときに約 270 ミリ秒以内であれば姿勢制御に利用できるといえる. しかし, MC による姿勢データの時間遅れにはむらがあり, 時間遅れが 30 ミリ秒を超えるときもある. この原因を調べ, 常に最短の時間遅れで姿勢を計測する手法を開発する必要がある.

5 まとめ

MC を用いた飛行ロボットの姿勢計測システムを開発した. リアルタイムに飛行ロボットが傾いた角度とそのときの角速度を計測することができた. しかし, MC を用いた姿勢計測では, 各センサを用いた姿勢計測に対して最短約 30 ミリ秒の時間遅れが生じることがわかった.

今後の課題として, 本システムを飛行ロボットの姿勢制御に利用する際, MC によって取得した姿勢データの, 各センサによって取得した姿勢データに対する時間遅れを常に最短に維持する必要がある. それを踏まえて, MC でリアルタイムに計測した姿勢データによって飛行ロボットの姿勢を制御するシステムの開発を行いたい.

参考文献

- [1] 浅田 稔, 國吉 康夫, “岩波講座 ロボット学 4 ロボットインテリジェンス”, 岩波書店, (2006).
- [2] 佐々木俊哉, 本田泰, 第 15 回交通流のシミュレーションシンポジウム論文集, 61-64, (2009).
- [3] 橋本理寛, 本田泰, 第 18 回交通流のシミュレーションシンポジウム論文集, 33-36, (2012).