

画像処理を用いたドローンの物体追尾

2017SC032 栗山佳樹

指導教員：大石泰章

1 はじめに

ドローンにカメラを搭載することで、対象物の観察、追跡、調査を行うことができると考えられる。その際には、対象物を認識し、得た情報からドローンを制御することが基本的技術となる。実際に、カメラ搭載のドローンを用いてヘリポートを認識し、ヘリポートに対する機体の相対位置を求めて制御を行い、ヘリポートの中心にドローンを着陸させる研究がされている [1]。

本研究では、カメラ搭載のドローンが撮影した物体の位置と大きさに基づいてドローンを制御し、物体を追尾することを試みる。具体的には、ドローンに搭載されているカメラを用いて画像を撮影し、撮影中の画像に対してリアルタイムに黄色のボールの認識を行う。さらに、認識したボールの位置に基づいてボールが画像の中心に見えるようドローンを移動させる。また、画像中のボールの大きさに基づいてドローンとボールの距離を一定に保つようにする。

2 使用するドローン

本研究では Ryze Technology 社の Tello というドローンを使用する (図 1)。外形寸法は幅 17.5cm、奥行き 17cm、高さ 4cm と小さく、質量もわずか 80g である。Tello では、4つのプロペラのモータの回転速度をそれぞれ独立して制御することができる。本研究では、それぞれのモータの回転速度を図 1 のように $\omega_1, \omega_2, \omega_3, \omega_4$ と書く。モータが機体に与えるトルクが相殺されるように、隣接するモータでは回転速度の正方向を逆向きにとっている。また、機体の正面にカメラを搭載しており (図 2)、640 画素 \times 480 画素の画像を撮影することができる。



図 1 Tello の外観



図 2 Tello 搭載のカメラ

3 システムの概要

本研究で用いるシステムの概要を説明する。Tello 本体にはプログラムを書き込むメモリ領域はないため、Tello のカメラ映像を PC が受け取り、PC 上でプログラミング言語 Python を使って画像処理、データ処理、制御入力の計算を行う。そして、制御入力の計算結果を Tello へ送り Tello を制御する。

4 ドローンの上下、左右方向の制御

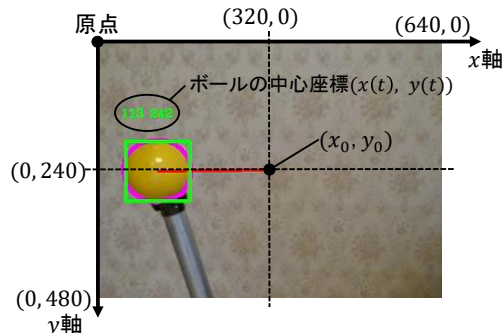


図 3 Tello のカメラ画像

4.1 制御方法

ボールが画像の中心に見えるようにモータ制御する方法を説明する。図 3 のように Tello のカメラが取得した 640 画素 \times 480 画素の画像の左上を原点とし、右向きを正方向として x 軸を、下向きを正方向として y 軸をとる。画像の中心の座標 (x_0, y_0) は $(320, 240)$ である。カメラがボールを認識した場合、ボールを枠で囲いボールの中心座標を測定する。

上下方向の制御は、ボールの中心の y 座標 $y(t)$ と画像の中心の y 座標 y_0 との差にゲイン k_1 を掛ける比例制御を行う。ゲイン k_1 の値は、カメラ画像からボールを見失わない範囲で調節し、0.10 とした。モータに与える回転速度 $\omega_i(t)$ を次の式で定める：

$$\omega_i(t) = u_0 - k_1(y(t) - y_0) \quad (i = 1, 2, 3, 4). \quad (1)$$

ただし、 u_0 は基準の入力であり、値は 150 とする。各モータの入力には 50 から 250 の値を入れることができる。

左右方向の制御は、ボールの中心の x 座標 $x(t)$ と画像の中心の x 座標 x_0 との差にゲイン k_2 を掛ける比例制御を行う。ゲイン k_2 の値は、カメラ画像からボールを見失わない範囲で調節し、0.12 とした。モータに与える回転速度 $\omega_i(t)$ を次の式で定める：

$$\omega_i(t) = u_0 + k_2(x(t) - x_0) \quad (i = 3, 4), \quad (2)$$

$$\omega_i(t) = u_0 - k_2(x(t) - x_0) \quad (i = 1, 2). \quad (3)$$

基準の入力 u_0 に対して、左右のモータで符号の異なる値を加えることでドローンを左右に移動させる。

斜め方向の制御は、式 (1) と式 (2)、または式 (1) と式 (3) を組み合わせた値をモータの入力として与える。すなわち、モータに与える回転速度 $\omega_i(t)$ を次の式で定める：

$$\omega_i(t) = u_0 - k_1(y(t) - y_0) + k_2(x(t) - x_0) \quad (i = 3, 4), \quad (4)$$

$$\omega_i(t) = u_0 - k_1(y(t) - y_0) - k_2(x(t) - x_0) \quad (i = 1, 2). \quad (5)$$

4.2 実験結果

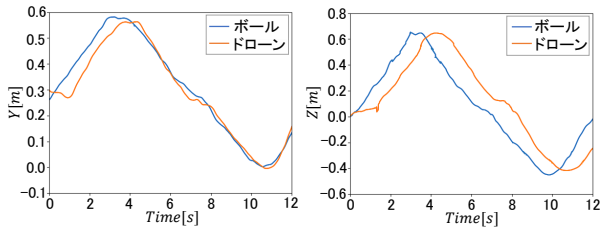


図4 上下方向の制御

図5 左右方向の制御

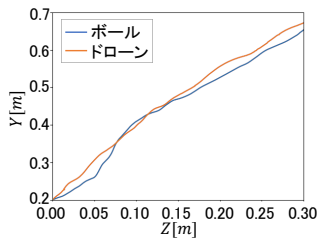


図6 斜め方向の制御

本研究では、Natural Point 社の光学的モーションキャプチャシステム OptiTrack[2] を使用して、3次元空間に基準となる原点を設定し、ドローンとボールの位置座標(前後方向を X 座標, 上下方向を Y 座標, 左右方向を Z 座標とする)を測定した。図4はボールを上下方向に動かしたときのドローンとボールの Y 座標を測定した結果, 図5はボールを左右方向に動かしたときのドローンとボールの Z 座標を測定した結果である。また, 図6はボールを斜め方向に動かしたときのドローンとボールの Y 座標と Z 座標を測定した結果である。図4, 図5, 図6よりボールの動きに追尾してドローンが動いていることが確認できる。上下方向の制御では, ボールの動きに対し, ドローンの動きは1秒を少しきる程度遅れている。それは, ドローンがPCへカメラ画像を送り, PC上で制御入力の計算を行ってその結果をドローンへ送るまでに時間がかかるためである。また, 上下方向に比べて左右方向はドローンの追尾の遅れが大きい。左右方向の制御では, 左右のモータで異なる値を加えることで機体を傾けて移動するためと考えられる。

5 ドローンの前後方向の制御

5.1 制御方法

ドローンとボールの距離を一定に保つようにモータ制御する方法を説明する。カメラがボールを認識した場合, カメラ画像中のボールに含まれる画素数を測定する。前後方向の制御は, 測定した画素数 $p(t)$ と目標とする画素数(10000画素)との差にゲイン k_3 を掛けることで比例制御を行う。カメラ画像中の画素数が10000画素とは実際の

ドローンとボールの距離にして約15cm離れていることを意味する。ゲイン k_3 の値は, ボールの認識が外れない範囲で調節し, 0.003とした。モータに与える回転速度 $\omega_i(t)$ を次の式で定める:

$$\omega_i(t) = u_0 + k_3(p(t) - 10000) \quad (i = 1, 4), \quad (6)$$

$$\omega_i(t) = u_0 - k_3(p(t) - 10000) \quad (i = 2, 3). \quad (7)$$

基準の入力 u_0 に対して, 前後のモータで符号の異なる値を加えることでドローンを前後に移動させる。

5.2 実験結果

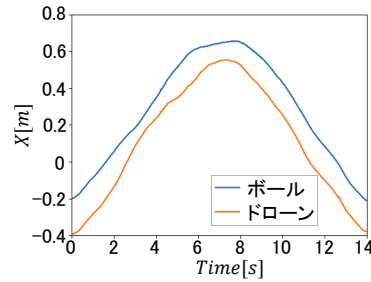


図7 前後方向の制御

図7はボールを前後に動かしたときのドローンとボールの X 座標をモーションキャプチャで測定した結果である。ボールを動かして始めてから約20cmの距離を保って追尾している。ただし, 3[s]~5[s]では, ドローンとボールの距離が9.1cmまで縮まっている。これは, 実験時の光の具合によって黄色のボールの一部が認識されず, カメラ画像中のボールに含まれる画素数が小さくなったことが原因と考えられる。これにより, モータに与える回転速度の値が大きくなりドローンが速く移動してしまった。

6 まとめ

本研究では, ドローンが撮影した画像から黄色のボールを認識し, カメラ画像中のボールの位置に基づいてボールが画像の中心に見えるようにドローンを移動させることで, ドローンの上下, 左右方向の制御を行った。また, カメラ画像中のボールに含まれる画素数に基づいてドローンとボールの距離を一定に保つことで, ドローンの前後方向の制御を行った。モータの回転速度を制御することで, ドローンの旋回の制御も同様に可能であると考えられる。

ドローンとボールの距離を一定に保つ実験は, 光の当たり具合に影響されることがわかった。今後は, 実験時の光に影響されることを防ぐために, 対象物をARマーカにするなどの工夫が必要であると考えられる。

参考文献

- [1] 岡崎豪・Rebull Oriol:『損害調査のための物体検出を用いた無人航空機のリアルタイム制御』, 人工知能学会全国大会論文集, 第32回全国大会(2018)
- [2] OptiTrack: <https://www.optitrack.co.jp/>