

画像処理に基づくドローンの自動操縦

2019SC038 森田大智 2019SC056 杉山航輝

指導教員：大石泰章

1 はじめに

ドローンは遠隔操作や自動操縦により飛行する無人航空機である。近年、ドローンは空中から農薬を散布したり、危険な場所での情報収集を行ったりと、様々な分野で貢献している [1]。ドローンを人間が立ち入るのが困難な場所で使用する場合や、ドローンが操縦者から確認できない場所に行ってしまった場合などのため、ドローン搭載のカメラで目標を見つけ、そこまで飛行するという研究がなされている [2][3]。

本研究では、ドローンに搭載されたカメラで動画を撮影し、これを PC に送信して処理することで、目標地点まで自動的に飛行させることを試みる。具体的には、ドローンのカメラで撮影した動画を PC に送信し、目標となる物体の色をもとづいて検出し、接近する。目標までの距離が一定値以下になったら、ドローンは着陸する。目標の色を検出し、目標へ接近する動作は Python を用いたプログラムで実現する。

2 使用するドローン

本研究では、図 1 に示す Ryze Technology 社の Tello というドローンを使用する。このドローンは長さが 9.2cm、幅が 9.8cm、高さが 4.1cm であり、重量は 81.6g と、小型かつ軽量であるため、無人航空機登録や、ドローン運用の免許の必要がない。このドローンは図 2 のように、機体の正面にカメラを搭載しており、毎秒 30 フレームの速度で撮影することができる。本研究では、このドローン搭載のカメラで撮影した動画を PC に送信し、検出した目標に向かって飛ばす処理を行う。



図 1 ドローン Tello

本研究のドローンには指令を与えることにより、離着陸、前進後退、左右旋回、上下移動をさせることができる。前



図 2 ドローンのカメラ

進後退や上下移動の指令ひとつで移動する距離は変更可能であり、小さくするほどより精密な指令が可能であるが、ドローンは空中で静止しないため、指令ひとつで移動する距離を小さくしすぎても無意味である。左右旋回の指令ひとつで旋回する角度についても同様である。ここでは前進後退や上下移動の指令ひとつで移動する距離は 20cm、左右旋回の指令ひとつで旋回する角度は 30 度と設定した。

3 実現を目指す動作

本研究で実現を目指す動作について説明する。ドローンは当初机の上に静止している。今回使用する目標は青色の A4 判の紙であり、ドローンの当初位置から 1.8m 離れたところにある壁の、机の上面から 1.2m の高さの場所に貼り付けてある (図 3)。動作開始の指示を与えると、ドローンは離陸して 30cm 上昇する。上昇した後ドローンはカメラを使い、目標の青色を認識することで、壁に貼り付けてある目標をとらえる。その後旋回と上下移動を適切に行い、目標がカメラで撮影した画像の中心に写るようにした後、目標に向かって前進する。最後に目標までの距離が 30cm 以下になったら着陸する。この動作の実現をめざして、研究を行った。

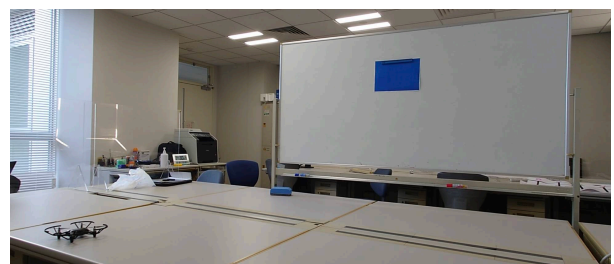


図 3 全体図

4 システムの概要

4.1 システムの全体構造

本研究で用いるシステムは図4のようにPCとドローンから構成される。PCとドローンはWi-Fiで接続されている。ドローンはカメラで動画を撮影し、これをPCに送信する。PCは言語Python上で画像処理ソフトウェアOpenCVを用い、送信された動画を処理してドローンに指令を送る。指令を受け取ったドローンはこれに従って動作する。

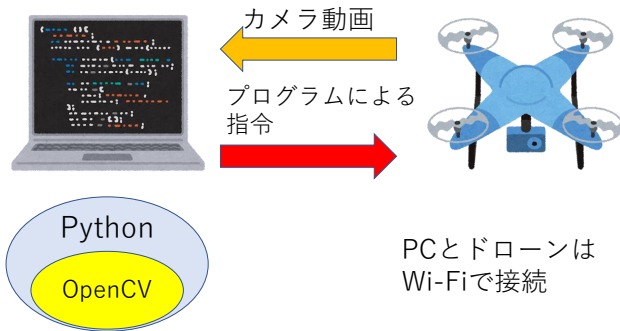


図4 システムの全体構造

4.2 OpenCV

Pythonを用いて画像処理を行う際、OpenCVとよばれるソフトウェアを使用する。OpenCVとは、米インテル社が開発したオープンソースソフトウェアであり、コンピュータで静止画や動画を処理し、色変換をはじめ、物体認識や物体追跡などを行うことができる。本研究では、動画の処理および、目標の色を検知する動作をこのOpenCVを用いて行っている[4]。

4.3 使用するPC

本研究で用いたPCは、CPUとしてCorei5を、メモリとして16GBを搭載する。

5 目標の検出

5.1 HSV色空間

HSV色空間とは、「色相」「彩度」「明度」の3つの基準により色を分類する方法である。色相とは、赤、緑、青といった様々な色を0度から360度の範囲で定義する要素である。色相の値により定められる色は図5のようになる。彩度とは、色相で定義した色の鮮やかさを0%から100%の範囲で表すもので、値が低くなる程白くなっていく。明度とは、色相で定義した色の明るさを0%から100%の範囲で表すもので、値が低くなる程暗くなっていく。OpenCVは本来RGB空間を用いるが、より特定の色を検出しやすいようここではHSV空間に変換している[5]。これにより、色相に注目して、特定の色の注目を行っている。

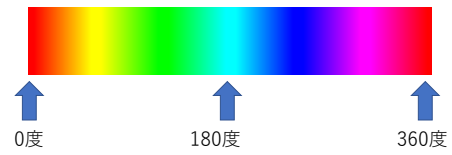


図5 HSV色空間における色相

HSV色空間を使い、指定した色を抽出するマスクを作成する。ここでは青色を抽出するため、OpenCVを使い、色相が90から150、彩度が64から255、明度が0から255の範囲を指定する。これを用いることで、図6(a)のようなドローンの目標である紙の画像から図6(b)のように青色のみを抽出することができた。

5.2 ラベリングによる目標の中心位置と面積の測定

ラベリングとは、2値化した画像において、同じ色が連続しているものごとに番号を割り当てる処理のことである。ラベリングを行うことで、図7(a)のように壁にはりつけた各々の図形の位置、高さ、幅、面積を図7(b)のように求めることができる。本研究では、この機能を用いることで、ドローンのカメラで検出した物体の中心位置や、面積を求めている。本研究では面積最大の物体を目標とするため、この場合の目標は四角となる。

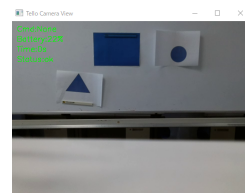


(a) 元の画像



(b) 青色を抽出した画像

図6 青色の抽出



(a) 元の画像



(b) ラベリング処理後

図7 ラベリング処理

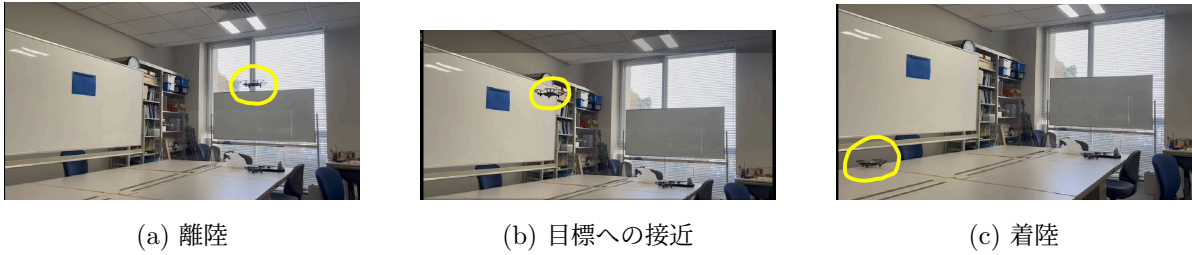


図8 実現したドローンの動作

6 目標への接近の実現

目標への接近は以下のように実現する。まずドローンに動作開始の指示を与える。これによりドローンは離陸して30cm上昇する。その後、ドローンからPCに送信された動画から青色を抽出し、ラベリング処理を行う。抽出した青色の領域は複数ありうるが、今回は面積が最大のものを目標とし、目標の領域の中心座標が画像の中心に位置するように移動を行う。具体的には、中心座標が画像中心の左側に位置していた際、ドローンは30度左に旋回する。それでも足りなかった際はさらに30度旋回する。右側に位置していた際も同様である。画像は縦が360画素、横が480画素であるが、目標の中心の横座標と画像中心の横座標との差が100画素以下になったら旋回を停止する。次に目標の中心が画像中心の上側に位置していた際、20cm上昇する。それでも足りなければさらに20cm上昇する。下降も同様に行い、目標の中心の縦座標と画像中心の縦座標との差が50画素以下になったら上下の移動を停止する。

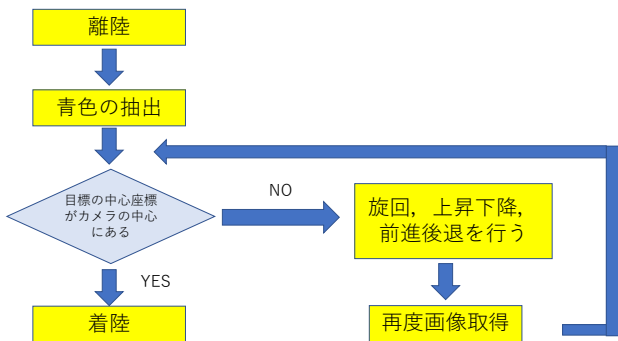


図9 目標への接近

こうして目標の領域の中心座標が画像の中心にくるようにした後、目標に向かって前進する。1回の移動で20cm前進し、たりなければ20cm追加して前進する。目標とする領域の面積が画素を単位として100000以上になった際、前進を停止し、着陸指示を与える。これらのアルゴリズムを以下の図9に、目標に接近する動作を図10にまとめる。

以上の方針のもとについて実験した結果、図8(a)のように離陸した後、図8(b)のように目標へ接近し、図8(c)の

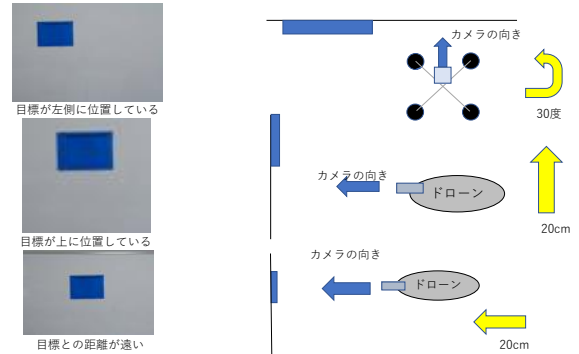


図10 目標への接近

ように着陸した。図では、ドローンの位置がわかりやすいように、黄色の丸で囲んでいる。このように、目指した動作が実現した。

7 PCからドローンへの指令の送信

本研究では、キー入力を用いて、離陸の指令を行い、ドローンの動作を開始する。このほかにも、ドローンが予期せぬ動作をした際には、キー入力を用いて緊急停止を行っている。キー入力で与えた指令をPCからドローンに送信し、所望の動作をさせる方法について説明する。この制御は、OpenCVモジュールの機能であるwaitkeyを用いて行う。waitkeyは画像を表示するウィンドウにおいて、キーボード入力を待ち受ける関数である。これを用いることで、PCで受信したドローンからの動画をリアルタイムで表示しながら、キー入力でドローンの操作を行うことができる。

7.1 キー入力と対応する動作

キー入力で与える指令と、対応するドローンの動作との関係を表1に示す。例えば、Wキーを入力するとドローンは20cm前進する。SキーからLキーまでについても、同様に1回のキー入力ですべて20cmずつ後退や左移動など対応する動作を行う。Uキーを入力すると90度左旋回する。Iキーの右旋回も同様である。Nキーを入力すると低速モードになり、最大傾斜角度が9度、最大速度は14.4km/hとなる。一方、Mキーを入力すると高速モードになり、最大傾斜角度を25度、最大速度を28.8km/hま

であげることが可能である。J キーを入力すると離陸し、K キーを入力すると着陸する。Q キーを入力すると通信が強制終了され、ドローンは緊急停止する。これは予期せぬエラーが発生した際の対策となる。

表 1 キーの割り当て

キー入力	対応するドローンの動作
W キー	前進
S キー	後退
A キー	左に移動
D キー	右に移動
H キー	上昇
L キー	下降
U キー	左旋回
I キー	右旋回
N キー	低速モード
M キー	高速モード
J キー	離陸
K キー	着陸
Q キー	緊急停止

- [4] 北山直洋：Python で始める OpenCV4 プログラミング，カットシステム，東京，2019
- [5] CRAFT GoGo, 【Python+OpenCV】 特定の色を検出するプログラム，<https://craft-gogo.com/python-opencv-color-detection/>, (参照 2023-01-26)

8 まとめ

本研究では、ドローンのカメラで撮影した動画を PC に送信し、目標となる物体を色にもとづいて検出し、接近する動作をドローンに行わせた。目標とする動作は完成したが、ドローンが空中で静止しないため、安定性は良いとはいえ、その影響で前進であれば 20cm のように設定した単位通り正確に動いてはいなかった。実際に行った結果、ドローンから PC に送られてくる動画にほとんど遅延はなく、PC 上で安定して動作していた。また、スマートフォンの専用アプリでドローンを操縦する際よりも、バッテリーの減りが速かったため、PC を用いた自動操縦はスマートフォンを用いた手動操縦と比べ、飛行可能な時間は短くなるのではないかと考える。

本研究ではひとつの目標に向かって接近し、着陸する動作を行わせた。これを応用し、目標物体を複数用意して、順番に目標へ接近していく動作ができると考えられる。

参考文献

- [1] 野波健蔵：世界のドローン開発動向と農業応用からみた課題と展望，計測と制御，Vol. 55, No. 9, pp. 780–787, 2016.
- [2] 江村優吾，小嶋和徳，伊藤慶明，馬淵浩司：製油所におけるドローン空撮画像からの安全領域判定手法の比較，情報科学技術フォーラム講演論文集，Vol. 19, No. 3, pp. 123–124, 2020.
- [3] 菊池慶仁，阿部太智：AR マーカーに基づくドローンの自律飛行 (第 2 報)，工学研究：北海学園大学大学院工学研究科紀要，Vol. 19, pp. 17–22, 2019