

動画内の物体検出によるクワドロータのビジュアルフィードバック制御

邵 露 (山口東京理科大学大学院), 永田 寅臣 (山口東京理科大学大学院)

渡辺 桂吾 (岡山大学大学院)

Visual Feedback Control of Quadrotor by Object Detection in Movies

Lu SHAO (Sanyo-Onoda City University), Fusaomi NAGATA (Sanyo-Onoda City University)

Keigo WATANABE (Okayama University)

Abstract : In the previous report, we developed a quadrotor mission planning function using a smart phone and an autopilot function using GPS signals. In this paper, we construct a system that can automatically recognize an object existing in images or movies measured by a quadrotor with a surveillance camera and utilize it for feedback control in order to track the object without using GPS signals. The visual feedback control is realized by automatic recognition of the object based on image processing technology, in which the center of gravity position in the image coordinate system can be obtained from color and shape information. A visual feedback application for flight control of DJI's Quadrotor Matrice 100, and image analysis software are developed using Xcode and MATLAB, respectively. The validity and effectiveness are evaluated by experiments.

1. 緒言

画像処理技術の発展に伴い、民生用カメラやスマートフォン (SP) を用いた様々なアプリケーションが開発・利用されている。例えば、監視カメラにおける人物検出や自動車の自動運転が挙げられる。運転の支援や自動化を実現するためには、自動車周辺の道路環境を認識する必要がある。そのための主要なセンサの1つとしてカメラによる画像認識が用いられている。また、外科医療の分野では一昔前のように切開手術をして病巣を切除するのではなく、可能な限り内視鏡による手術が行われている。その基本システムとなるのが胃カメラや大腸カメラなど消化管の診断を目的として開発されてきた内視鏡である。現在医者が直接行っている病巣の確認、すなわち画像に映っている物体が何かを識別する処理は人間が持つ視覚機能による認識処理であり、これを計算機上で実現することにより、様々な応用が可能となる。芹沢らは、飛行ロボットに搭載したステレオカメラを操縦者の頭部の動きに伴い、回転・並進移動する映像システムを開発した [1]。大塚らは、基準マーカを地面に設置し、UAV を移動させ、基準マーカがカメラの視野領域にあるうちに新たなマーカを落とし、2つのマーカの相対位置を測ることで UAV の基準マーカからの位置を推定できるようにした [2]。斉藤らは、地上に複数の二次元マーカを配置し、UAV 搭載カメラからそれらを撮影することによって UAV の位置推定を行うとともに、同一画像内に写る地上の人物の位置推定を行うことで絶対座標系における人物の移動軌跡の観測を可能とした [3]。魯らは iPhone など iOS 端末を使って操作者の周りを旋回するラウンディングと複数の飛行タスクを実行できるミッションプランニングの機能について検討し、クワドロータ (QR)



Fig. 1: Quadrotor Matrice 100 used in experiments.

に実装した [4, 5]。

筆者らはこれまでに、スマートフォンを用いたクワドロータのミッションプランニング機能と、GPS 信号を使ったオートパイロット機能を開発した [6]。本研究では、GPS 信号を用いることなく、クワドロータが監視カメラを使って観測する環境内に存在する対象物を自動認識し、自動追従機能などのフィードバック制御に利用できるシステムの構築を行う。動画内の対象物の自動認識にはカラー情報と形状情報を用い、その重心位置からフレーム内での対象物の座標を獲得できるようにし、ビジュアルフィードバック制御の実現を試みる。実験では図 1 に示す DJI 社のクワドロータ Matrice 100 を用い、飛行制御と画像解析のソフトウェアはそれぞれ、Xcode 及び MATLAB を使用して開発した。

2. 実験システム

まず、図 2 のように iOS 上で利用できる開発環境である Xcode と、DJI 社が提供する Mobile SDK を用いて、スマートフォンを使ってホバリング、目標ロック制御、距離制御の機能を実行できるモバイルアプリケーションを開発した。これは、クワドロータに搭載された Windows PC が DJI Onboard SDK の Onboard API 機能を使用しながら飛行動作を遠隔監視制御し、内蔵のスマートナビゲーションモード (Intelligent Navigation Mode) を使用して事前に設定した経路に沿った自律的な飛行をできるように設計した「スマートフォンから直接実行できるユーザインタフェースの開発」である。開発言語は Swift であり、以降、スマートフォンアプリと呼ぶ。スマートフォンの操作時、スマートフォンからの指令はシリアル通信を介して専用コントローラである C1 コントローラへ送られる。C1 コントローラは無線 (922.7-927.7 MHz, 出力電力: 9 W, 通信距離: 2 km) により、クワドロータに内蔵された Windows PC へと指令を送信する。一方、クワドロータに搭載されている GPS センサで得られる位置情報や、超音波センサで計測される高度情報、障害物との距離情報 (有効距離: 4 m 以内) はスマートフォンにフィードバックできる。

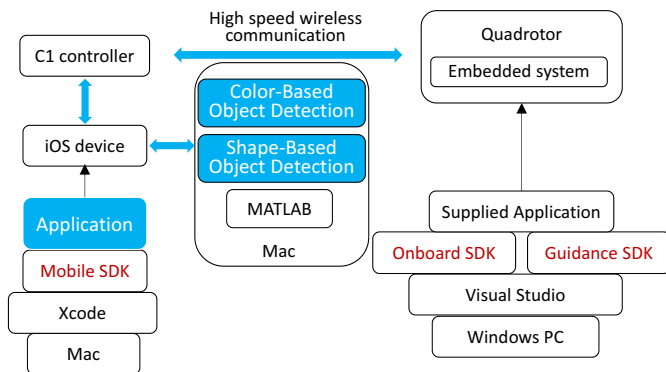


Fig. 2: Software development environment based on three SDKs provided by DJI and using MATLAB for video and image processing.

3. MATLAB 上で開発した画像処理機能

MATLAB は高級プログラミング言語であり、アルゴリズム開発、データ解析、可視化、数値計算のインタラクティブな環境を提供しており、静止画と動画の処理や解析に応用できる。本章では、MATLAB 上で開発した画像処理の機能について述べる。クワドロータで撮影された画像や動画は解析用の Mac に送信され、MATLAB 上で開発

した画像処理ソフトによりカラーや形状をもとに検出したオブジェクトの重心位置を算出できるようにした。この重心位置を参照することでクワドロータのビジュアルフィードバック制御が可能になる。

3.1 カラーで画像内のオブジェクトの重心位置を抽出する機能

3.1.1 画像内のオブジェクトの連結処理

図 3 には、クワドロータから Mac に送られてくる画像や動画内のオブジェクトを抽出するために MATLAB システムで開発したユーザフレンドリな解析ツールを示す。このツールによりカラーあるいは形状をもとにオブジェクトの位置を抽出する。本節では、赤緑青白あるいは黒のカラー指定によるターゲットイメージ内のオブジェクトの重心座標 (x, y) の求め方について述べる。RGB 画像から赤緑あるいは青のカラーで指定されたオブジェクトを検出するために、まず、RGB 画像から R,G,B それぞれ 256 階調のチャンネルを抽出する。このチャンネルは 256 階調のグレースケール画像となる。次に下位側と上位側の閾値 Th_1 と Th_2 を設定しておく。ただし、 $0 < Th_1 < Th_2 < 1$ とする。

例えば、R のオブジェクト検索が指定された場合、 Th_2 をもとに二値化された R のチャンネルを RB とし、 Th_1 の値をもとに二値化された G と B のチャンネルをそれぞれ GB, BB する。次に、RB のチャンネルで 1 を持ち、かつ、GB と BB のチャンネルでともに 0 を持つピクセルを 1 とし、それ以外のピクセルを 0 とした新たな二値画像 BW を生成する。この BW 内において図 4 のような 4 連結あるいは 8 連結で指定する二次元の連結性をもとにオブジェクトを生成させ、最も大きい面積を持つオブジェクトの重心位置を求めるようにしている。今回の実験では 8 連結を用いる。G あるいは B のオブジェクトが指定された場合も同様の処理によりオブジェクトを生成し、重心位置を出力できるようにした。

なお、白である W のオブジェクトが指定された場合は RB, GB, BB の全てにおいて 1 の値を持つピクセルを 1 に、また、黒である Bk のオブジェクトが指定された場合は RB, GB, BB の全てにおいて 0 の値を持つピクセルを 1 に設定することで二値画像 BW を生成できるようにした。この BW に対しても 4 連結や 8 連結で指定する二次元の連結性をもとにオブジェクトの候補を生成させ、最も大きい面積を持つものを最終的なオブジェクトとしてその重心位置 (x, y) を求めることで、同様に画像処理からの出力としている。

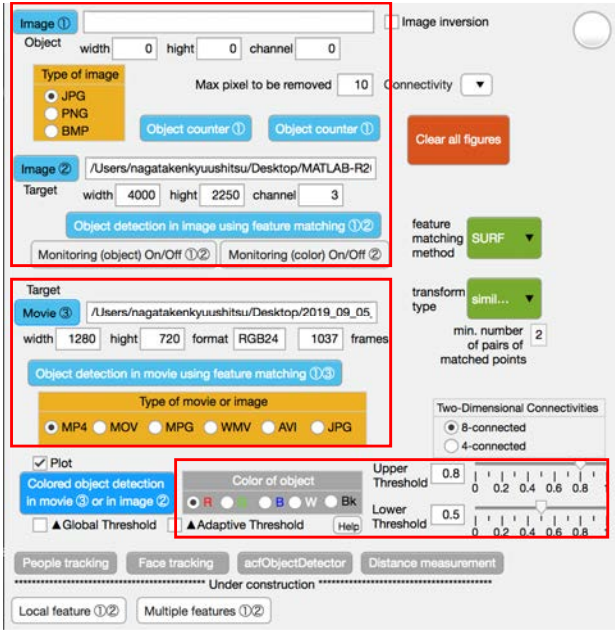


Fig. 3: Developed application for object detection in images and movies.

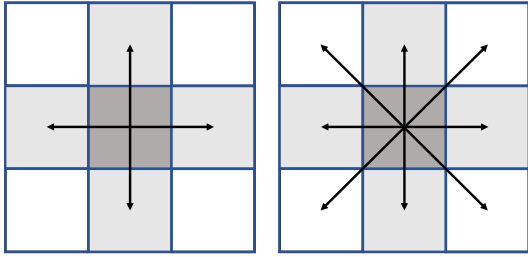


Fig. 4: Two-dimensional connectivities named 4-connected and 8-connected, respectively.

3.1.2 オブジェクトの重心座標の計算

前節で構成された二値画像 BW 内においてオブジェクトの領域のピクセルは 1 の値を持ち、それ以外の領域のピクセルは 0 の値を持つ。この二値画像 BW に対するオブジェクトの重心位置 (G_x, G_y) は次式から求めることができる。

$$G_x = \frac{\sum_{x=1}^X \sum_{y=1}^Y x f(x, y)}{S} \quad (1)$$

$$G_y = \frac{\sum_{x=1}^X \sum_{y=1}^Y y f(x, y)}{S} \quad (2)$$

ここで、 (X, Y) は画像のフルサイズであり、 (x, y) は画像内の座標変数である。また、 $f(x, y)$ は座標 (x, y) のバイナリ値であり、1 あるいは 0 の値を持つ。さらに、 S はオブジェクトの面積であり、オブジェクトを構成する 1 の値を持つピクセルの総数である。クワドロータはカラー指定やテンプレート画像の指定によりターゲットの画像内から検出されたオブジェクトの重心位置の変化をモニターする

ことでビジュアルフィードバック制御を行うことが可能となる。

3.1.3 静止画による実験

開発したスマートフォン (SP) アプリ内のカメラコントロールボタンで「QR が一定間隔で複数の画像を撮影できるモード」を選択している場合、図 5 のように QR から送信された写真のリストが Mac 上にリアルタイムでプレビューできる。実験では SP 上に 2 秒毎に自動撮影の画像を表示させた。QR に接続されたカメラは、最大 4000×3000 ピクセルの解像度の写真を撮影でき、QR のカメラのメモリカードに自動的に保存される。今回の実験では 4000×2250 のサイズの画像を用いた。表 1 には、連続する 8 枚のフレーム内でそれぞれ抽出された重心位置を示している。

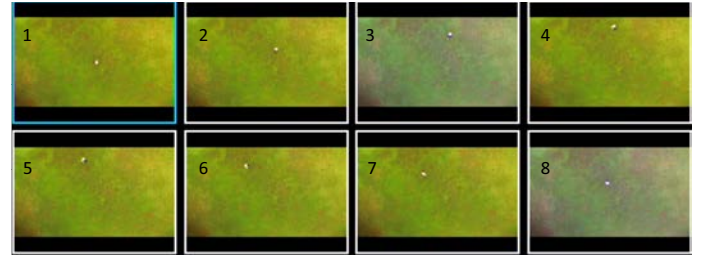


Fig. 5: Preview function of continuous images obtained by the quadrotor every 2 seconds.

Table 1: Variation of COG position obtained using Eqs. (1) and (2).

x	y
1830.0	1337.0
2259.5	799.3
2330.1	388.3
2136.0	215.4
1754.4	285.2
1446.5	418.0
1604.5	647.5
2249.7	1084.0

3.1.4 動画による実験

実験で使用したカメラの場合、24 フレーム/秒のレートで解像度 4000×2250 の動画を撮影できる。図 6 には動画のフレームを分析することにより得られた赤いカラーを持つオブジェクトの重心位置の変化を示している。実験結果の (c) と (d) のように Th_1 と Th_2 をそれぞれ 0.95 付近に調整すると赤いカラーのオブジェクトがより良好に検出さ

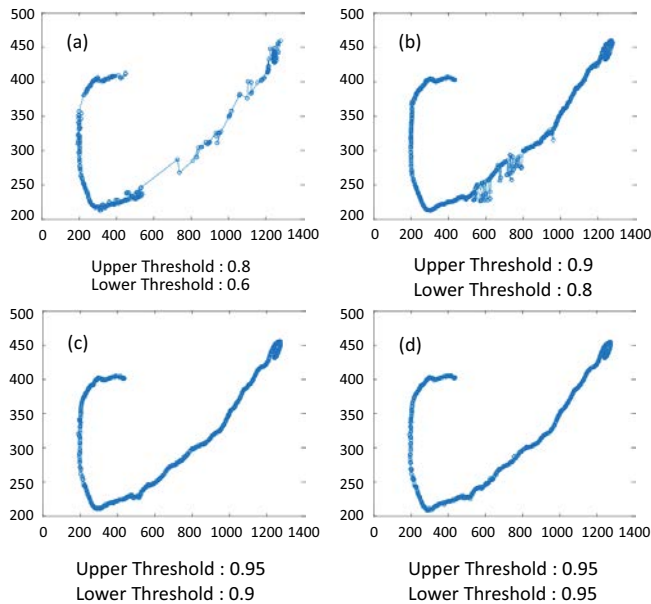


Fig. 6: Variation of COG position obtained by using the proposed color-based object detection.

れたことが確認できた．このように下位側と上位側の閾値である Th_1 と Th_2 を変化させることによる効果を確認できた．

3.2 形状でオブジェクトの位置を抽出する機能

一方，動画内に含まれるオブジェクトの形状をもとにその動きを検出するために，まず，DJI が提供する Mobile SDK に含まれるターゲットトラッキング機能を用いた．図 7 には Swift を使ったプログラムコードの例を示しており，QR は矩形形状の追跡ターゲットを検索し，認識後にロックし，設定された目標高度を保ちながら自動的に追従が開始される．例として，図 8 にはターゲットトラッキング機能により自動制御された QR の飛行軌跡を示す．この SDK の問題点として矩形や円形以外の任意の形状の認識はできないことがわかった．DJI が提供する SDK のこ

```
func initWithNib() -> TargetTrackingConfigView {
    let mainView: UIView =
        NSBundle.mainBundle().loadNibNamed("TargetTracking
        ConfigView", owner: self, options: nil)[0] as! UIView
    self.frame = mainView.bounds
    self.addSubview(mainView)
    self.Search=Searchtarget
    self.target = locktarget(Shape:rectangle)
    self.distance = 50 %Unit: cm
    self.tack = true
    self.startPoint = DJITargetTracking.North
    self.heading =
        DJITargetTrackingHeading.AlongCircleLookingForward
    return self
}
```

Fig. 7: Example of Swift program code for Xcode.

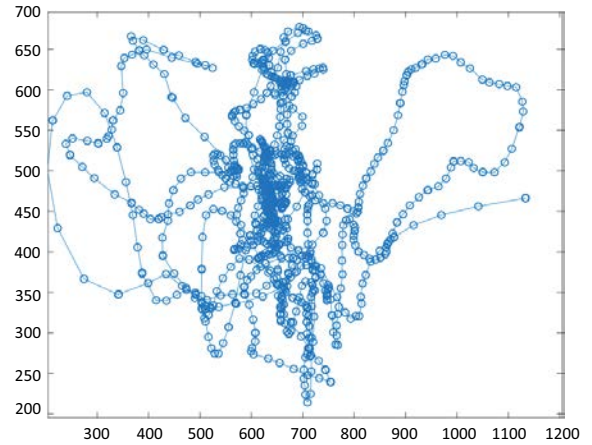


Fig. 8: Variation of object's position in case of being controlled by DJI's SDK, in which only rectangle and circle shapes are available.

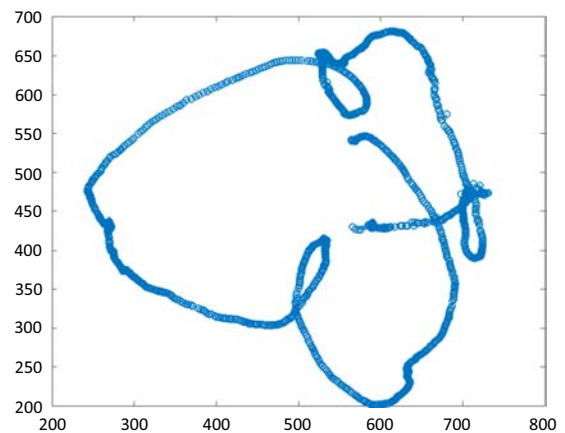


Fig. 9: Variation of COG position obtained by using the proposed shape-based object detection developed on MATLAB.

のような問題点を解決するために，MATLAB 上でテンプレートで指定した任意形状のオブジェクトの自動追従機能を開発した．

テンプレート画像の指定によるターゲットイメージ内のオブジェクトの検出方法については，テンプレート画像をターゲット画像内の左上から右下まで 1 ピクセルずつスタースキャンさせながら，テンプレート真下の領域との相関係数を基に最もマッチングの高い領域を抽出し，その重心位置を求め，出力するようにしている．動画は複数のフレーム（静止画像）に分割できるので，前節で述べた静止画像の処理をそのまま適用できる．図 9 には動画のフレームを解析することにより検出されたオブジェクトの重心位置の変化を示す．この実験から提案手法の有効性が確認された．また，図 10 にはターゲットトラッキング機能の実験風景を示す．なお，安全対策として突然オブジェクトを見失った場合は，ホバリング機能にモードが移るようにしている．



Fig. 10: Experimental scene of shape-based target tracking function.

4. 結言

前報では，スマートフォンを用いたクワドロータのミッションプランニング機能と，GPS 信号を使ったオートパイロット機能を開発した [6]．本研究では，GPS 信号を用いることなく，クワドロータが監視カメラを使って観測する環境内に存在する対象物を自動認識し，自動追従機能などのフィードバック制御に利用できるシステムの構築を行った．動画内の対象物の自動認識にはカラー情報と形状情報を用い，その重心位置からフレーム内での対象物の座標を獲得できるようにし，ビジュアルフィードバック制御が可能になった．

参考文献

- [1] K. Serizawa, R. LADIG, K. Shimonomura, “Movable camera and stereoscopic image system for teleoperation of aerial robot working in high place,” *Procs. of the 2019 JSME Conference on Robotics and Mechatronics*, 1P2-N08, 3 pages, 2019 .
- [2] 大塚光, 谷島諒丞, 永谷圭司, 久保大輔, “投下型 AR マーカを用いた小型マルチロータ機の自己位置推定”, 2A2-G02, ロボティクス メカトロニクス講演会講演概要集, 2015.
- [3] H. Saito, T. Takubo, A. Ueno, K. Cai, R. Miyamoto, S. Hara, “Recognizing own UAV Position and Measuring position of a person using camera image,” *Procs. of the 2018 JSME Conference on Robotics and Mechatronics*, 1P1-C08, 3 pages, 2018 .
- [4] Z. Lu, F. Nagata, K. Watanabe, M.K. Habib, “iOS Application for Quadrotor UAV Remote Control – Implementation of Basic Functions with iPhone –,” *Artificial Life and Robotics*, Vol. 22, No. 3, pp. 374–379, Springer, 2017.
- [5] Z. Lu, F. Nagata, K. Watanabe, “Mission Planning of iOS Application for a Quadrotor UAV,” *Artificial Life and Robotics*, Vol. 23, No. 3, pp. 428–433, 2018.
- [6] 邵 露, 永田寅臣, 渡辺桂吾, “iOS と Android に対応したクワドローターのリモートコントロールアプリケーション,” ロボティクス・メカトロニクス講演会 2019 講演論文集, 1P1-N06, 2 pages, 2019 .