

# 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

## Real-time control of unmanned aerial vehicle by object detection for loss assessment

岡崎 豪<sup>\*1</sup>

Takeshi Okazaki

レブル オリオル<sup>\*1</sup>

Oriol Gaspa Rebull

<sup>\*1</sup> エーオンベンフィールドジャパン株式会社  
Aon Benfield Japan Ltd.

During a loss assessment after a disaster, a drone is useful to take the photos of the building roof. However, there is a possibility of touching with obstacles and persons when using drone. Therefore, in this study, we developed the system that can control a drone in real-time by the object detection called YOLOv2. By outdoor test flights, we confirmed that the system can identify the person's position and control the drone according to the position.

### 1. はじめに

損害保険会社や共済団体(以下、まとめて損保業界と呼ぶ)は災害が発生した際、適切な保険金を支払うための調査(以下、損害調査と呼ぶ)を行っている。台風を例とした場合、損害調査の手法は大きく2つに分けられる。1つは修理業者が作成した見積書を鑑定人が精査し、保険金を決定する手法である。この手法は業者の見積書に基づくことから、実際の修理費用と支払保険金が近い金額となる利点がある。一方、大災害時には一度に多くの家屋で被害が生じるため、見積書の作成までに多くの日数を要してしまう問題がある。もう1つの手法は、鑑定人などが現地を訪問し、家屋の被害箇所の面積などから支払保険金を積算する方法である。この手法は修理業者の見積書を待つ必要がないため、早期に支払いができる利点がある。一方、実際の修理費用と支払保険金に差異が生じてしまう問題がある。どちらの手法も一長一短であり、会社によって採用する手法は異なるが、災害の規模が大きくなるほど、後者の手法の必要性が高まる。そして、後者の手法で屋根の被害を確認する際、高所に登ることは危険を伴うため、無人飛行機(以下、ドローンと呼ぶ)による撮影が有効となる。

2015年から国内の大手損保会社の間で、ドローンを活用する動きが広がっている。ただし、現状は主に企業向けの物件調査や人が近づけない場所の調査などに活用されている。米国の損保業界では、FAA(連邦航空局)による運用規則の策定が遅れたため、規則が発表された2016年以降に各社での導入が進んだ。テキサス州で2番目に大きい損害保険会社であるAllstateは、米国でも特にドローンの活用に積極的であり、2016年のハリケーン・ハービーの直後、ドローンを週に数千回飛ばしている。米国の住宅の屋根は大きく、屋根の被害状況の確認に時間を要していたが、ドローンの活用により作業時間を大幅に短縮している。Farmers Insuranceは、従来1日に3件の損害調査しかできなかったが、ドローンの活用により1時間に3件の損害調査ができるようになったと述べている[Slate 17]。

国内の住宅物件向けの調査にドローンを活用する場合、技術的な難易度は高くなる。基本的にはマニュアル操縦ではなく、自動操縦システムを使用するべきであるが、現在の一般的なシステムはGPSの位置情報に基づいて決められたルートを移動するものであり、準自律飛行と呼ぶべきものである。そのため、

障害物が多い空間では、それらに接触する危険性がある。ドローンで屋根を鮮明に撮影するためには、ドローンを建物に近づける必要があるが、都市部は階数の異なる建物が密集している。都市部以外は建物の密集度が低くなるものの、電柱や電線、防風林などが存在する。これら建物の階数の違いによる高低差や障害物はGoogle Mapsなどの地図アプリでは正確に判別できない。また、災害後に飛ばすため、住民の不在中などにより、ドローンを飛行させることの事前周知が十分にできないことがあり、ドローンが第三者の上空を飛行してしまう状況が起こりうる。これらの課題に対して、ドローンの映像をリアルタイムに解析し、障害物や人などの物体を検出できれば、障害物を避けるなどの適切な飛行が可能となり、危害の発生が低減すると考える。

ドローンと物体検出を組み合わせ、自律飛行させた先行事例として、Leeらはドローンのカメラによる映像からリアルタイムで物体を検出し、検出結果に基づいて機体を制御した研究を報告している[Lee 17]。物体検出の手法にはFaster R-CNNやYOLOなどを用いている。ただし、同研究はAR.Drone 2.0を用いた室内実験であり、基礎研究の位置付けとなる。

本研究では、国土交通省がドローンの飛行許可・承認申請において「資料の一部を省略することが出来る無人航空機」と認定された機体を選択し、実際の損害調査を前提としたシステムを構築した。物体検出の手法にはYOLOv2、ドローンのカメラはGoPro Hero 4、ドローンの制御にはDronekitを選択した。

以下、本システムの構成と屋外でのテスト飛行の結果を示す。また、実際の損害調査でドローンを活用するにあたっての課題を示す。

## 2. 物体検出を用いたリアルタイム制御

### 2.1 ドローンの制御方法

ドローンをリアルタイムに自律制御するためには、ドローンのカメラから得られた映像を解析し、その解析結果に基づいてドローンをコントロールする必要がある。映像データを解析する方法は、大きく3つに分けられる。1つ目はドローンの機体に搭載された演算装置で画像を解析する方法である。この場合、外部に映像データを送信する必要がないため、システムは機体のみで完結し、実務的に扱いやすいものとなる。しかし、現在のドローンに搭載されたCPUやメモリのスペックは低く、ディープラーニングなどの負荷の大きい計算は処理できない。2つ目は携帯電話の電波を通じて、クラウドサーバーに映像データを送り、クラウドサーバー側で解析する方法である。しかし、携帯電話の

連絡先: 岡崎 豪, takeshi.okazaki@aonbenfield.com, エーオンベンフィールドジャパン株式会社, 03-4589-4237, 東京都千代田区永田町 2-10-3 キャピトルタワー11階

電波はドローンに搭載して使用することが原則として禁じられている。2016年7月に実用化試験局という扱いのみ、試験的に使用できる制度が整ったものの、現時点では限られた事業者のみが取り組める状況である。3つ目は機体から映像データを wi-fi で Laptop に送り、Laptop 側で画像を解析する方法である。この手法は規制上の問題もなく、インターネット網を介さないため、映像データの送信に伴う遅延が小さい利点がある。以上の3つの方法を比較し、本研究ではこの3番目の方法を選択した。図1に映像データと制御の全体図を示す。また、選択したドローン、開発環境、Laptopなどを次節以下に示す。



図1 リアルタイム制御の全体構成

## 2.2 物体検出手法

物体検出には、物体の位置を検出する手法、物体の形状を検出する手法の2通りがある。前者は画像内で検出された矩形の物体領域ごとにラベルを分類する方法であり、後者は画像内の pixel ごとにラベルを分類する方法である。pixel 単位での検出は物体の形状まで把握できるが、計算負荷が大きい。ドローンが飛行し、障害物を回避するには、pixel 単位ではなく、物体領域が判別できれば十分であるため、物体位置の検出を選択した。

ディープラーニングによる物体検出は活発な研究分野であり、毎年のように新しい手法が提案されている。表1に Redmon が最新の手法を比較した結果を示す [Redmon 17]。同表の mean Average Precision (mAP) はクラス毎の物体検出の精度の平均であり、Frame Per Second (FPS) は1秒間に処理できるフレーム数である。mAP と FPS の両方が高ければ、1秒間に多くのフレームを高い精度で評価できることになる。

ドローンのリアルタイム制御には FPS の高さが重要である。ドローンの移動速度は遅くとも 2~3m/s はあるため、FPS が低ければ、数メートル前の位置の画像を分析することになる。そこで、本研究では、現状で FPS が最も高く、mAP も他とほぼ同水準である YOLOv2 と Tiny YOLOv2 を選択した。ただし、YOLOv2 は画像をグリッドで分割し、グリッド毎にクラスを分類している。そのため、画像に多数のオブジェクトが含まれている場合、すべてのオブジェクトを検出できない問題がある。ただし、ドローンを損害調査に使用する際、多くの人や障害物が同時に画面内に映るケースは少ないため、支障はない。

表1 物体検出手法の比較(実行環境は NVIDIA の GeForce GTX Titan X, Train のデータセットは Pascal VOC)

Detection Frameworks	Train	mAP	FPS
Fast R-CNN	2007 + 2012	70.0	0.5
Faster R-CNN VGG-16	2007 + 2012	73.2	7
Faster R-CNN ResNet	2007 + 2012	76.4	5
SSD500	2007 + 2012	76.8	19
Tiny YOLOv2 (416 x 416)	2007 + 2012	76.8	67
YOLOv2 (544 x 544)	2007 + 2012	78.6	40

## 2.3 ドローンの機体

ドローンの機体は 3D Robotics 社が開発した Solo を使用した。主なスペックを表2に示す。Solo は産業用ドローンや DJI の INSPIRE と比較すると、小型に分類されるドローンである。Solo の特徴として、耐風性能が高く、少し強い風が吹いても墜落の危険性が低いこと、ソフトウェアとハードウェアの両方のカスタマイズ性の高いことが挙げられる。

表2 3D Robotics 社の Solo のスペック

重量	1.8 kg (カメラを含む)
最高速度	25.5 m/s
電波到達距離	800 m
飛行可能風速	11 m/s
最大搭載可能重量	500g
最大使用可能時間	20 分間
搭載カメラ	GoPro Hero 4

## 2.4 開発環境

プログラムの開発環境には Dronekit を選択した。Dronekit は 3D Robotics 社が提供する API や SDK がパッケージされたプラットフォームである。主な特徴として、ドローンのみならずヘリコプターやローバー、小型飛行船などの制御プログラムも開発できる点、MIT ライセンスであることから、無料で商用利用できる点、コンピュータやスマートフォンなどの様々なデバイス向けの開発が可能である点が挙げられる。現状、ドローンのオープンソースとしては、最も安定した制御ができるプラットフォームである。Dronekit はプログラミング言語に Python を使用していることから、ディープラーニングのプログラムとの組み合わせが容易な利点もある。Dronekit の各種 API を使用し、ドローンの水平方向の移動、上昇下降、ジンバルの角度などを操作した。

## 2.5 HDMI キャプチャ

Febon 168 は USB2.0 対応の HDMI キャプチャであり、UVC に対応しているため、コンピュータからは webcam として認識される。webcam として認識されることにより、OpenCV による画像処理が容易となることから、このデバイスを介して、コントローラーとコンピュータ間で映像を共有した。

## 2.6 Laptop

YOLOv2 を高速で実行させるため、NVIDIA の GPU が搭載されたコンピュータが必要である。また、屋外で飛ばすため、持ち運びができる Laptop が望ましい。そこで、GeForce GTX 1060 が搭載された Razer Blade 2017 を選択した。スペックを表3に示す。

表 3 Laptop の主なスペック

製品名	Razer Blade 2017
CPU	Intel® Core™ i7 -7700HQ クアッドコア プロセッサ ー HT 対応 (定格 2.8 GHz / TB 時 3.8 GHz)
GPU	NVIDIA® GeForce® GTX 1060 (6GB GDDR5 VRAM)
RAM	16GB デュアルチャンネル オンボードメモリー (DDR4, 2400MHz)
Storage	SSD 256 GB (PCIe M.2 接続)
OS	Ubuntu 16.04 LTS

ただし, Razer Blade はバッテリーの持続時間が 60~80 分間であり, 運用面で支障がある. そこで, Macbook Air も用いて, 状況によって使い分けた. Macbook Air のスペックを表 4 に示す. Razer Blade を使用したときの物体検出は YOLOv2, Macbook Air を使用したときの物体検出は Tiny YOLOv2 とした. また, Amazon の EC2 上で分析する環境も試みたが, 画像を送信した際のレイテンシが大きかったため, 候補から外した.

表 4 Macbook Air の主なスペック

製品名	Macbook Air 2017
CPU	Intel® Core™ i5 デュアルコア プロセッサ (定格 1.8 GHz / TB 時 2.9 GHz)
GPU	Intel HD Graphics 6000
RAM	8GB 1,600MHz LPDDR3 オンボードメモリー
Storage	SSD 256 GB (PCIe 接続)
OS	macOS High Sierra

### 3. テスト飛行

本システムを用いて 2 通りのテスト飛行を行い, 物体検出に基づいて, ドローンが制御できるかを検証した. 一つはドローンのヘリポートへの着陸時の制御であり, もう一つは映像内に人を捉えたときの制御である.

#### 3.1 ヘリポートへの着陸時の制御

ヘリポート(ランディングパッド)からドローンを離陸させて, 10m の高度まで上昇する. そして, 下降運動に移り, 離陸したときと同じヘリポートに着陸させる. この下降運動中, ヘリポートの位置をドローンのカメラで検出し, 位置を調整することを試みた. 図 2~4 にドローンのカメラで撮影した画像を示す. 図 2 は高度 8 m, 図 3 は高度 5 m, 図 4 は高度 3 m のときの映像である. 四角枠が YOLOv2 によりヘリポートがあると認識した位置である. YOLOv2 は常にヘリポートの位置を正しく認識しており, ドローンもそのヘリポートの中心に向かって移動している. ヘリポートに近づきすぎると, 少しポジションがずれたときにカメラの映像内にヘリポートが映らない問題が生じる. そのため, 高度 3 m で水平方向の位置を固定し, それから垂直に降下させるプログラムとした. レイテンシは Razer Blade を使用したときが 1.1 秒, Macbook Air を使用したときが 3.0 秒である. HDMI キャプチャによるレイテンシが約 1 秒, 残りが YOLOv2 (Razer Blade) もしくは Tiny YOLOv2 (Macbook Air) によるものである.

都市部などの遮蔽物が多い地域では十分な衛星が捕捉できずに測位の精度が落ちる. 物体検出に基づいた着陸であれば, この問題を回避できると考える.



図 2 カメラの向きを真下にしたときの映像 (高度 8 m)



図 3 カメラの向きを真下にしたときの映像 (高度 5 m, 図 2 の状態から 6 秒後)



図 4 カメラの向きを真下にしたときの映像 (高度 3 m, 図 2 の状態から 12 秒後)

#### 3.2 人を映像内に捉えたときの制御

ドローンのカメラが人を映像内に捉えた際, 適切な回避運動が求められる. 人の位置を認識した運動が可能であるかをテストするため, 物体検出の結果に基づいて人が画面の中央に位置するようにドローンを制御するテストを行った. 図 5 が高度 6 m でホバリングしたときの映像であり, 図 6 が図 5 の 6 秒後, 図 7 が図 5 の 8 秒後の映像である. 高度は変えずに水平方向のみを変えてポジションを調整している. 結果, 人の位置を認識し, 映像の中心に人を捉えることが可能であることを確認した. なお, 映像ではコントローラーを手にしているが, 緊急時にドローンを操作するためであり, コントローラーによるマニュアル操作は行っていない. 映像で人を認識し, それに応じた制御をすることにより, 第三者の上空を飛行する状況を回避できると考える.



図 5 物体検出により人を認識したときの映像（高度 6 m）



図 6 物体検出により人を認識したときの映像（高度 6 m, 図 5 から 6 秒後）



図 7 物体検出により人を認識したときの映像（高度 6 m, 図 5 から 8 秒後）

#### 4. 実務上の課題

本システムは損害保険の実務に適用する前提で構築し、一定の成果を得た。しかし、ドローンを実務に導入するためには、まだ多くの課題が残されている。主な課題を以下に整理する。

まず、ドローンの準備に要する時間である。台風後の住宅物件の損害調査に要する時間は会社により異なるが、早い会社であれば 1 件あたり 30 分間で終わる。そのため、ドローンの撮影も短時間で終わることが求められる。理想的にはすべての作業を 5 分以内で終わることが望ましい。しかし、ドローンの準備にはプロペラガードの装着、衛星の捕捉、キャリブレーションなどがあり、起動するまでも 10 分間は要している。本システムは Laptop を使用するため、接続や起動などで更に準備時間の掛かる点が課題となった。

次が天候の問題である。経験上、気象庁の予報で風速（10 分間平均風速）が 3 m/s を超えた場合、安定した飛行が困難となる。またドローンはモーターがむき出しの状態であるため、僅かでも雨が降っていると飛ばせない。穏やかな風の日のみで使用

できる。損害調査にドローンを導入する場合、訪問時に常に飛ばせるとは限らない点が問題となる。

最後が電波干渉の問題である。都市部では wi-fi の 2.4GHz が混信しているため、通信が途切れるケースがある。現在の多くのドローンは wi-fi でコントローラーと通信しているため、この問題を有している。その他、航空法規制による運用上の問題、プライバシーの問題にも配慮する必要がある。

#### 5. まとめ

物体検出の YOLOv2 とドローンの開発環境である Dronekit により、ドローンのリアルタイム制御のシステムを構築し、屋外でのテスト飛行によりシステムを検証した。本研究により、以下の結果が得られた。

- ドローンの下降運動中、ヘリポートの位置を物体検出し、ヘリポートの中心に着陸するように位置を補正しながら着陸できることを確認した。
- ドローンのホバリング中、人の位置を認識し、人が画面の中央に位置するようにドローンを制御できることを確認した。

また、損害調査にドローンを導入する際の課題を整理した。主な課題は飛行準備に要する時間、天候、電波干渉であり、その他に規制上の問題、プライバシーの問題がある。損害調査の際に屋根の上に登ることは危険が伴う。現状のドローンには課題が残されているものの、損保業界として引き続きドローンの導入を模索したいと考える。

#### 参考文献

[Slate 17] Slate, Insurance Companies Are Preparing Fleets of Drones to Assess the Damage of Harvey, アクセス日時 2018 年 2 月 5 日

[http://www.slate.com/blogs/future\\_tense/2017/08/29/insurance\\_companies\\_will\\_fly\\_drones\\_after\\_hurricane\\_harvey.html](http://www.slate.com/blogs/future_tense/2017/08/29/insurance_companies_will_fly_drones_after_hurricane_harvey.html)

[Lee 17] Jangwon Lee, Jingya Wang, David Crandall, Selma Sabanović, and Geoffrey Fox, Real-Time, Cloud-Based Object Detection for Unmanned Aerial Vehicles, First IEEE International Conference on Robotic Computing (IRC), 2017

[Redmon 17] Joseph Redmon, Ali Farhadi, The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 7263-7271, 2017.