

# ドローンの羽音を利用して個体識別及び位置情報を検出するシステムの提案

小野 龍一<sup>1,a)</sup> 室崎 之典<sup>1</sup> 齊藤 佑祈<sup>1</sup> 中野 亜希人<sup>1</sup> 大淵 康成<sup>1</sup> 羽田 久一<sup>1,b)</sup>

**概要:** ドローンの位置を音で取得する手段としてマイクロフォンアレイを用いた方法がある。しかしその方法では、位置しか取得できず個体の識別はできない。そこで本研究ではマイクを装着した複数のマイコンボードと、羽音から特徴量を抽出することで個体識別と位置推定を可能とするシステムを提案する。その結果、少ない特徴量でもドローンの羽音から高確率に個体識別を行うことができ、2mの範囲において20cmから30cmの誤差での位置推定に成功した。

## Proposal of a system for detecting individual identification and position information using drone's feathers

ONO RYUICHI<sup>1,a)</sup> MUROSAKI YUKINORI<sup>1</sup> SAITO YUKI<sup>1</sup> NAKANO AKITO<sup>1</sup> OBUCHI YASUNARI<sup>1</sup>  
HADA HISAKAZU<sup>1,b)</sup>

**Abstract:** Detecting a position of a drone has been conducted by using microphone array in previous researches. However, this method could only acquire a position of drone, and this does not identify each drone. In this research, we propose a system that distinguishes individual drones and their estimated location, using propeller hum sound obtained by microphone implemented to multiple microcomputer boards. To distinguish such difference, we first made the micro computer to learn the different propeller sound of each drone. As a result, we successfully identified position of drone with high accuracy, with only small amount of selected feature. We have also succeeded to identify drone's position with a measurement error of 20cm to 30cm in 2 meters range.

### 1. はじめに

近年ドローンを用いた演出表現が普及してきている。ドローンを使用した演出の例として24台の小型ドローンが機械学習の手法を取り入れることで、お互いにぶつからない安全な距離を常に保ちつつ3人のダンサーとパフォーマンスを行ったElevenplayとRhizomatiksResearchの24drones[1]。富士山を舞台とし、20台を超えるドローンの編隊飛行、三味線の生演奏、MIDIコントロールされた1万6500個のLEDライトでライブ演出を行ったSky Magic live at Mt. Fuji[2]。500機のドローンを使ったライトショーでギネス記録を更新したIntelのIntel's 500

Drone Light Show[3]などがある。これらの演出はドローンの位置を取得する方法にOptiTrackなどの光学式の方法を採用したトラッキングシステムやGPSを用いておりあらかじめ反射マーカを装着する手間やキャリブレーションを行う必要がある。GPSに関しては屋内では使えない。そこでドローン本体に何も施すことなく機体の識別と位置推定をドローンの羽音から導き出すシステムを提案する。本システムはドローンに何も装着する必要がない為、機体の重量が変化せず入手時の状態のまま位置と個体識別が可能となる。

### 2. 関連研究

#### 2.1 音の位置推定

ドローンに関する位置推定は、様々な手法が研究されている。その中でドローンの位置を推定した先行研究

<sup>1</sup> 東京工科大学 メディア学部  
Tokyo University of Technology School of Media Science  
a) m011410763@edu.teu.ac.jp  
b) hadahskz@edu.teu.ac.jp

を紹介する。岡田らは、マイクロホン感度の単一指向性を利用し、3次元での音源方向を検出する研究 [4] を行った。3次元空間内での音源方向を検出には、マイクが四面体頂点になるように配置する必要があることを示し、それぞれのマイクが取得する音圧（振幅）の強度差を用いて音源方向が検出できることを提案した。マイクロフォンアレイによる位置推定は、数多く研究されている手法である。マイクロフォンアレイを用いると、ドローンの音源の方向とマイクから音源までの距離が得られる。川岸らは、マイクロフォンアレイを用いてドローンのローター音を計測し、音が到達するまでの時間差によって位置を推定するシステムを開発 [5] した。このシステムを用いた結果では、5m 間隔でマイクを設置することで 0.7m 以内の誤差で位置推定可能になったことがわかった。萬造寺らは、パッシブ音響センサー方式によるドローンの探知システムを開発 [6] した。三方向に放射状に伸ばしたマイクで音を取得し、音の伝達時間差により方向を検出する。ドローンの位置は検出できないが、カメラによって飛行方向を検出することが可能である。これらの先行研究はドローン本体に何も装着する事なく簡易的に位置を推定できる。

## 2.2 機械学習によるパターン認識

機械学習によるパターン認識を行った研究を紹介する。音の特徴量を抽出し機械学習で分類した例として大淵らの機械学習による硬貨落下音の分析の研究 [7] がある。この研究ではピッチ抽出やパワーなどの特徴量を使用して硬貨の判別を試みている研究である。実験には特徴量の抽出に openSMILE、機械学習に weka を使用しており、最終的な結果として三種類の硬貨の音を 80 % の正答率で識別している。機械学習を用いたジェスチャ認識の事例に佐藤らのユーザの”触り方”を検知するデバイス Touché の研究 [8] がある。ユーザがデバイスに触れることで予め流入した周波数帯の変化を周波数解析し、99 % のジェスチャー分類精度の結果をだしている。

## 3. 羽音による位置推定と個体識別の提案

光学式の方法では、キャプチャーさせるために反射マーカーを取り付ける必要がある。そのため装着する手間がかかる事や重量が重くなるといった問題がある。更にライブ演出用途を想定した場合、ドローン本体の造形も評価の対象となり得るので、この要素もデメリットとなる。前章で述べたように、位置推定を行うと同時に個体識別ができるシステムはまだ存在していない。そこで、ドローン本体に特別な処理を施す事なく、位置推定と羽音による個体認識を可能とするシステムを提案する。複数のドローンを制御する際、位置情報がわかっても個体の区別がなければ、演出等での制御に支障が出る。ドローンに制限をかけないようにするため、どのドローンも一様に備わっている特徴

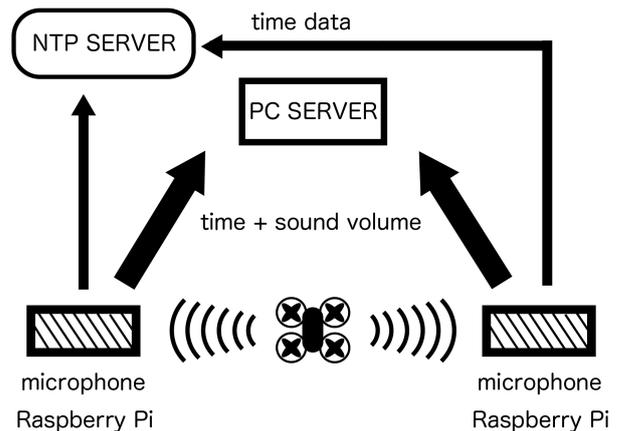


図 1 ネットワークの構成

Fig. 1 Network configuration

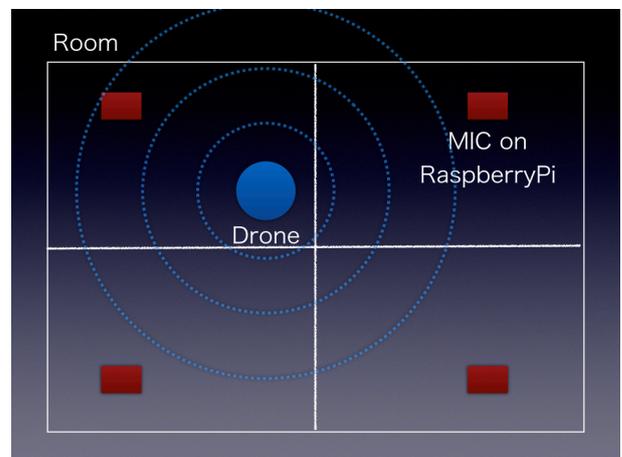


図 2 位置推定のイメージ図

Fig. 2 Image drawing of position estimation

としてプロペラの羽音に注目した。そのため本研究ではプロトタイプとして、大まかな位置推定と羽音による個体識別システムの実装と実験を行う。

## 4. 羽音による位置推定と個体識別システムの実装

### 4.1 システム概要

システム全体としてはまず2つのマイコンボードで音量を測定した値を PC サーバーに送信し、PC サーバー内のアプリで受信した値を元に位置推定処理を行い、音源のおおよその位置を導き出す。同時に PC サーバーのマイクロフォンでドローンの羽音を録音して特徴量を抽出し、あらかじめ登録しておいたドローンの羽音の特徴量と比較し類似度を計算した。ネットワークの構成は図 1 のとおりである。

### 4.2 音源の推定

ドローンの位置をプロペラ音で特定するために、音量を

利用する手法を用いた。この位置推定システムは図 2 のように、ドローンのプロペラ音を異なる位置に設置されたマイクにより音量を測定し、複数のマイクから取得した音量を比較することでどのマイクの近くにドローンが存在するかを推定するものである。このシステムの特徴は、ドローンに対する拘束がなく、マイクを取り付けたデバイスを位置推定したい場所に置くだけの簡易的なものであるところにある。今回のプロトタイプでは 2 つのマイクデバイスで実装、実験を行った。大まかな処理の流れは、マイクで取得した音量を時刻 (マイクロ秒) データと合わせて PC サーバに送信し、PC サーバで 2 つの受信したデータの同じ時刻同士の音量を比較する。そしてその取得音量の大きな方の側に音源となるドローンが存在すると特定するといった流れとなっている。ここで時刻データも同時に送信するのはマイクロ秒単位での通信となるため、より細かいタイミングで判定できるようにするためである。以降詳しくシステムの仕組みを見ていく。本システムではマイクを取り付け、音量情報を取得するためにマイコンボードの Raspberry Pi 3 を用いた。マイクは高感度な小型の電気リックコンデンサマイクを使用した。ドローンの小さい音を拾うにあたり、マイクの振幅信号を図 3 の通り、オペアンプにより増幅させる。増幅率の設定は、微量の音まで拾うためオペアンプで増幅させた。マイクの出力を、オシロスコープを使って増幅前と増幅後の比較を行なった。これにより、最大電圧が 5V のうち、測定した電圧の最大値はオペアンプなしでは 500mV、オペアンプありが 3.44V となり、約 70 倍に増幅された。同じ音に対して Raspberry Pi 3 はアナログ信号の入力がないため、A/D コンバーターを介してデジタル信号として Raspberry Pi に送る。Raspberry Pi 内の処理としては音データを音量として取得すると同時に現在時刻を取得する。その際、複数のマイコンボード間で時刻のズレが生じるのを防ぐため、同一ネットワーク内に構築した NTP(Network Time Protocol) サーバの時刻データを有線 LAN 経由で参照する。NTP サーバには CentOS を使用した。次に、それらのデータを PC サーバに送信する手段として、OSC 通信を用いた。OSC(Open Sound Control) はデータ量が非常に少ないため高速に通信ができ、送信手段もシンプルなため、この通信規格を採用した。Python 環境で提供されている OSC ライブラリの pyliblo を使用し、Python プログラムによって一連の取得から送信を行う。OSC で送信する情報には、異なる Raspberry Pi からのデータを区別するためにラベルを付与する。また、音量情報と時刻情報は一つの OSC データとして送信する。受信側の PC サーバには Mac OS を使い、openFrameworks によりサーバを構築する。受信した OSC データの時刻と音量を取り出し、受け取った二つの Raspberry Pi の時刻が同じなら音量比較の処理をする。そして最終的に、音量が大きい方のデバイス側の近くにドローンが存在している

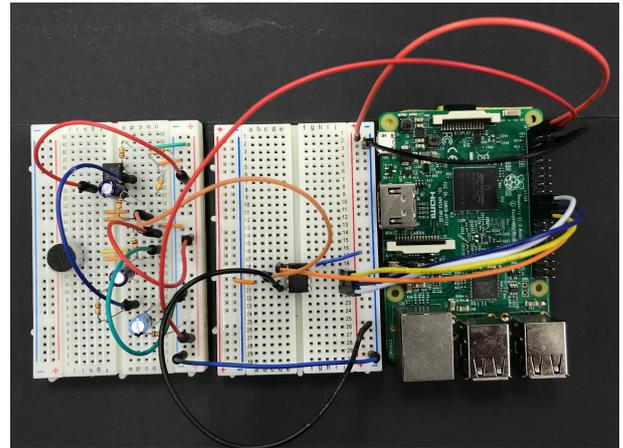


図 3 ラズベリーパイに接続されたマイク  
Fig. 3 Microphone connected to raspberry pie

という結果を返す。

### 4.3 羽音のパターンの認識

#### 4.3.1 構成

本研究では羽音の特徴量抽出に openSMILE, パターン認識に Weka を使用した。openSMILE とは「音声認識」「音楽認識」「パラ言語認識」の研究向けに作られた特徴量を抽出するユーティリティで weka は機械学習ソフトウェアである。羽音の録音には PC サーバに接続した USB マイクと Maxmsp を用いた。Maxmsp で行った処理は定期的に音声を録音し wav ファイルに変換し、指定したフォルダに保存するといった内容である。録音した wav ファイルは、openSMILE を用いて特徴量が記録された学習用の arff ファイルと予測用の arff ファイルの作成に使用した。作成した学習用 arff ファイルを weka を用いてモデルデータに変換し、そのモデルデータと予測用 arff ファイルを比較し個体の予想結果の出力を行った。テストに用いたドローンは図 4 に載せた Crazyflie 2.0, PARROT の MiniDrones Rolling Spider, Makeblock の Airblock である。

#### 4.3.2 音データ

本研究で用いる音データは Crazyflie 2.0, MiniDrones Rolling Spider, Airblock の羽音を PC サーバに取り付けた USB マイクと Maxmsp, Audacity を用いて録音した。学習用の音データの作成には Audacity を用いた。録音の長さは 1 秒にし音声ファイルの形式には wav を用いた。録音は 1 機体につき 10 回おこなった。予測用の音データの作成には Maxmsp を用いた。定期的に指定のディレクトリに録音した wav ファイルを保存するプログラムを作成した。

#### 4.3.3 特徴量抽出

特徴量抽出の流れは図 5 のとおりである。具体的には音データを短時間のフレームに分割し、フレームごとに特徴量抽出を行い、その後それぞれの特徴量を全フレーム分

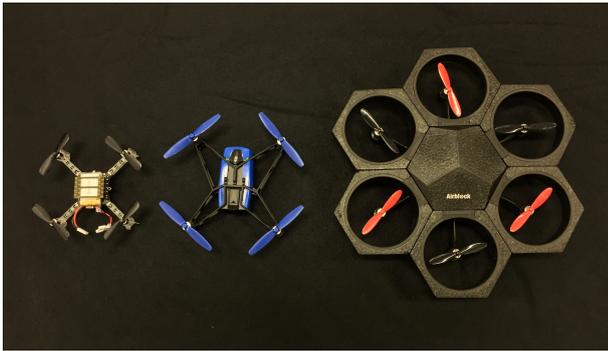


図 4 使用したドローン

Fig. 4 Drone used

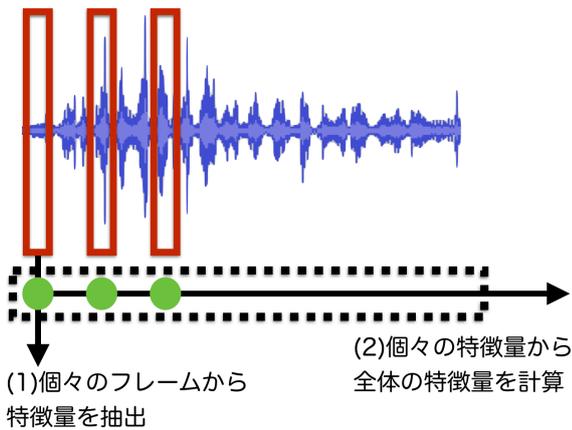


図 5 特徴量抽出の流れ

Fig. 5 Flow of feature quantity extraction

表 1 個々のフレームからの特徴量

Table 1 Feature amount from each frame

名称	抽出内容
F0	基本周波数
centroid	スペクトルの重心
entropy	スペクトルのエントロピー
variance	スペクトルの分散
skewness	スペクトルの歪度
kurtosis	スペクトルの尖度

集め全体の特徴量の抽出を行う。最初に欲しい特徴量を設定をする為に必要な conf ファイルの作成を行った。この conf ファイルは openSMILE が音データから特徴量を抽出するのに必要な指示が記されている。今回取得した特徴量は表 1 と表 2 のようになっている。特徴量の種類の選別は既存の研究を参考に必要な項目を予測し行った。まず学習用の arff ファイルを作成する為に Audacity から出力された wav ファイルを使用し特徴量の抽出を行った。次に予想用の arff ファイルの作成は Maxmsp から出力された wav ファイルを使用し特徴量の抽出を行った。

#### 4.3.4 分類

まず Weka を使用し学習用の音データの特徴量を記録し

表 2 全体の特徴量

Table 2 Overall feature quantity

名称	抽出内容
Max	最大値
Maxpos	最大値を取るフレーム
amean	平均
linregc1	線形回帰係数 (傾き)
linregc2	スペクトルの尖度 (オフセット)
linregerrQ	線形回帰二乗誤差

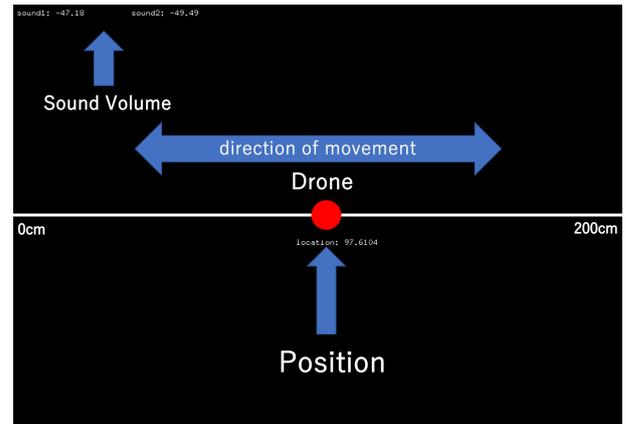


図 6 アプリケーション画面

Fig. 6 Application screen



図 7 実験風景

Fig. 7 Experimental landscape

た arff ファイルからモデルデータを作成し、このモデルデータと予想用の arff ファイルを比較することでドローンの個体の予測結果の出力を行う。

#### 4.4 アプリケーション

推定した位置の結果を可視化するために、openFrameworks を用いて図 6 に示したアプリケーションを作成した。受け取った音量の値を、図 8 のように円の X 座標に反映させることで、直線上を推移させる。その際、左端から 200cm までの推定した位置を cm 単位で表記する。それによってリアルタイムにドローンの一次元的な位置を視覚的に把握できるようにした。

### 5. 実験と結果

#### 5.0.1 位置推定

プロトタイプ実験として、図 7 のように 2 m の範囲で位置推定の実験を行なった。図 8 に示したように 2 m の間隔を空けた位置に実装したマイクデバイスを設置し、そ

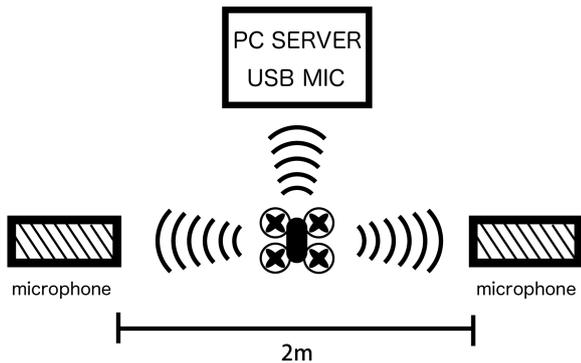


図 8 実験設備図

Fig. 8 Laboratory equipment drawing

表 3 変更後の個々のフレームからの特徴量

Table 3 Characteristics from each frame after change

名称	抽出内容
F0	基本周波数
centroid	スペクトルの重心
entropy	スペクトルのエントロピー

の間の地点でドローンを飛行させた。そして一連の処理を行い、アプリケーションで視覚的にドローンの位置情報を提示させた。2 mの間隔を空けて設置したマイクの間には c m 単位の目盛をつけ、システムによる推定と実際の位置の誤差を把握できるようにした。また、ドローンの飛行は手で操縦し、マイクの間を行き来させるように移動させた。そして、アプリケーション上での音源の移動による位置の推移を観察した。その結果、ドローンの位置を 20~30 c m の誤差での位置推定に成功した。しかし、ドローンの早い動きに逐一对応することはできなかった。ドローンの移動が早いと、データの取得が追いつかず正確に推定することはできなかった。

### 5.0.2 個体識別

マイクから半径 2m 以内の範囲で三機種のドローンを飛行させ録音した。そして録音した音声データを arff に変換し予測データを一機種につき 10 個ほど作成し計 30 回モデルデータとの比較を行った。その結果、識別率は 30 回とも全て正確であった。次に更に計算量を減らすために取得する特徴量の項目を削減した conf ファイルを作成することにした。削減した特徴量の項目はフレームではスペクトルの分散、歪度、尖度を全体では最大値、最大値を取るフレームを削減した。項目を削減した conf ファイルを元にモデルデータと予測データを作成し比較を前回と同様に行った。結果、30 回とも正確な識別結果が得られた。削減後の特徴量の項目の表は表 3 と表 4 である。

## 6. 評価

今回の結果から、音量のみの少ない情報でも音源の位置

表 4 変更後の全体の特徴量

Table 4 Overall feature amount after change

名称	抽出内容
amean	平均
linregc1	線形回帰係数 (傾き)
linregc2	スペクトルの尖度 (オフセット)
linregerrQ	線形回帰二乗誤差

が推定できることが分かった。このシステム構成では設置位置が既知であれば、方向を把握せずとも距離が推定できる。また、マイクロフォンアレイより安価かつ小型なシステムでも実現が可能となる。個体識別は特徴量の抽出を削減した状態でも複数のドローンの羽音から高確率に個体識別を行う事ができた。その為、ドローンの判別に必要な特徴量はフレームの場合は基本周波数、スペクトルの重心、スペクトルのエントロピーである。全体の場合は平均、線形回帰係数、スペクトルの尖度、線形回帰二乗誤差で必要な結果は得られる。

## 7. おわりに

本稿では、ドローンの羽音による音源の位置推定と個体識別のシステムの提案と実装、実験を行なった。位置推定の精度実験の結果、20~30 c m の誤差で位置推定が可能となり、個体識別の実験では特徴量を削減した状態でも削減しない状態と同じ正答率という結果が得られた。位置推定に関しては 2 m 範囲での実験であったが、今後の取り組みとして詳細な距離測定による位置の推定と、周波数フィルタリングによる雑音除去を行う必要がある。今回の結果より、マイクを 2 つ用いることで一次的に位置推定が可能となることから、マイクを 3 つに増やすことで二次元的、4 つに増やすことで 3 次的に座標を取ることも十分可能だと考えられる。個体識別に関しては実験に用いたドローンの機種が全て別であった為、同一の機種を用意して同様の実験を行った場合でも識別が可能か検証したい。またドローンの羽音を操作し新たな羽音の特徴をもたせる手法が考えられる。これらの改善点を含め、さらなる精度向上を目指す。

### 参考文献

- [1] Elevenplay x Rhizomatiks Research 24 drones 入手先 (<https://research.rhizomatiks.com/s/works/24drones/>)2015).
- [2] マイクロアド Sky Magic Live at Mt. Fuji : Drone Entertainment Show 入手先 (<https://www.youtube.com/watch?v=aOd4-Tp5fA>)2016).
- [3] intel Intel's 500 Drone Light Show 入手先 (<https://www.youtube.com/watch?v=aOd4-Tp5fA>)2016).
- [4] 岡田徳次, 佐藤秀幸, 三富堅太郎: マイクロホン感度の単一指向性を利用する 3 次元音源方向の検出, 計測自動制御学会論文集 vol. 37 (2001), No. 1, 13/20.

- [5] 川岸 卓司, 水谷 孝一, 若槻 尚斗: マイクロフォンアレイによるドローン飛行音の位置推定, SAT テクノロジー・ショーケース 2016, p95.
- [6] 萬造寺博, 望月大志, 神林誠, 進士浩樹: 空中音響技術を利用したドローン探知システム, OKI テクニカルレビュー vol. 82(2015), No. 2.
- [7] 横田 渉・大淵康成: 機械学習による硬貨落下音の分析, 映情学技報, vol. 41, no. 12, AIT2017-119, pp. 261-264, 2017年3月.
- [8] Munehiko Sato, Ivan Poupyrev, Chris Harrison. "Touché: Enhancing Touch Interaction on Humans, Screens, Liquids, and Everyday Objects. " In Proceedings of the 30th Annual SIGCHI Conference on Human Factors in Computing Systems (Austin, TX, May 5- 10, 2012). CHI ' 12. ACM, New York, NY.