

## 管理領域に侵入する小動物を撃退するロボットシステム：飛行ロボットによる監視・撃退システム

前川, 真吾 / MAEKAWA, Shingo

---

(出版者 / Publisher)

法政大学大学院理工学・工学研究科

(雑誌名 / Journal or Publication Title)

法政大学大学院紀要. 理工学・工学研究科編 / 法政大学大学院紀要. 理工学・工学研究科編

(巻 / Volume)

55

(開始ページ / Start Page)

1

(終了ページ / End Page)

5

(発行年 / Year)

2014-03-24

(URL)

<https://doi.org/10.15002/00010373>

# 管理領域に侵入する小動物を撃退するロボットシステム -飛行ロボットによる監視・撃退システム-

A ROBOT SYSTEM THAT REPELS SMALL ANIMALS TRESPASSING INTO MANAGED TERRITORIES  
- SURVEILLANCE AND REPELLING SYSTEM USING A FLYING ROBOT-

前川 真吾  
SHINGO MAEKAWA  
指導教員 高島 俊

法政大学大学院工学研究科機械工学専攻修士課程

Recently there are many cases that little animals such as homeless cats intrude into home gardens or garages and do harm to them, such as dropping animal excreta or making scratch damages on cars. Therefore, there are many needs for keeping little animals away from home area and therefore, many kinds of cat-repelling goods are sold in the city. In this report, a new type of patrol robot that can repel little animals without treating them cruelly is proposed. The robot can recognize little animals, track them and repel them from a home garden by some harmless method.

**Key Words:** Patrol Robot, Home Garden, Repelling Animals, Flying Robot, AR Drone

## 1. 緒 論

現在、土地開発や環境変化による弊害で農地に野生動物が、一般家庭の庭に猫などの小動物が侵入し被害を与えるといった報告が増えている。また動物によっては人的被害に及ぶものまで報告されている。このような状況を受けて、それらに対する柵や金網といったような防犯システムも多数設置されているが、効果が一時的であったり、広範囲にわたりすぎて設置が困難であったりなど様々な問題が存在し確実な成果を上げている物がないのが現状である。そこで移動可能なロボットにより、少ない時間で設置でき、かつ広範囲をカバーできる小動物撃退ロボットシステムの実現を目指す。また、小動物の撃退方法は動物に危害を加えずに、追い払うことを第1目標とする。

本報告は、主に家庭の庭に侵入する小動物を寄せ付けなためロボットの開発を試みた、福井らの文献[1][2]による「お庭番ロボット」の継続研究として、家庭の庭に侵入する猫を撃退するロボットシステムを考察することで小動物撃退ロボットシステム全体の監視・撃退システムの考察・検討した。

## 2. システム概要

管理領域に侵入する小動物を撃退するには大きく分けて4つのシステムが必要である。まず、管理領域になんらかの生命体が侵入したことを感知するシステム、次に侵入した生命体が小動物かどうかを確認するシステム、そして

確認した生命体が小動物であった場合にそれを追跡するシステム、最後に追跡している小動物が管理領域に被害を及ぼす前に追い払うシステム、そしてこれら4つのシステムを制御用コントローラとして用意した一つのPC(パーソナルコンピュータ)で管理・制御することで小動物の監視・撃退を行っていく。Fig.1にシステムの全体図を示す。本報告ではこの4つのシステムの内、感知システムと認識システム、そして追跡システムについてと制御用コントローラについて考察・検討した。

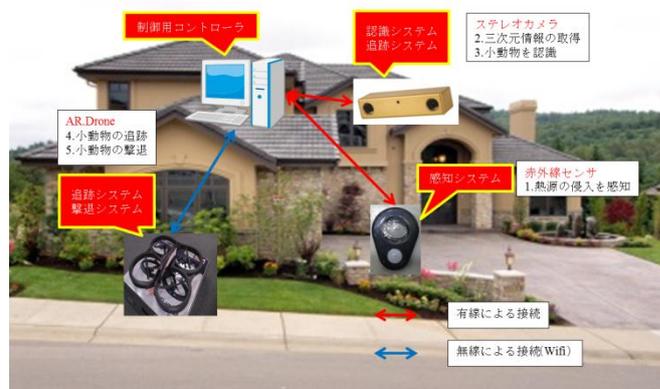


Fig.1 General view of the system

### 3. 感知システム

このシステムは管理領域内に侵入した生命体を感知するシステムである。想定される小動物は恒温動物であるので一定の温度を持つ物体の移動を検知できる赤外線センサを使用する。実験に使用するセンサにはアイリスオーヤマ社製の乾電池式 LED センサーライト LSL-0.5 を採用した。このセンサーライトは感知範囲内で一定温度以上の熱源が移動するとライトを点灯させる。後述する動物認識システムに光源が必要なため、このタイプの既製品を侵入感知センサとして使用する。このセンサーライトの感知範囲は水平前方方向に中心角約  $130^\circ$  半径約 3m の扇形状、鉛直下方向に約 2m である。Fig.2 に感知範囲の図を示す。

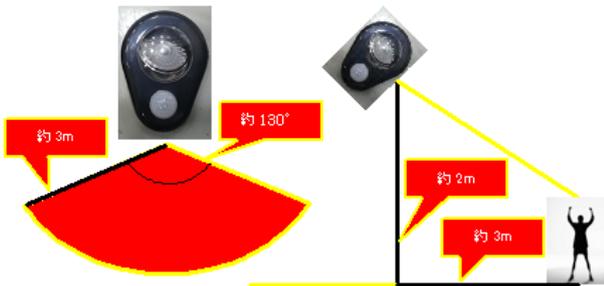


Fig.2 Perception range of the infrared sensor

このセンサは日中にライトが点かないように光センサで一定の明るさ以上のときは感知しない設定になっている。そのため、光センサの回路を切断することで日中でもセンサーライトを使用できるようにした。また、熱源が範囲内に入ったことをコントローラに伝えるため、センサ内の抵抗器と AD 変換モジュールを繋いだ。これは熱源を感知していないときはライトが光らないため抵抗器に電圧が発生せず、熱源を感知しているときはライトが光り抵抗器に電圧が発生するためである。この電圧の有無で侵入を判断する。使用する AD 変換モジュールは interface 社製の PCI-3168C で、このモジュールは  $\pm 10V$  を検出することができる。このモジュールを使用し、複数の赤外線センサを管理領域全体に配置することで小動物の侵入を感知する。またどの赤外線センサが感知したかを制御システムに伝えることで小動物がどのあたりに居るかを推定させることができる。

### 4. 認識システム

このシステムは感知システムで感知した熱源を持つ移動物体を認識するシステムである。今回は一般家屋に侵入する猫を例として目的とするため、認識システムで猫を認識させる。また、後述する追跡システムで位置を確認する絶対的な三次元距離情報が必要なため、認識システムのカメラには三次元情報を取得できる Point Gray Research 社製の二眼ステレオカメラ Bumblebee2 を使用する。

二眼ステレオカメラ Bumblebee2 は付属されている

TRICKLOPS SDK を使用することにより、C++ のプログラムでカメラ映像の取得と三次元演算をすることができる。

SDK とは Software Development Kit の略でカメラ情報の取得や三次元演算をするプログラムをまとめた Library file とまとめられたプログラムを必要に応じて関数として使用できるように定義する Header file, そしてそれらを使用して作成されたサンプルプログラムが入っている開発用のキットである。

#### (1) 三角測量

ステレオカメラの三次元演算は三角測量を採用している。詳解 OpenCV[3]によると三角測量は完全に歪み補正され、行がそろい、正確に計測された二つの画像があると仮定して行う測量である。その仮定条件の状態を Fig.3 に示す。

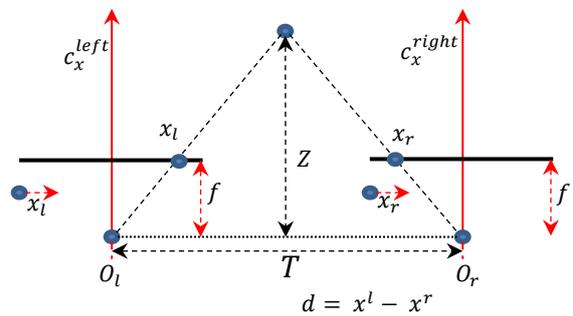


Fig.3 Method of triangulation in 2-D plane

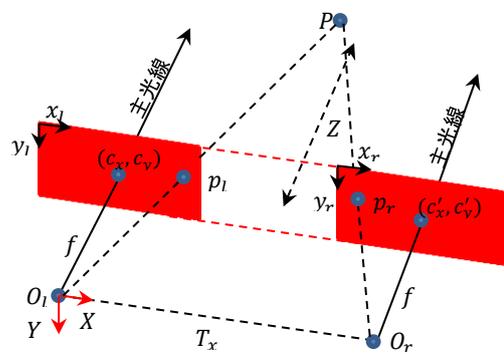


Fig.4 3D drawing of triangulation

Fig.3 より、平面図の相似比から式 (1), (2) が得られる。

$$\frac{T - (x_l - x_r)}{Z - f} = \frac{T}{Z} \quad (1)$$

$$Z = \frac{fT}{x_l - x_r} \quad (2)$$

つまり、焦点距離  $f$  とカメラ間距離  $T$  と視差  $(x_l - x_r)$  が分かればカメラから目標点  $P$  までの距離を測定することができる。SDK の Library では Fig.2 ではなく、Fig.4 を理

想的な設定とすることで三次元的な位置情報を取得している。Fig.4はFig.3を立体的に表現した図である。この状態での計算式は再投影行列  $Q$  を使用して計算処理する。式(3)に  $Q$  を示す。

$$Q = \begin{bmatrix} 1 & 0 & 0 & -c_x \\ 0 & 1 & 0 & -c_y \\ 0 & 0 & 0 & f \\ 0 & 0 & -\frac{1}{T_x} & \frac{(c_x - c'_x)}{T_x} \end{bmatrix} \quad (3)$$

理想的な状態の場合  $(c_x - c'_x)$  は 0 となる。

この再投影行列を使用して左画像の  $X, Y$  位置と視差  $d$  を式(4)に代入すると 3次元で表現された式(2)が得られ、変形すると最終的に 3D カメラ座標  $(X/W, Y/W, Z/W)$  つまりはカメラ座標系が得られる。

$$Q \begin{bmatrix} x \\ y \\ d \\ 1 \end{bmatrix} = \begin{bmatrix} X \\ Y \\ Z \\ W \end{bmatrix} \quad (4)$$

これによりカメラ座標が得られるため、画像から画像内の物体までの距離が得られる。

## (2) 猫の認識

認識システムの本题である。感知した熱源を持つ移動物体認識するシステムには MATLAB を使用する。MATLAB は数値計算、可視化、プログラミングのための高水準言語による開発環境のことで、この MATLAB を使用して物体認識を行うプログラムが Ross B. Girshick's Home Page [4] にて無料で配布されている。このプログラムは Discriminatively trained deformable part models と言って認識目標をいくつかのパーツに分けそのパーツとパーツの位置関係を学習し、それらを個別に認識することでその認識目標が画像内のどこにあるかを認識するプログラムである。Fig.5にこのプログラムを用いて猫を認識した画像を示す。

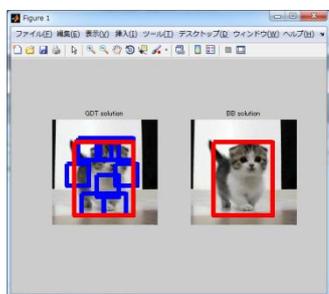


Fig.5 A picture of the result of recognizing a cat

このプログラムを C++ に移植することで、入力したカメラ画像内における猫の位置を取得し、その位置を三角測量で三次元位置情報に変換することで絶対的な猫の位置情報を得ることができる。

しかし、このプログラムの欠点として画像内における認

識目標が小さいと認識できないという問題点とプログラムの処理時間が長いこと、常時認識し続けることが難しいという問題点がある。これらを解決する手段として、物体認識のアルゴリズムを自作し、より良い学習データを作成するなどの手段が考えられる。

## (3) 追跡ロボットの認識

追跡ロボットに追跡を行わせるには追跡ロボット自体の位置を認識する必要がある。今回は ARToolKit を使用した認識方法を採用した。ARToolKit は AR アプリケーションの実装を手助けする C 言語用のライブラリである。AR とは拡張現実感 (Augmented Reality) の略語である。このライブラリを利用すると事前に登録したパターンを画像内から発見、認識し、そのパターンの画像内の位置、傾きに合わせて 3D キャラクターなどを表示することができる。追跡ロボットに登録したパターンを取り付けることで、追跡ロボットが画像内のどの位置に、どの傾きで存在するかをコントローラに認識させることができる。このプログラムの作成は「工学ナビ「ARToolKitを使った拡張現実感プログラミング」」[5]、「俺 CG 屋」[6]、「The Sixwish project」[7]を参考に作成した。Fig.6に認識した AR マーカーを白線で囲った図を示す。

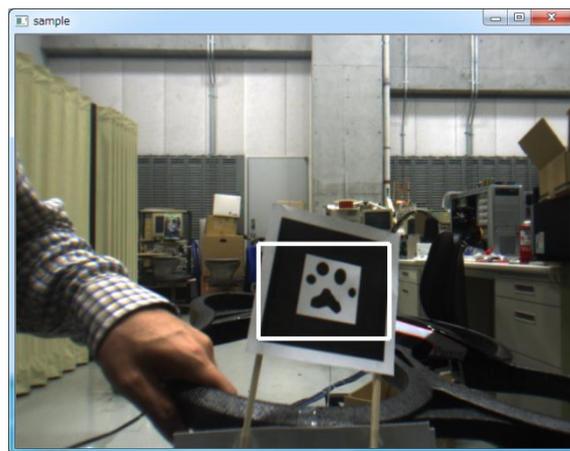


Fig.6 Example marker pattern of ARToolKit

## 5. 追跡システム

このシステムは前述した認識システムで認識した物体を追跡するシステムである。

### (1) 画像データ内における追跡

前述した認識システムでは、物体認識の処理時間が長いので猫を認識し追跡し続けることはできない。そのため、認識した猫を追跡するプログラムを新たに作成した。

この追跡プログラムにはパーティクルフィルタという時系列フィルタを使用する。特定物体認識に有効な特徴量 [8]によると時系列フィルタとは簡単に述べると未来予測のようなもので、ノイズを含む対象の観測値から対象の現在の状態を取得し、次の状態を予測して再度観測、そして予測した状態と観測した状態を評価することで予測精度を上げていく理論のことである。これを画像追跡に応用したものを使用する。この中でもカルマンフィルタとパーテ

ィクルフィルタなどが存在するが、今回はパーティクルフィルタを使用する。理由としてカルマンフィルタは状態推定する状態方程式を綿密に決定する必要が有り実装が難解である、それに対してパーティクルフィルタは尤度関数さえ定義できれば実装が簡易であるためこれを採用した。

パーティクルフィルタとは次の状態の予測にパーティクル(粒子)をしようする理論で、最初に画像内にパーティクルを一様に散布し、各パーティクルを設定した状態方程式に基づいて移動させる。その後観測した画像の状態と予測したパーティクルの状態の尤度を計算する。この尤度を重みとすることで、再度パーティクルを散布するときのあたりに散布するかを決定する。パーティクルフィルタが計算するのはパーティクルの分布によって表される確率密度であるのでどこが対象の位置かを決定するには重み付け平均を取ったり、最大重みを持つパーティクルを使ったりなどの方法を使用する必要がある。今回は正確に場所を特定したいため、目標位置にパーティクルが散布されていない可能性を考慮し重み付け平均で位置を決定する方法を採用した。Fig.7にパーティクルフィルタの説明図を示す。

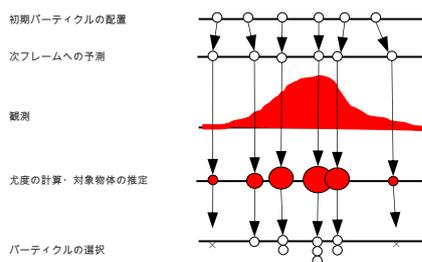


Fig.7 Explanation of Particle filter

尤度の計算には色ヒストグラムを使用する。ステレオカメラで取得した画像はBGRA画像(青, 緑, 赤, 透明度でデータが分けられた画像)なのでこれをHSV画像(色相, 彩度, 明度で分けられた画像)に変換し、色相と彩度のヒストグラムで尤度を計算する。BGRA画像で計算しない理由はBGRでヒストグラムを作成すると白で構成された画像と赤と緑と青で構成された画像が一致してしまうためであり、明度を計算しない理由は明度まで計算すると対象に影が掛かるだけで一致していないことになってしまうためである。

これらの画像処理にはOpenCVというオープンライブラリを使用する。OpenCVとはインテル社が開発・公開した無料で得られるコンピュータビジョン向けライブラリでこれを使用することで、複雑な画像処理を単純な関数のみで使用することができる。

現在の問題として使用する状態方程式がサンプルプログラムの単純なものなので追跡対象が不規則な動きをすると追跡できない点である。よって今後は動物の動きに合わせた状態方程式を立てる必要がある。

## (2) 実空間における追跡

これまでのシステムによりステレオカメラで計測した空間内のどこに猫が居るかをコントローラに認識させることができた。侵入した猫を撃退するにはこのデータを基にロボットで実際に猫を追跡する必要がある。

### a) 追跡ロボット

追跡ロボットには飛行ロボットAR.Droneを使用する。三次元的に空間を移動することができるため飛行ロボットを採用した。AR.DroneはフランスのParrot社が開発したSmart Phone等のタブレットデバイスを使い、Wi-Fi通信で操縦する4軸ヘリコプタのラジコンモデルである。Wi-Fi通信で操縦できるのでC++によるプログラミングで操縦することも可能である。Fig.8にAR.Droneを示す。このAR.Droneには様々なセンサが搭載されている。まず、鉛直下方向きに水平速度測定用のボトムカメラ(60fpsQVGA)や対地面の高度測定用超音波センサが搭載されており、内部のオンボードでボトムカメラ画像内のオプティカルフロー計算、超音波センサの高度計算を行うことで、安定したホバリングを実現している。他にも角速度を毎秒2000°測定可能な3軸ジャイロスコープや誤差50mgの3軸加速度計、6°の正確性を持つ磁気センサなどがあり、これらによって、通常のラジコンヘリよりも正確かつ精密な飛行をすることが出来る。さらに簡単な自動飛行プログラムや速度測定プログラムが内蔵されているため、C++プログラムによる自動制御飛行が可能である。



Fig.8 Flying robot, AR.Drone2.0(Parrot.inc)

### b) 追跡ロボットの位置追跡

前述したARToolKitを使用した認識を使用することでAR.Droneをコントローラに認識させることができた。しかし、このARToolKitも前述した認識システムと同様に画像内のパターンが小さいと認識することができないので、追跡はAR.Droneのオンボードから速度や角度の情報を得てそれを積分演算することで位置と姿勢を計測するデッドレコニングで行う。AR.Droneからの情報の取得やAR.Droneへの移動の指示を行うプログラムにはPuku's Laboratory[9]というサイトにあるオープンソースCV Droneを使用する。CV DroneはOpenCVとAR.Droneを組み合わせ、AR.Droneの操作や情報の取得といった操作を一つのクラスにまとめることでより簡易にAR.Droneを

C++環境下で操作できるようにしたプログラムである。

c) 猫の追跡

画像内の猫の位置追跡とロボットの位置追跡に基づいて実空間でロボットに猫を追跡させる。Fig.9 は三次元座標を持つ2つの物体間の単位ベクトルの計算方法を示したものである。図にあるような両者の三次元位置情報はステレオカメラで計測したものが使えるので式(5)によってステレオカメラから見た目標位置までの単位ベクトルを得ることができる。

$$l = \frac{1}{\sqrt{(x_2-x_1)^2+(y_2-y_1)^2+(z_2-z_1)^2}} \begin{bmatrix} x_2-x_1 \\ y_2-y_1 \\ z_2-z_1 \end{bmatrix} \quad (5)$$

この単位ベクトルの向きに AR.Drone が移動すれば目標点に到達することができる。しかし、この単位ベクトルは前述したとおりステレオカメラから見た単位ベクトルである。よってこの計算したベクトルを AR.Drone から見た単位ベクトルに変換する必要がある。

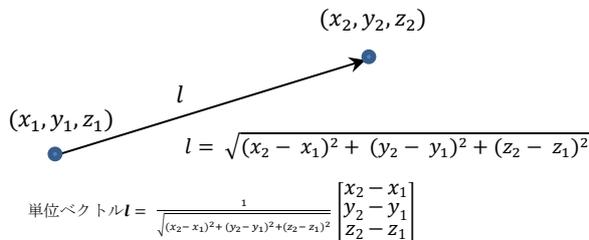


Fig.9 Unit vector

Fig.10 はロボット制御基礎論[10]を参考に作成した座標系の違う物体それぞれから見たベクトルの関係を示した図である。r<sup>A</sup>, r<sup>B</sup>はそれぞれ座標系 A から見た目標までのベクトルと座標系 B から見た目標までのベクトルを示している。<sup>A</sup>p<sub>B</sub>は座標系 A から見た座標系 A の原点O<sub>A</sub>から座標系 B の原点O<sub>B</sub>までのベクトルを示している。<sup>A</sup>R<sub>B</sub>は座標系 A から座標系 B への座標変換行列を示しており、<sup>B</sup>R<sub>A</sub>はその逆行列を示している。

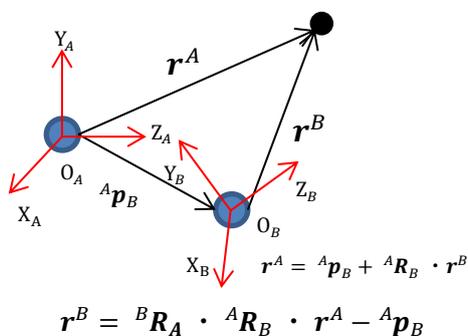


Fig.10 Relation between coordinate systems, A and B

この図に示した座標変換式で計算することで AR.Drone から見た単位ベクトルを得ることができる。式(6), (7)に

ベクトル変換式を示す。

$$r^A = {}^A p_B + {}^A R_B \cdot r^B \quad (6)$$

$$r^B = {}^B R_A \cdot r^A - {}^B R_A \cdot {}^A p_B \quad (7)$$

式(6)はベクトルr<sup>A</sup>をベクトル<sup>A</sup>p<sub>B</sub>とベクトルr<sup>B</sup>で表したもので、式(7)は式(6)からベクトルr<sup>B</sup>を導き出したものである。座標系 A をステレオカメラの座標系とし、座標系 B を AR.Drone の座標系とすることでこの 2 式からステレオカメラから見た単位ベクトルを AR.Drone から見た単位ベクトルに変換することができる。この計算はコントローラ上で行われ、この計算結果を元に AR.Drone を操作することで小動物の追跡を実現する。

6. 結論

本報告では、管理領域に侵入する小動物の被害を減らし、人と小動物の安全な生活を目的に研究を行った。その中で、飛行ロボットによる監視・撃退システムを検討してきた。その結果、全体的な基礎システムの構築と限定的な条件下における猫の追跡が達成できた。しかし、前述した各システムの欠点や制御用コントローラの制御プログラム等、様々な課題が数多く残っている。今後はそのような課題点をどう解決していくかということが研究の中心となる。

謝辞 本研究の遂行にあたり終始、御指導、御鞭撻下さいました、法政大学理工学部機械工学科 高島俊教授には、深く感謝の意を表すとともに、厚く御礼申し上げます。

参考文献

- 1) 福井春彦, 高島俊,他, “庭に進入する小動物を撃退するロボットについての基礎研究”,Robomec 2011 講演論文集,2011
- 2) 前川真吾,高島俊,他, “庭に進入する小動物を撃退するロボットについての基礎研究”,Robomec 2013 講演論文集, 2013
- 3) 著 Gary Bradski, Adrian Kaebler, 訳 松田晃一: 詳解 OpenCV コンピュータビジョンライブラリを使った画像処理・認識,オライリージャパン,pp423-443,2009
- 4) Ross B. Girshick’s Home Page, <http://www.cs.berkeley.edu/~rbg/>
- 5) 工学ナビ「ARToolKitによる拡張現実感プログラミング」, <http://kougaku-navi.net/ARToolKit/>
- 6) 俺 CG 屋, <http://www.cg-ya.net/>
- 7) The Sixwish project, <http://sixwish.jp/>
- 8) 山下隆義, 藤吉弘亘, "特定物体認識に有効な特徴量", 情報処理学会 研究報告 CVIM 165,2008
- 9) Puku’s Laboratory, <http://pukulab.blog.fc2.com/>
- 10) 吉川恒夫: ロボット制御基礎論, コロナ社,1988