

飛行ロボットの自律制御 -小型カメラとDeep learningによる自己位置推定-

○大山 裕通^{*1} 田中 康浩^{*2}
入江 寿弘^{*3} 新宮 清志^{*4}

キーワード：飛行ロボット 自律制御 慣性センサー カメラ

1. はじめに

飛行ロボットは上空から情報収集できるため、災害現場の調査、物資の輸送、環境や施設の監視など幅広い用途で需要がある。しかし、飛行ロボットを正確に操縦するためには熟練した操縦技術が必要である。そのため、ロボット自身が状況を判断し自律飛行することが求められている。現在運用されている飛行ロボットの多くは、慣性センサーとGPSを用いて自律制御を行っているが高層ビル群や建物内では、GPSの電波が乱反射を起こし、衛星が建物の陰になるため受信が難しく正確な制御は困難である。慣性センサーにより加速度・角速度を測定し姿勢や軌道を求めることができるが、加速度計や角速度計にはバイアス・トレンドが伴い、長時間にわたって積分を行うと誤差が蓄積され精度が落ちてしまう^[1]。本研究では慣性センサーの精度補正と障害物認識のため、小型カメラを併用する飛行ロボットの制御方法の検討を行う。慣性センサーから得る加速度情報を積分し速度としたものと、カメラの画像から得る内部パラメータより算出した位置を微分し速度としたものとを比較し調整を行う事で慣性センサーの誤差・外乱の精度補正を行う。また、ニューラルネットワークに景色を学習させることで、カメラから得られた景色によって自己位置の推定と対象物の判別が可能かどうか試みる。

2. 小型カメラによるカメラ移動量の推定

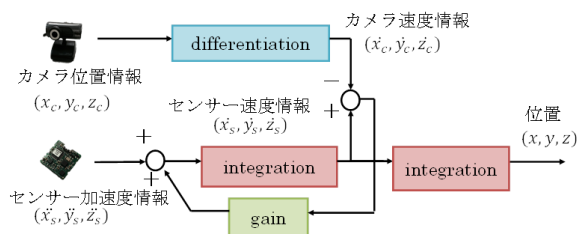


図-1 カメラと慣性センサーによる自己位置推定

慣性センサーの加速度計から得られる加速度を2回積分することで位置を算出できるが、積分を行うと誤差が

蓄積されるため正確な自己位置推定ができない。そこで、図-1の通り加速度計から得た加速度を積分し速度としたものとカメラの画像情報より算出した位置を微分し速度としたものとを比較することで誤差・外乱の明確化を考える。その後、誤差・外乱を取り除き得た速度をもう一度積分することでより正確な位置情報を取得することが可能である。以上のことを行うには、カメラでの自己位置推定が行なえていることが大前提なので、カメラを用いての自己位置推定を試みる。

2.1 カメラモデル

カメラで撮影する時、カメラの位置や姿勢によって撮影される写真は変わる。また、カメラの位置や姿勢が同じでもカメラやレンズが異なると同様に写真は変わってしまう。したがって、カメラの性質を決定するカメラキャリブレーションが重要となる。ここでは、ピンホールカメラモデルに基づくカメラキャリブレーション(以下キャリブレーション)を取り扱うこととする。カメラの位置・姿勢や特性を表すパラメータは次の通りである。外部パラメータ

- ・回転(3次元空間におけるレンズ光軸ベクトルの向き)
- ・平行移動(3次元空間におけるレンズ中心の位置座標)

内部パラメータ

- ・焦点距離(レンズ中心から投影画像平面までの距離)
- ・レンズ歪み係数(レンズによる像の歪み)
- ・画像主点位置(レンズ中心を通り、投影画像平面に垂直に交わる光線と、投影画像平面との交点)
- ・画素サイズ(画素の縦および横方向の大きさ)

2.2 ピンホールカメラモデルによる

画像平面への射影

まず、世界座標系における点 $P(x_w, y_w, z_w)$ と、画像座標系における点 P の投影点 $p(u_{ud}, v_{ud})$ をそれぞれ同次座標表現 $P=[x_w \ y_w \ z_w \ 1]^T$ 及び $p=[u \ v \ 1]^T$ で表し、 P と p の関係を(1)式で表す。

$$sp = A[R|T]P \quad (1)$$

ここで s は画像のスケール係数, $[R|T]$ はカメラの外部パラメータ行列, A はカメラの内部パラメータ行列である. 外部パラメータは回転 R と平行移動 T から成り, それぞれ

$$R = \begin{bmatrix} C\theta_z & -S\theta_z & 0 \\ S\theta_z & C\theta_z & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} C\theta_y & 0 & S\theta_y \\ 0 & 1 & 0 \\ -S\theta_y & 0 & C\theta_y \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & C\theta_x & -S\theta_x \\ 0 & S\theta_x & C\theta_x \end{bmatrix} \\ = \begin{bmatrix} \gamma_1 & \gamma_2 & \gamma_3 \\ \gamma_4 & \gamma_5 & \gamma_6 \\ \gamma_7 & \gamma_8 & \gamma_9 \end{bmatrix} \\ T = \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix} \quad (2)$$

と表せる. ただし, $C = \cos$, $S = \sin$ とする. 内部パラメータ行列 A は,

$$A = \begin{bmatrix} \alpha_u & \gamma & u_0 \\ 0 & \alpha_v & v_0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} fs_u & fk_s & u_0 \\ 0 & fs_v & v_0 \\ 0 & 0 & 1 \end{bmatrix} \quad (3)$$

と表す事が出来る.

ここで, f は焦点距離, (s_u, s_v) はそれぞれ u 軸及び v 軸方向画素の大きさ, (a_u, a_v) は画像主点位置, k_s は画像平面における u 軸に対する v の傾きを表すせん断係数である. 次に, 実際に世界座標系の点 P が, デジタル画像座標系 p_0 に射影される過程を述べる. それぞれの座標系の対応関係を図-2に示す.

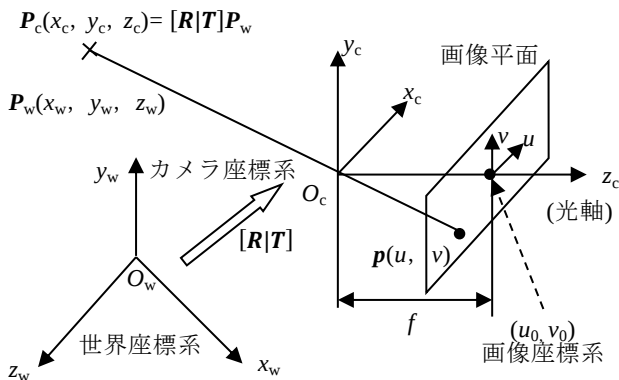


図-2 ピンホールカメラモデルで取り扱う座標系の関係

(1) 世界座標系からカメラ座標系への変換

世界座標系で表わされる点 $P(x_w, y_w, z_w)$ を, カメラ座標系 $P_c(x_c, y_c, z_c)$ に変換する. 変換は外部パラメータ $[R|T]$ を用い以下のように表わす.

$$\begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} = [R|T] \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix} \quad (4)$$

(2) 画像平面上への射影

カメラ座標系で表現された点 P_c を, ピンホールカメラモデルに従い画像平面に射影する. このとき, 射影された画像平面上での座標値 $p(u, v)$ はそれぞれ以下のよう求められる^[2].

$$u = f \frac{x_c}{z_c} + u_0 \quad (5)$$

$$v = f \frac{y_c}{z_c} + v_0 \quad (6)$$

(3) カメラ移動量の推定

本研究では2次元画像上の特徴点の移動量から自己位置の推定を行なう. カメラで撮影した2次元画像の複数の特徴点を取得し, 動画のフレーム間の特徴点の2次元位置の変化を元に, カメラの3次元座標での移動量の推定を行う. カメラの角速度および加速度は慣性センサーより検出できるので, 動画から得られる姿勢角と移動量の情報と合わせて特徴点の3次元位置情報とカメラの位置, 姿勢角の推定を行う方法について検討した.

初期状態でカメラの捉えている特徴点の奥行き方向の距離は未知であるが, 任意の距離 L の初期値を設定し, 新たな2次元画像とセンサーの姿勢情報を得るたびに更新を行い, 逐次特徴点の3次元座標とカメラ姿勢角をより正確な値とする事が可能と考えられる.

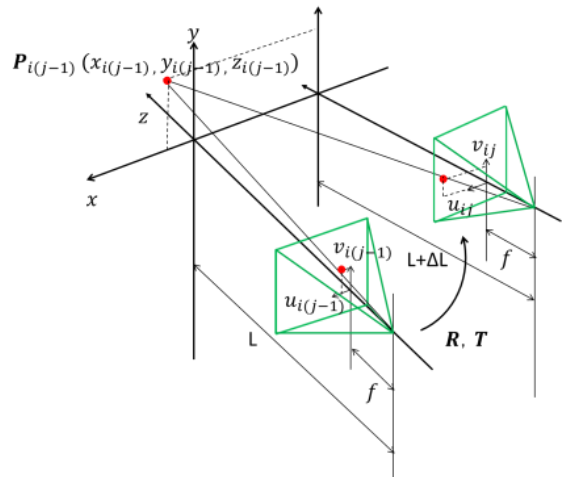


図-3 2次元座標系と世界座標系の関係

図-3より式(7), (8)が成り立つ.

$$\frac{L}{f} = \frac{x_{i(j-1)}}{u_{i(j-1)}} = \frac{y_{i(j-1)}}{v_{i(j-1)}} \quad (7)$$

$$\frac{L+\Delta L}{f} = \frac{x_{ij}}{u_{ij}} = \frac{y_{ij}}{v_{ij}} \quad (8)$$

まず, 2次元座標 $\mathbf{p}_{i(j-1)}(u_{i(j-1)}, v_{i(j-1)})$ と距離 L を元に, フレーム $j-1$ のカメラ座標における特徴点 $\mathbf{P}_{i(j-1)}(x_{i(j-1)}, y_{i(j-1)}, z_{i(j-1)})$ の3次元位置を式(7), (8)より求める.

$$\begin{bmatrix} \gamma_1 & \gamma_2 & \gamma_3 & t_x \\ \gamma_4 & \gamma_5 & \gamma_6 & t_y \\ \gamma_7 & \gamma_8 & \gamma_9 & t_z \end{bmatrix} \begin{bmatrix} x_{i(j-1)} \\ y_{i(j-1)} \\ z_{i(j-1)} \\ 1 \end{bmatrix} = \begin{bmatrix} x'_{ij} \\ y'_{ij} \\ z'_{ij} \end{bmatrix} \quad (9)$$

(9)式より任意の回転・並進を仮定しフレーム j のカメラ座標 $(x'_{ij}, y'_{ij}, z'_{ij})$ を推定する.

(5), (6)式より2次元座標は

$$u'_{ij} = f \frac{x'_{ij}}{z'_{ij}} + u_0 \quad (10)$$

$$v'_{ij} = f \frac{y'_{ij}}{z'_{ij}} + v_0 \quad (11)$$

式(10), (11)となり, フレーム j の時の2次元座標 (u'_{ij}, v'_{ij}) を求める.

$$f(\theta_x, \theta_y, \theta_z, t_x, t_y, t_z) = \sum_{i=0}^n \{(u'_{ij} - u_{i(j-1)})^2 + (v'_{ij} - v_{i(j-1)})^2\} \quad (12)$$

n : 2次元画像の中の特徴点の数

現在のフレーム j とフレーム $j-1$ の2次元座標を式(12)で評価し, 回転・並進を微小量ずつ逐次更新して, 最適な回転・並進を推定する. 評価式は現在の2次元画像中のすべての特徴点 n に対して一致する時の回転・並進を得たとき最少となる様に設定している. 実際にはセンサーの計測値と合わせて推定を行う予定である.

本研究ではABCアルゴリズムやPSOアルゴリズムなど群知能アルゴリズム等を用いて, 評価式(12)が最小になるときの各フレーム間の姿勢角および移動量の推定を行う. 動画像を元に逐次最新の画像に更新し, カメラの移動の軌跡を求めることが出来る.

3. ディープラーニングによる自己位置の推定

撮像物判別, 自己位置推定の補助としてディープラーニングを用いる. ディープラーニングとは, ニューラルネットワーク (以下NN) の重みの層が多層になっている

ものである. NNとは, 人間の神経回路を模した数学モデルである. 正解とするデータ (教師データ) を用意したうえで, NNにある入力が入ると重みを加え特徴を抽出して出力を出す. この出力と正解データとの誤差をもとに重みを調整し直すことで識別度を上げることが可能となる. つまり学習が可能となる. 著者は, NNに景色の画像を学習させることで入力された景色に対して学習した複数の画像の中から近い景色の画像を抽出できるのではないかと考えた. ここでは, ディープラーニングでの画像問題を得意とする畳込みニューラルネットワーク (Convolutional Neural Network, CNN) を使用した^[3].

3.1 ニューラルネットワーク

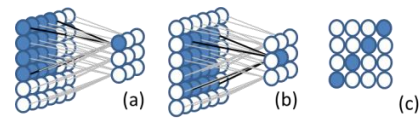


図-4 単層ニューラルネットワーク

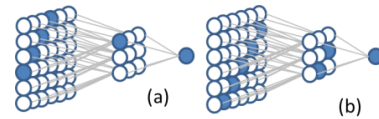


図-5 多層ニューラルネットワーク

図-4に示す単層ネットワークでは, 左層が入力で右層が出力である. 各層は2次元的に配置され(a), (b)のように右層のユニットは左層の4x4のユニット群とのみ結合を持ち, (c)のような特定パターンが入力されると活性化する. 一方, 図-5の多層ネットワークでは右層は, 中間層の3x3のユニット群と結合を持ち, このユニット内1つでも活性化すると自身も活性化する. このようにNNは, ニューロン毎, 活性化関数を持っており, 入力→出力で得たそれら誤差を逆方向に伝播し調整を行う. これを誤差逆伝播法と呼び, NNの学習である^[4].

3.2 自己位置の推定

著者は, ある空間内において多方向から撮影し景色をNNに学習させることで, ロボット自身が空間内のどこにいるか判別することができ, 自己位置の推定が可能か検討を行った. 学習に用いる画像は研究室内で撮影した.

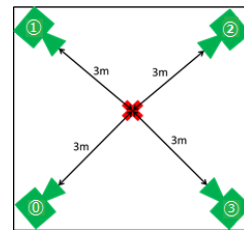


図-6 実験概略図

図-6に示した通り，研究室にある地点を定め，そこから3m離れた①の位置にカメラを設置した．その場所から左右・上下方向に0.3m水平移動させながら動画を撮影し，キャプチャソフトで毎秒4回の間隔で静止画に編集した．次にカメラを①，②，③の位置で同様に撮影した．0：カメラ0，1：カメラ1，2：カメラ2，3：カメラ3のようにカテゴリを作成し，それぞれの判別を行うNNを構築した．ネットワークの構成は図-8のように畳込み層→プーリング層のペア3層に全結合層の組み合わせである．学習パラメータは表1の通りである．教師用データはそれぞれ2000枚ずつ計8000枚格納し，評価データは800枚格納した．

表1 学習パラメータ

Batchサイズ	100 枚
テスト間隔	100 iteration
学習率	0.0001
教師データ	8000 枚
評価データ	800 枚
学習iteration数	5000 回

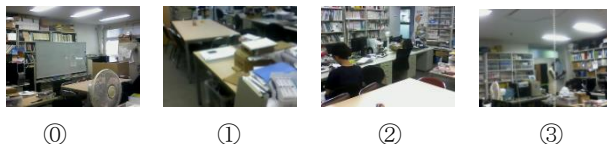


図-7 サンプル教師データ

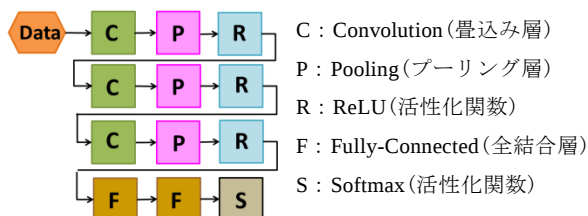


図-8 ネットワークの構成

3.3 判別結果

図-9に教師用データ，評価用データ以外の判別用データを入力した結果を示す．

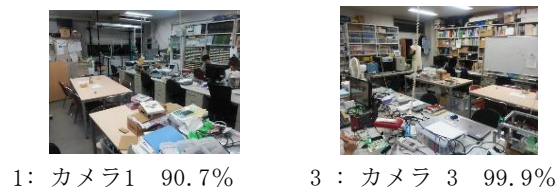


図-9 判別の一例

表2にカメラ位置が①，②，③付近である判別用画像を入力したときのそれぞれの判別結果を示す．

表2 判別結果

		判別用画像の位置			
		①	②	③	
カメラ	0	0.1 %	0.0 %	0.0 %	0.0 %
	1	6.6 %	90.7 %	0.0 %	0.0 %
	2	69.9 %	0.0 %	1.0 %	0.0 %
	3	23.4 %	9.3 %	99.0 %	99.9 %

4. まとめ

2次元画像中の特徴点の移動量からカメラの移動量の推定を試みた．今後はABCアルゴリズムやPSOアルゴリズムなど群知能アルゴリズムを用い，最適解を求める予定である．その上でセンサーからの情報を組み合わせ，カメラの移動量の推定精度を向上させる予定である．

また，CNNを用いて決められた空間内での自己位置の推定を可能とした．しかし，現状では判別用データの種類や解像度によっては，位置の推定を誤ってしまうこともある．精度を高めるため，さらに多くの教師データを用い新たなNNを構築する必要がある．

本研究では4方向の位置を判別したが，今後はより多方向の位置におけるデータをNNに学習させ，自己位置の推定を可能とする予定である．そして，慣性センサーとカメラ，NNを組み合わせた飛行ロボットの自律飛行を実現する予定である．

【参考文献】

- 1) 大山裕通，黒田 晟一郎，入江 寿弘，新宮 清志：カメラと慣性センサーを用いた飛行ロボットの研究，第21回計算工学講演会論文集，日本計算工学会，Vol.21，2016.
- 2) OpenCVプログラミングブック制作チーム，OpenCV2プログラミングブック，マイナビ，pp.177-184，2011.
- 3) 荒木雅弘：フリーソフトではじめる機械学習入門，講談社，pp.2-18，2014.
- 4) 岡谷貴之：深層学習，森北出版，pp.79-110，

- *1 日本大学大学院 理工学研究科 精密機械工学専攻 院生
- *2 日本大学 理工学部 精密機械工学科 学部生
- *3 日本大学 理工学部 精密機械工学科 教授 博士(工学)
- *4 日本大学 名誉教授 工学博士，総合資格学院 顧問