

小型自律飛行ロボットを用いた見守りシステムの研究 ～対象人物の位置推定と追従制御～

横田 恵助^{†1}, 正 大矢 晃久^{‡1}

Children Watching System Using a Small UAV -Position Estimation And Following Control of a Target Person-

*Keisuke YOKOTA^{‡1}, Akihisa OHYA^{‡1}

Abstract— In this study, we are aiming to develop a children watching system using a small UAV. As necessary functions, we discuss detection of a targeted person, position estimation of the person and following control of the UAV, in this report. The target person wearing a colored helmet is detected from the image taken on the UAV. Distance and direction to the person are calculated from the position and size of the helmet in the image. Some experimental results are also shown.

Keywords: UAV, Child Watching System, Human Tracking

1. 緒言

日本の少子化が進むにつれ、次代を担う子どもたちの命を守ることが我々の使命の1つとなっている。しかし近年、子どもが交通事故や犯罪に巻き込まれるケースが多発しており、子どもの安全が脅かされている。特に、6～12歳の歩行者の交通事故による死者が平成21年から増加し続けており、その約4割が登下校中に発生した事故である[1]。また、平成23年には小学生の犯罪被害件数が約2万件に上っており[2]、1日に約57人の小学生が狙われている計算になる。子どもが1人で登下校する場合、子どもの不注意を監視する人がいないため、飛び出しを始めとした交通事故や、連れ去りや声かけ、恐喝などの事件が発生しやすい。子どもの死傷者を減らすためには、こうした交通事故や事件の遭遇をできる限り回避するよう対策を講じる必要がある。

現在各地域で行われている安全対策として、集団登下校や、保護者もしくはボランティアによる見守りなどがある。集団登下校の場合、連帯行動により事故や事件に巻き込まれる危険性は大きく減る一方、列が乱れることによる通行の妨げや、その他子ども同士の対人関係といった新たな問題が発生する[3]。最近では、栃木・鹿沼市クレーン車事故(2011年4月18日)や亀岡市登校中児童ら交通事故死事件(2012年4月23日)など、集団登校中の小学生の列に車が突っ込み、複数人が同時に死亡・重軽傷を負う事例も見られる。また、保護者やボランティアによる見守りについても、活動できる時間・場所に限界があったり、目配りが難しいなどの問題が起きている[3]。このように、有人による安全対策では、交通事故や事件の件数は減らせるものの、別の新たな問題が数多く発生する。こうした報告から、筆者らは無人による見守りシステムを構築することにより、問題をよりスマートに解決できるのではないかと考えた。

本研究では1つのアプローチとして、小型自律飛行ロボットを用いた見守りシステムの構築を目指す。本稿ではその1つの機能として、ロボットに取り付けたカメラにより見守り対象者を認識し、追跡するシステムについて議論する。

2. 人間追跡システム

2.1 概要

無人による見守りシステムを実現するには、見守り対象者を認識するためのイメージセンサが不可欠である。本研究では、飛行ロボット前面にカメラを取り付け、得られる視覚情報をもとに対象者を発見し、追従させる。この際、対象者にマーカーを付けることにより、視覚情報からの対象者の識別がより単純化できる。本研究では小学生が登下校時に単色の帽子をかぶっていることに着目し、この帽子の色と形状を手がかりに対象者の発見を試みることにした。なお、本研究ではマーカーの色をその場の環境に応じて適切な色に変更できるように、帽子の代わりにヘルメットを用い、その上に単色の布をかぶせた。ヘルメット単体を画像に収めると楕円に近似でき、距離の大小によってその楕円の面積が変化する。これを利用し、画像上での近似楕円の位置とその面積より、ヘルメットをかぶっている人間の飛行ロボットから見た3次元位置を求める。

飛行ロボット操作による追従の手法としては、ロボットの高さとヘルメットまでの距離がほぼ一定となるように速度制御をかける。この時、図1のように斜め後ろから追従させることで、万が一飛行ロボットが墜落した時の怪我を予防する。

2.2 ヘルメット検出アルゴリズム

前面カメラの画像からヘルメットを検出し、画像上でのヘルメットの位置とその大きさを求めるアルゴリズムを次のように実装した。この時、画像処理ライブラリとしてOpenCVを用いた。なお、ここではヘルメットにかぶせる布の色はピンクであるという前提で話を進める。

- 1) 元の画像をHSV変換し、H成分とS成分に分解する。
- 2) H成分のうち300～360°、S成分のうち0.25～1.00の部分をそれぞれ2値画像として抽出し、2つの2値画像の共通部分を抽出する。
- 3) 得られた2値画像から、輪郭をすべて抽出し、楕円フィッティングする。
- 4) 検出された楕円群の中から、真円に近く、かつ面積の大きいものをヘルメットを示す楕円とする。具体的には、評価関数を(面積)×(短径÷長径)²と定め、この値が最も大きい楕円をヘルメット楕円として扱う。

^{†1} 筑波大学 大学院 システム情報工学研究科

^{‡1} Graduate School of Systems and Information Engineering,
University of Tsukuba

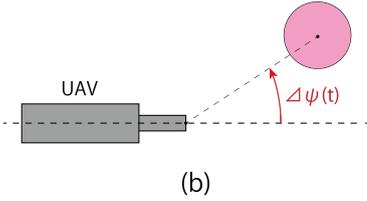
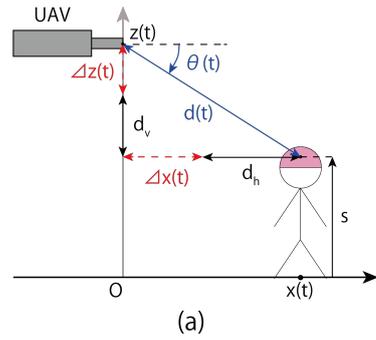


Fig. 1 Abstract of AR.Drone control

5) ヘルメット楕円の中心座標と面積を取得する。

このアルゴリズムを屋内にて適用すると、図2の画像から図3のようにヘルメットの近似楕円を取り出すことができる。



Fig. 2 Helmet (indoor)

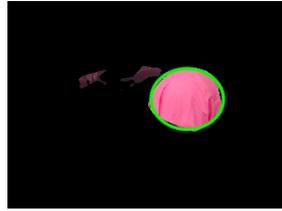


Fig. 3 Detected helmet (indoor)

2.3 前面カメラから見たヘルメットの3次元位置の導出

ロボットの高さとヘルメットまでの距離を一定に保つためには、ロボットとヘルメットとの位置関係を知る必要がある。そのためには、図1中のうち、距離 $d(t)$ 、俯角 $\theta(t)$ 、相対方位角 $\Delta\psi(t)$ が分かればよい。これら3つのパラメータを以下のアルゴリズムで求める。

1) 仮想スクリーンの対角線を求める

カメラの前方 d [m] の位置に、カメラ全体に写る仮想的なスクリーン S_d を考え、このスクリーンにカメラ画像やその近似楕円 E が描画されるものとする [図4]。カメラの対角線画角を α とすると、画角と仮想スクリーンの関係は、図4(b)より以下の式で対角線の長さが求まる。

$$D = 2d \tan\left(\frac{\alpha}{2}\right) \quad (1)$$

2) ヘルメット楕円の中心座標を長さに変換する

ピクセル座標を仮想スクリーン上でのメートル座標に変換し、図4(a)の $x'y'$ 座標系に以下の式で貼り直す。ここで、仮想スクリーンの幅・高さ・対角線の長さをピクセル

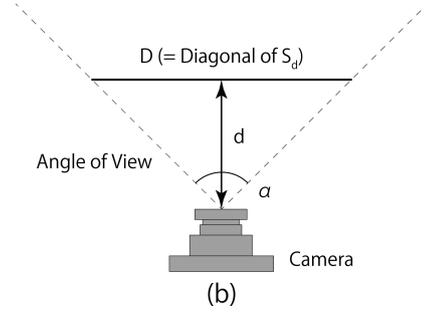
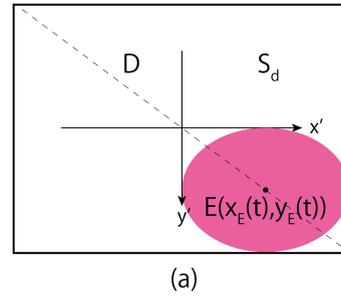


Fig. 4 Virtual screen S_d

数で表したものをそれぞれ W_p, H_p, D_p とする。

$$x_E(t) = \frac{D}{D_p} \left(\text{近似楕円の } x \text{ 座標} - \frac{W_p}{2} \right) \quad (2)$$

$$y_E(t) = \frac{D}{D_p} \left(\text{近似楕円の } y \text{ 座標} - \frac{H_p}{2} \right) \quad (3)$$

3) 俯角 $\theta(t)$ ・相対方位角 $\Delta\psi(t)$ を求める

前面カメラから見たヘルメットの俯角 $\theta(t)$ と相対方位角 $\Delta\psi(t)$ は以下の式で求められる。

$$\theta(t) = \tan^{-1} \left(\frac{y_E(t)}{d} \right) \quad (4)$$

$$\Delta\psi(t) = \tan^{-1} \left(-\frac{x_E(t)}{d} \right) \quad (5)$$

4) カメラとヘルメットとの距離 $d(t)$ を求める

まず、ヘルメットの断面積 a_0 [m^2] を予め求めておく。ここでいう断面積とは、ヘルメットの中身を詰め、斜めにスライスしてできる近似楕円の面積のことである。

次に、 S_d に貼り付けてある楕円 E の面積 $a_E(t)$ を、公式 $\pi \times (\text{長径} \div 2) \times (\text{短径} \div 2)$ を用いて求める。またカメラから楕円 E の中心までの距離 $d_E(t)$ を以下の式で求める。

$$d_E(t) = \frac{1}{\cos \Delta\psi(t) \cos \theta(t)} = \sqrt{1 + x_E(t)^2 + y_E(t)^2} \quad (6)$$

最後に、 a_0 と $a_E(t)$ の面積比を使って、カメラとヘルメットの距離を求める。 $d'(t) = d(t) \cos \Delta\psi(t) \cos \theta(t)$ として、カメラから距離 $d'(t)$ の位置に新たにスクリーン S' を設置し、仮想スクリーンの楕円 E を透視投影させると、スクリーンに映し出される楕円 (楕円 E' とする) の面積は a_0 と等しくなるはずである [図5]。この性質を利用すると、 $a_E : a_0 = 1 : d'(t)^2 = d_E(t)^2 : d(t)^2$ の関係が得られ、これと $d(t) > 0$ より、距離 $d(t)$ は以下のように求めることができる。

$$d(t) = d_E(t) \sqrt{\frac{a_0}{a_E(t)}} = \sqrt{\frac{a_0}{a_E(t)}} (1 + x_E(t)^2 + y_E(t)^2) \quad (7)$$

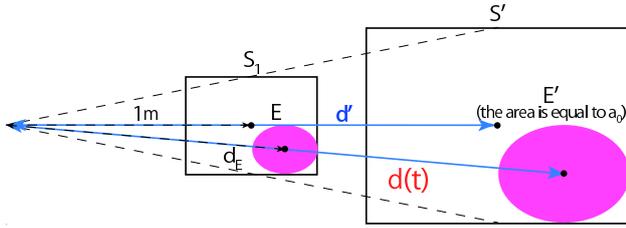


Fig. 5 Correspondence of the area ratio of the ellipse and the distance between the camera and the ellipse

2.4 速度制御による追従アルゴリズム

人間の地面から頭部までの高さを s としたとき、ロボットが常に人間のほうを向き、後方 d_h 、高さ $s + d_v$ の位置を維持するようにする。つまり図 1 において、 $\Delta x(t), \Delta z(t), \Delta \psi(t)$ をそれぞれ 0 にするような制御を考える。 $\Delta x(t), \Delta z(t)$ に関しては、上で求めた $d(t), \theta(t)$ より求められる。PID 制御による速度制御を考えるとき、前後方向、上下方向、yaw 方向の各入力速度・角速度を $u_x(t), u_z(t), u_\psi(t)$ とし、以下のような式を立てることができる。

$$u_x(t) = K_x^p \left\{ \Delta x(t) + \frac{1}{T_x^i} \int \Delta x(t) dt + T_x^d \frac{d(\Delta x(t))}{dt} \right\} \quad (8)$$

$$u_z(t) = K_z^p \left\{ \Delta z(t) + \frac{1}{T_z^i} \int \Delta z(t) dt + T_z^d \frac{d(\Delta z(t))}{dt} \right\} \quad (9)$$

$$u_\psi(t) = K_\psi^p \left\{ \Delta \psi(t) + \frac{1}{T_\psi^i} \int \Delta \psi(t) dt + T_\psi^d \frac{d(\Delta \psi(t))}{dt} \right\} \quad (10)$$

ここで、パラメータ K^p は比例ゲイン、 T^i は積分時間、 T^d は微分時間である。この 3 つのパラメータは、安定制御になるまで値を変えつつ繰り返し実験によって求める。

3. 関連研究

小型自律飛行ロボットに対する視覚フィードバック制御は、現在盛んに研究されている。Altug らは、小型自律ロボットに取り付けられたカメラと地面に設置されたカメラの 2 つを用いて位置を推定する手法を提案し、実験によりその有効性を示した [4]。Azrad らはカルマンフィルタと提案する画像トラッキングアルゴリズムにより 1 つのカメラで位置を推定し、その有効性を屋外での実験により示している [5]。また鶴尾らは、飽和型三角関数を用いることで、速度入力が高時に引き起こす望ましくない摂動を抑制し、その手法による制御則の安定性を証明した [6]。

しかしながら、これらの研究ではロボットが到達すべき目標地点が動かないことを前提に議論が進められている。本研究は歩行者を観測対象とし、見守りという目的の達成に主眼を置いている点において、これまでの研究とは異なる。

4. 実験システムの構成

4.1 AR.Drone

クアッドロータは、4 つの回転翼を持ち、飛行ロボットの中でも特に安定性が高いとされ、近年 UAV (無人航空機) の

研究分野において注目を集めている。本研究では、小型クアッドロータの中でも比較的安価で普及率の高い、Parrot 社の AR.Drone [図 6] を使用する。AR.Drone には、ジャイロセンサや加速度センサが搭載されており、機体の姿勢や速度を知ることができる。また、機体の先端に小型の CMOS イメージセンサ (カメラ) が搭載されている。カメラから得られるストリームデータのピクセルサイズは QVGA (320 × 240) で、圧縮方式は UVLC (MJPEG-like) である。

AR.Drone にはコマンドによって機体を制御するための CPU が搭載されている。離陸・着陸やホバリング、速度変更などを、専用のコマンドを送ることによって実現できるため、姿勢制御・速度制御が非常に簡単に行える。



Fig. 6 AR.Drone 1.0

4.2 ノート PC

AR.Drone との無線通信と、本研究で使用するプログラムの実装および実行のために使用する。本研究で使用するプログラムはすべて ROS と呼ばれるフレームワーク上で稼働し、それぞれのノード (プロセス) は、roscore というプログラムを通してデータをやり取りする。

4.3 PC - AR.Drone 間の通信方法

PC と AR.Drone は、Wi-Fi による無線通信によりデータをやり取りする。AR.Drone との接続が確立すると、AR.Drone は Navdata と呼ばれるデータを約 5ms 周期で PC に送る。この Navdata には、AR.Drone の姿勢を示す roll 角 $\phi(t)$ 、pitch 角 $\theta(t)$ 、yaw 角 $\psi(t)$ 、x (前後) 方向の速度 $v_x(t)$ 、y (左右) 方向の速度 $v_y(t)$ が含まれる。また、それと同時に、前面カメラの映像を 15fps で PC に送る。離陸、着陸、速度変更などを行う場合は、それぞれ対応する AT コマンドを AR.Drone に送る。速度変更に必要なパラメータ (目標速度・入力速度) は、x 方向の速度 $u_x(t)$ 、y 方向の速度 $u_y(t)$ 、z 方向の速度 $u_z(t)$ 、および yaw 方向の速度 $u_\psi(t)$ である。

5. 検証実験

5.1 ヘルメット検出アルゴリズム

ピンクと黄色のヘルメットをそれぞれ大学キャンパス内の路面上に置き、距離を 3m ほど離れた上で本アルゴリズムを適用した。このとき、カメラ画像のフレームごとに正検出か、誤検出か、あるいは未検出かを目で見分けて判断した。ここでいう正検出・誤検出・未検出とはそれぞれ、楕円がヘルメットを覆うこと [図 7]、楕円がヘルメットを覆わないこと [図 8]、ヘルメット楕円が求められないことである。この判断を 1,000 フレームに対して行い、各色のヘルメットでの正検出率・誤検出率・未検出率を求めた結果が表 1 である。

ピンクのヘルメットに関しては、黄色に比べて比較的高い正

Table 1 Result of helmet detection experiment

ヘルメットの色	正検出率	誤検出率	未検出率
ピンク	98.5%	0.3%	1.2%
黄色	82.6%	17.2%	0.2%

検出率を出していることが分かる。また誤検出が起こる原因の1つとして、図8のように建物の壁が誤って認識されてしまうことがピンク・黄色両方において確認された。本アルゴリズムの各パラメータやアルゴリズムそのものを見直し、正検出率を上げるように改良していくことが今後の課題である。



Fig. 7 Detected helmet (outdoor)



Fig. 8 Mis-detected helmet

5.2 ヘルメットの3次元位置の導出

前面カメラからヘルメットまでの距離 d と、俯角 θ_v がどの程度正しく求められるか、検証実験をおこなった。実験方法としては、図9のようにカメラを固定し、ヘルメットを動かして x の値を 0.5~5m 間で 0.5m ずつ段階的に変化させる。各 x ごとに 60 フレーム分の画像データを取得し、 a_0 を適当において本アルゴリズムを適用し、俯角 θ_v (単位は $^\circ$) と距離 d を求めた。実験結果を図11に示す。

図10の曲線は、図9より求めた俯角 θ_v の式、 $\theta_v = \frac{180^\circ}{\pi} \tan^{-1} \left(\frac{0.6}{x} \right) - 30^\circ$ のグラフ (理想曲線) である。また、図11の曲線は、図9より求めた距離 d の式、 $d = \sqrt{x^2 + 0.6^2}$ のグラフ (理想曲線) である。

図10を見ると、本アルゴリズムにより、理想曲線に極めて近い値が求められていることが分かる。よって、前面カメラから見たヘルメットの俯角についてはおよそ正しく求められることがわかる。

次に、図11を見てみると、点は理想曲線の近くにあるものの、 x が大きくなるにしたがって点が分散していることが分かる。分散そのものに関しては、カメラやヘルメットは固定され、検出処理は各フレームごとに独立しているので、照明の変化や、画像の圧縮処理による劣化などの影響が考えられる。また、ヘルメットが遠い位置にあるほど、面積に対する1ピクセルの割合が大きくなるため、距離の振れ幅が大きくなると考えられる ($x = 5$ の時に分散が著しく大きいのは、ヘルメット全体がうまく検出できなかったことも原因の1つである)。しかし、ヘルメットとの距離が近くなればその誤差も縮まるので、ヘルメット検出アルゴリズムの精度を向上させ、ヘルメットの断面積 a_0 の値を調節すれば、ヘルメット間距離も本アルゴリズムによりおよそ正しく求められると考える。

6. 結言

本稿では、見守りシステムに必要な機能の1つとして、小型自律飛行ロボットを用いた対象人物の位置推定およびその追

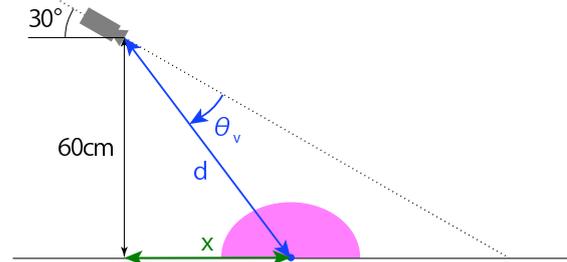


Fig. 9 The method of verification experiment of the helmet positioning algorithm

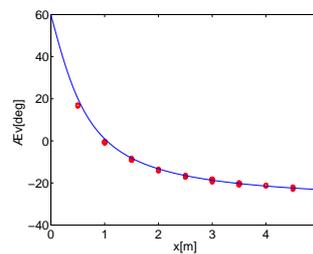


Fig. 10 Relationship between x and θ_v for the verification experiment

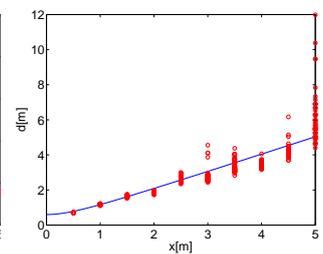


Fig. 11 Relationship between x and d for the verification experiment

従制御法について議論した。位置推定の手法として、ロボット前方に取り付けられたカメラから対象人物にかぶせた色付きのヘルメットを認識し、近似楕円のカメラ上での位置と大きさから対象人物の3次元位置を求め、その有効性を示した。また追従制御法として、前後方向、上下方向、yaw方向の目標速度および目標角速度をPID制御により求め、対象人物と一定の距離を保って追従する手法を示した。今後の予定として、本手法の改良をしつつ、追従制御を実装し、実機による自律的な人間追従の検証を行っていく。

参考文献

- [1] 公益財団法人交通事故総合分析センター: "通学時の交通事故の特性と対策", pp. 8-9, August 8, 2012.
- [2] 警視庁: "平成23年の犯罪情勢", pp. 119-122, 2012.
- [3] 学校たんけん隊 - 集団登下校の現状と問題点: <http://www.5d.biglobe.ne.jp/tanken/danger/dad-gro.htm>, Jan 7, 2013.
- [4] E. Altug, J. P. Ostrowski and C. J. Taylor: "Control of a Quadrotor Helicopter Using Dual Camera Visual Feedback", The International Journal of Robotics Research, 24- 5, 329/341, 2005.
- [5] S. Azrad, F. Kendoul and K. Nonami: "Visual Servoing of Quadrotor Micro-Air Vehicle Using Color-Based Tracking Algorithm", Journal of System Design and Dynamics, 4-2, 255/268, 2010.
- [6] 鶴尾 有生, 河合 宏之, 村尾 俊幸: "入力飽和を考慮した動的視覚オブザーバによる小型自律飛行ロボットの位置姿勢制御", 第12回計測自動制御学会システムインテグレーション部門講演会 (SI2011), pp. 951-954(1M2-3), 2011.