

東京大学大学院新領域創成科学研究科  
社会文化環境学専攻

2015 年度  
修 士 論 文

飛行型プラットフォームによる  
移動物体のリアルタイム識別手法に関する研究  
Study on a real-time method to detect moving objects  
from flying platform-borne monocular images

2016 年 1 月 25 日提出  
指導教員 柴崎 亮介 教授

木 村 元 紀  
Kimura, Motoki

## 論文要旨

近年、小型の無人航空機（UAV）が市場に出回るようになった。UAV は、従来のバルーンなどのプラットフォームに加えて、空撮のプラットフォームとして利用され、上空からの広範囲にわたる情報収集を可能とする。本研究では、これらの UAV やバルーンによって撮影された画像から、リアルタイムに移動物体を検出する手法の開発を行う。空中を移動しながら撮影を行うことによって、より広い範囲を選択的に観測することが可能となり、イベント会場などにおける雑踏の流動の把握および異常行動の監視や、災害時における避難経路の最適化などへの応用が期待できる。

本論文では、飛行型プラットフォーム自身の動きを生かした、移動体の検出方法を提案する。提案手法では、移動しながら撮影した 2 枚の画像における幾何的な拘束条件を考慮することによって環境中を移動している移動物体を検出する。

提案手法の精度評価実験を行うため、小型のクアッドローター型 UAV を実環境で飛行させた。複数の歩行者が存在する実環境下では、画面内に歩行者が特に多く存在する状況下において、毎秒 15 フレームの処理速度の下、65% の割合で歩行者を検出することができ、提案手法の有用性が示された。

## 論文の構成

本論文の構成は、以下のとおりである。

### 第 1 章

序章として、研究の背景と目的を述べる。また、飛行型プラットフォームによる空撮画像から移動物体を識別する関連研究の概要を記す。章の最後では、これらの内容に基づいて、本研究の寄与と意義を整理する。

### 第 2 章

本論文で提案する移動物体の識別手法に関して説明する。最初に、提案手法の処理の全体像について、UAV による実際の空撮画像を用いて説明する。続いて、特徴点抽出とその追跡、複数画像間の幾何など、提案手法を理解するために必要なコンピュータ・ビジョンにおける理論を記述する。これらの理論に基づき、静止物体に対してのみ成立する幾何学的な拘束条件について詳細を述べる。さらに、これらの幾何学的拘束条件を移動物体の識別に用いる評価関数の定式化を行う。最後に、これらの UAV に最適化された手法を、バルーンによる空撮画像に適用するための手法を述べる。

### 第 3 章

提案手法の精度評価を行うため、小型のクアッドローター型 UAV を実環境で飛行させた。この精度評価実験のコンフィギュレーションや、画像をはじめとする実験によって得られたデータ、提案手法のパフォーマンスを掲載する。また、精度の評価方法および、その評価指標に基づいた実験結果を記述する。精度の劣化が見られる箇所においては、その原因を考察した。

### 第 4 章

精度評価実験の結果を受けて、提案手法の実用化に関する考察を行う。現段階の精度とパフォーマンスの下で、適用可能ないくつかのアプリケーションおよび、提案手法とそのさらなる改善によって実現できる発展的な応用を述べる。続いて、そうしたアプリケーションの実現に向けた今後の課題について説明する。

### 第 5 章

本論文の結論として、提案手法の利点や精度、期待できる実用例を中心として総括を行う。

# 目次

論文要旨	1
第 1 章 序論	7
1.1 研究の背景	7
1.2 研究の目的	8
1.3 関連研究	10
1.4 本研究の寄与	11
第 2 章 移動物体識別手法	12
2.1 移動体識別手法の概観	12
2.2 画像間でのテクスチャの対応付け	15
2.3 幾何学的な拘束条件	17
2.4 外部パラメーターの算出	23
2.5 幾何学的拘束条件の評価	27
2.6 バルーン空撮画像への適用のための拡張	30
第 3 章 実験と精度検証	34
3.1 実験のコンフィギュレーション	34
3.2 実験結果	35
3.3 精度評価と考察	41
第 4 章 実用に関する考察	45
4.1 提案手法の利用可能性	45
4.2 実用に向けての課題	46
第 5 章 結論	47
参考文献	48



# 表目次

3.1	True Positive/Negative および False Positive/Negative の定義 . . . . .	41
3.2	実験で取得した画像を処理するのに用いたラップトップの性能 . . . . .	44

# 目次

1.1	飛行型プラットフォーム（マルチコプター）	7
1.2	UAV による移動体モニタリングの利点（左：広域観測，右：多視点観測）	7
1.3	飛行型プラットフォーム（バルーン）	8
1.4	2 種類の飛行型プラットフォームを用いた移動体モニタリングシステム	9
2.1	UAV の撮影した画像上での特徴点追跡結果（赤：特徴点，緑：追跡結果）	12
2.2	提案する移動物体識別処理の流れ	13
2.3	ステップ 1: 特徴点の抽出	13
2.4	ステップ 2: 特徴点の追跡（マッチング）	14
2.5	ステップ 3: 幾何学的拘束条件の計算	14
2.6	ステップ 4: 幾何学的拘束条件の評価（移動物体/静物体の識別）	14
2.7	画像の構成とピクセルが持つ色情報 [1]	15
2.8	KLT-Tracker による特徴点抽出の結果 [2]	17
2.9	ピンホールカメラモデル	18
2.10	エピポーラ幾何	20
2.11	PTAM によってカメラ画像から抽出された特徴点	25
2.12	PTAM によって作成された環境地図とカメラの軌跡	25
2.13	エピポーラ拘束の評価関数 $f_{EP}$ の形状	29
2.14	フローベクター・バウンドの評価関数 $f_{FV}$ の形状 ( $d_{min} = 0$ , $d_{max} = 20.0$ )	29
2.15	バルーンによる 1 枚目の空撮画像 $I_1$	30
2.16	バルーンによる 2 枚目の空撮画像 $I_2$	31
2.17	バルーンによる 3 枚目の空撮画像 $I_3$	31
2.18	バルーンによる 2 枚目の空撮画像の投影変換結果 $I'_2$	32
2.19	バルーンによる 3 枚目の空撮画像の投影変換結果 $I'_3$	32
2.20	前処理によって得られた $I_2$ における移動物体の候補領域	33
3.1	AR.Drone 2.0	34
3.2	カメラの搭載方法およびカメラ座標系	34
3.3	UAV を飛行させる場所および世界座標系	35
3.4	UAV により撮影された画像の抜粋（フレーム 0 から 80）	36
3.5	UAV により撮影された画像の抜粋（フレーム 90 から 170）	37
3.6	UAV により撮影された画像の抜粋（フレーム 180 から 240）	38
3.7	UAV 飛行軌跡の XY 平面への投影 ( $x=0.75$ , $y=4.95$ がフレーム 0 における位置)	39

3.8	UAV の飛行高度の時間推移 . . . . .	39
3.9	UAV の飛行速度の時間推移 . . . . .	40
3.10	UAV の姿勢（クォータニオン）の時間推移 . . . . .	40
3.11	Precision と Recall の時間推移 . . . . .	42
3.12	評価値 F の時間推移 . . . . .	42
3.13	フレーム 77 における特徴点（赤）とその追跡結果（緑） . . . . .	43
3.14	フレーム 77 において移動物体として識別された特徴点（赤） . . . . .	43
3.15	追跡に成功した特徴点の個数（赤）と移動物体に属する特徴点の個数 . . . . .	44

# 第 1 章

## 序論

### 1.1 研究の背景

近年，電子機器の小型化，高性能化に伴い，オートパイロット機能を備えた小型の無人航空機 (UAV: Unmanned Aerial Vehicle) が市場に出回るようになった．これらの UAV は，空撮のための安価かつ手軽なプラットフォームとして，災害時の情報収集等をはじめとして，様々な用途で利用がなされている [3] [4] [5]．図 1.1 に示したのは，マルチコプター型の UAV である．UAV による空撮画像には，図 1.2 に示すように，広範囲を選択的にセンシングすることができる，機動力が高く一つの物体をいろいろな視点で観察できるという 2 つの利点がある．その一方で，消費電力の大きさとバッテリー積載量の制約から，その持続飛行時間が長くないという欠点がある．



図 1.1 飛行型プラットフォーム（マルチコプター）

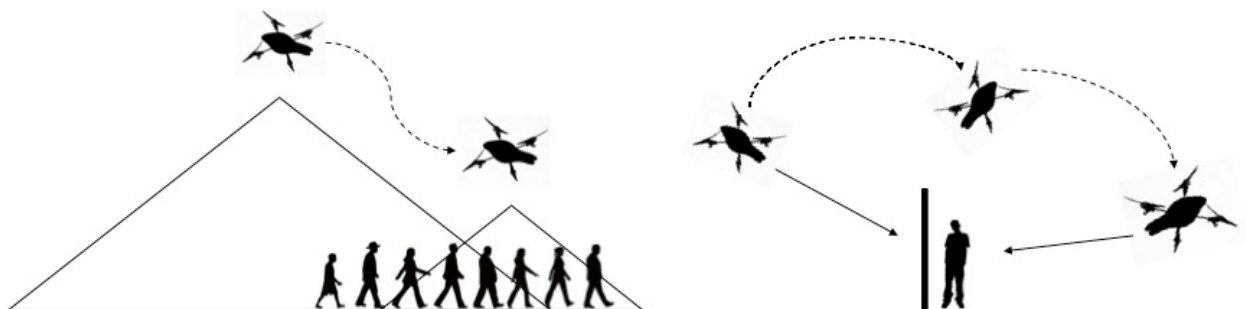


図 1.2 UAV による移動体モニタリングの利点（左：広域観測，右：多視点観測）



図 1.3 飛行型プラットフォーム（バルーン）

一方，UAV の登場以前から係留型の気球（バルーン）による空撮も多数行われてきた [6] [7]。バルーンは，基本的に地上の一ヶ所から繋ぎ止めて使用するものであり，観測中にその位置（視点）を自由に変更することはできないため，機動力の点で UAV には大きく劣る。しかしながら，内蔵するヘリウムガスの浮力によって飛行するため，飛行に電力を必要とせず，持続飛行時間はマルチコプターに比べて格段に長い。さらに，飛行と姿勢の安定化のために常に制御が必要であるマルチコプターに対して，基本的には制御を必要としないため，一度設置してしまえば運用が比較的容易であるという利点がある。そのため，比較的高高度を飛行させ，長時間にわたって広範囲を観測する用途に向いている。

本論文では，このようなマルチコプター型の UAV とバルーンをまとめて，飛行型プラットフォームと呼ぶことにする。

## 1.2 研究の目的

本論文では，飛行型プラットフォームが持つ前述のような利点に着目し，飛行型プラットフォームに搭載した単眼のカメラで撮影した複数枚の画像を用いて，歩行者や自動車などの移動物体をリアルタイムに検出する手法を提案する。

これまでに移動物体のセンシング手法として，防犯カメラやレーザースキャナーなど（固定型センサ）を用いる方法や，複数種類の固定センサを組み合わせた方法が数多く提案されているが，こうした既存手法には2つの制限がある。まず，これらのセンサは環境に固定されるセンサであるため，計測範囲が限定されてしまう。また，これらの固定センサは設置の制約上，斜め上から対象を撮影するものが多いため，混雑した環境においては移動物体同士の重なりによる隠蔽（オクルージョン）の影響を受けやすく，このような状況下では，移動物体の検出精度が劣化すると考えられる。

本研究では，上で述べた飛行型プラットフォームが持つ特性を活用することで，従来型の固定センサが持つこのような欠点を解消した，移動物体のモニタリングシステムの実現に寄与する。まず，広範囲を選択的にセンシングできるという UAV の利点と，バルーンによる広範囲の空撮画像を活用することで，移動物体がより集中している場所を優先的に撮影することができ，より効率的に移動物体をモニタリングすることが可能となる。また，隠蔽の影響を受けにくい上空からの撮影という特性や，一つの物体をいろいろな視点から観察できるという UAV の利点を活用することで，移動物体同士または障害物との重なり

よる遮蔽に対処することができ、移動物体のモニタリングをより確実なものとするのが可能である。

先に述べたように、固定カメラから撮影した画像から移動物体を検出する研究は数多く行われてきた。固定カメラによって撮影された画像から移動体を検出する代表的な方法としては、背景差分法 [8] や、オプティカルフローを利用したもの [9] が挙げられる。これらの手法では、移動体によって画面内に現れる変化を識別することにより、移動体を検出している。このように移動体検出に関する既存手法の多くは、移動体によって画面内に現れる変化を識別することにより移動体を検出している。しかしながら、これらの手法の大部分は、監視カメラなど環境に固定されたカメラを対象としており、飛行型プラットフォームにカメラを搭載する場合など、カメラ自身が動く場合には移動体を画像から正しく検出することができない。なぜならば、カメラが動くことによって移動物体がなくとも画面に変化が生じ、カメラ自身の動きによる画面の変化と、移動物体による画面の変化を分離することが出来ないからである。特に、飛行型プラットフォームは3次元空間を移動・回転するため地上を走るロボットなどに比べて位置・姿勢の自由度が高い上に、突風などの環境外乱によって姿勢が大きく変化することもありうる。本研究では、このような飛行型プラットフォームの動きによる画面の変化と、移動物体による画面の変化を分離し、移動物体をリアルタイムに検出するための手法を提案する。

このような移動体検出手法と2種類の飛行型プラットフォームを利用することにより、将来的には図1.4に示すような移動物体のモニタリングシステムも実現可能となろう。このシステムでは、広範囲を一度に観測できるバルーンと、機動力の高いUAVを組み合わせることにより、移動物体を広範囲かつ詳細にモニタリングすることを実現する。すなわち、バルーンを用いた空撮により観測領域から移動物体の分布を計測し、移動体が多く存在する場所にUAVを向かわせ、移動体をより近距離から詳細に観測するというものである。発展として、2台以上のバルーンの観測により地上をステレオ観測し、構築された立体地図の情報をもとにUAVの軌道を制御することも可能である。このようなシステムは、イベント会場における異常行動の監視や、スポーツの実況中継、災害時の状況把握等に役立てることができると考えられる。

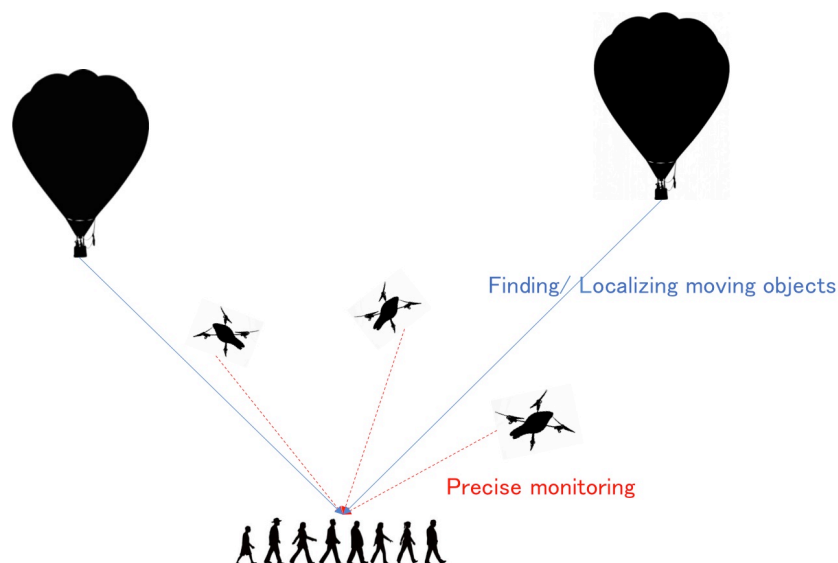


図 1.4 2 種類の飛行型プラットフォームを用いた移動体モニタリングシステム

### 1.3 関連研究

次節で研究の寄与を明らかなものとするため、本節では飛行物体に搭載されたカメラの画像から移動物体を検出する既存研究についてその概要を述べる。

Jung らは、動的環境下におけるロボットナビゲーションを目的として、ロボットによって撮影された複数枚の画像から移動体を認識・追跡する手法を提案した [10]。Jung らの手法では、カメラの動きに起因する各画素の画面内での移動をバイリニア変換によってモデル化する。バイリニア変換のパラメータは、画像間で追跡した特徴点を利用して推定がなされる。その後、バイリニア変換によってカメラの移動に起因する画面の変化を補償し、背景差分を観測量としたパーティクルフィルターを適用することにより、移動体の存在領域を識別する。Jung らの手法は、無人の小型ヘリコプターで実験がなされ、8割以上という高い割合で移動体（人）を認識・追跡できている。しかしながら、同時に複数の移動体を追跡する場合、処理に時間がかかる上に検出精度が大きく低下するという欠点がある。

Rodriguez らの研究では、防犯用途での利用を目的として、飛行ロボットによる移動体の認識・追跡手法を提案している [11]。この手法では、PTAM[12] と呼ばれる SLAM (Simultaneous Localization And Mapping) の手法によってカメラの外部パラメータを推定し、これと画像から抽出された特徴点を利用することによって、疑似的なオプティカルフローを作成する。この疑似的なオプティカルフローと特徴点追跡によって得られた実際のオプティカルフローを比較することにより、移動体を検出する。この手法の最大の利点は計算効率の高さである。本手法は、回転翼型の小型 UAV 搭載のコンピュータで実験がなされ、視野内に存在する人を 3 人までリアルタイムに認識・追跡できている。しかし、疑似的なオプティカルフローの計算時に、地表面を完全な平面と仮定したホモグラフィーを利用するため、建物などによって急激な高低差が現れるような場面には適用できないという制限がある。

Siam らは、UAV で撮影した画像から複数の移動物体をリアルタイムに識別・追跡する枠組みを提案している [13]。彼らの枠組みは、特徴量計算と、ホモグラフィーと呼ばれる静物体に対して成立する投影幾何に基づいて行われる。つまり、UAV によって異なる位置から撮影した 2 枚の画像に対して特徴点の追跡を行い、得られた特徴点の各ペアがホモグラフィーを遵守しているかを、最小メディアン推定によって計算する。そして、ホモグラフィーを遵守していないと判定された特徴点を、移動物体の領域から抽出されたものとして扱うというものである。このようにして抽出された特徴点はクラスタリングがなされ、移動物体としてクラスター化がなされる。各クラスターに対してはカルマンフィルタによる移動体追跡、および一貫性による誤認識の排除といった処理がなされる。

Sirmacek らは、解像度の低い航空写真や衛星画像から、群衆の分布と流動を推定する手法を提案している [14],[15]。解像度の低い航空写真および衛星画像では、人間の体の部位を判別することはできないため、頭部などに着目した人物の認識手法を用いることが出来ない。そこで、Sirmacek らは、人が高密度に分布している領域と、疎に分布している領域とでは、その色が明瞭に異なっていることに着目し、ピクセルの色彩の変化に基づいた人物の認識手法を提案した。Sirmacek らの提案手法では、FAST [16] [17] と呼ばれる手法を用いて画像中からサンプリングされた特徴点に対して、適応的なカーネル密度推定を適用し、画像中の各ピクセルに人物が存在する確率を推定する。この確率密度の推定は、人物が密に分布している領域では色彩の変化が大きいため、このような領域では、人物が疎に分布している領域に比べて、より多くの特徴点が抽出されるという仮定に基づいている。最初に、推定された確率密度は、高密度に人が集まっている場所の境界を推定するのに使用される。次に、その領域の色彩をもとに、それ以外の領域に分布している人物を認識する。このような処理を各フレームに対して繰り返し、人物が疎に分布する領

域にカルマンフィルタを用いた人物追跡を行う．2.5km から 8km 四方という広範囲を撮影した航空写真に対して実験を行った結果，15cm から 50cm という低い解像度にも関わらず，87 パーセントという高い精度で人数のカウントを行うことができた．

## 1.4 本研究の寄与

先に述べた既存研究では，観測された画面の変化から，飛行型プラットフォーム自身の動きの効果による画面の変化を分離することによって移動体の存在を検出するという点では似通っている．すなわち，これらの手法は，飛行型プラットフォームの動きによってもたらされる画面の変化をいかにして分離するかに重きを置いており，飛行型プラットフォーム自身の動きを移動体検出における不利な点としてとらえている．

これに対して本研究では，飛行型プラットフォームの動きを生かした移動体の検出方法を提案する．提案手法では，移動しながら撮影した 2 枚の画像において幾何的な拘束条件を考慮することによって，環境中を移動している点（移動点）を検出する．飛行型プラットフォームから撮影した多視点の画像に対して，幾何的な拘束条件を利用した移動体検出の手法を適用するのは初めての試みであり，これは本研究の寄与の一つである．



## 第 2 章

# 移動物体識別手法

### 2.1 移動体識別手法の概観

研究の目的でも説明した通り，監視カメラなど環境に固定されたセンサに比べて，飛行型プラットフォームから撮影した画像から環境中を移動している物体を抽出することは難しい．図 2.1 は，UAV から数秒の時間差をおいて撮影した 2 枚の画像に対して，特徴点と呼ばれる代表となる点を複数抽出し，その追跡（マッチング）を行ったものである．赤い点が画像から抽出された特徴点であり，緑色の直線がその追跡結果，すなわち UAV のカメラ視野内での特徴点の動きを表している．図 2.1 から明らかなように，走行中の自動車から抽出された特徴点だけではなく，その手前の芝生等，静止している物体から抽出された特徴点でさえも，UAV のカメラの視野内では移動が見られる．これは，UAV 自身，つまりはカメラが環境中を移動しているためである．

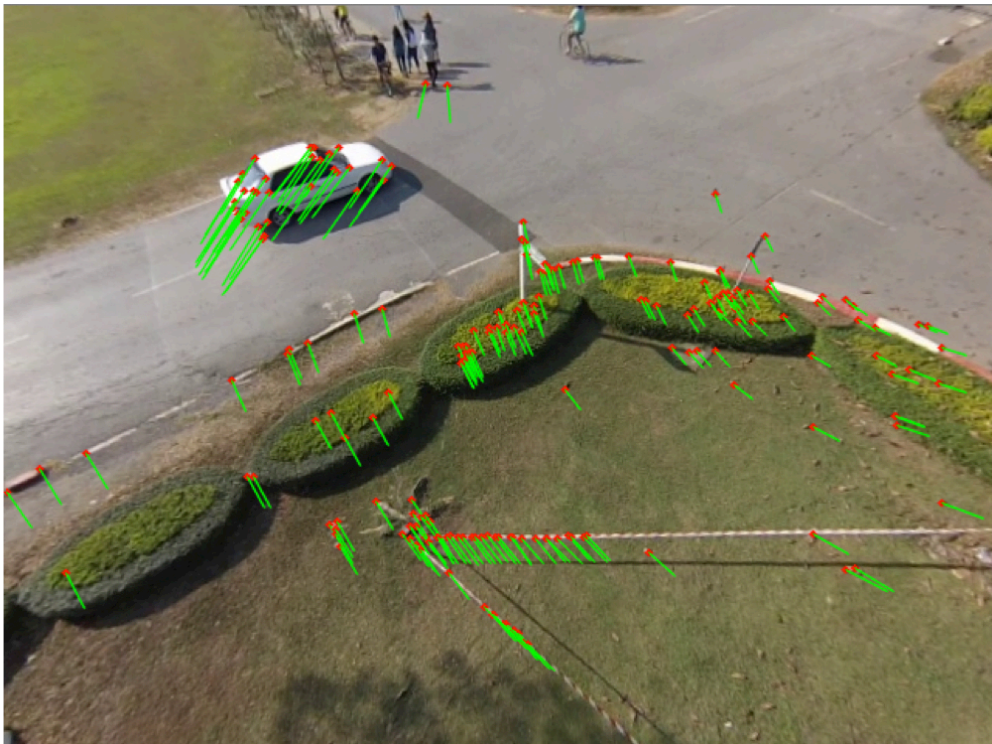


図 2.1 UAV の撮影した画像上での特徴点追跡結果（赤：特徴点，緑：追跡結果）

このように、飛行型プラットフォームから撮影した画像から移動物体を識別する場合、単純に視野内（画像平面上）での物体の動きだけではなく、飛行型プラットフォーム自身の動きを加味した上で、移動物体/静物体の識別を行う必要がある。本節の以降では、本研究で提案する移動体識別手法に関して、その概観を述べる。図 2.2 は、提案手法で行われる処理を表したものである。

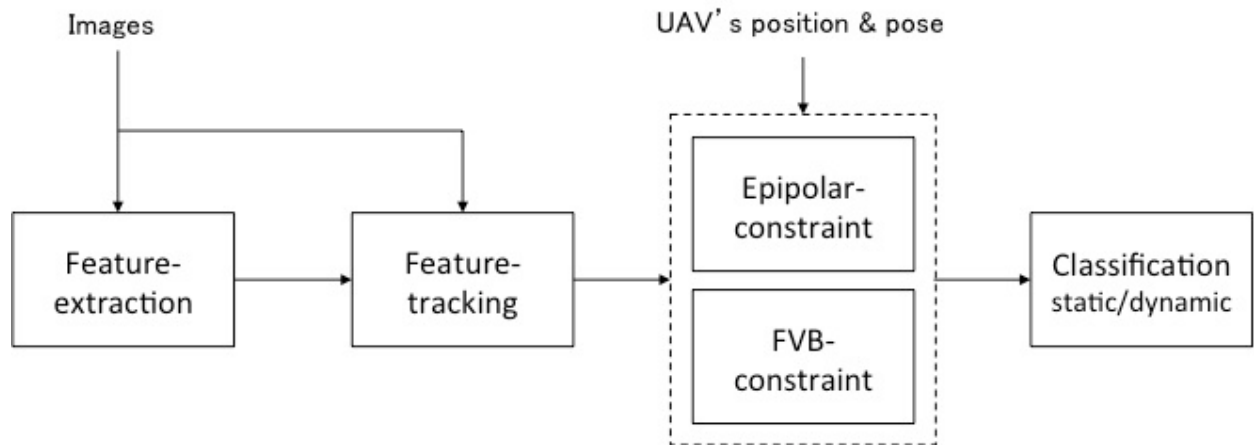


図 2.2 提案する移動物体識別処理の流れ

まず、この識別手法は数秒の時間間隔において撮影した2枚の画像を入力とし、特徴点（代表点）の抽出と画像間での特徴点の追跡を行う。図 2.3 には抽出された特徴点を赤色で、図 2.4 にはその追跡結果を緑色の直線で示している。

次に、追跡に成功した特徴点のペアと、2枚の画像間における飛行型プラットフォームの相対運動量を入力とし、幾何学的な2つの拘束条件を計算する。この2つの拘束条件は、いずれも空間中を静止している物体に対してのみ現れる拘束条件であり、空間中を動いている物体には適用されない。図 2.5 には、拘束条件の計算結果を白い直線で示している。拘束条件の計算方法について、詳細は後述する。

最後に、それぞれの特徴点に対してこれらの拘束条件の遵守の度合いを評価することによって、各々の特徴点が移動物体から抽出されたものなのか、静物体から抽出されたものであるかを識別する。図 2.6 には、この評価の結果、移動物体から抽出されたと判断された特徴点のみを赤色で示している。



図 2.3 ステップ 1: 特徴点の抽出

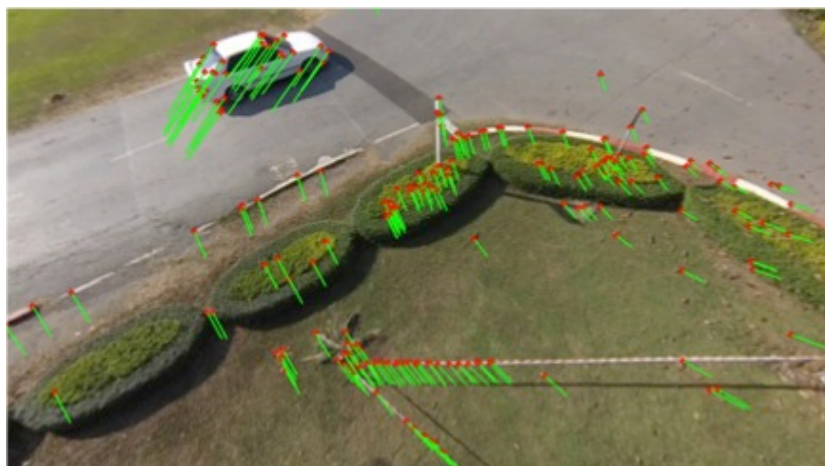


図 2.4 ステップ 2: 特徴点の追跡 (マッチング)

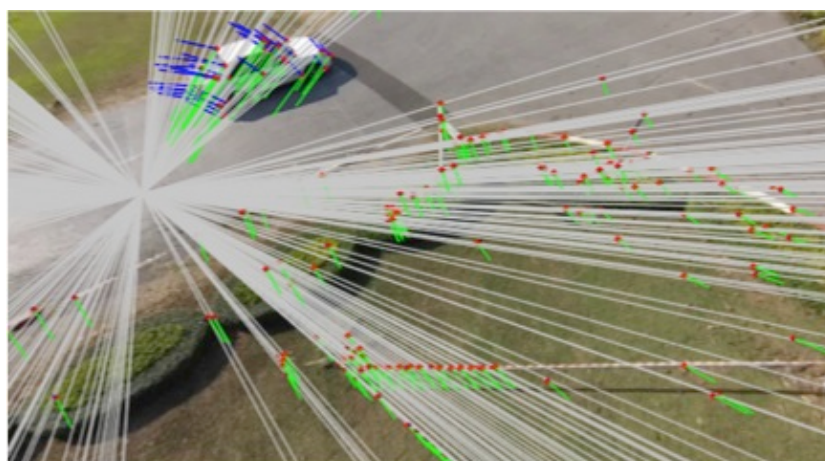


図 2.5 ステップ 3: 幾何学的拘束条件の計算

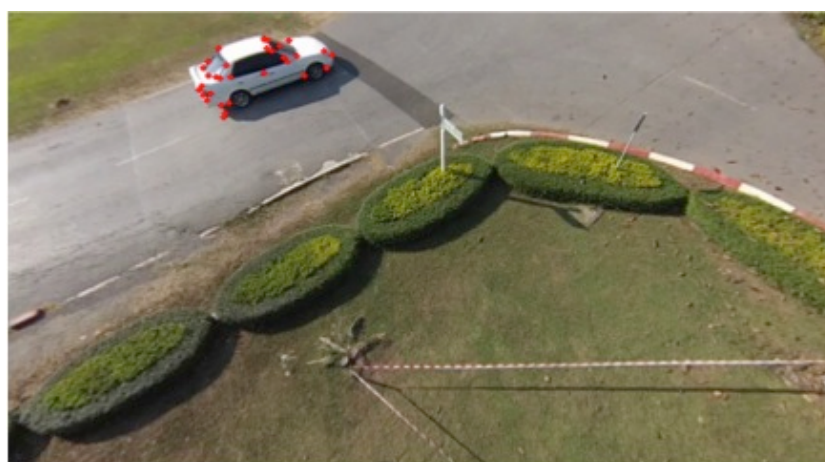


図 2.6 ステップ 4: 幾何学的拘束条件の評価 (移動物体/静物体の識別)

以降に続く節では、提案手法の各ステップに関して、詳細を述べる。



## 2.2 画像間でのテクスチャの対応付け

先に説明したとおり，本研究では，UAV を用いて異なる位置（視点）から撮影した対象物に関して，幾何学的な拘束条件を調べることによって，静物体と移動物体を識別する．よって，1 枚目に撮影した画像，2 枚目に撮影した画像間において，映り込んだ物体の対応付けを行う必要がある．移動物体の識別率向上のためには，2 枚の画像上の全ての物体に対して対応付けを行うのが理想的ではあるが，取得した画像列には非常に多くの情報が含まれているため，そのような処理は時間がかかり過ぎ，実用に耐えない．そこで，画像中のテクスチャから特徴点と呼ばれる代表点を抽出し，その特徴点に関してのみ，画像列間での対応付けを行う．

### 2.2.1 特徴点の抽出と追跡

図 2.7 に示すように，カメラから取得する画像はピクセルと呼ばれる小さな長方形の集合であり，それぞれのピクセルは色や輝度などの情報を持つ．テクスチャの強さは，画像中のある小さな領域で，ピクセルの輝度値の変化が特徴的でありエッジやコーナーが強く抽出できることや，領域内の他のピクセルと比較した際に輝度の変化が大きいことで判断される．こうした強いテクスチャは，異なる視点から撮影した画像間において追跡を行うことができる可能性が高く，画像間でその推移を調べることによって，テクスチャの対応付けが実現できる．

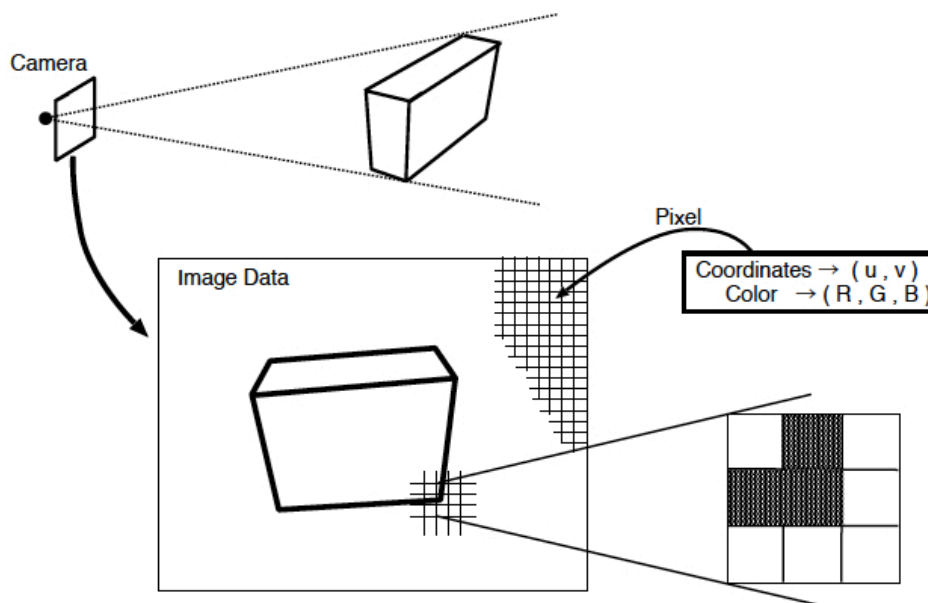


図 2.7 画像の構成とピクセルが持つ色情報 [1]

強いテクスチャを特徴点として抽出および追跡する手法としては，Harris オペレータを用いた方法 [18]，FAST 法 [16]，SIFT 法 [19] など，いくつかの優れた提案がある．本研究では，これらの提案と同様に高い実績を持つ KLT(Kanade-Lucas-Tomasi)-Tracker[20]，[21] による画像特徴点の抽出および追跡を行う．

### 2.2.2 KLT-Tracker による特徴点追跡

KLT-Tracker では、以下の2つの仮説のもと、特徴点の対応付け（追跡）を行う。

- 対応付けを行う2枚の画像間で、対応するテクスチャの輝度変化しない
- となり合うピクセルの画像平面上での動きは類似する

以降では、KLT-Tracker による特徴点追跡の理論について説明する。なお、KLT-Tracker の特徴点の選択方法についても、この理論を追うことで明らかとなる。上記の1つ目の仮説により、式2.1が成立する。

$$I(x, y, t) = I(x + dx, y + dy, t + dt) \quad (2.1)$$

ここで、 $I(x, y, t)$  は、時刻  $t$  における画像平面上の座標  $(x, y)$  のピクセルが持つ輝度を表す。式2.1は、このピクセルが、微小時間  $dt$  の間に  $(x + dx, y + dy)$  の位置に移動したということを表す。

右辺をテイラー展開し、2次以降の微少量を無視すると、式2.2を得る。

$$I(x + dx, y + dy, t + dt) = I(x, y, t) + \frac{\partial I}{\partial x} dx + \frac{\partial I}{\partial y} dy + \frac{\partial I}{\partial t} dt \quad (2.2)$$

式2.1と両辺を引き算し、両辺を  $dt$  で割ることにより、式2.3を得る。

$$I_x u + I_y v + I_t = 0 \quad (2.3)$$

ただし、式2.4および2.5の定義を用いた。

$$I_x = \frac{\partial I}{\partial x}, \quad I_y = \frac{\partial I}{\partial y}, \quad I_t = \frac{\partial I}{\partial t} \quad (2.4)$$

$$u = \frac{dx}{dt}, \quad v = \frac{dy}{dt} \quad (2.5)$$

画像の勾配  $I_x$ ,  $I_y$ ,  $I_t$  は既知であるから、未知数はテクスチャの移動量  $u$ ,  $v$  の2つである。未知数が2つに対して方程式（式2.3）が1つのみであるため、ここで2つめの仮定を導入する。すなわち、追跡対象となるピクセルの周りに小領域（パッチ）を定義し、そのパッチ内のすべてのピクセルはすべて同じ動きをすると仮定する。

説明のため、ここでは、 $3 \times 3$  程度の小領域（パッチ）を定義したとする。仮定により、パッチ内の9つのピクセルの移動量  $u$ ,  $v$  は同じ値を持つため、未知数  $u$ ,  $v$  に対して、2.3と同様の式が9つ定義できることになる。この連立方程式を最小二乗法で解くことにより、以下の式2.6を得る。

$$\mathbf{G} \begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} -\sum_i I_{xi} I_{ti} \\ -\sum_i I_{yi} I_{ti} \end{bmatrix} \quad (2.6)$$

ただし、 $\sum_i$  はパッチ内のすべてのピクセル（今の例では9つのピクセル）についての総和を表し、行列  $\mathbf{G}$  を式2.7によって定義する。

$$\mathbf{G} = \begin{bmatrix} \sum_i I_{xi}^2 & \sum_i I_{xi} I_{yi} \\ \sum_i I_{xi} I_{yi} & \sum_i I_{yi}^2 \end{bmatrix} \quad (2.7)$$

式 2.6 を安定して解くためには、行列  $G$  が逆行列を持つ必要がある。すなわち、 $G$  の 2 つの固有値が十分に大きく、両者の値が大きく変わらないことが条件である。このような条件を満たす点が特徴点として抽出され、それらに対して式 2.6 を解くことにより、画像間での対応をとることが可能となる。図 2.8 には、このような条件を満たすピクセル、すなわち特徴点をオレンジ色の丸で示している。図 2.8 からわかるとおり、特徴点はコーナーにあたる点から多く抽出されている。このような点はテクスチャのない平面やエッジにあたるピクセルよりも追跡がしやすいという直感に沿っており、この条件は妥当であるといえる。



図 2.8 KLT-Tracker による特徴点抽出の結果 [2]

以上のように特徴点検出と追跡を統合した手法を、KLT(Kanade-Lucas-Tomasi)-Tracker と呼ぶ。

実用的には、移動量  $u$ ,  $v$  が微小であるときにしか式 2.6 が成り立たないため、繰り返し計算を行うことによって結果が収束するまで追跡を続ける。また、対象となる 2 枚の画像それぞれを何段階かダウンサンプリング（低解像度化）し、まず低解像度化した画像に対して特徴点追跡を行い、その結果を初期値として高解像度の画像に対して追跡を行う、といった計算を繰り返し行うことにより、特徴点追跡の高速化を図るという提案もある [21]。

## 2.3 幾何学的な拘束条件

本節では、異なる視点から撮影された静止物体に対して成立する幾何学的な拘束条件について説明する。直前の節で説明した特徴点追跡によって追跡に成功したテクスチャに対してこれらの拘束条件を調べることにより、追跡されたテクスチャが静止物体であるか、移動物体であるかを識別する。まず、こうした幾何学を理解するために必要なカメラの投影モデルについて述べ、その後、本研究で用いる幾何学的な拘束条件について説明を行う。

### 2.3.1 カメラの投影モデル

カメラから得られる画像は、3次元空間の物体がレンズを通してカメラの撮像面に射影されたものであり、2次元の画素平面として構成される。このカメラの投影モデルとしては、ピンホールカメラという理想のカメラが用いられる。ピンホールカメラによる射影は中心投影と呼ばれ、我々の目から普段見えている光景と同じ投影モデルである。

図 2.9 に示すようなカメラ座標系を定義する。このカメラ座標系はカメラに固定された左手座標系であり、カメラの前方を  $Z$  方向、右を  $X$  方向、下方を  $Y$  方向とする。カメラの撮像面の2次元座標（画像座標）は、原点が画像平面とカメラ座標系との交点、 $X$  方向および  $Y$  方向はそれぞれカメラ座標系の  $X$  方向および  $Y$  方向と一致するものとする。

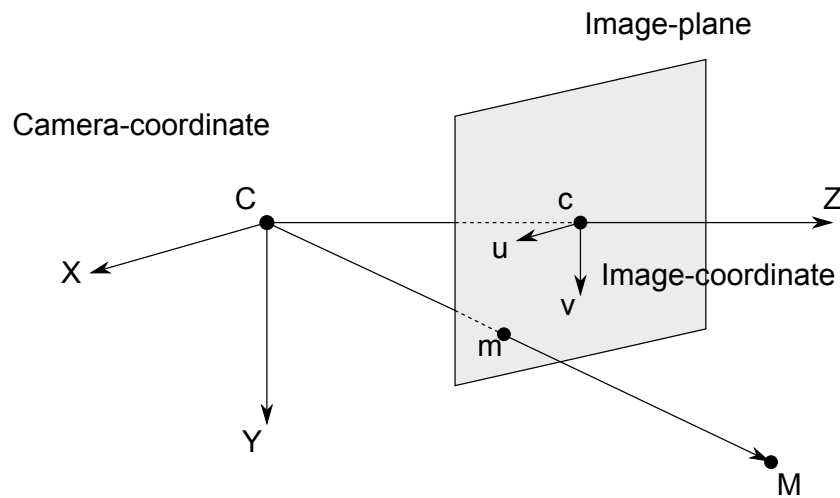


図 2.9 ピンホールカメラモデル

図 2.9 から明らかなように、ピンホールカメラモデルにより、カメラ座標系で測った空間中の3次元座標  $(x, y, z)^T$  と撮像面に投影された2次元座標  $(u, v)^T$  には、式 2.8 で表される関係が成り立つ。なお、 $f_x$ ,  $f_y$  はカメラの焦点距離（カメラ中心から撮像面までの距離）であり、カメラによって決まる定数である。

$$\begin{aligned} u &= f_x \frac{x}{z} \\ v &= f_y \frac{y}{z} \end{aligned} \quad (2.8)$$

非線形の式 2.8 は式 2.9 のように線形表現することができる。なお、前述のとおり、 $(x, y, z)^T$  は3次元空間の点をカメラ座標系で測った値、 $(u, v)^T$  はその画像への投影の座標である。 $\cong$  はスケールを除いて両辺の値が等しいことを表す記号である。式 2.9 のように、任意の座標ベクトル  $\mathbf{a} = (a_1, a_2, a_3)$  に対して、末尾に新しい要素として1を加えたベクトルを、 $\mathbf{a}$  の同次座標または拡張ベクトルと呼び、記号  $\tilde{\mathbf{a}}$  で表す。

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \cong \begin{bmatrix} f_x & 0 & 0 & 0 \\ 0 & f_y & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \quad (2.9)$$

式 2.9 は、一般に式 2.10 のように記述される。

$$\widetilde{\mathbf{m}} \cong \mathbf{P} \widetilde{\mathbf{x}} \quad (2.10)$$

ただし、式 2.11 および式 2.12 の定義を用いた。

$$\widetilde{\mathbf{m}} = \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}, \quad \widetilde{\mathbf{x}} = \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \quad (2.11)$$

$$\mathbf{P} = \begin{bmatrix} f_x & 0 & 0 & 0 \\ 0 & f_y & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad (2.12)$$

ここで、 $\mathbf{P}$  は透視投影行列といい、カメラ座標系を画像座標系に変換するカメラによって固有の値を持つ行列である。また、カメラの内部構造によって決定されるため、内部パラメータ行列とも呼ばれる。透視投影行列の値は、チェッカーボードなど 3 次元空間内での座標が既知の物体を撮影するなど、カメラキャリブレーションの手法によって正確に決定することができる [22] [23]。

ところで、今までは 3 次元空間内の点をカメラ座標系において記述してきた。環境に固定された世界座標系において記述された値から画面への投影（画像平面でのその点の位置）を計算したい場合には、その点の座標を世界座標系からカメラ座標系へと変換し、式 2.10 を適用すればよい。

そのためには、まず世界座標系とカメラ座標系の方向が一致するように座標系を回転させ、次にその姿勢を保ったまま平行移動を行う。一般に回転は  $3 \times 3$  の行列  $\mathbf{R}$ 、並進は  $3 \times 1$  のベクトル  $\mathbf{t}$  で記述される。回転行列  $\mathbf{R}$  は式 2.13、式 2.14 のような性質を持つ。

$$\mathbf{R}\mathbf{R}^T = \mathbf{R}^T\mathbf{R} = \mathbf{I} \quad (2.13)$$

$$\det(\mathbf{R}) = 1 \quad (2.14)$$

ここで、 $\mathbf{I}$  は単位行列である。式 2.13、式 2.14 は 6 つの独立した拘束条件となるため、 $\mathbf{R}$  の自由度は 3 である。

以上の内容を式で表すと、式 2.15 となる。ここで、3 次元空間内の任意の点  $M$  に関して、 $\mathbf{x}_c$  はカメラ座標系における点  $M$  の座標、 $\mathbf{x}_w$  は世界座標系における点  $M$  の座標を表す。右辺では、 $\mathbf{x}_w$  の同次座標表現  $\widetilde{\mathbf{x}}_w$  によって方程式を記述している。なお、 $[\mathbf{R}|\mathbf{t}]$  は  $\mathbf{R}$  と  $\mathbf{t}$  を横に並べた  $3 \times 4$  の行列である。カメラの内部パラメータ行列  $\mathbf{P}$  に対して、 $[\mathbf{R}|\mathbf{t}]$  はカメラの外部パラメータ行列と呼ばれる。

$$\mathbf{x}_c = \mathbf{R}\mathbf{x}_w + \mathbf{t} = [\mathbf{R}|\mathbf{t}] \widetilde{\mathbf{x}}_w \quad (2.15)$$



### 2.3.2 エピポーラ幾何

カメラの画像は3次元世界の2次元平面への投影であり，この投影からは奥行きという1次元分の情報が失われている．このように3次元世界の完全な記述に関して不定性が残っているために，1つのカメラから得られた画像だけでは，3次元物体の形状を完全に決定することはできない．しかし，ステレオカメラのように2つの異なる視点（位置）から撮影された画像を用いることで，このような不定性を取り除き，物体の3次元的な形状を決定することができる．このとき，異なる視点におかれたカメラの位置関係や相対姿勢などの情報が必要になる．

このように，複数の視点から撮影した画像を用いて3次元情報を復元する基本的な手法の一つとして，エピポーラ幾何があげられる．エピポーラ幾何はステレオビジョン（異なる視点におかれた2つのカメラを用いて物体を立体視する手法）に用いられる基本的な幾何関係であり，同一物体の画像上の投影位置に拘束を与える．このようなエピポーラ幾何が与える拘束（エピポーラ拘束）は，一般に3次元空間中に静止している物体に対してのみ成立し，3次元中を移動している物体にはあてはまるとは限らない．本研究では，エピポーラ幾何が静止物体のみに成立することを利用し，移動物体と静止物体の識別を行う．以降では，エピポーラ幾何の導出を行い，その詳細を明らかにする．

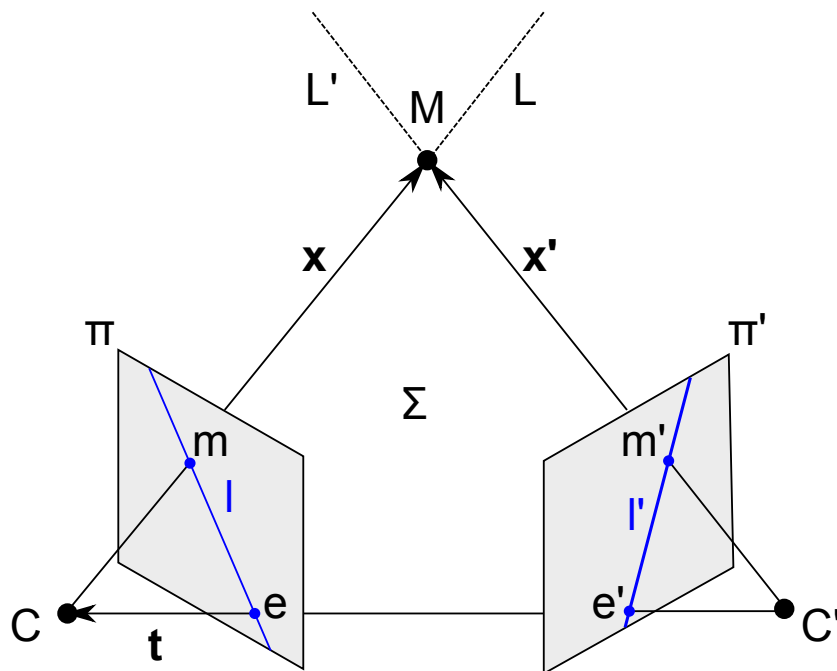


図 2.10 エピポーラ幾何

いま，図 2.10 に示すように，3次元空間内のある点  $M$  がカメラ中心  $C$  の画像上の点  $m$  に投影されているものとし，カメラ  $C$  の画像平面を  $\pi$  と定義する．同様に，点  $M$  はカメラ中心  $C'$  の画像上の点  $m'$  に投影されているものとし，カメラ  $C'$  の画像平面を  $\pi'$  と定義する．

このとき， $C$ ， $C'$ ， $M$  の3点は3次元空間内に1つの平面  $\Sigma$  を定義する．この平面  $\Sigma$  は，エピポーラ平面と呼ばれる．エピポーラ平面  $\Sigma$  と画像平面  $\pi$  が交差してできる直線  $l$  はエピポーラ線と呼ばれる．同様にエピポーラ平面  $\Sigma$  と画像平面  $\pi'$  が交差してできる直線  $l'$  は，画像平面  $\pi'$  におけるエピポーラ

ラ線である。

視点  $C$  で画像点  $m$  が観測されたとき、これに対応する空間中の点は半直線  $CM$  (半直線  $L$ ) 上に存在するので、半直線  $L$  は点  $m$  に対応する点の集合であると考えられる。図 2.10 から明らかなように、この半直線  $L$  の画像平面  $\pi'$  上への投影がエピポーラ線  $l'$  にほかならない。すなわち、エピポーラ線  $l'$  は、画像平面  $\pi$  上の点  $m$  に対応する画像平面  $\pi'$  上の対応点の集合である。同様に、エピポーラ線  $l$  は、画像平面  $\pi'$  上の点  $m'$  に対応する画像平面  $\pi$  上の対応点の集合である。このように、エピポーラ線は異なる画像平面上への投影の位置関係に関して拘束条件を与えている。また、視点  $C$  と視点  $C'$  を結ぶ直線が、画像平面  $\pi$  および  $\pi'$  と交差してできる点  $e, e'$  をエピポールと呼ぶ。

ここまで、3次元空間内の1点  $M$  についてのみ考えてきたが、3次元空間内の任意の点に対して、エピポーラ平面が考えられ、またエピポーラ線が存在する。これらすべてのエピポーラ線は対応する画像平面上のエピポールを通る。このように3次元空間内の対象物を2つのカメラで撮影した場合には、画像平面上での投影に特有の幾何学的拘束が現れる。このような幾何のことを、エピポーラ幾何という。

ここで、エピポーラ幾何を記述する方程式 (エピポーラ方程式) を導出する。図 2.10 に示すように、3つのベクトル  $\mathbf{x}, \mathbf{x}', \mathbf{t}$  を定義する。 $\mathbf{x}$  は視点  $C$  を始点とし点  $M$  を終点とするベクトル、同様に  $\mathbf{x}'$  は視点  $C'$  を始点とし点  $M$  を終点とするベクトル、 $\mathbf{t}$  は視点  $C'$  を始点とし点  $C$  を終点とするベクトル (すなわち並進ベクトル) である。この3つのベクトルは、すべて1つの平面  $\Sigma$  上に存在する。同一平面上に3つのベクトルが存在するとき、そのうちの任意のベクトルの外積と残る1つのベクトルの内積は0となる。したがって、式 2.16 が成り立つ。

$$\tilde{\mathbf{x}}' \cdot (\mathbf{t} \times \mathbf{R}\tilde{\mathbf{x}}) = \tilde{\mathbf{x}}'^T (\mathbf{t} \times \mathbf{R}\tilde{\mathbf{x}}) = 0 \quad (2.16)$$

ただし、行列  $\mathbf{R}$  は  $C$  で記述された座標を  $C'$  での記述に変換する回転行列である。ここで、 $\mathbf{x}$  はカメラ座標系  $C$  によって成分が記述されているが、 $\mathbf{x}'$  および  $\mathbf{t}$  はカメラ座標系  $C'$  によって成分が記述されていることに注意が必要である。

式 2.16 の外積を行列演算に置き換えると、式 2.17 のようになる。

$$\tilde{\mathbf{x}}'^T [\mathbf{t}]_{\times} \mathbf{R}\tilde{\mathbf{x}} = 0 \quad (2.17)$$

ここで、 $[\mathbf{a}]_{\times}$  は、任意のベクトル  $\mathbf{a} = [a_1, a_2, a_3]^T$  に対する、式 2.18 で定義されるような行列 (歪対象行列) を表すものとする。この歪対象行列を用いることで、任意のベクトル  $\mathbf{b}$  との外積が、式 2.19 のように記述できる。

$$[\mathbf{x}]_{\times} = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} \quad (2.18)$$

$$\mathbf{a} \times \mathbf{b} = [\mathbf{a}]_{\times} \mathbf{b} \quad (2.19)$$

ここで、行列  $\mathbf{E}$  を式 2.20 のように定義すると、式 2.17 は式 2.21 のように表すことができる。

$$\mathbf{E} = [\mathbf{t}]_{\times} \mathbf{R}\tilde{\mathbf{x}} \quad (2.20)$$

$$\tilde{\mathbf{x}}'^T \mathbf{E} \tilde{\mathbf{x}} = 0 \quad (2.21)$$

式 2.21 はエピポーラ方程式と呼ばれ、行列  $E$  は基本行列と呼ばれる。異なる視点から得られた 2 つの画像において、対応する点同士は必ず式 2.21 を満たす。この拘束をエピポーラ拘束という。

なお、画像から直接得られるのは物体の 3 次元座標  $x, x'$  ではなく、その画像座標  $m, m'$  である。式 2.10 により、画像座標が与えられたとき、式 2.21 は式 2.22 のように変形できる。

$$\widetilde{m}'^T F \widetilde{m} = 0 \quad (2.22)$$

ただし、 $F$  は式 2.23 で定義される  $3 \times 3$  の行列であり、基礎行列と呼ばれる。

$$F = P^{-T} E P^{-1} \quad (2.23)$$

ここで、画像平面  $\pi'$  上におけるエピポーラ線  $l'$  は、式 2.24 で記述できる。また、画像平面  $\pi'$  上におけるエピポーラ線  $l'$  と対応する点  $m'$  との距離を  $h'$  とすると、 $h'$  は式 2.25 のように表すことができる。

$$l' = \frac{F \widetilde{m}}{\|F \widetilde{m}\|} \quad (2.24)$$

$$h' = \widetilde{m}'^T l' \quad (2.25)$$

すなわち、式 2.22 は、式 2.24 で定義されるエピポーラ線と対応する点  $m'$  との距離はゼロ ( $h'=0$ ) であることを示している。

ここで、式 2.22 の両辺で転置をとると、画像平面  $\pi$  においても同様に、エピポーラ線  $l$  と点  $m$  の距離  $h$  がゼロであるということがわかる。なお、エピポーラ線  $l$  と距離  $h$  は式 2.26 および式 2.27 によって定義される。

$$l = \frac{F^T \widetilde{m}'}{\|F^T \widetilde{m}'\|} \quad (2.26)$$

$$h = \widetilde{m}^T l \quad (2.27)$$

エピポーラ幾何は静止物体に対してのみ働くため、距離  $h$  および  $h'$  がゼロより大きな値であればあるほど、対応する 3 次元空間内の点は移動物体から抽出されたものである可能性が高い。このような  $h$  および  $h'$  の値の評価に関しては、節 2.5 で説明する。

### 2.3.3 フローベクター・バウンド

前述したエピポーラ幾何を利用した移動物体の識別は、カメラと移動物体が、3 次元空間内で平行に動いている場合には利用することができない。このような場合、移動物体はエピポーラ平面内を移動し、その画像上への投影はエピポーラ線に沿ったものとなるためである。また、カメラが並進していない場合にはエピポーラ平面が定義できないため、この場合もエピポーラ幾何した移動物体の識別は不可能である。このような状況下においても移動物体を正しく検出するために、もう一つの幾何学的な拘束条件を導入する。

いま、図 2.10 に示すように、3 次元空間内のある点  $M$  がカメラ中心  $C$  の画像上の 2 次元座標  $m$  に投影されているものとする。また、カメラ座標系  $C$  で測った点  $M$  の 3 次元座標を  $x$  とする。同様に、点  $M$  は別の視点に置かれたカメラ  $C'$  の画像上の 2 次元座標  $m'$  に投影されているものとする。ここで、カ

メラ  $C$  および  $C'$  から点  $M$  までの奥行き（深度）は等しいものと仮定し、その値を  $z$  とおく。ほぼ同じ高度を保って飛行しながら、上空から地上の移動物体を撮影する場合、この仮定は妥当である。以上の条件のもと、式 2.10 および式 2.15 より、式 2.28 および式 2.29 が成り立つ。

$$z\tilde{\mathbf{m}} = \mathbf{P}_{3 \times 3} [\mathbf{I} | \mathbf{0}] \tilde{\mathbf{x}} \quad (2.28)$$

$$z\tilde{\mathbf{m}}' = \mathbf{P}_{3 \times 3} [\mathbf{R} | \mathbf{t}] \tilde{\mathbf{x}} \quad (2.29)$$

なお、 $\mathbf{P}_{3 \times 3}$  はカメラの透視投影行列  $\mathbf{P}$ （式 2.12）から左上  $3 \times 3$  成分を抜き出した  $3 \times 3$  の行列である。また、 $\mathbf{t}$  はカメラ座標系  $C$  で測ったカメラ  $C'$  の位置（並進ベクトル）、 $\mathbf{R}$  は、座標系  $C$  で記述された座標を座標系  $C'$  での記述に変換する回転行列である。 $\tilde{\mathbf{a}}$  は、任意の座標  $\mathbf{a}$  の同次座標表現である。

式 2.28 および式 2.29 から  $\mathbf{x}$  を消去すると、式 2.30 を得る。

$$\mathbf{m}' - \mathbf{P}_{3 \times 3} \mathbf{R} \mathbf{P}_{3 \times 3}^{-1} \mathbf{m} = \frac{1}{z} \mathbf{P}_{3 \times 3} \mathbf{t} \quad (2.30)$$

式 (2.30) の左辺は、カメラの回転の効果を補正した、点  $M$  の投影のエピポーラ線に沿った移動量を表す。以降では、この左辺の値をフローベクターと呼ぶ。

ここで、深度  $z$  がとりうる最小値  $z_{min}$  と最大値  $z_{max}$  を設定することにより、式 2.31 のようにフローベクターの大きさ  $d$  に関する拘束条件を得ることができる。UAV を使って地上を撮影する際には、 $z_{min}$  および  $z_{max}$  の値は、UAV の高度を参照するなどして容易に設定することができよう。

$$d_{min} < d < d_{max} \quad (2.31)$$

ただし、

$$d = \|\mathbf{m}' - \mathbf{P}_{3 \times 3} \mathbf{R} \mathbf{P}_{3 \times 3}^{-1} \mathbf{m}\| \quad (2.32)$$

$$d_{min} = \frac{1}{z_{max}} \|\mathbf{P}_{3 \times 3} \mathbf{t}\| \quad (2.33)$$

$$d_{max} = \frac{1}{z_{min}} \|\mathbf{P}_{3 \times 3} \mathbf{t}\| \quad (2.34)$$

以上より、3次元空間内に固定された任意の点  $M$  から抽出された点は、フローベクターの絶対値に関する拘束条件（フローベクター・バウンド）である式 2.31 を必ず満足する。よって、フローベクター絶対値  $d$  と、その最大値  $d_{max}$  との差、および最小値  $d_{min}$  との差を、移動物体検出のもう一つの基準とすることができる。この  $d$  の値の評価については、節 2.5 で説明する。

## 2.4 外部パラメーターの算出

前述で説明した幾何学的拘束条件を利用するためには、カメラ座標系  $C$  および  $C'$  を関係づけるカメラの外部パラメーター（回転行列  $\mathbf{R}$  および並進ベクトル  $\mathbf{t}$ ）が必要である。本節では、このカメラの外部パラメーターをカメラの画像から推定する手法について説明する。

### 2.4.1 PTAM (Parallel Trackign And Mapping)

本研究では、カメラの画像を用いた SLAM により、この外部パラメーターを計算する。SLAM (Simultaneously Localization And Mapping) とは、ロボットが各種センサから取得した情報から、自己位置推定と地図作成を同時に行う手法のことである。未知の環境下でロボットが行動するとき、環境に

対する自己の位置を把握する必要がある。しかし、自己位置を推定するためには環境の地図が必要であり、逆に地図を得るためには自己位置を知る必要がある。各種センサを用いてこの矛盾する要求を解決する手法を総称して、SLAM と呼ぶ。

SLAM には様々なタイプが存在し、大きく3つの分類方法がある。まず、使用されるセンサによる分類としては、レーザーレンジファインダーや超音波センサなどの測距センサを用いる手法 [24] と、カメラを用いる手法 [25][26]、およびそれらの異種センサを組み合わせた手法 [27] の3種類がある。測距センサは周辺の壁や障害物までの距離を高速にスキャンすることができ、また、出力（スキャン方向と距離の集合）を処理することが比較的容易である。しかしながら、測距センサは環境中を2次的にスキャンするものが多く、UAV のように3次元空間内を移動するロボットには主たるセンサとしては適さない。それに対して、カメラを用いる SLAM は、3次的に情報を得ることが可能であり、UAV などの飛行ロボットにも適用が可能である。しかしながら、画像から特徴点を抽出したり、異なる画像間でそのマッチングを行う必要があるなど、センサの出力の処理にかかるコストが大きいという欠点がある。

次に、作成する地図の種類による分類としては、グリッド・ベースの SLAM とランドマーク・ベースの SLAM [28] がある。グリッド・ベースの SLAM では、作成する地図を格子状に表現する [29]。これに対して、ランドマーク・ベースの SLAM では、周辺環境に存在する特徴的な物体をランドマークとして認識し、そのランドマークの位置座標の集合としての地図を作成する [30]。

最後に、センサの出力を処理する方法による分類としては、フィルタリング・ベースの SLAM と、バッチ・ベースの SLAM が存在する。フィルタリング・ベースの SLAM では、拡張カルマンフィルタやパーティクルフィルタなどのフィルタリング手法によってセンサデータが得られるたびに逐一処理を行い、ロボット位置と環境地図を即座に更新する。対して、バッチ・ベースの SLAM では、異なる時刻に得られたセンサデータをまとめて処理する手法である。フィルタリング・ベースの SLAM はリアルタイムでの処理に適しており、代表的な手法である拡張カルマンフィルタを用いた EFK-SLAM[31] とパーティクルフィルタを用いた Fast-SLAM[32][33] は、多くのロボットの行動計画に適用されている。バッチ・ベースの SLAM では、センサデータをまとめて処理するため時間遅れが存在することが欠点ではあるが、画像間の対応づけをより正確に行うことができるため、推定結果がより正確なものとなりやすい。

本研究では、Klein らによって開発された PTAM (Parallel Trackign And Mapping)[12] によってカメラの外部パラメーターを計算する。PTAM は、カメラ画像を用いた SLAM 手法の1つであり、もともとはマーカーレス AR（拡張現実）への適用を目的として開発がなされた。しかし、そのロバストさと位置推定精度の高さから、飛行ロボットのナビゲーションへの応用も複数報告されている [34], [35], [36]。

PTAM はランドマーク・ベースの SLAM であり、カメラの画像中から特徴点の抽出と追跡（マッチング）をとることにより、これらの特徴点をランドマークとして3次元地図の作成を行う。PTAM では、撮影された画像に対して、キーフレームと呼ばれる画像に対してのみバッチ処理を行い、カメラの位置と環境地図の更新を行う。これに対して、キーフレーム以外の画像に対してはバッチ処理ではなく逐一処理を行い、カメラの位置のみを更新する。このバッチ処理と逐一処理はマルチスレッドとして実装されており、それぞれ独立して実行される。一般に、SLAM においては自己位置推定よりも地図更新の計算コストが格段に大きい。PTAM では、前述のように自己位置推定と地図更新の処理を分離することにより、高速な自己位置推定と、正確な地図作成の両立を実現している。また、地図の推定精度が高いだけでなく、PTAM は従来のフィルタリング・ベースの SLAM に比べて、位置の推定結果もより正確であるという研究結果も報告されている [37]。

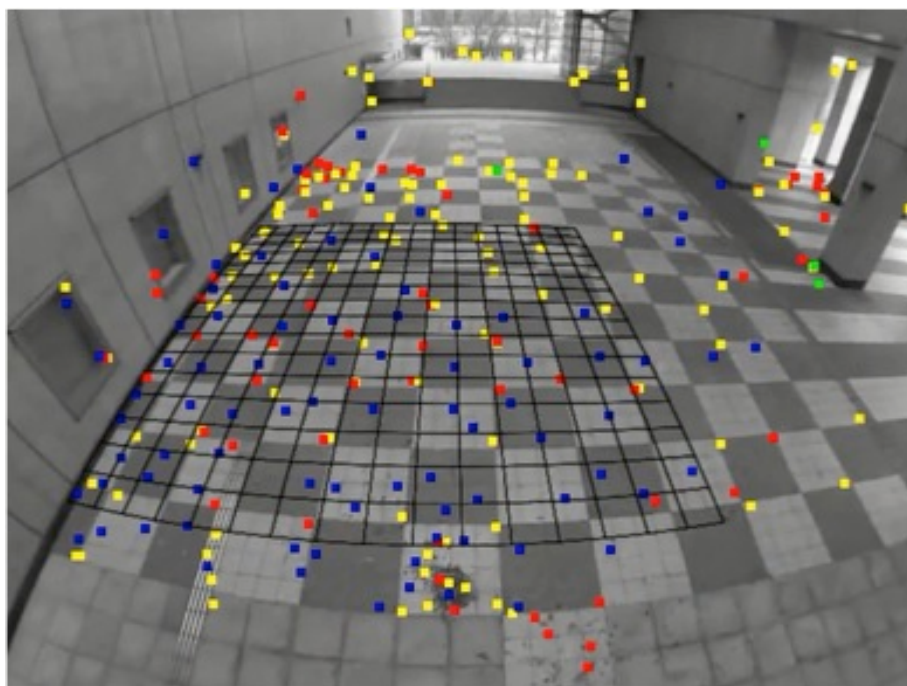


図 2.11 PTAM によってカメラ画像から抽出された特徴点

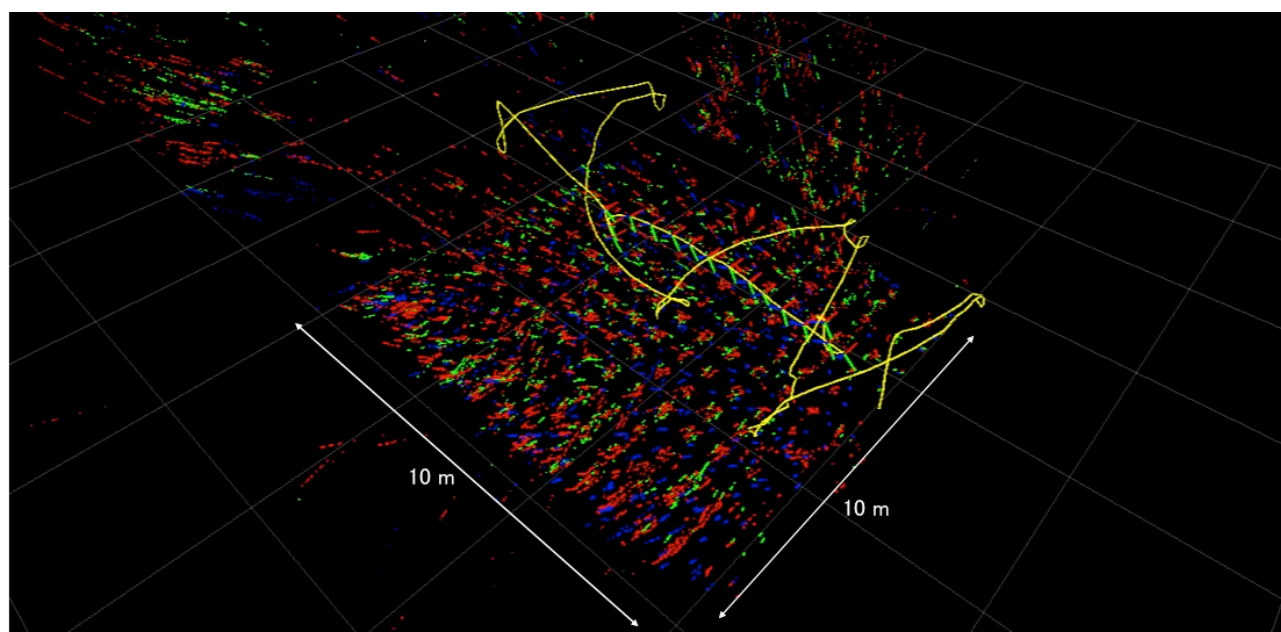


図 2.12 PTAM によって作成された環境地図とカメラの軌跡

図 2.11 には、PTAM によってカメラ画像から抽出された特徴点を示した。赤、緑、および青色の四角で示された点のそれぞれが、PTAM によって抽出された特徴点である。図 2.12 には、これらの特徴点をランドマークとして作成された環境地図を赤、緑、および青色の点群で、推定されたカメラの軌跡を黄色の曲線で表している。

### 2.4.2 クォータニオンによるカメラの姿勢演算

視点  $C$  で撮影した画像を PTAM に入力すると、ある基準座標系  $W$  で測ったカメラの位置と姿勢が出力される。この PTAM の出力（観測値）を  $z$  とおくと、 $z$  は式 2.35 で表される。なお、 $r$  はカメラの位置、 $q$  はカメラの姿勢を規定するクォータニオンである。

$$z = \begin{bmatrix} r \\ q \end{bmatrix} \quad (2.35)$$

同様に、視点  $C$  で撮影した画像を PTAM に入力した結果、基準座標系で測ったカメラの位置  $r'$  とカメラの姿勢を表すクォータニオン  $q'$  が観測値  $z'$  として得られるものとする。

$$z' = \begin{bmatrix} r' \\ q' \end{bmatrix} \quad (2.36)$$

ここで、クォータニオン  $q$  とは、回転軸（3 成分のベクトル  $\mathbf{q} = [q_1, q_2, q_3]^T$ ）と回転角（スカラー  $q_0$ ）からなる 4 成分で 3 次元空間内の任意の回転を表現する複素数を拡張した数体系である [38][39]。クォータニオンは 4 成分から成るが、そのノルムは 1 であるため独立な変数の個数は 3 であり、これは 3 次元空間内での回転の自由度 3 と同じである。

$$q = \begin{bmatrix} q_0 \\ \mathbf{q} \end{bmatrix} = \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} \quad (2.37)$$

$$\|q\|^2 = q_0^2 + \|\mathbf{q}\|^2 = q_0^2 + q_1^2 + q_2^2 + q_3^2 = 1 \quad (2.38)$$

3 次元空間内の回転は、オイラー角と呼ばれる基準座標系を基準とした 3 つの角度によって表現されることが多い。こうしたオイラー角による回転表現は直感的ではあるが、回転を定義することができない特異点が存在する。これに対して、クォータニオンによる回転の表現では、オイラー角による回転表現で生じる特異点が存在しない。この性質は、姿勢の自由度が大きい UAV を含む飛翔体や宇宙機の姿勢を表現する上では特に重要である。

PTAM から出力された観測値を用いると、カメラ  $C$  から  $C'$  に対する 外部パラメーターは、式 2.39、式 2.40 のように計算できる。

$$\mathbf{t} = -\mathbf{R}(q) \times (\mathbf{r}' - \mathbf{r}) \quad (2.39)$$

$$\mathbf{R} = \mathbf{R}(q^* \otimes q') \quad (2.40)$$

ここで、 $\mathbf{R}(q)$  は、基準座標系  $W$  で記述された座標を、クォータニオン  $q$  で定義される座標系で測った値に変換する回転行列であり、式 2.41 のように計算できる。

$$\mathbf{R}(q) = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1q_2 + q_0q_3) & 2(q_1q_3 - q_0q_2) \\ 2(q_1q_2 - q_0q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2q_3 + q_0q_1) \\ 2(q_1q_3 + q_0q_2) & 2(q_2q_3 - q_0q_1) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix} \quad (2.41)$$

$q^*$  は,  $q$  の共役クォータニオン (逆クォータニオン) を表す. 共役クォータニオンとは, あるクォータニオンとは逆向きの回転を表すクォータニオン, すなわちベクトル部の正負を逆転したものである.

$$q^* = \begin{bmatrix} q_0 \\ -\mathbf{q} \end{bmatrix} = \begin{bmatrix} q_0 \\ -q_1 \\ -q_2 \\ -q_3 \end{bmatrix} \quad (2.42)$$

また, 記号  $\otimes$  は, クォータニオン積を表す. 2つの任意のクォータニオン  $q, p$  に関して, クォータニオン積は式 2.43 のように定義できる.

$$\begin{aligned} q \otimes p &= \begin{bmatrix} q_0 & -q_1 & -q_2 & -q_3 \\ q_1 & q_0 & -q_3 & q_2 \\ q_2 & q_3 & q_0 & -q_1 \\ q_3 & -q_2 & q_1 & q_0 \end{bmatrix} \begin{bmatrix} p_0 \\ p_1 \\ p_2 \\ p_3 \end{bmatrix} \\ &= \begin{bmatrix} p_0 & -p_1 & -p_2 & -p_3 \\ p_1 & p_0 & p_3 & -p_2 \\ p_2 & -p_3 & p_0 & p_1 \\ p_3 & p_2 & -p_1 & p_0 \end{bmatrix} \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} \end{aligned} \quad (2.43)$$

式 2.40 について少し捕捉すると,  $q^* \otimes q'$  は, カメラ座標系  $C$  を基準とするカメラ  $C'$  の相対姿勢を表すクォータニオンである. したがって,  $\mathbf{R}(q^* \otimes q')$  とは, カメラ座標系  $C$  で記述された座標を, カメラ座標系  $C'$  で測った値に変換する回転行列に他ならない.

## 2.5 幾何学的拘束条件の評価

本節では, 前述した2つの幾何学的拘束条件の遵守の度合いから, 対象となる特徴点が移動物体から抽出されたものである確率を計算する評価関数を定義する.

前節で述べた通り, 幾何学的拘束条件を計算するためには, カメラの外部パラメーター (回転行列  $\mathbf{R}$  および並進ベクトル  $\mathbf{t}$ ) が必要であり, これらの値は PTAM によって出力された観測値から計算される. しかし, 観測値には計測誤差が存在し, このような計測誤差の影響を考慮しない場合, 外部パラメーターの誤差により, 移動体検出の精度低下につながる. 本研究では, 3次元空間中の点  $M$  が環境中に固定された点である確率を, 与えられた観測値  $\mathbf{z}, \mathbf{z}'$  に対する条件付き確率  $P(M = \text{static} | \mathbf{z}, \mathbf{z}')$  として定義することにより, このような問題に対処する. この条件付き確率を直接計算可能とするため, 以下のような変形を行う.

$$\begin{aligned} P(M = \text{static} | \mathbf{z}, \mathbf{z}') \\ = \sum_{\mathbf{R}, \mathbf{t}} P(M = \text{static} | \mathbf{R}, \mathbf{t}) P(\mathbf{R}, \mathbf{t} | \mathbf{z}, \mathbf{z}') \end{aligned} \quad (2.44)$$

条件付き確率  $P(\mathbf{R}, \mathbf{t} | \mathbf{z}, \mathbf{z}')$  は, 計測モデルから計算することができる. なお, 本研究では PTAM の出力を使用しているが, IMU やその他のセンサの出力から外部パラメーターを計算することも可能である.

条件付き確率  $P(M = \text{static} | \mathbf{R}, \mathbf{t})$  は, エピポーラ拘束および, フローベクター・バウンドの遵守度合いに基づいて, 対象の点が3次元空間内の静止物体の投影である確率を与える関数である. この条件付き確率を, 本研究では式 2.45 のように定義する.



$$P(M = static|\mathbf{R}, \mathbf{t}) = f_{EP} \times f_{FV} \quad (2.45)$$

ここで、 $f_{EP}$  はエピソード拘束の遵守度合いを条件付き確率に反映する関数であり、 $f_{FV}$  はフローベクター・バウンドの遵守度合いを条件付き確率に反映する関数である。幾何的な拘束から類似の条件付き確率をモデル化する試みは、Kundu らによってなされている [40]。Kundu らによって考案されたモデルを参考にして、関数  $f_{EP}$  および関数  $f_{FV}$  を、以下のように定義する。

$$f_{EP} = e^{-\alpha(h+h')} \quad (2.46)$$

$$f_{FV} = \frac{1}{1 + \left(\frac{d-d_{mean}}{d_{range}}\right)^{2\beta}} \quad (2.47)$$

ただし、

$$d_{mean} = \frac{d_{min} + d_{max}}{2} \quad (2.48)$$

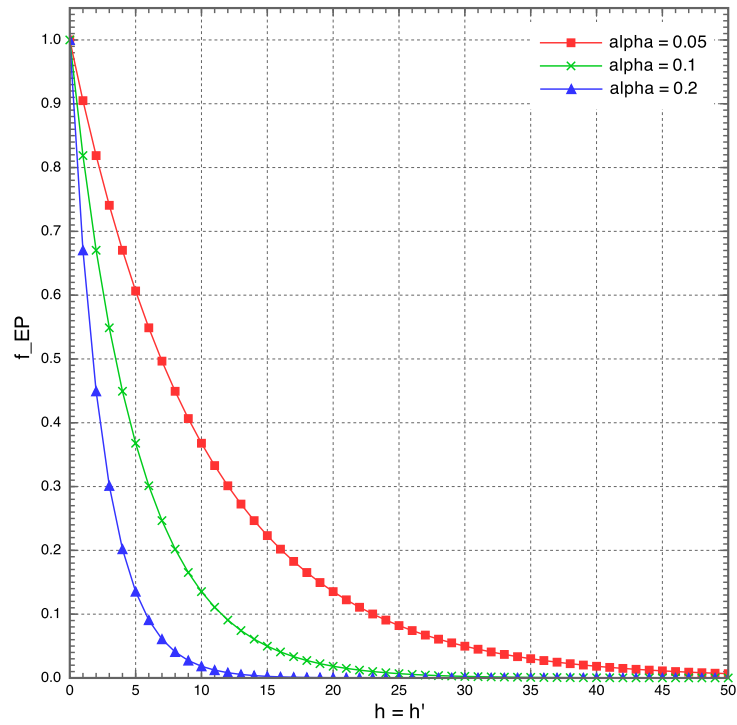
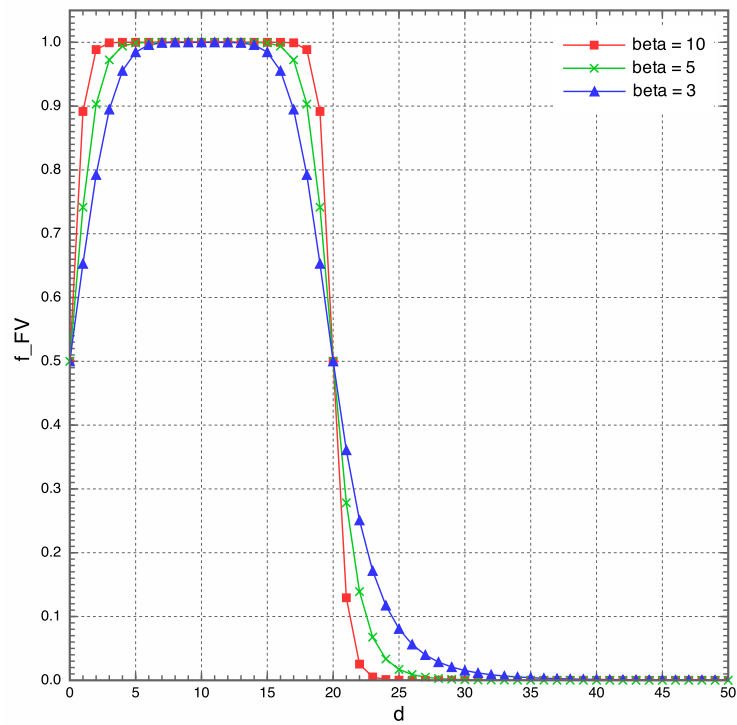
$$d_{range} = \frac{d_{max} - d_{min}}{2} \quad (2.49)$$

なお、 $\alpha, \beta$  はスムージングファクターであり、本研究では  $\alpha = 0.1$ ,  $\beta = 10$  程度の値に設定した。 $d_{min}$  および  $d_{max}$  の定義については、式 2.33 および式 2.34 の通りである。

図 2.13 は、 $h = h'$  の仮定のもと、エピソード線からの距離  $h(= h')$  の値を横軸に、関数  $f_{EP}$  の値を縦軸にとったグラフである。 $f_{EP}$  の値は、距離  $h$  に対して単調減少する。すなわち、対象の点がエピソード線から離れれば離れるほど、その点が移動物体に属する点である確率が高いということを示唆する。

同様に、図 2.14 は、 $d_{min} = 0$ ,  $d_{max} = 20.0$  の仮定のもと、フローベクターの長さ  $d$  の値を横軸に、関数  $f_{FV}$  の値を縦軸にとったグラフである。 $f_{FV}$  の値は、フローベクターの長さ  $d$  が  $d_{min} < d < d_{max} = 20.0$  の範囲内にあるときには 1 に近い値をとるが、その範囲を外れると急激に 0 に近づく。すなわち、フローベクターの長さが、想定範囲内から離れれば離れるほど、その点が移動物体に属する点である確率が高いということを示唆する。

以上の定義に基づいて、式 2.44 により計算された条件付き確率  $P(M = static|\mathbf{z}, \mathbf{z}')$  が、あらかじめ定めた閾値を下回った点  $M$  を、3次元空間内を移動している物体に属する点として識別する。本研究では、この条件付き確率の閾値を 0.7 に程度の値に設定した。

図 2.13 エピポーラ拘束の評価関数  $f_{EP}$  の形状図 2.14 フローベクター・バウンドの評価関数  $f_{FV}$  の形状 ( $d_{min} = 0$ ,  $d_{max} = 20.0$ )

## 2.6 バルーン空撮画像への適用のための拡張

上で説明した手法は、移動物体が画像の中である程度の面積を占めている場合には有効に働くが、移動物体の画像内での大きさが小さすぎる場合にはうまく移動物体を検出することができない。なぜならば、移動物体の面積が小さすぎる場合、画像から特徴点を抽出する際に、移動物体から十分な数の特徴点が抽出されないからである。これは、バルーンのように比較的高高度を飛行するプラットフォームを利用する場合に問題となる。このような広域画像に対しては、以下で説明する前処理を施してから提案手法を適用することにより、移動物体の識別を行う。

### 2.6.1 バルーン空撮画像に対する前処理の概要

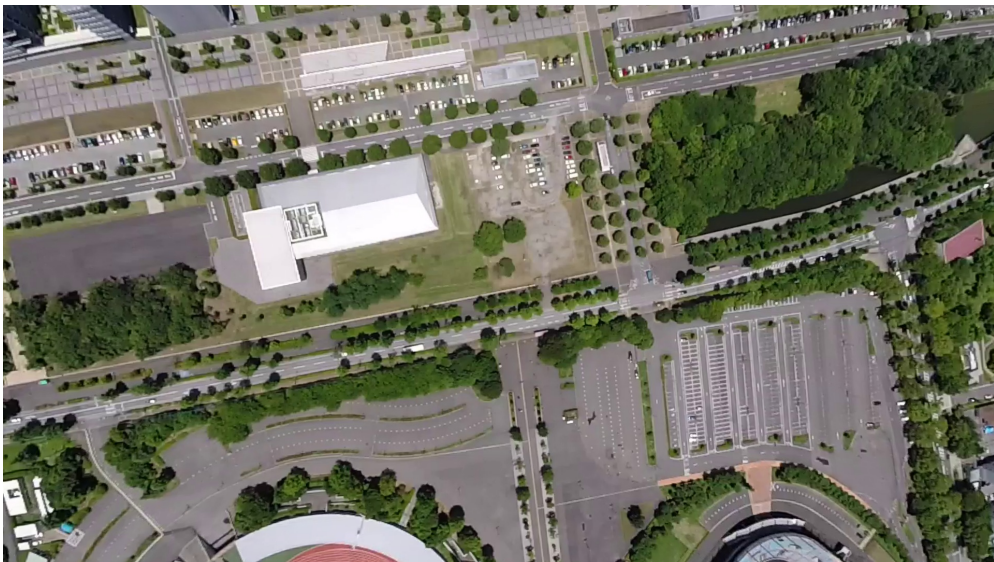
この前処理においては、バルーンから一定のインターバルをおいて撮影した3枚の画像  $I_1$ ,  $I_2$ ,  $I_3$  を用いる。いま、例として図 2.15, 図 2.16, 図 2.17 に示す3枚の画像から移動物体を識別することを考える。図 2.15, 図 2.16, 図 2.17 からは、風の影響などによりバルーンの位置が変化しているため、画面内の視野に変化が現れていることがわかる。

この3枚の画像に対する前処理においては、まず1枚目の画像  $I_1$  を基準とした投影変換を行い、バルーンによる画面内の視野の変化を補正する。この投影変換においては、地面とその上に存在するオブジェクトを1つの平面と仮定した上でホモグラフィ行列を計算する。次に、こうして得られた補正画像  $I_2'$ ,  $I_3'$  と画像  $I_1$  に対してフレーム間差分を行い、 $I_2'$  における移動物体の候補領域を計算する。

後述するように、この前処理によって得られた候補領域には投影変換の誤差などに起因するノイズが多く含まれているが、提案手法で行う特徴点の抽出をこの候補領域に限定することにより、十分な個数の特徴点を移動物体の領域から抽出することができ、移動物体の検出率が向上する。



図 2.15 バルーンによる1枚目の空撮画像  $I_1$

図 2.16 バルーンによる 2 枚目の空撮画像  $I_2$ 図 2.17 バルーンによる 3 枚目の空撮画像  $I_3$ 

以降では、前処理における投影変換とフレーム間差分について、詳細を述べる。

### 2.6.2 平面の投影変換による幾何補正

3 枚の画像  $I_1$ ,  $I_2$ ,  $I_3$  の視野の変化を補正するため、 $I_2$ ,  $I_3$  に対して、 $I_1$  を基準としたホモグラフィーによる投影変換を施す。ホモグラフィーとは、射影変換を用いてある平面を別の平面に射影することであり、二次元画像の変形などに使用される [41] [42]。

ホモグラフィーによる投影変換には、ホモグラフィー行列と呼ばれる 8 自由度の  $3 \times 3$  行列を求める必要がある。変換を行いたい 2 枚の画像において、1 つの点と点の関係がわかれば 2 本の独立な方程式を立てることができるため、4 組の点の対応関係を求めることができれば、ホモグラフィー行列を計算することができる。この前処理においては、KLT-Tracker によって特徴点の抽出と追跡を行い、追跡に成功し



たすべての特徴点と、ロバストなパラメータ推定手法である RANSAC[43] を用いることにより、ホモグラフィー行列の推定を行った。

ホモグラフィーによって  $I_2$  と  $I_3$  を投影変換した結果（画像  $I'_2$  と画像  $I'_3$ ）を、図 2.18、図 2.19 に示す。図 2.15 に示した画像  $I_1$  と、画像  $I'_2$  および画像  $I'_3$  を比較すると、バルーンの視野の変化が補正されている、つまり画面内でのオブジェクトの位置がほとんど一致するように投影変換が行われていることが確認できる。

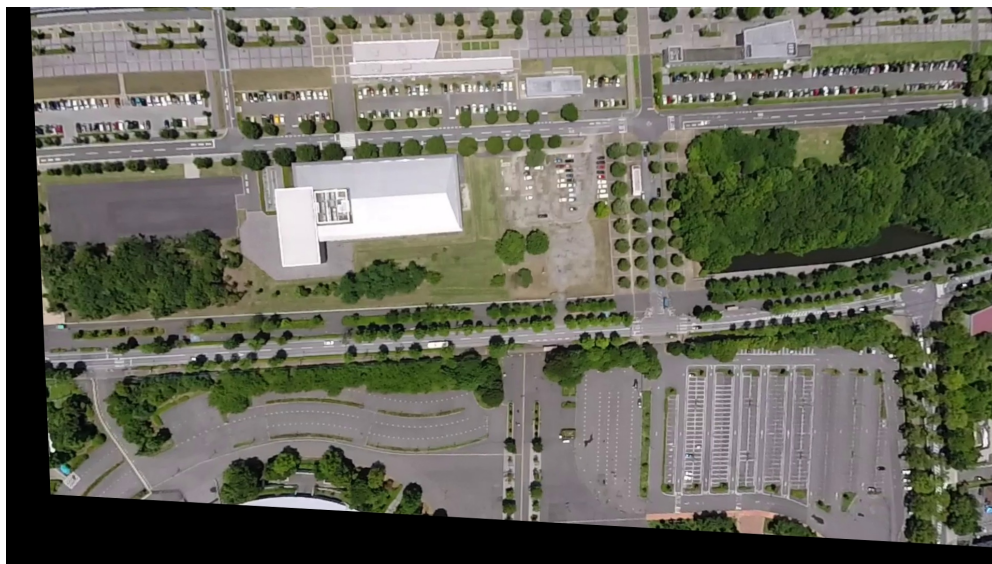


図 2.18 バルーンによる 2 枚目の空撮画像の投影変換結果  $I'_2$



図 2.19 バルーンによる 3 枚目の空撮画像の投影変換結果  $I'_3$

### 2.6.3 フレーム間差分法による移動物体候補領域の抽出

こうして視野の変化補正が行われた3枚の画像に対してフレーム間差分を適用することにより、画像  $I_2$  における移動物体の候補領域を抽出する。ここで、フレーム間差分とは、ある固定点からインターバルをおいて撮影した3枚の画像を用いて画面内の移動物体領域を計算する手法である [44]。

フレーム間差分においては、まず、撮影した3枚の画像において、1枚目の画像と2枚目の画像、2枚目の画像と3枚目の画像でそれぞれ各ピクセルの輝度値の差分を計算する（こうして得られた差分の2次元配列を、差分画像と呼ぶ）。次に、それぞれの差分画像において、各ピクセルの値に閾値処理を施し、差分画像の2値化を行う。こうして得られた差分画像は、もとの2枚の画像の動領域の和集合となっているため、差分画像1枚だけでは1つの画像内の移動物体のみを検出することができない。そこで、得られた2つの2値化された差分画像に対して、各ピクセルの論理積をとることによって、2枚目の画像中における移動物体領域のみを得る。

このフレーム間差分を、画像  $I_1$ ,  $I_2$ ,  $I_3$  に対して行った結果を、図 2.20 に示した。図 2.20 には、フレーム間差分によって計算された、画像  $I_2$  における移動物体の候補領域以外の領域を黒塗りで示している。図 2.20 と図 2.18 を比較すると、画像の中心を横に走る道路上において、乗用車と大型バスの動領域がうまく抽出できていることがわかる。また、画像の中央よりやや右を縦に走る道路上においても、2台の乗用車の動領域が抽出できている。この画像には、主に投影変換の誤差に起因するノイズが多く含まれているが、このような前処理によって計算された動領域に対してのみ特徴点の抽出を行うことにより移動物体からより多くの特徴点を抽出でき、移動物体の検出率を向上させることができる。



図 2.20 前処理によって得られた  $I_2$  における移動物体の候補領域

## 第 3 章

# 実験と精度検証

歩行者や自動車が往来する場所で実際に小型のカメラを搭載した UAV を飛行させ、提案手法の評価を行った。本章では、実験のコンフィギュレーションおよび、取得画像と提案手法に対して行った精度評価の方法とその結果について詳細を述べる。

### 3.1 実験のコンフィギュレーション

実験に使用する UAV として、Parrot 社が販売する AR.Drone 2.0 を使用した。AR.Drone 2.0 は、大きさ約 45cm×45cm×20cm、重量約 500g のホビー用途の小型 UAV である。その外観を、図 3.1 に示す。AR.Drone 2.0 は WiFi を通じて、スマートフォンやラップトップから操縦を行ったり、内蔵カメラの画像やセンサーデータなどのテレメトリを取得することができる。この実験を行うにあたって、ラップトップのキーボード入力によって AR.Drone 2.0 を操作し、また一定周期でセンサーデータを取得する Windows アプリケーションを開発した。

AR.Drone 2.0 には機体前方と下方にそれぞれ 1 つずつカメラが搭載されており、飛行中にその画像を取得することができる。しかし、この画像は WiFi を通じて送られてくるため、電波の状況によってコマ落ちが発生することがある。今回の実験では、画像の取得を確実なものとするため、GoPro 3 という小型のアクションカメラを AR.Drone 2.0 に搭載した。



図 3.1 AR.Drone 2.0

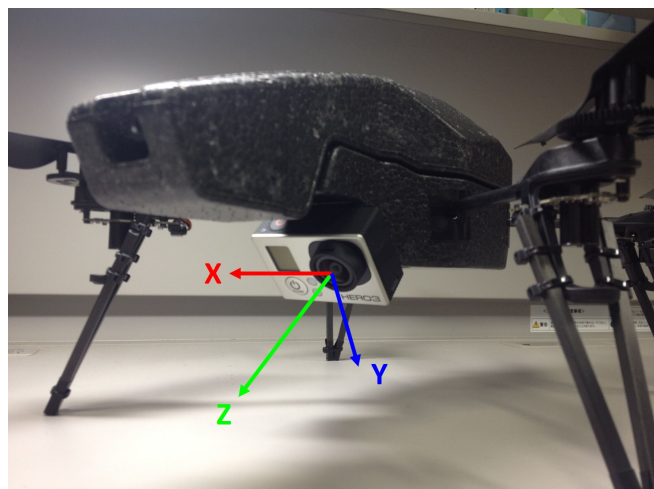


図 3.2 カメラの搭載方法およびカメラ座標系



図 3.2 に示すように、Gopro 3 は AR.Drone 2.0 の前方から下斜め 45 度程度の方角に向けて搭載されている。図 3.2 には、この実験で定義するカメラ座標系を合わせて示した。すなわち、カメラの光軸方向が  $Z$  方向、右方が  $X$  方向、下方が  $Y$  方向である。飛行中、ビデオモードにより毎秒 15 フレームの速度で撮影された動画データを内蔵の SD カードに保存し、実験の終了後に、動画データの画像化、640pixel×480pixel サイズへのリサンプリング、およびレンズの歪み補正を行う。

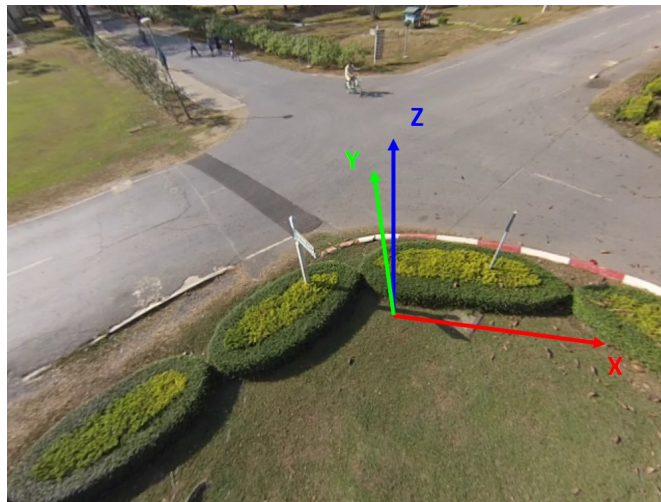


図 3.3 UAV を飛行させる場所および世界座標系

図 3.3 に示すような 2 車線の道路が交わる交差点にて、UAV を飛行させた。図 3.3 には、環境に固定された世界座標系を示している。今後、とくに断りのない限り、UAV の位置や速度、姿勢はこの世界座標系を基準とする。

墜落など万が一の場合を考慮し、交差点の真上を飛ぶことは避け、手前の芝生の上空から、交差点を斜め下に見下ろすように飛行させた。また、接触すると怪我の恐れがあるプロペラ部位に関しては専用の発泡スチロール製ガードを装着し、万全を期した。

## 3.2 実験結果

前述のような設定の下、飛行実験を行った。図 3.4、図 3.5、および図 3.6 には、UAV 搭載の Gopro 3 によって毎秒 15 フレームの速度で撮影された画像を、10 フレーム毎に抜粋して示している。

図 3.7 には、UAV の飛行軌跡を世界座標の  $XY$  平面（地面）に投影したものを示す。飛行領域は、およそ  $3\text{m} \times 4.5\text{m}$  の領域に収まっていることがわかる。また、図 3.8 には、UAV の飛行高度の時間推移を示した。飛行高度は、おおむね  $6\text{m}$  から  $8\text{m}$  である。これらの位置情報は、Gopro 3 の画像を PTAM に入力して得られた値を使用している。

図 3.9 には、UAV の飛行速度の時間推移を示す。UAV は地上のラップトップからの制御入力や、風などの外乱の影響により、空中で完全に静止することはできない。速度の範囲はどの方向も  $\pm 2\text{m}$  の範囲に収まっているが、最初の 40 フレームは制御入力によって水平方向の動きが大きくなっており、またフレーム 150 からフレーム 210 の間は風によって高さ方向の動きが大きくなっている。

最後に、図 3.10 には、UAV の姿勢（クォータニオン）の時間推移を示した。図 3.10 から、姿勢に関してはそれほど大きな変化はなく、ほぼ一定の姿勢を保ちながら飛行していることがわかる。



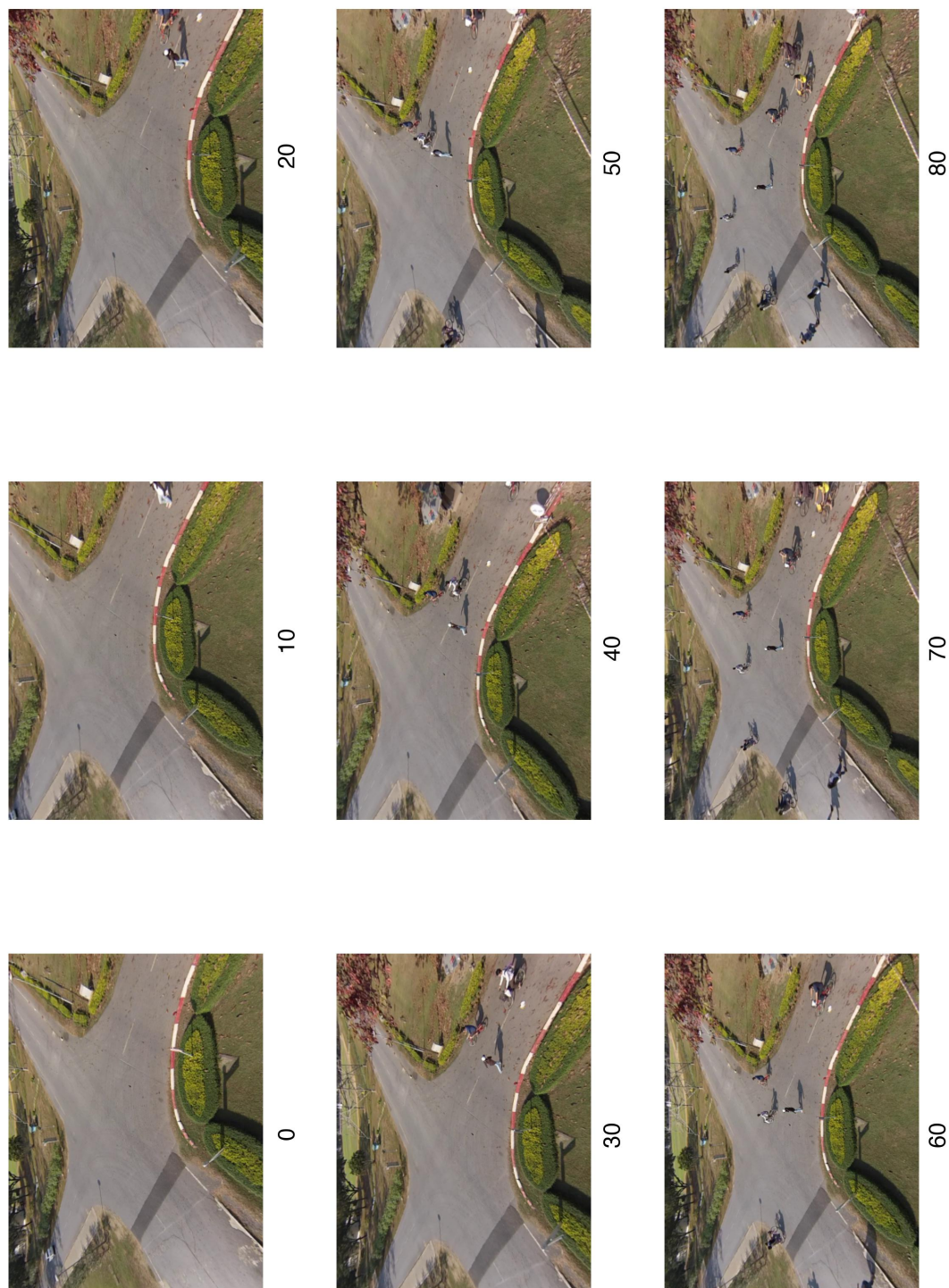


図 3.4 UAV により撮影された画像の抜粋 (フレーム 0 から 80)

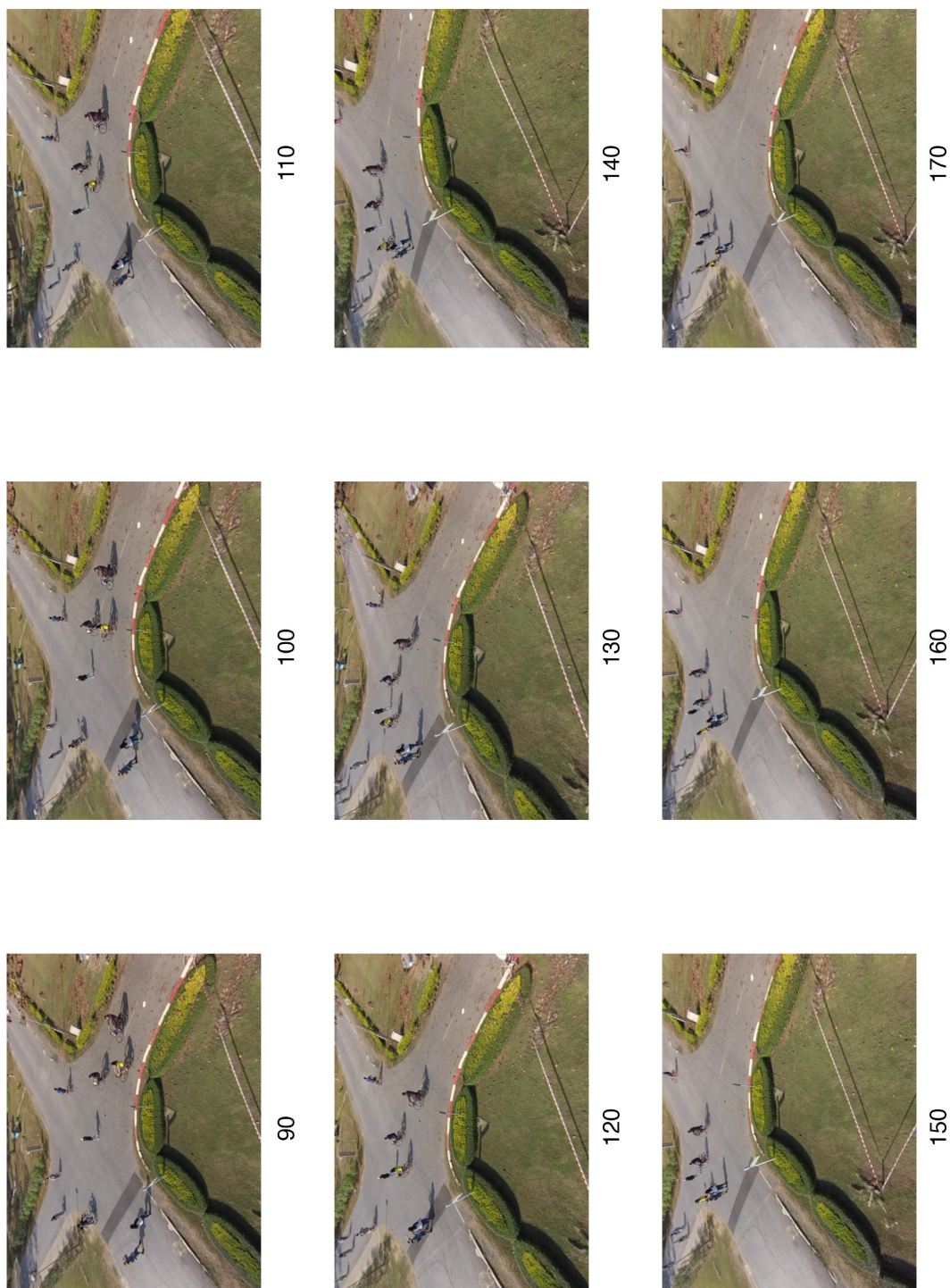


図 3.5 UAV により撮影された画像の抜粋 (フレーム 90 から 170)



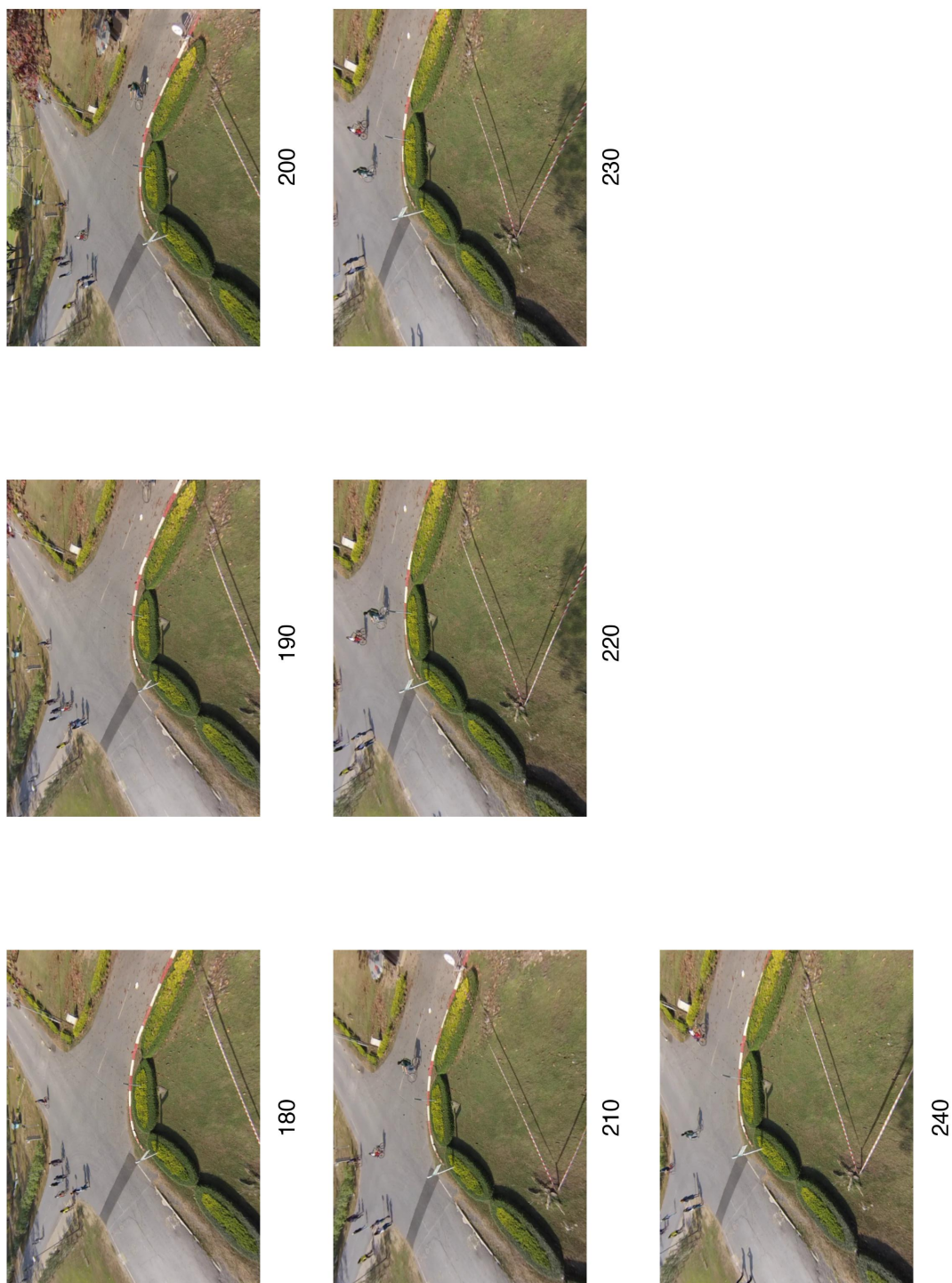


図 3.6 UAV により撮影された画像の抜粋 (フレーム 180 から 240)

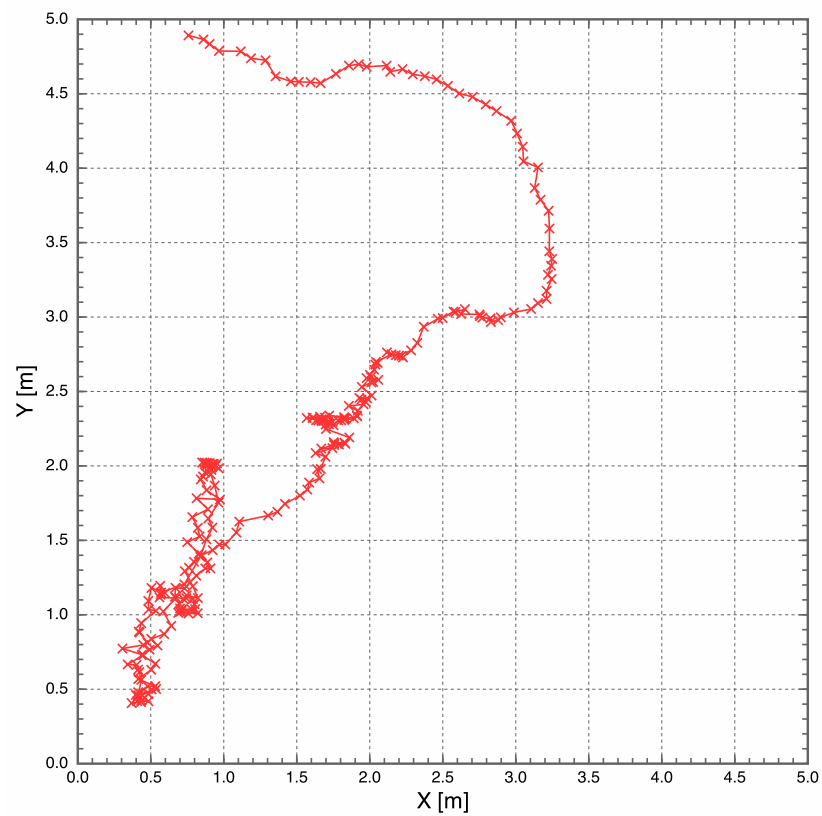


図 3.7 UAV 飛行軌跡の XY 平面への投影 ( $x=0.75$ ,  $y=4.95$  がフレーム 0 における位置)

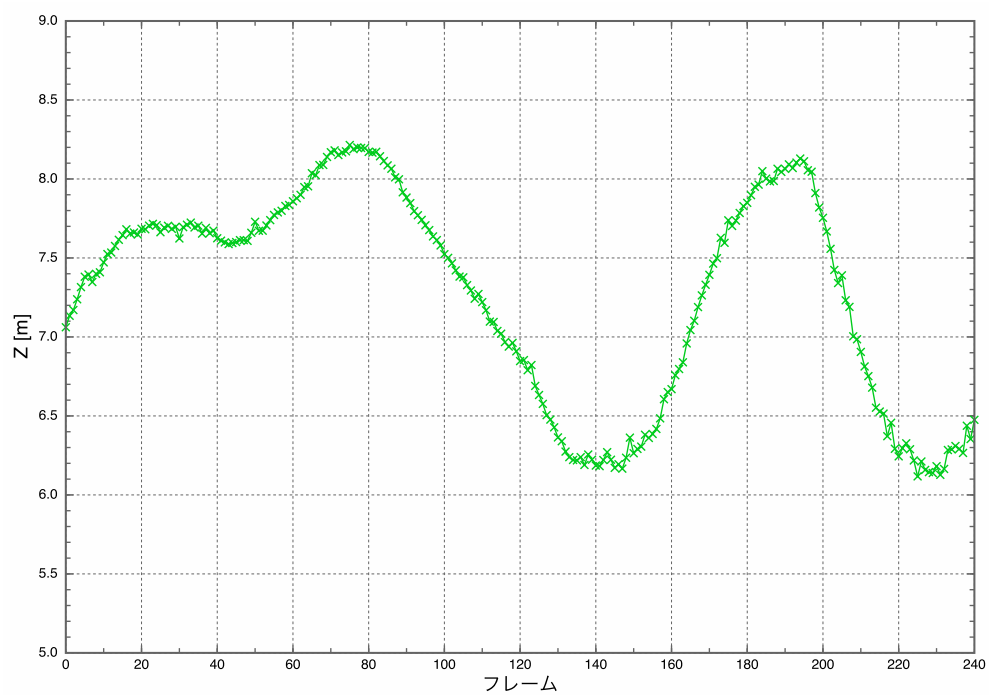


図 3.8 UAV の飛行高度の時間推移

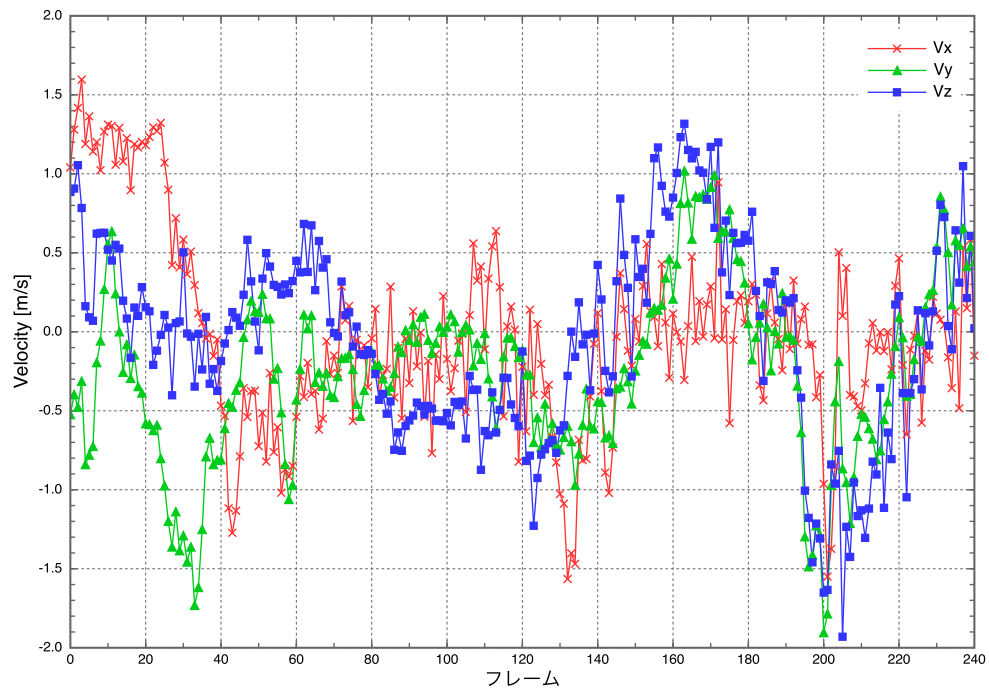


図 3.9 UAV の飛行速度の時間推移

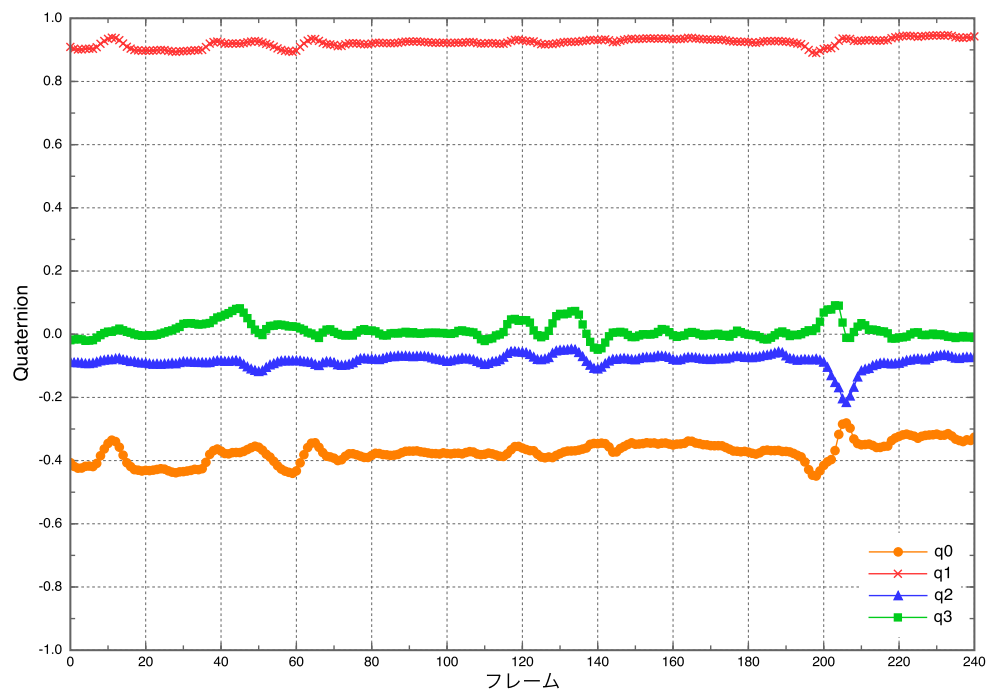


図 3.10 UAV の姿勢（クォータニオン）の時間推移

### 3.3 精度評価と考察

前述の画像および UAV の位置・姿勢データに対して提案手法を適用し、追跡に成功した特徴点のそれぞれについて、移動物体/静止物体の分類を行った。本節では、この移動体検出の精度評価について、その方法と結果を述べる。

表 3.1 True Positive/Negative および False Positive/Negative の定義

分類結果 \ 実際の値	移動物体	静止物体
移動物体	True Positive (真陽性)	False Positive (偽陽性)
静止物体	False Negative (偽陰性)	True Negative (真陰性)

このような分類問題に対して、一般に実際の値と分類結果の組み合わせによって、表 3.1 のような 4 つのグループを定義する。すなわち、陽性および陰性の 2 つの値を持ちうるものを、その 2 つのどちらかに分類する問題を考える。実際に陽性であるものを陽性に分類したとき、この結果は真陽性 (True Positive) と呼ばれる。逆に、実際には陽性であるものを陰性に分類したとき、この結果は偽陽性 (False Positive) と呼ばれる。同様に、実際に陰性であるものを陰性に分類したときは真陰性 (True Negative)、実際には陰性であるものを陽性に分類したときは偽陰性 (False Negative) という。今回の場合、それぞれの特徴点は移動物体から抽出されたもの (=陽性)、もしくは静止物体から抽出されたもの (=陰性) という 2 つの値を持つ。

このような分類を行う手法を評価するとき、一般に Precision および Recall という 2 つの評価指標が利用できる。Precision は、陽性と分類したもののうち、実際にどれだけ陽性が含まれるかを割合で示したものであり、式 3.1 によって定義される。一方、Recall は、実際の陽性の数に対して、陽性であるとどれだけ正しく分類できたかを割合で示したものであり、3.2 によって定義される。

$$Precision = \frac{TruePositive}{TruePositive + FalsePositive} \quad (3.1)$$

$$Recall = \frac{TruePositive}{TruePositive + FalseNegative} \quad (3.2)$$

Precision と Recall はそれぞれ 0 から 1 までの値をとり、信頼できる分類手法はどちらも高い値を示す。この 2 つの評価指標を統合した指標として、式 3.3 に示すような値  $F$  が定義される。 $F$  は 0 から 1 までの値をとり、その値が大きいほど、その分類手法は信頼できるといえる。

$$F = \frac{2}{\frac{1}{Precision} + \frac{1}{Recall}} \quad (3.3)$$

図 3.11 には、各フレームにおいて、それぞれの特徴点を提案手法によって移動物体 (=陽性) と静止物体 (=陰性) のいずれかに分類した結果の Precision および Recall を示した。図 3.12 には、この Precision と Recall を用いて計算された各フレームにおける信頼度係数  $F$  の値を示した。

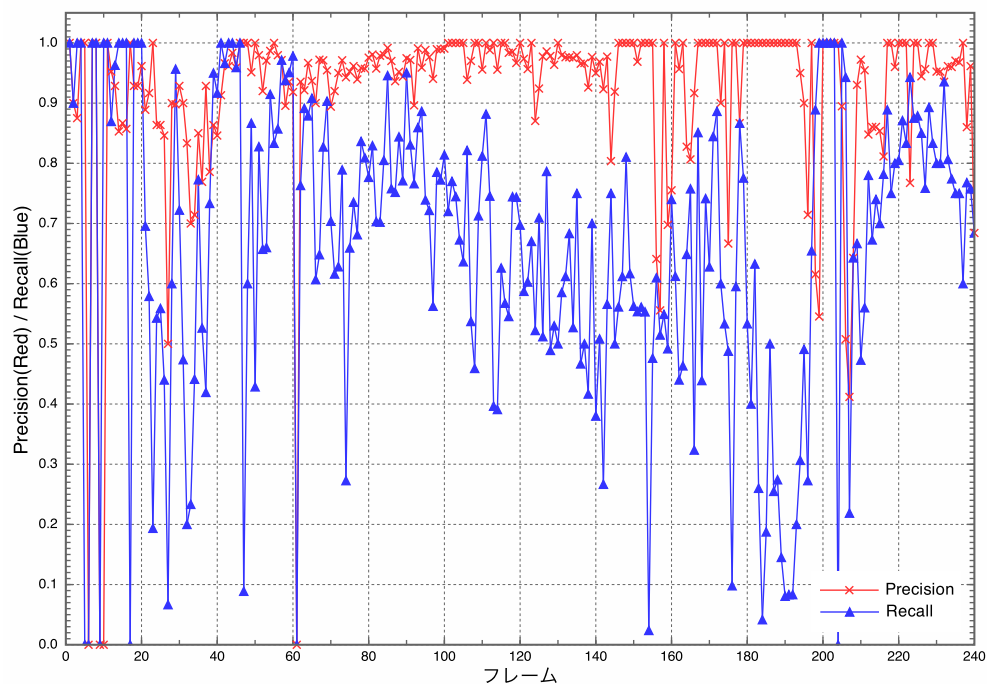


図 3.11 Precision と Recall の時間推移

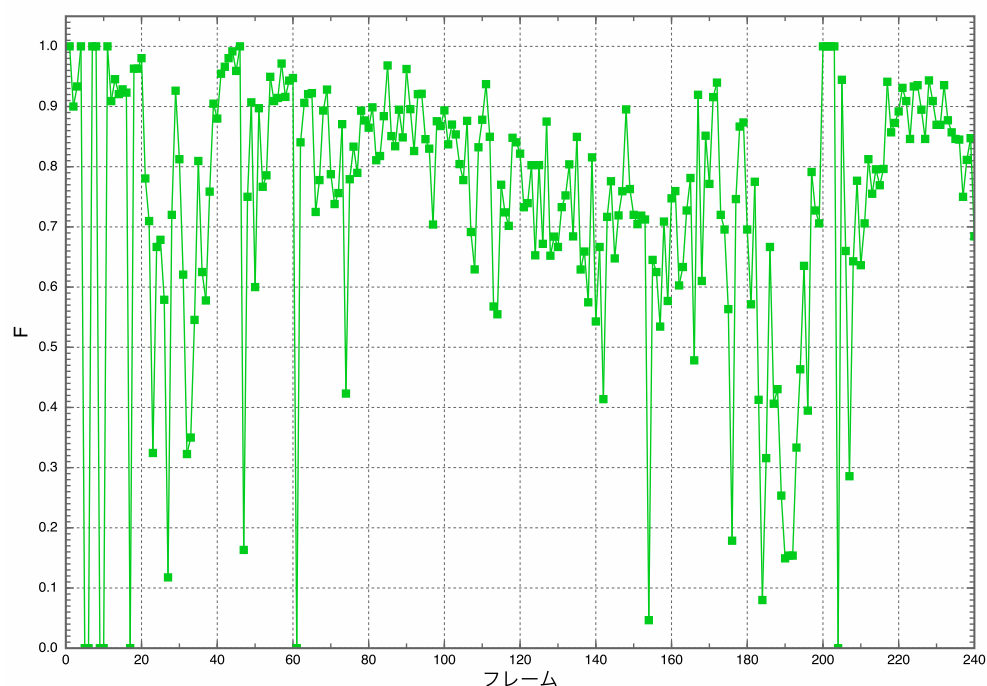


図 3.12 評価値 F の時間推移

図 3.5 より、画面内に移動物体が多いのはフレーム 70 からフレーム 130 の区間であり、図 3.11 より、この間、Precision は 1 に近い値、Recall も平均して 0.65 程度の値をマークしている。すなわち、移動物体として分類された結果に誤回答はかなり少なく、さらに、割合にして 68% 程度の移動物体を識別できている。また、3.9 より、この間、UAV は平均して毎秒 0.65m 程度、瞬時毎秒 1.5m 程度の速度で動い



ている。参考として、この区間に属するフレーム 77 における分類結果を、図 3.13 および図 3.14 に示す。前述のとおり、移動物体に属する特徴点のうち 70% 程度の識別にとどまっているが、すべての移動物体から 1 点以上の特徴点を正しく検出できており、誤回答は少ない。こうした点を踏まえて、移動物体が画面内に特に多い区間において、提案手法は有効に働いているといえる。



図 3.13 フレーム 77 における特徴点（赤）とその追跡結果（緑）



図 3.14 フレーム 77 において移動物体として識別された特徴点（赤）

その一方で、以下のような傾向がみられる。

- 最初の 60 フレームで、Precision がところどころ 0 となる箇所がある
- 最初の 80 フレームで、Recall がところどころ 0.5 以下の低い値をとる
- フレーム 90 からフレーム 190 の区間では、Precision の値はおおむね高いまま保たれているが、Recall の値が次第に減少している

以下では、それぞれの原因について考察を行う。まず、最初の 60 フレームにおいて Precision がところどころ 0 となる問題については、移動体に属する特徴点の個数が 0 であることが原因である。図 3.15 には、各フレームにおける追跡に成功した特徴点の個数と、そのうち移動物体に属するものの個数を示している。最初の 80 フレームにおいて、図 3.11 と図 3.15 を見比べると、Precision がゼロとなる場所では、移動物体に属する特徴点が 0 個である。移動体に属する特徴点の個数が 0 である場合、True Positive は必然的に 0 個となり、False Positive が 1 つでもあれば Precision の値は 0 になってしまう。実際、Precision が 0 となっている箇所では False Positive が 1 個から 2 個程度存在するため、このような結果となっている。しかし、実用上、False Positive が 1 個から 2 個というのは大きな問題とはなりえないと考えられる。

次に、最初の 80 フレームで、Recall がところどころ低い値をとる問題については、この間、UAV とそれぞれの移動物体の動きがほとんど平行であることが原因である。図 3.4 より、この間、移動体のほとんどは画面の右下から左上に向かって移動している。一方、図 3.3 の世界座標系とにおけるこの移動物体の動きと、図 3.7 の UAV の動きを見比べると、両者はほとんど平行に動いている。この場合、移動物体は UAV の視点で作るエピポーラ平面内を動くため、エピポーラ拘束による移動物体の識別が有効に働かない。UAV をいろいろな方向に動かすといった行動計画を組み込むなどすれば、このような問題は解決できると考えられる。



最後に、フレーム 90 からフレーム 190 の区間では、Recall の値が次第に減少するという問題に関しては、図 3.4 および図 3.5 からわかるとおり、この間においては移動物体がカメラから次第に離れていき、それとともに画面内での移動物体の動きが徐々に小さくなっていくことが原因である。今回は安全のため斜め上から見下ろすように空撮を行っているが、安全面の問題をクリアした上で移動物体を真上から撮影できれば、このような問題は解決できると考えられる。

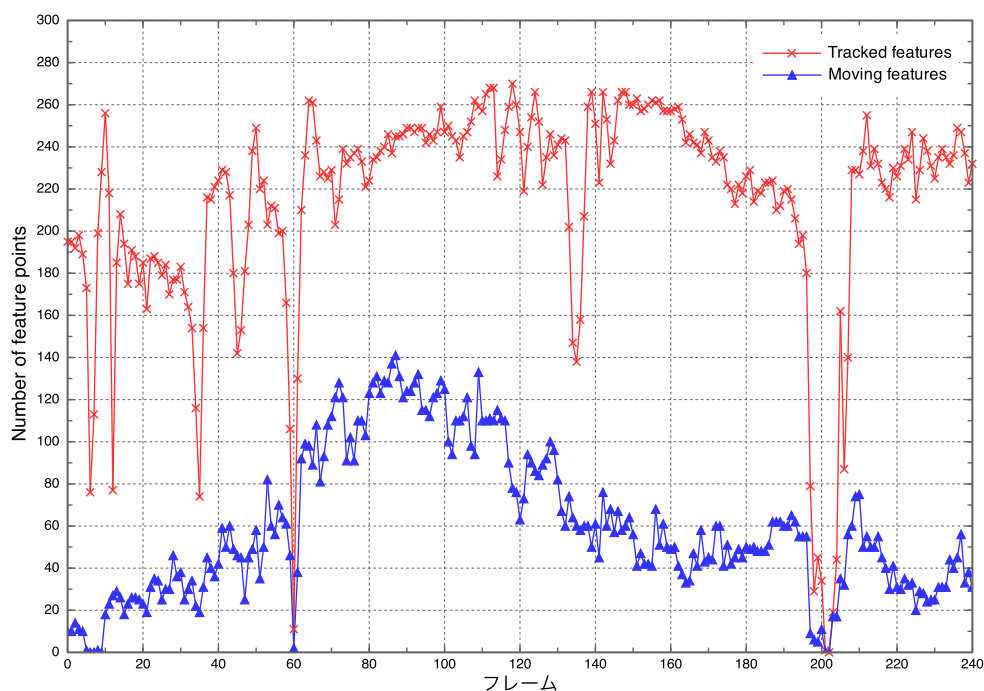


図 3.15 追跡に成功した特徴点の個数（赤）と移動物体に属する特徴点の個数

最後に、処理時間に関しては、サイズ 640pixel×480pixel の画像を、表 3.2 で示すようなラップトップで処理する場合、毎秒 15 フレームの速度で処理を行うことが可能であった。これは、十分にリアルタイムでの処理に耐えうると考えられる。なお、処理の実装は C++ 言語を用いて行い、特徴点の追跡に関しては OpenCV のライブラリを使用した。

表 3.2 実験で取得した画像を処理するのに用いたラップトップの性能

CPU	Intel Core i5-2540M
クロック周波数	2.6GHz
メモリ サイズ	4GB

## 第 4 章

# 実用に関する考察

第 3 章で検証したとおり，提案手法は画面内での移動体の動きが大きい状況において，多数の移動物体を平均 65% 程度の精度で検出することができた．また，プラットフォームが移動をしているような状況においても精度を劣化させず，移動物体の検出を行うことができた．さらに，その処理時間は毎秒 15 フレーム程度であり，十分にリアルタイム・アプリケーションへの適用が期待できる．本章では，この結果に基づいて提案手法の利用可能性と，その実用に向けての課題を議論する．

### 4.1 提案手法の利用可能性

提案手法に関して，現段階で実用化が可能であるアプリケーションとしては，移動物体の流動のリアルタイムでの可視化が挙げられる．PTAM によって出力された飛行型プラットフォームの相対位置と姿勢から，特徴点追跡によって得られた画面内での移動物体の移動量を地面を模した 2 次元平面に投影することで，移動物体の地面上での動きをリアルタイムに可視化することが可能である．このような流動の可視化は，イベント会場における群衆の誘導に役立てることができる．前述したように，UAV をプラットフォームとして扱えば，移動をしながら撮影を行うことによって会場全体の状況を把握したり，その結果から混雑している箇所を集中的に観測したりすることも可能である．また，バルーンをプラットフォームとして用いる場合，このようなイベント会場の全体の状況把握への利用の他に，交通量の多い交差点等，ある一定範囲内における車両の流動を把握することができ，交通状況の調査の自動化に役立てることも可能であろう．さらに，提案手法がリアルタイムでの処理が可能であることを考慮すると，UAV のようなロボットが（特に室内を）移動する際の移動物体の回避や，SLAM で構築する地図の高精度化など，ロボット・ナビゲーションへの応用が期待できる．提案手法はカメラ以外のセンサーを必要としないため，特に飛行ロボットなどの機体重量に関する制限が大きいロボットのナビゲーションに関して，有用であると考えられる．

本論文で提案した手法では，移動物体から抽出された特徴点を識別するまでにとどまったが，個々の特徴点の集合を 1 つの移動物体にクラスタリングすることができれば，スポーツのブロードキャスティングへの応用が期待できる．UAV を用いたスポーツのブロードキャスティングを行う動きは既にある [45][46] が，提案手法ではプラットフォームを移動させながら移動物体の認識を行うことが可能である．したがって，提案手法を用いてプレイヤーやボールを移動物体として認識できれば，それを UAV の制御フィードバックに入力することで，ボールやそれを運ぶプレイヤーを追いつづけることができる．このような UAV の制御により，決定的なシーンにより臨場感を持たせて放送することが可能となろう．

最後に，発展としては，バルーンと UAV を組み合わせることで，それぞれの短所を補い合った移動物

体のモニタリング・システムを構築することが可能である。例えば、バルーンによって得られた広範囲の画像を用いて、移動物体の全体の動向を把握し、より移動物体が集中している箇所を UAV によって詳細にモニタリングするといったものである。さらに、2 機以上のバルーンを用いた立体視（ステレオ視）によって 3 次元空間の地図を作成し、その地図を UAV の飛行制御に役立てることで、障害物の確実な回避などを行わせることも可能である。このようなシステムは、災害時に地形崩壊後の状況を把握と同時に、動くことができる生存者の発見に役立てられる。

## 4.2 実用に向けての課題

実用に向けての課題としては、まず、移動物体の検出精度の向上が挙げられる。特に、プラットフォームと検出対象の移動物体が空間内を平行に動いているとき、エピポーラ拘束の限界から検出精度が低下しているが、先に述べたような流動の可視化やスポーツのブロードキャスティングでは、移動物体がどんな方向に動いている場合でもそれを検出できることが重要である。このような状況下での検出精度を向上するためには、プラットフォームを様々な方向に動かすことや、3つのカメラ視点に現れるとトライ・フォーカル・テンサー（Tri-focal Tensor）[47] [48] をエピポーラ幾何に代わって導入することなどが解決策として考えられる。

また、検出された点を1つの移動体にクラスタリングし、クラスター化された移動体をカルマンフィルタなどによって複数フレーム間にわたって追跡を行うことができれば、先に述べたスポーツのブロードキャスティングをはじめとする、多くのアプリケーションがひらける。また、複数フレーム間での追跡（複数フレーム間での一貫性の確認）を行うことによって、1フレームでは検出できなかった移動物体や、逆に誤検出を減らすことができ、信頼性を向上することも可能となる。

最後に、これは手法の内容とは直接関係しないが、歩行者や自動車などの上空を UAV が飛行することに関する規制も、実用化に関する大きな課題である。ドローン（UAV）の規制に関して改訂された航空法によると、人または家屋の密集している地域の上空での UAV の飛行は制限されており、国土交通省の許可を得ない限りはそれらの地域の上空で UAV を飛行させることはできない [49]。現在、日本は UAV に関する法整備を進めている段階であると考えられる。今後、UAV 自体の信頼性の向上や、UAV の飛行ライセンスなどの法整備により、UAV をこうした場所で安全に飛行させる仕組みが構築されることを望む。

## 第 5 章

# 結論

本論文では，飛行型プラットフォームから撮影された画像に対して，多視点画像に関する幾何学的な拘束条件を利用した移動体検出手法を提案した．多視点の画像を利用する提案手法はプラットフォームを移動させながら移動物体の検出を行うことが可能であり，空間内での動きの自由度が大きい UAV やバルーンの特性をいかした飛行型プラットフォームと非常に相性の良い手法である．

また，小型のクアッドローター型 UAV を実環境で飛行させ，提案手法の精度評価実験を行った．複数の歩行者が存在する実環境下での実験では，画面内に歩行者が特に多く存在する状況下において，65% の精度で歩行者を識別することができ，提案手法の有用性が示された．

提案手法は毎秒 15 フレームの速度で処理が可能であり，移動物体の流動のリアルタイムでの可視化や，飛行ロボットの移動物体回避のためのナビゲーション，さらには，より臨場感を持たせたスポーツのブロードキャスティングなど，多岐にわたる応用が期待できる．

課題としては，異なる拘束条件の導入による検出率の更なる向上や，検出された特徴点のクラスタリング，複数フレーム間にわたる移動体のトラッキング手法の開発が挙げられる．

## 参考文献

- [1] 山崎公俊. カメラを搭載した移動ロボットによる画像列を用いた物体の3次元形状モデリング. p. 15. 筑波大学, 2004.
- [2] コーナー検出 (eigenvalue, harris, fast) . [http://opencv.jp/opencv2-x-samples/corner\\_detection](http://opencv.jp/opencv2-x-samples/corner_detection) 2015 年 12 月 19 日に閲覧.
- [3] Stuart M Adams and Carol J Friedland. A survey of unmanned aerial vehicle (uav) usage for imagery collection in disaster research and management. In *9th International Workshop on Remote Sensing for Disaster Response*, 2011.
- [4] Dieter Hausamann, Werner Zirrig, and Gunter Schreier. Monitoring of gas transmission pipelines—a customer driven civil uav application. In *ODAS Conference*, 2003.
- [5] J Everaerts, et al. The use of unmanned aerial vehicles (uavs) for remote sensing and mapping. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. 37, pp. 1187–1192, 2008.
- [6] Julia Boike and Kenji Yoshikawa. Mapping of periglacial geomorphology using kite/balloon aerial photography. *Permafrost and periglacial processes*, Vol. 14, No. 1, pp. 81–85, 2003.
- [7] Michiru Miyamoto, Kunihiro Yoshino, Toshihide Nagano, Tomoyasu Ishida, and Yohei Sato. Use of balloon aerial photography for classification of kushiro wetland vegetation, northeastern japan. *Wetlands*, Vol. 24, No. 3, pp. 701–710, 2004.
- [8] Ahmed Elgammal, David Harwood, and Larry Davis. Non-parametric model for background subtraction. *Computer Vision—ECCV 2000*, pp. 751–767. Springer, 2000.
- [9] Rita Cucchiara, Costantino Grana, Massimo Piccardi, and Andrea Prati. Detecting moving objects, ghosts, and shadows in video streams. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, Vol. 25, No. 10, pp. 1337–1342, 2003.
- [10] Boyoon Jung and Gaurav S Sukhatme. Detecting moving objects using a single camera on a mobile robot in an outdoor environment. In *International Conference on Intelligent Autonomous Systems*, pp. 980–987, 2004.
- [11] Gonzalo R Rodríguez-Canosa, Stephen Thomas, Jaime del Cerro, Antonio Barrientos, and Bruce MacDonald. A real-time method to detect and track moving objects (datmo) from unmanned aerial vehicles (uavs) using a single camera. *Remote Sensing*, Vol. 4, No. 4, pp. 1090–1111, 2012.
- [12] Georg Klein and David Murray. Parallel tracking and mapping for small ar workspaces. In *Mixed and Augmented Reality, 2007. ISMAR 2007. 6th IEEE and ACM International Symposium on*, pp. 225–234. IEEE, 2007.
- [13] Mennatullah Siam and Mohamed ElHelw. Robust autonomous visual detection and tracking

- of moving targets in uav imagery. In *Signal Processing (ICSP), 2012 IEEE 11th International Conference on*, Vol. 2, pp. 1060–1066. IEEE, 2012.
- [14] Beril Sirmacek and Peter Reinartz. Kalman filter based feature analysis for tracking people from airborne images. In *ISPRS Workshop High-Resolution Earth Imaging for Geospatial Information, Hannover, Germany*, 2011.
- [15] Beril Sirmacek and Peter Reinartz. Feature analysis for detecting people from remotely sensed images. *Journal of Applied Remote Sensing*, Vol. 7, No. 1, pp. 073594–073594, 2013.
- [16] Edward Rosten and Tom Drummond. Fusing points and lines for high performance tracking. In *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, Vol. 2, pp. 1508–1515. IEEE, 2005.
- [17] Edward Rosten and Tom Drummond. Machine learning for high-speed corner detection. In *Computer Vision–ECCV 2006*, pp. 430–443. Springer, 2006.
- [18] I Pratikakis, M Spagnuolo, T Theoharis, and R Veltkamp. A robust 3d interest points detector based on harris operator. In *Eurographics Workshop on 3D Object Retrieval*, Vol. 1. Citeseer, 2010.
- [19] Huiyu Zhou, Yuan Yuan, and Chunmei Shi. Object tracking using sift features and mean shift. *Computer vision and image understanding*, Vol. 113, No. 3, pp. 345–352, 2009.
- [20] Carlo Tomasi and Takeo Kanade. *Detection and tracking of point features*. School of Computer Science, Carnegie Mellon Univ. Pittsburgh, 1991.
- [21] Jean-Yves Bouguet. Pyramidal implementation of the affine lucas kanade feature tracker description of the algorithm. *Intel Corporation*, Vol. 5, pp. 1–10, 2001.
- [22] Zhengyou Zhang. Flexible camera calibration by viewing a plane from unknown orientations. In *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, Vol. 1, pp. 666–673. IEEE, 1999.
- [23] Zhengyou Zhang. A flexible new technique for camera calibration. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, Vol. 22, No. 11, pp. 1330–1334, 2000.
- [24] David M Cole and Paul M Newman. Using laser range data for 3d slam in outdoor environments. In *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on*, pp. 1556–1563. IEEE, 2006.
- [25] Andrew J Davison, Ian D Reid, Nicholas D Molton, and Olivier Stasse. Monoslam: Real-time single camera slam. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, Vol. 29, No. 6, pp. 1052–1067, 2007.
- [26] Nikolas Engelhard, Felix Endres, Jürgen Hess, Jürgen Sturm, and Wolfram Burgard. Real-time 3d visual slam with a hand-held rgb-d camera. In *Proc. of the RGB-D Workshop on 3D Perception in Robotics at the European Robotics Forum, Vasteras, Sweden*, Vol. 180, 2011.
- [27] Paul Newman, David Cole, and Kin Ho. Outdoor slam using visual appearance and laser ranging. In *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on*, pp. 1180–1187. IEEE, 2006.
- [28] Kai M Wurm, Cyrill Stachniss, and Giorgio Grisetti. Bridging the gap between feature-and grid-based slam. *Robotics and Autonomous Systems*, Vol. 58, No. 2, pp. 140–148, 2010.
- [29] Hans P Moravec and Alberto Elfes. High resolution maps from wide angle sonar. In *Robotics*

- and Automation. Proceedings. 1985 IEEE International Conference on*, Vol. 2, pp. 116–121. IEEE, 1985.
- [30] John J Leonard and Hugh F Durrant-Whyte. Mobile robot localization by tracking geometric beacons. *Robotics and Automation, IEEE Transactions on*, Vol. 7, No. 3, pp. 376–382, 1991.
- [31] Sebastian Thrun, Wolfram Burgard, and Dieter Fox. *Probabilistic robotics*. MIT press, 2005.
- [32] Michael Montemerlo, Sebastian Thrun, Daphne Koller, Ben Wegbreit, et al. Fastslam: A factored solution to the simultaneous localization and mapping problem. In *AAAI/IAAI*, pp. 593–598, 2002.
- [33] Michael Montemerlo and Sebastian Thrun. Fastslam 2.0. *FastSLAM: A Scalable Method for the Simultaneous Localization and Mapping Problem in Robotics*, pp. 63–90, 2007.
- [34] Stephan Weiss, Davide Scaramuzza, and Roland Siegwart. Monocular-slam-based navigation for autonomous micro helicopters in gps-denied environments. *Journal of Field Robotics*, Vol. 28, No. 6, pp. 854–874, 2011.
- [35] Stephan Weiss, Markus W Achtelik, Simon Lynen, Margarita Chli, and Roland Siegwart. Real-time onboard visual-inertial state estimation and self-calibration of mavs in unknown environments. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pp. 957–964. IEEE, 2012.
- [36] Jakob Engel, Jürgen Sturm, and Daniel Cremers. Camera-based navigation of a low-cost quadcopter. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, pp. 2815–2821. IEEE, 2012.
- [37] Hauke Strasdat, JMM Montiel, and Andrew J Davison. Real-time monocular slam: Why filter? In *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pp. 2657–2664. IEEE, 2010.
- [38] Jack CK Chou. Quaternion kinematic and dynamic differential equations. *Robotics and Automation, IEEE Transactions on*, Vol. 8, No. 1, pp. 53–64, 1992.
- [39] James Diebel. Representing attitude: Euler angles, unit quaternions, and rotation vectors. *Matrix*, Vol. 58, pp. 15–16, 2006.
- [40] Abhijit Kundu, K Madhava Krishna, and Jayanthi Sivaswamy. Moving object detection by multi-view geometric techniques from a single camera mounted robot. In *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*, pp. 4306–4312. IEEE, 2009.
- [41] Anubhav Agarwal, CV Jawahar, and PJ Narayanan. A survey of planar homography estimation techniques. *Centre for Visual Information Technology, Tech. Rep. IIIT/TR/2005/12*, 2005.
- [42] Rahul Sukthankar, Robert G Stockton, and Matthew D Mullin. Smarter presentations: Exploiting homography in camera-projector systems. In *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, Vol. 1, pp. 247–253. IEEE, 2001.
- [43] Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, Vol. 24, No. 6, pp. 381–395, 1981.
- [44] Karan Gupta and Anjali V Kulkarni. Implementation of an automated single camera object tracking system using frame differencing and dynamic template matching. In *Advances in*

- Computer and Information Sciences and Engineering*, pp. 245–250. Springer, 2008.
- [45] Inkyu Sa and Ho Seok Ahn. Visual 3d model-based tracking toward autonomous live sports broadcasting using a vtol unmanned aerial vehicle in gps-impaired environments. *International Journal of Computer Applications*, Vol. 122, No. 7, 2015.
- [46] Use of drones in sports broadcasts. <http://www.stats.com/blog/2015/12/16/use-of-drones-in-sports-broadcasts/> 2016 年 1 月 12 日に閲覧.
- [47] Richard I Hartley. Lines and points in three views and the trifocal tensor. *International Journal of Computer Vision*, Vol. 22, No. 2, pp. 125–140, 1997.
- [48] Anders Heyden. A common framework for multiple view tensors. *Computer Vision—ECCV’98*, pp. 3–19. Springer, 1998.
- [49] ドローン・小型無人機への規制に関して閣議決定された航空法改正の内容. <https://drone.beinto.xyz/rule/revise-aviation-law-for-drone/> 2016 年 1 月 10 日に閲覧.



# 謝辞

本修士論文は、筆者が東京大学大学院 新領域創成科学研究科 社会文化環境学専攻修士課程在学中に柴崎・関本研究室において行った研究をまとめたものです。本研究を進めるにあたり、指導をしていただいた柴崎亮介先生に深謝致します。柴崎先生からは、研究の内容に関するものだけではなく、自身のバックグラウンドを生かした研究テーマの選定から始まり、その分野を専門とする方々への紹介等、実験の方法やその手続きなど、研究を進めるにあたって、広範にわたるご指導をいただきました。また、G-SPASEをはじめとするいくつかの研究会や活動に参加するきっかけとご支援をいただきました。そうした活動を通して、自身の研究を多くの視点から捉え直すことができ、研究をはじめとする大学院での生活をより豊かなものにすることが出来ました。研究生時代を含めて3年間にわたってご指導をいただけたことに、深く感謝しております。

お忙しい中、本論文をご精読いただきました瀬崎薫教授に深謝致します。また、副指導をしていただいた早川裕弐准教授には、UAVの今後の利用可能性をはじめ、深く考察するためのアドバイスを数多くいただきました。ここに感謝申し上げます。

実験に多大なるご協力をいただきました長井正彦先生とその研究室の皆様、アジア工科大学の学生の皆様に感謝致します。3週間にわたるバンコクでの実験は、研究の内容を充実させるだけではなく、アジア各国出身の同年代の学生と議論をしたり、他のグループの研究活動に参加したりと、自身の見聞を大きく広める貴重な機会となりました。

研究室秘書の秋枝さん、本間さん、川端さんには、実験に必要な機材の購入や、学会への参加手続き、その他の各種手続きに関しまして、様々なご支援をいただきました。心から感謝申し上げます。

最後になりますが、ゼミを通じて活発な議論にお付き合い頂いた柴崎・関本研究室の皆様には感謝します。