Visual SLAM

斎藤英雄（慶應義塾大学）
hs@keio.jp

1. Visual SLAM（Simultaneous Localization and Mapping）とは

SLAM（自己位置推定と環境地図の同時生成手法）は1980年代後半から，ロボット分野で重要な技術として位置付けられている．この技術は，超音波など様々なセンサ情報を用いて，ロボットの自己位置と環境地図を生成する手法である．特に，カメラなどのビジョンセンサを利用したSLAMを「Visual SLAM」と呼ぶ．

2. Visual SLAMの基本原理

図1に示すように，移動するカメラで撮影された画像列（画像1，画像2，画像3，…）では，カメラの移動に伴い三次元位置 $\boldsymbol{P}_i$ にある点が異なる二次元位置 $\boldsymbol{p}_i^j$ に撮影される．これらの二次元位置 $\boldsymbol{p}_i^j$ は，撮影された点の三次元位置 $\boldsymbol{P}_i$ と，各画像を撮影したカメラの三次元位置 $\boldsymbol{t}^j$・姿勢 $\boldsymbol{R}^j$（三次元ポーズ）に依存する．単一点の二次元位置 $\boldsymbol{p}_i^j$ からこれらを全て推定することは不可能だが，多数の三次元点を複数の位置姿勢から撮影した画像で検出し，画像間でマッチングすることにより，移動中のカメラの三次元ポーズと全ての点の三次元マップを推定することができることが，多視点幾何に関する研究により明らかになっている[(1)]．これがVisual SLAMの基本原理である．

3. Visual SLAMのアルゴリズム

3.1 基本要素

Visual SLAMは，以下のような基本要素により実現される．

（1） 特徴点マッチングによる特徴点の追跡

図1に示すように，カメラの移動に伴って撮影される各画像 j で検出した特徴点をマッチングし，空間の同一の三次元位置 $\boldsymbol{P}_i$ に対する特徴点の二次元位置 $\boldsymbol{p}_i^j$ として複数の画像にわたって追跡する．このプロセスには以下の主要な2ステップが必要である．

特徴点の検出と特徴記述：「特徴点」とは，異なる画像間で同一の点であるかどうかを判断するのに十分な特徴を持つ点であり，これを検出し，その特徴をベクトル量として記述する．

特徴点のマッチング：上記の特徴点検出と特徴記述に基づき，異なる画像間での特徴点のマッチングが行われる．

この分野の研究は，SIFT（Scale-Invariant Feature Transform）[(2)]の登場をきっかけに盛んになり，それ以降，多様な手法が提案されている．

（2） 三次元マップの初期化

Visual SLAMの開始時に最初に取得された画像1と，カメラ移動後に得られる次の画像2の間の二次元特徴点ペア群（図1では $[\boldsymbol{p}_1^1, \boldsymbol{p}_1^2]$，$[\boldsymbol{p}_2^1, \boldsymbol{p}_2^2]$，$[\boldsymbol{p}_3^1, \boldsymbol{p}_3^2]$，$[\boldsymbol{p}_4^1, \boldsymbol{p}_4^2]$，$[\boldsymbol{p}_5^1, \boldsymbol{p}_5^2]$）を用いて，画像1を撮影したカメラの位置姿勢を基準とした三次元座標系において，画像2を撮影したカメラの三次元ポーズ $(\boldsymbol{R}^2, \boldsymbol{t}^2)$ を推定する．この推定を基に，各二次元特徴点ペアから三角測量計算を行い，それぞれの特徴点ペアが示す三次元位置 $(\boldsymbol{P}_1, \boldsymbol{P}_2, \boldsymbol{P}_3, \boldsymbol{P}_4, \boldsymbol{P}_5)$ を推定することで，三次元マップの初期化を行う．

（3） 特徴点追跡による三次元ポーズ推定と三次元マップの更新

三次元マップ初期化後，カメラが移動して得られる新しい画像（画像3）と，三次元マップ生成時の画像（画像2）の間で特徴点を追跡する．図1では，$[\boldsymbol{p}_2^2, \boldsymbol{p}_2^3]$，$[\boldsymbol{p}_4^2, \boldsymbol{p}_4^3]$，$[\boldsymbol{p}_5^2, \boldsymbol{p}_5^3]$が追跡されている．この追跡により，三次元マップの各点と画像3に撮影された二次元点との「3D-2Dマッチング点群」（図1では，$[\boldsymbol{P}_2, \boldsymbol{p}_2^3]$，$[\boldsymbol{P}_4, \boldsymbol{p}_4^3]$，$[\boldsymbol{P}_5, \boldsymbol{p}_5^3]$）が得られ，これを基にして画像3を撮影したカメラの三次元ポーズ $(\boldsymbol{R}^3, \boldsymbol{t}^3)$ を推定する．この推定は，PnP（Perspective-n-Point）問題[(1)]として知られ，多くの手法が提案されている．

画像3のカメラ三次元ポーズが得られると，画像1から3にわたる画像群の中で，以前の三次元マップに含まれなかった新しい特徴点についてもマッチングが行われる（図1では，$[\boldsymbol{p}_6^2, \boldsymbol{p}_6^3]$，$[\boldsymbol{p}_7^2, \boldsymbol{p}_7^3]$）．これらの特徴点の三次元位置 $\boldsymbol{P}_6, \boldsymbol{P}_7$ は三角測量によって推定され，これにより三次元マップが更新される．

このプロセスは，引き続き新しい画像（画像4，画像5など）が得られるたびに繰り返される．

3.2 性能向上のための構成要素

以上をまとめると，Visual SLAMは，特徴点マッチングを画像列間で繰り返しながら追跡した特徴点群の二次元位置

本会ハンドブック「知識の森」
https://www.ieice-hbkb.org/portal/doc_index.html

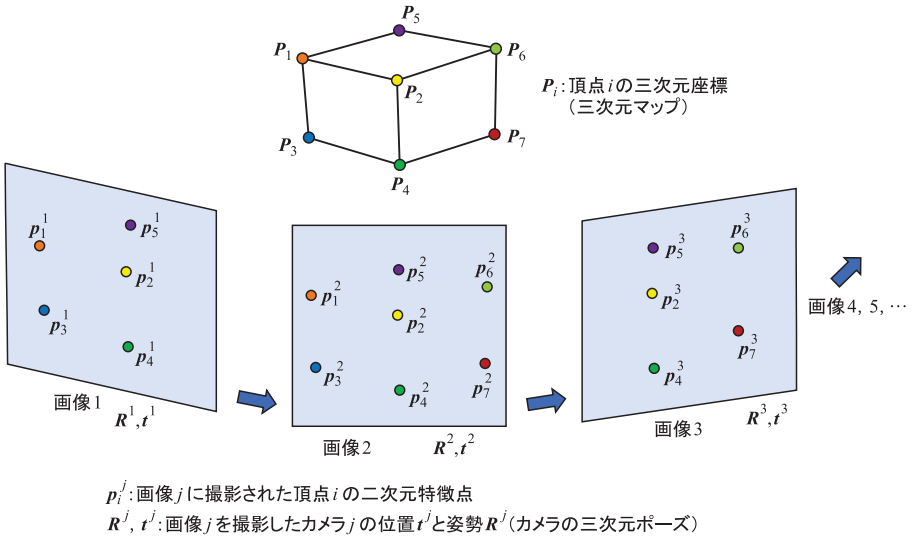

図1　Visual SLAM の基本原理　Visual SLAM は，特徴点マッチングを画像列間で繰り返しながら追跡した特徴点群の二次元位置 $\boldsymbol{p}_i^j$ から，最初に三次元マップの初期化を行った後に，カメラの三次元ポーズ推定と三次元マップ更新を繰り返し行う形で進行する．画像1のカメラが世界座標系の基準となるため，$\boldsymbol{R}^1$ は単位行列で，$\boldsymbol{t}^1$ はゼロベクトルである．

$\boldsymbol{p}_i^j$ から，最初に三次元マップの初期化を行った後に，カメラの三次元ポーズ推定と三次元マップ更新を繰り返し行う形で進行する．

ただし，このプロセスを繰り返すたびに発生する誤差がカメラの移動とともに蓄積してしまう．この誤差蓄積を防ぐために，三次元マップ中の各点を各画像撮影時のカメラの三次元ポーズで再投影した二次元位置と，特徴点として検出された二次元位置の差異（再投影誤差）を最小化し，三次元マップと三次元ポーズの精度を向上させる．この処理を「Bundle Adjustment」[1] と呼ぶ．

更に，カメラの移動に伴い，三次元マップ内の点がカメラの視野から外れ，結果としてカメラの三次元ポーズを見失うことがある．このような場合には，カメラの三次元ポーズを見失った画像について，以前に撮影した全ての画像から同一の二次元点を検出し，三次元マップと対応する点を再検出して，その位置と姿勢を PnP 解法により推定する．この処理は「Relocalization」と呼ばれる．同様に，カメラ移動開始時の画像とマッチングすることで，蓄積されたカメラ移動の誤差を最小化し，これまでに得られた三次元マップと三次元ポーズ全体を更新する「Loop Closure」処理も重要である．

また，実際の処理では，撮影される全ての画像を利用すると計算量が膨大になり，新たな画像に対する処理が間に合わなくなる可能性がある．このため，推定精度や処理速度を考慮しながら，上記の処理に利用する画像を適切に選択する「キーフレーム選択処理」が必要となる．

4. 様々な VSLAM アルゴリズムの例

上記の基本アルゴリズムに対して，各処理で様々な工夫がなされ，これまでに大量の VSLAM アルゴリズム[3] が提案されてきた．下記がその代表的なものである．

- MonoSLAM：空間を動き回る1台のカメラ画像から実時間で Visual SLAM が可能になることを初めて示した．
- PTAM：特徴点追跡スレッドとマップ生成・更新スレッドを分離して処理の高速化を実現した．
- DTAM：特徴点だけではなく，画素全てについて三次元位置を推定することにより，密な三次元マップ生成を実現した．
- LSD-SLAM：特徴点追跡の代わりに，画像間の画素値が一致するようなカメラの三次元ポーズ推定を行うことにより，密な三次元マップ生成を実現した．
- ORB-SLAM：特徴点マッチングに ORB 特徴記述法を利用し，公開ソースコードが使いやすいため広く活用されている．

更に，下記のような深層学習を活用した Visual SLAM が次々と生み出されている．

- CNN-SLAM[4]：CNN で推定したデプス画像を利用することにより，密な三次元マップ生成を実現した．
- DeepSLAM[5]：2台のステレオカメラから深層学習により得られる深度画像を利用して Visual SLAM の性能を改善した．
- NeRF-SLAM[6]：視線方向に応じて空間の各点がどのように見えるかを学習により獲得する NeRF を活用して密な三次元マップを得ることができる．

5. 今後の展開

現在も Visual SLAM の研究は盛んに行われており，性能の改善が進んでいる．それに伴って，スマートフォン上で動作し，AR アプリケーション開発用のライブラリ（Google の ARCore，Apple の ARKit）や，Meta の MR/VR 向け HMD の Quest 等で実用に供されるようにもってきた．特にスマートフォンや HMD にはカメラだけではなく多くのセンサが内蔵されており，その中の加速度センサも活用した Visual Inertial SLAM[7] も大きく進歩している．

文　　献

（1） R. Hartley, and A. Zisserman, Multiple view geometry in computer vision, Cambridge University Press, Cambridge, UK, 2nd edition, 2004.

（2） D. Lowe, “Object recognition from local scale-invariant features,” Proc. Int. Conf. Computer Vision. vol. 2, pp. 1150-1157, 1999.

（3） T. Taketomi, H. Uchiyama, and S. Ikeda, “Visual SLAM algorithms : a survey from 2010 to 2016,” IPSJ Trans. Computer Vision Applications, vol. 9, article no. 16, 2017.

（4） K. Tateno, F. Tombari, I. Laina and N. Navab, “CNN-SLAM : Real-time dense monocular SLAM with learned depth prediction,” 2017 IEEE Conference on Computer Vision and Pattern Recognition（CVPR）, pp. 6565-6574, 2017.

（5） R. Li, S. Wang and D. Gu, “DeepSLAM : A robust monocular SLAM system with unsupervised deep learning,” IEEE Trans. Industrial Electronics, vol. 68, no. 4, pp. 3577-3587, 2021.

（6） A. Rosinol, J.J. Leonard, and L. Carlone, “NeRF-SLAM : Real-time dense monocular SLAM with neural radiance fields,” 2023 IEEE/RSJ Int. Conf. Intelligent Robots and Systems（IROS）, pp. 3437-3444, 2023.

（7） Myriam Servières, Valérie Renaudin, Alexis Dupuis, and Nicolas Antigny, “Visual and visual-inertial SLAM : State of the art, classification, and experimental benchmarking,” Journal of Sensors, vol. 2021, Article ID 2054828, p. 26, 2021.

（2023 年 12 月 16 日受付）