Dimensionality reduction in visual-inertial SLAM

Laurent Kneip
http://mpl.sist.shanghaitech.edu.cn

Macao, November 8th 2019
Who are we?
Research@MPL: Visual SLAM

• An enabler of new technologies
  • Factory automation
  • Service robotics
  • Augmented reality
  • Intelligent transportation

Research@MPL: Surround-view camera systems

• Origins: V-CHARGE
  • EU FP7 project
    • ETH Zurich
    • Volkswagen
  • AVP with vision only
    • Close-to-market sensors
Research@MPL: Surround-view camera systems

• Early research (2012):
  • Non-overlapping stereo

• Inspired by nature
  • Field of view of humans
  • Field of view of pigeons

[Kazik, Kneip, Nikolic, Pollefeys, Siegwart, Real-Time 6D Stereo Visual Odometry with Non-Overlapping Fields of View, CVPR’12]

Dataset 1: Circular Motion
Research@MPL: Surround-view camera systems

- Now: Joint work with Motovis Intelligent Technologies, Co Ltd. (Shanghai)
Research@MPL: OpenGV

- Open-source, hosted on github
- Widely used in both academia and industry

[GitHub: https://github.com/laurentkneip/opengv]
Is visual SLAM a solved problem?

• Traditional SLAM is solved as a graph optimization problem using sparse feature correspondences
  • 2D-2D correspondences for bootstrapping
  • 2D-3D correspondences for tracking
  • Entire SfM view-graph for mapping
Is visual SLAM a solved problem?

• Traditional SLAM is solved as a graph optimization problem using sparse feature correspondences

• Issues:
  • Feature-poor scenarios
  • Bad feature distributions
  • Blur/high disparity
  • Meaningless maps
  • Poor long-term stability
Is visual SLAM a solved problem?

• Traditional SLAM is solved as a graph optimization problem using sparse feature correspondences

• Issues for geometric relative pose computation (used in bootstrapping):
  • Planar/degenerate point distributions
  • Pure rotation, rotation/translation ambiguity
Inertial-assisted visual odometry

- **Idea:** Use relative rotation priors from IMU
  - Short-term integration of gyroscopic signals for full 3D orientation change
  - Integration typically accomplished inside IMUs
    - Orientation drifts only slowly
    - Short-term relative rotations recoverable from consecutive IMU measurements

- **Assumption**
  - Extrinsic transformation parameters are known

- **Delta-rotation prior given by:**
  \[
  R_{c'}^c = R_c^i U_c^T U_{c'} R_c^i
  \]
Inertial-assisted visual odometry

• Relative pose: Computation of translation using 2-point algorithm
  • Epipolar plane normals: \( n_i = f_i \times R_{c'} f_i' \)
  • Translation direction: \( d_{c'}^c = n_1 \times n_2 \)
  • Translation vector: \( t_{c'}^c = +/ - \frac{d_{c'}^c}{\|d_{c'}^c\|} \)

• Absolute pose:
  • Becomes a 1 ½ point algorithm!

[L Kneip, M Chli, and R Siegwart. Robust real-time visual odometry with a single camera and an IMU. In Proceedings of the British Machine Vision Conference (BMVC), 2011]
Inertial assisted visual odometry

• Results on MAV dataset
  • Challenging motion (human pilot, high velocity, full 3D)
  • Challenging structure (moving, planar degeneracy, specularity)
  • FOV 100, 2.8 GHz machine, ~80 Hz

Dimensionality reduction in SLAM

Representing *motion* using

- higher-order
- low-dimensional
- robust
- implicitly smooth

prior models
Vehicle motion is non-holonomic

• Exploitation of the Ackermann-steering model
  • Rotation and translation of a vehicle are coupled
  • Leads to local parametrization of motion by arc of circle
  • Solution in the space of rotations only
Vehicle motion is non-holonomic

• An n-view 1-point algorithm
  • [K Huang, Y Wang, and L Kneip. Motion estimation of non-holonomic ground vehicles from a single feature correspondence measured over n views. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, USA, June 2019]
Vehicle motion is non-holonomic

• Cameras distributed over tractor-trailer system
  • Again, a 1-point algorithm!

Dimensionality reduction in SLAM

Representing **structure** using

- higher-order
- low-dimensional
- robust
- implicitly smooth

prior models
What do we mean by higher-order priors?

Classical SLAM formulation:
Measurements = f(T_j, p_i) + noise
Residuals = Measurements - f(T_j, p_i)

• Classical formulation ignores higher level information
• Dense representation = “many pts constrained by local smoothness”
What do we mean by higher-order priors?

• Detect object(s) in image
• Segment/annotate in 3D
• Use semantic knowledge to reconstruct object (i.e. use shape models)
What do we mean by higher-order priors?

- Find the nearest shape on “a shape manifold” that agrees with the measurements!

Modern SLAM formulation:

Measurements = \( f(T_j, \lambda) + \text{noise} \)  \( \text{dim}(\lambda) = d \ll n \)

Residuals = Measurements - \( f(T_j, \lambda) \)
Spatial AI  (Term “coined” by Andrew Davison)

• An evolution of SLAM
  • Joint geometric-semantic scene understanding at the level of objects
    • What objects? Where? What is their shape?
      “old” style map
      = a ‘primitive’ point cloud, mesh etc.
      “new” style map
      = HD map

lkneip@shanghaitech.edu.cn http://mpl.sist.shanghaitech.edu.cn
A hierarchy of scene element models

Points

Lines, Planes

Shape primitives

Full object shapes

Dimensionality

Implicit smoothness

Semantic meaning

Ability to handle partial measurements

Ability to handle noise

Ability to handle outliers
Deep-SLAM++

• Hybrid Graph
  • Includes both low and high-level (object-level) features
  • Has pose and shape parameters for high level landmarks

\[ E = \sum_{i=1}^{n} \sum_{j=1}^{m} A_{ij} (T_{Wi}, T_{Wj}, r_{ij}, r_{i}, h_{j}) + \sum_{i=1}^{n} B_{i} (T_{Wi}, n) + \sum_{j=1}^{m} \sum_{i=1}^{n} C_{ij} (T_{Wj}, p_{j}) \]

High-level features (objects with pose and shape)  
Mid-level features (e.g. planes, lines)  
Low-level features (e.g. points)

\[ \theta_1 = \{T_{WO1}, r_{11}, r_{12}, h_1 \} \]

\[ \theta_2 \]

\[ \{p_1, \ldots, p_m \} \]

Deep-SLAM++

• Low-dimensional, differentiable complex shape representations?
  • Dimensionality reduction/shape manifold learning with auto-encoders

Deep-SLAM++

- Results on an indoor scenario
  - Chairs and tables are generated by a neural network
  - Reasonable geometries are obtained by differentiating latent variables w.r.t the measurements

Deep-SLAM++

- Underground parking lot mapping for AVP
  - Only surround-view fish-eye images; **AI embedded into FPGA**
  - Full optimization over pose and higher level shape parameters (lanes, parking lots, ...)

lkneip@shanghaitech.edu.cn  
http://mpl.sist.shanghaitech.edu.cn
Deep-SLAM++

- Towards AVP
  - Online localization and autonomous driving based on high-level feature map
Thank you!