Visual-Inertial Localization and Mapping for Robot Navigation

Dr. Guillermo Gallego

Robotics & Perception Group
University of Zurich
Mocular Vision-inertial Navigation of Mobile Robots

https://www.youtube.com/watch?v=vHpw8zc7-JQ

Keyframe-based Visual Odometry

PTAM (Parallel Tracking & Mapping) [Klein, ISMAR’08]
Feature-based methods

1. Extract & match features (+RANSAC)
2. Minimize Reprojection error minimization

\[ T_{k,k-1} = \arg \min_T \sum_i \| u'_i - \pi(p_i) \|_\Sigma^2 \]

Direct methods

1. Minimize photometric error

\[ T_{k,k-1} = \arg \min_T \sum_i \| I_k(u'_i) - I_{k-1}(u_i) \|_\sigma^2 \]

where \( u'_i = \pi(T \cdot (\pi^{-1}(u_i) \cdot d)) \)

[Jin,Favaro,Soatto’03] [Silveira, Malis, Rives, TRO’08], [Newcombe et al., ICCV ‘11], [Engel et al., ECCV’14], [Forster et al., ICRA’14]
Feature-based methods

1. Extract & match features (+RANSAC)
2. Minimize Reprojection error minimization

\[ T_{k,k-1} = \arg \min_{T} \sum_{i} ||u'_i - \pi(p_i)||_{\Sigma}^2 \]

Direct methods

1. Minimize photometric error

\[ T_{k,k-1} = \arg \min_{T} \sum_{i} ||I_k(u'_i) - I_{k-1}(u_i)||_{\sigma}^2 \]

where \[ u'_i = \pi \left( T \cdot (\pi^{-1}(u_i) \cdot d) \right) \]

[\text{Jin,Favaro,Soatto’03} \text{ [Silveira, Malis, Rives, TRO’08]}, \text{ [Newcomer [Engel et al., ECCV’14], [Forster et al., ICRA’14]}

\checkmark \text{ Large frame-to-frame motions}
\times \text{ Slow due to costly feature extraction and matching}
\times \text{ Matching Outliers (RANSAC)}

\checkmark \text{ All information in the image can be exploited (precision, robustness)}
\checkmark \text{ Increasing camera frame-rate reduces computational cost per frame}
\times \text{ Limited frame-to-frame motion}
Feature-based methods

1. Extract & match features (+RANSAC)
2. Minimize Reprojection error minimization

\[
T_{k,k-1} = \arg \min_T \sum_i \| I_k(u_i) - I_{k-1}(u_i) \|_\sigma
\]

where \( u'_i = \pi(T \cdot (\pi^{-1}(u_i) \cdot d)) \)

\[
\checkmark \text{Large frame-to-frame motions}
\]
\[
\times \text{Slow due to costly feature extraction and matching}
\]

Our solution:

**SVO**: Semi-direct Visual Odometry [ICRA’14]

Combines feature-based and direct methods

\[
\text{Limited frame-to-frame motion}
\]

[Jin,Favaro,Soatto’03] [Silveira, Malis, Rives, TRO’08], [Newcombe [Engel et al., ECCV’14], [Forster et al., ICRA’14]
SVO Results: Fast and Abrupt Motions
https://www.youtube.com/watch?v=2YnIMfw6bJY

[Forster, Pizzoli, Scaramuzza, «SVO: Semi Direct Visual Odometry», ICRA’14]
Processing Times of SVO

Laptop (Intel i7, 2.8 GHz)

400 frames per second

Embedded ARM Cortex-A9, 1.7 GHz

Up to 70 frames per second
Probabilistic Depth Estimation

Depth-Filter:

- **Depth Filter** for every feature
- **Recursive Bayesian** depth estimation

Mixture of Gaussian + Uniform distribution

\[
p(\tilde{d}_i^k | d_i, \rho_i) = \rho_i \mathcal{N}(\tilde{d}_i^k | d_i, \tau_i^2) + (1 - \rho_i) \mathcal{U}(\tilde{d}_i^k | d_i^{\text{min}}, d_i^{\text{max}})
\]

[Forster, Pizzoli, Scaramuzza, SVO: Semi Direct Visual Odometry, IEEE ICRA’14]
Visual-Inertial Fusion via Optimization [RSS’15]

- Fusion solved as a *non-linear optimization problem* (no Kalman filter):
  - Increased accuracy over filtering methods

\[
\sum_{(i,j) \in \mathcal{K}_k} \| r_{L_{ij}} \|^2_{\Sigma_{ij}} + \sum_{i \in \mathcal{K}_k} \sum_{l \in \mathcal{C}_i} \| r_{C_{il}} \|^2_{\Sigma_C}
\]

*IMU residuals*  
*Reprojection residuals*

Comparison with Previous Works

Forster, Carlone, Dellaert, Scaramuzza, IMU Preintegration on Manifold for efficient Visual-Inertial Maximum-a-Posteriori Estimation, *Robotics Science and Systems’15*

*Proposed* ASLAM

Accuracy: 0.1% of the travel distance

Open Source

https://www.youtube.com/watch?v=CsJkci5lfco
Integration on a Quadrotor Platform
Quadrotor System

Odroid U3 Computer
- Quad Core Odroid (ARM Cortex A-9) used in Samsung Galaxy S4 phones
- Runs Linux Ubuntu and ROS

Global-Shutter Camera
- 752x480 pixels
- High dynamic range
- 90 fps

450 grams
Quadrotor System

Odroid U3 Computer
- Quad Core Odroid (ARM Cortex A-9) used in Samsung Galaxy S4 phones
- Runs Linux Ubuntu and ROS

450 grams
Indoors and outdoors experiments

RMS error: 5 mm, height: 1.5 m – Down-looking camera

Speed: 4 m/s, height: 1.5 m – Down-looking camera

Visual-Inertial Odometry - Results
https://www.youtube.com/watch?v=6KXBoprGaR0

SVO with a single camera on Euroc dataset

Forster, Carlone, Dellaert, Scaramuzza, IMU Preintegration on Manifold for efficient Visual-Inertial Maximum-a-Posteriori Estimation, *Robotics Science and Systems’15*, Best Paper Award Finalist
Application: Autonomous Inspection of Bridges and Power Masts

Project with Parrot: Autonomous vision-based navigation

[https://www.youtube.com/watch?v=BxVoaLrKJ4U](https://www.youtube.com/watch?v=BxVoaLrKJ4U)
From Sparse to Dense 3D Models

[M. Pizzoli, C. Forster, D. Scaramuzza, REMODE: Probabilistic, Monocular Dense Reconstruction in Real Time, ICRA’14]
Dense Reconstruction Pipeline

- **Local methods**
  - Estimate depth for every pixel independently using **photometric cost aggregation**

- **Global methods**
  - Refine the depth surface as a whole by enforcing smoothness constraint (**“Regularization”**)

\[
E(d) = E_d(d) + \lambda E_s(d)
\]

- Data term
- Regularization term: penalizes non-smooth surfaces

[Newcombe et al. 2011]
**REMODE**: Probabilistic Monocular Dense Reconstruction [ICRA’14]

- **Track independently** every pixel using the same recursive **Bayesian depth estimation of SVO**

\[
p(d_i^k | d_i, \rho_i) = \rho_i \mathcal{N}(d_i^k | d_i, \tau_i^2) + (1 - \rho_i) \mathcal{U}(d_i^k | d_i^{\min}, d_i^{\max})
\]

- A **regularized depth map** \( F(u) \) is computed from the noisy depth map \( D(u) \) as

\[
\min_F \int_{\Omega} \{ G(u) \| \nabla F(u) \|_\epsilon + \lambda \| F(u) - D(u) \|_1 \} \, du
\]

where

\[
G(u) = E_\rho[q](u) \frac{\sigma^2(u)}{\sigma_{\max}^2} + \{ 1 - E_\rho[q](u) \}
\]

Minimization is done using [Chambolle & Pock, 2011]

[M. Pizzoli, C. Forster, D. Scaramuzza, REMODE: Probabilistic, Monocular Dense Reconstruction in Real Time, ICRA’14]
REMODE: Probabilistic Monocular Dense Reconstruction [ICRA’14]
https://www.youtube.com/watch?v=QTKd5UWCG0Q
Running at 50 Hz on GPU on a Lenovo W530, i7

Monocular dense reconstruction in real-time from a hand-held camera

Open Source

[M. Pizzoli, C. Forster, D. Scaramuzza, REMODE: Probabilistic, Monocular Dense Reconstruction in Real Time, ICRA’14]
Autonomus, Flying 3D Scanning

https://www.youtube.com/watch?v=7-kPiWaFYAc

• Sensing, control, state estimation run onboard at 50 Hz (Odroid U3, ARM Cortex A9)
• Dense reconstruction runs live on video streamed to laptop (Lenovo W530, i7)

Autonomous, Flying 3D Scanning [JFR’15]

https://www.youtube.com/watch?v=7-kPiWaFYAc

- Sensing, control, state estimation run onboard at 50 Hz (Odroid U3, ARM Cortex A9)
- Dense reconstruction runs live on video streamed to laptop (Lenovo W530, i7)

Autonomous Landing-Spot Detection and Landing

https://www.youtube.com/watch?v=phaBKFWfCJ4

Outlook
To go faster, we need faster sensors!

- At the current state, the agility of a robot is limited by the latency and temporal discretization of its sensing pipeline.

- Currently, the average robot-vision algorithms have latencies of 50-200 ms. This puts a hard bound on the agility of the platform.

- Can we create a low-latency, low-discretization perception pipeline?
  - Yes, if we use **event-based cameras**

[Censi & Scaramuzza, «Low Latency, Event-based Visual Odometry», ICRA’14]
Human Vision System

- 130 million photoreceptors
- But only 2 million axons!
Dynamic Vision Sensor (DVS)

- **Event-based camera** developed by Tobi Delbruck’s group (ETH & UZH).
- Temporal resolution: **1 μs**
- High dynamic range: **120 dB**
- Low power: **20 mW**
- Cost: **2,500 EUR**

![DVS Camera Image](image)

Image of the solar eclipse (March’15) captured by a DVS (courtesy of IniLabs)

Camera vs Dynamic Vision Sensor

https://www.youtube.com/watch?v=LauQ6LWTkxM
Camera vs Dynamic Vision Sensor

https://www.youtube.com/watch?v=LauQ6LWTkxM
High-speed State Estimation

https://www.youtube.com/watch?v=iZZ77F-hwzs

[Event-based Camera Pose Tracking using a Generative Event Model, Under review. Available at arXiv]

[Censi & Scaramuzza, Low Latency, Event-based Visual Odometry, ICRA’14]
Conclusions

- Visual-inertial state estimation is crucial for GPS-denied navigation.
- More accurate than GPS, DGPS, and RTK-GPS.
- Pending challenges: robustness to changing illumination and high-speed motion.
- Event cameras are revolutionary visual sensors that can address such challenges where standard cameras fail.