PIVO: Probabilistic Inertial-Visual Odometry for Occlusion-Robust Navigation

Arno Solin[1] Santiago Cortés[1] Esa Rahtu[2] Juho Kannala[1]

[1]Aalto University, Finland
[2]Tampere Technical University, Finland

TAMPERE UNIVERSITY OF TECHNOLOGY

Aalto University

Introduction

- Novel visual-inertial odometry method
- Information fusion for the low-cost IMU sensors (gyroscope and accelerometer) and the monocular camera in a smartphone
- **Problem**: Previous methods visual-heavy and thus sensitive to the visual environment
- **Novelty**: Takes into account all the cross-terms in the visual updates
- Thus propagating the inter-connected uncertainties throughout the model
- Robustness against occlusion and feature-poor environments
- Stronger coupling between the inertial and visual data
  → "Inertial-visual odometry"

Information Fusion

- The IMU data drives the dynamics
- Complemeted by visual updates
- Formulated as a statistical information fusion problem
- Non-linear filtering by an extended Kalman filter (EKF)
- Exact up to the first-order linearizations in the EKF

IMU propagation model

- We leverage on recent advances in inertial navigation on smartphones [2]
- For each time step t_k, the state holds the position $\mathbf{p}_k$, velocity $\mathbf{v}_k$, orientation $\mathbf{q}_k$, and sensor biases
  $\mathbf{x}_k = (\mathbf{p}_k, \mathbf{q}_k, \mathbf{v}_k, \mathbf{b}^{\mathrm{a}}_k, \mathbf{b}^{\omega}_k, \mathbf{T}^{\mathrm{a}}_k, \boldsymbol{\pi}^{(1)}, \boldsymbol{\pi}^{(2)}, \ldots, \boldsymbol{\pi}^{(n_\mathrm{a})})$
- A trail of past poses $\boldsymbol{\pi}^{(j)}$ to be coupled by the visual updates
- IMU propagation is done with the mechanization equations

$$\begin{pmatrix}\mathbf{p}_k \\ \mathbf{v}_k \\ \mathbf{q}_k\end{pmatrix} = \begin{pmatrix} \mathbf{p}_{k-1} + \mathbf{v}_{k-1}\Delta t_k \\ \mathbf{v}_{k-1} + [\mathbf{q}_k(\tilde{\mathbf{a}}_k + \varepsilon^{\mathrm{a}}_k)\mathbf{q}^{\star}_k - \mathbf{g}]\Delta t_k \\ \Omega[(\tilde{\boldsymbol{\omega}}_k + \varepsilon^{\omega}_k)\Delta t_k]\mathbf{q}_{k-1} \end{pmatrix}$$

- Double-integrating the accelerations $\mathbf{a}_k$ corrected by the gyroscope rotations $\boldsymbol{\omega}_k$

Algorithm 1: Outline of the PIVO method.

```
Initialize the state mean and covariance
foreach IMU sample pair (a_k, ω_k) do
    Propagate the model with the IMU sample                  see Sec. 3.2
    Perform the EKF prediction step
    if new frame is available then
        track visual features
        foreach feature track do
            Jointly triangulate feature using poses in state and
              calculate the visual update proposal            see Sec. 3.4
            if proposal passes check then
                Perform the EKF visual update
        Update the trail of augmented poses                  see Sec. 3.3
```

Visual updates

- Features tracked by Good features to track and a pyramidal Lucas–Kanade tracker
- Visual update performed per tracked feature
- The observed data are the pixel coordinates of the feature trail
- The 3D location of the feature point is triangulated by Gauss–Newton minimization of reprojection error
- The feature location couples the augmented poses in the state
- The entire update procedure differentiated (including the Gauss–Newton iteration) for the EKF update
- This way the 3D position of the feature is integrated out in the update
- Outlier rejection by innovation tests

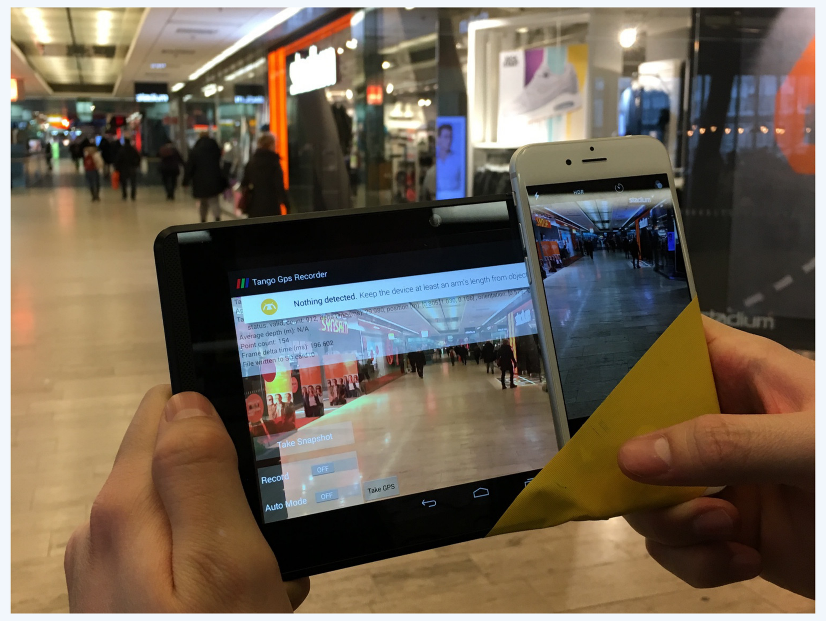

Figure 2: Test setup for comparing the Tango device and an iPhone.

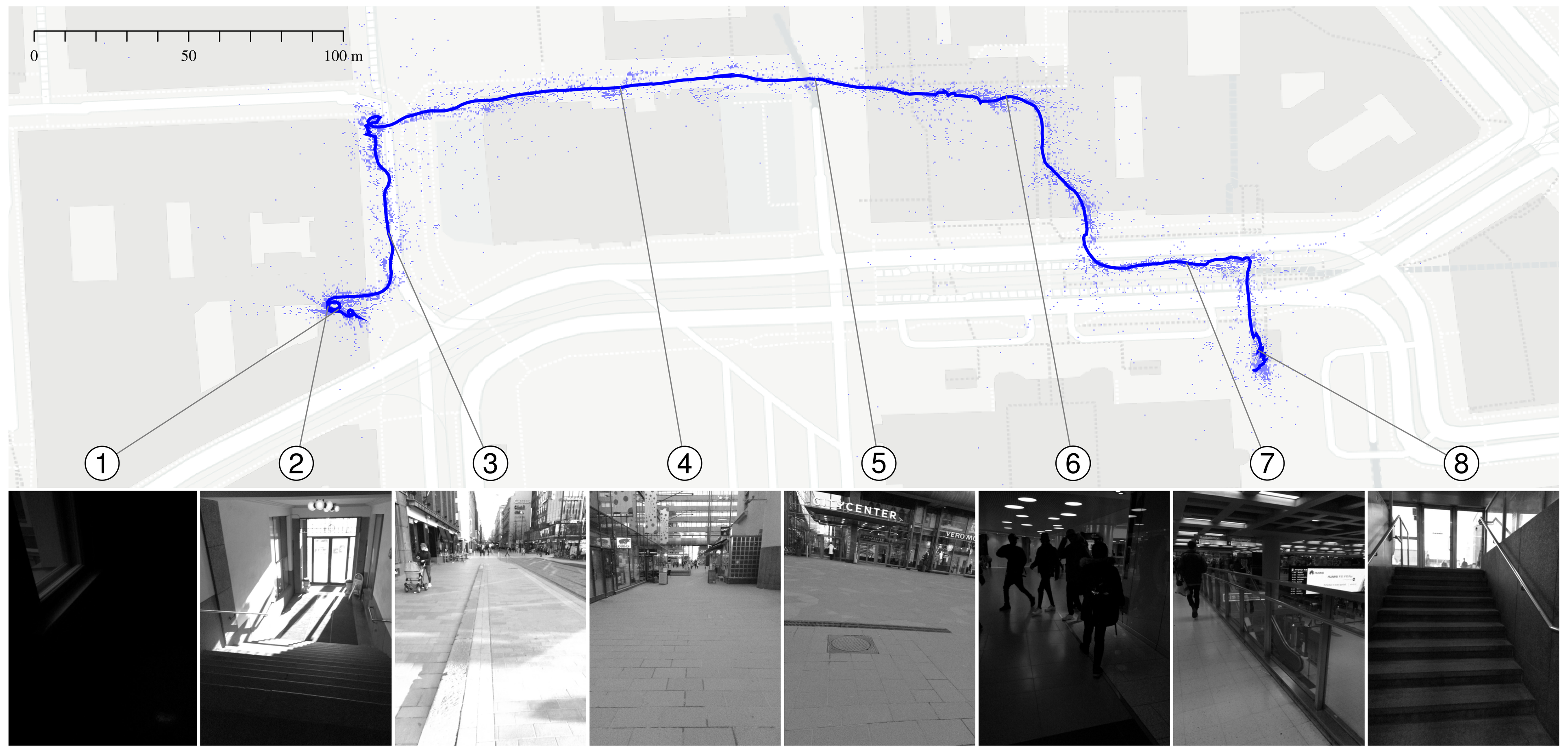

Figure 1: PIVO tracking on a smartphone (iPhone 6) starting from an office building (1–2), through city streets (3–5), a shopping mall (6), and underground transportation hub (7–8). Path length ∼600 meters.

- We perform comparisons of the visual update model to
  (i) brute-force Monte Carlo simulation
  (ii) the MSCKF method [3]
- Comparison examples in Figure 3

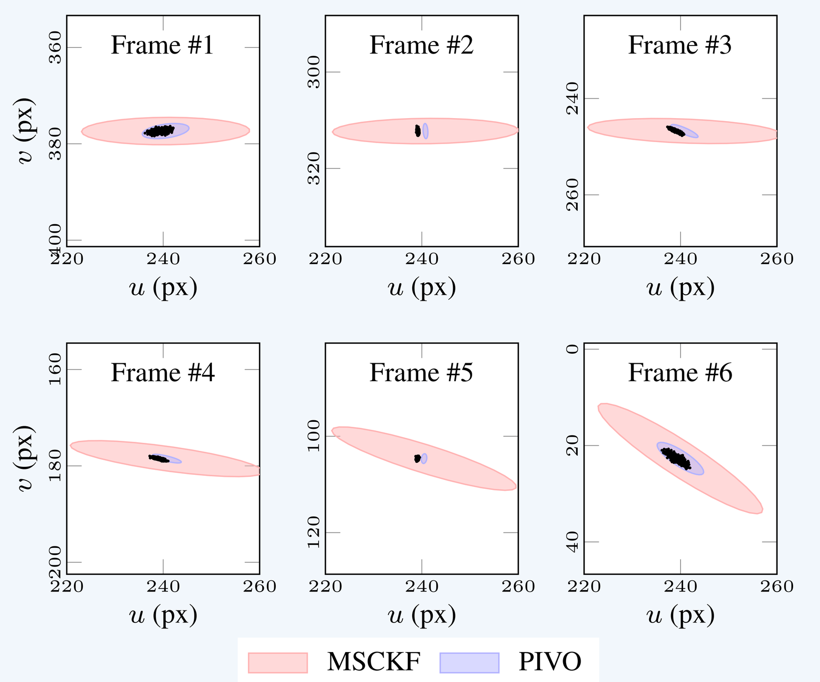

Figure 3: A visual feature observed by a trail of camera poses with associated uncertainties. The black dots are the 'true' distributions for the visual update model. The red patch shows shape of the Gaussian approximation used by the MSCKF [3] visual update, and blue shows the shape of the approximation used by PIVO.

Benchmarks

- EuRoC MAV data
- Dataset of a micro aerial vehicle with a mounted stereo camera and IMU and external ground-truth
- Comparable RMSE error with state-of-the-art
- Several passes of the same scene better suited for map building algorithms

Occlusion Experiment

- Robustness to occlusion compared to the Google Tango device
- Experiment setup in Figure 2
- A small scene is traversed.
- For some portions of the walk, the camera is completely occluded
- The odometry system keeps correct motion and is not confused by the occlusion
- **Note:** No map building done

Large Experiment

- City-wide navigation: Figure 1
- Walking through a busy city center, indoors/outdoors, with partial occlusions and dynamic objects in the scene
- Used hardware: Apple iPhone 6
- Path length: ∼600 meters
- Manual alignment with city map shows the trajectory remains consistent in scale and orientation

Discussion

- Principled approach for fusing inertial and visual information
- PIVO shows robustness to occlusion
- Robustness dynamic objects moving in the scene
- PIVO is comparable with state-of-the-art algorithms in ideal scenes
- Improved performance in challenging conditions

References

[1] A. Solin, S. Cortés, E. Rahtu, and J. Kannala (2018). PIVO: Probabilistic inertial-visual odometry for occlusion-robust navigation. *Proceedings of WACV*.

[2] A. Solin, S. Cortés, E. Rahtu, and J. Kannala (submitted). Inertial odometry on handheld smartphones. *arXiv preprint arXiv:1703.00154*.

[3] A.I. Mourikis and S.I. Roumeliotis (2007). A multi-state constraint Kalman filter for vision-aided inertial navigation. *Proceedings of ICRA*.

Project Page

https://aaltovision.github.io/PIVO

Scan me

Contact details: arno.solin@aalto.fi, web address: http://arno.solin.fi, WACV 2018, USA.