Velocity estimation from IMU and camera

Oscar Fischerström, Isac Svensson

Introduction

When landing or launching helicopters, especially on unsteady surfaces such as at sea, a good velocity estimation is essential. Our research question was to find out if camera measurements could complement IMU measurements to improve the velocity estimation in a GNSS-free environment.

Digital image processing

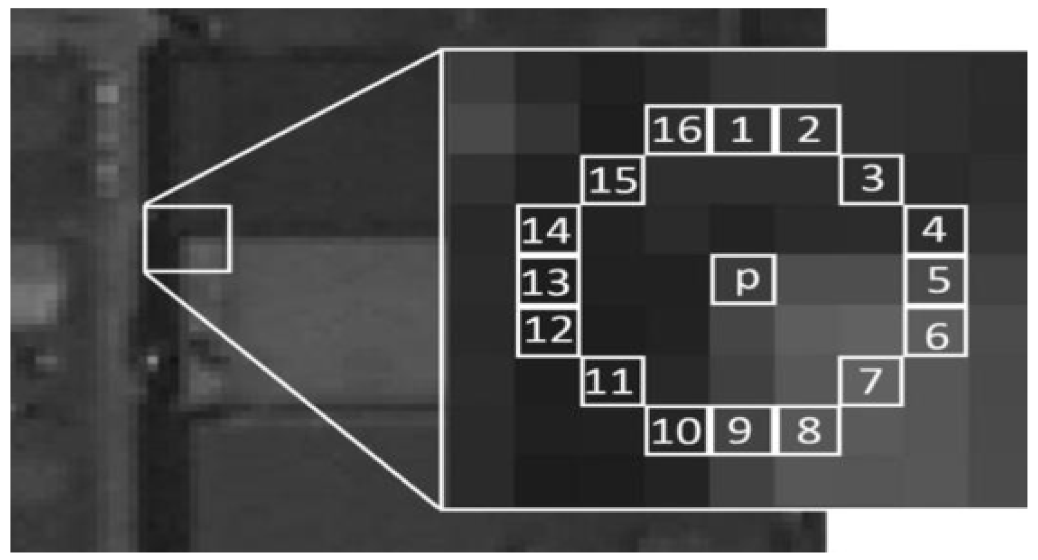

Figure 1: A detected feature

A helicopter is equipped with a downward-facing camera. To use this video feed as input data we found distinct points (features) in each frame, using ORB feature detector. Features are detected in places like corners and other regions with high intensity variance. These features' movement between two frames are then tracked by matching them with features detected in the next frame. The matching is done by comparing their surrounding area (descriptors) to each other.

A different approach to the tracking problem is the Kanade-Lucas-Tomasi-algorithm (KLT). This method instead of using descriptors, assumes that the feature has a similar intensity and only moves a small distance. Therefore KLT only needs ORB features in the first frame, which it can then track over time. KLT works well for video feeds with relatively high frame rate.

Figure 2: Feature tracking in video feed

Model descriptions

Two separate models were implemented. The first had the measurement equation

$$0 = h(\mathbf{x}_t, \mathbf{y}_t) = \dot{\tilde{\mathbf{m}}}_n^\top(\mathbf{v}_C \times \tilde{\mathbf{m}}_n) + \tilde{\mathbf{m}}_n^\top(\boldsymbol{\omega}_C \times (\mathbf{v}_C \times \tilde{\mathbf{m}}_n)), \quad (1)$$

The system dynamics model

$$\mathbf{p}_t^W = \mathbf{p}_{t-1}^W + T\,\mathbf{v}_{t-1}^W + \tfrac{T^2}{2}(\mathbf{R}_{t-1}^{WB}(\mathbf{a}_t^B + \mathbf{n}_t^a) + \mathbf{g}^W), \quad (2)$$

$$\mathbf{v}_t^W = \mathbf{v}_{t-1}^W + T(\mathbf{R}_{t-1}^{WB}(\mathbf{a}_t^B + \mathbf{n}_t^a) + \mathbf{g}^W), \quad (3)$$

$$\boldsymbol{\theta}_t = \boldsymbol{\theta}_{t-1} + T\,\mathbf{R}^{WB}(\boldsymbol{\omega}_t^B + \mathbf{n}_t^\omega), \quad (4)$$

The second model used SLAM to estimate position and from that indirectly estimated velocity. The system dynamics model was the same as the other model, aside from the fact that landmark positions also were part of the states. This model had the measurements described as

$$\begin{bmatrix} X_t^c \\ Y_t^c \\ Z_t^c \end{bmatrix} = \boldsymbol{R}_{cw}(\boldsymbol{\theta}_t)(\boldsymbol{p}_{t,l,i}^W - \boldsymbol{p}_{t,c}^W) + \boldsymbol{r}^{cb} \quad (5)$$

$$\begin{bmatrix} u_i \\ v_i \\ 1 \end{bmatrix} = \boldsymbol{K} \begin{bmatrix} -X_i^c/Z_i^c \\ -Y_i^c/Z_i^c \\ 1 \end{bmatrix} \quad (6)$$

Results

In Figure 3 we can see the resulting velocity estimate of a UAV flying in a square pattern with constant heading. In this result the estimated velocity is quite close to ground truth.

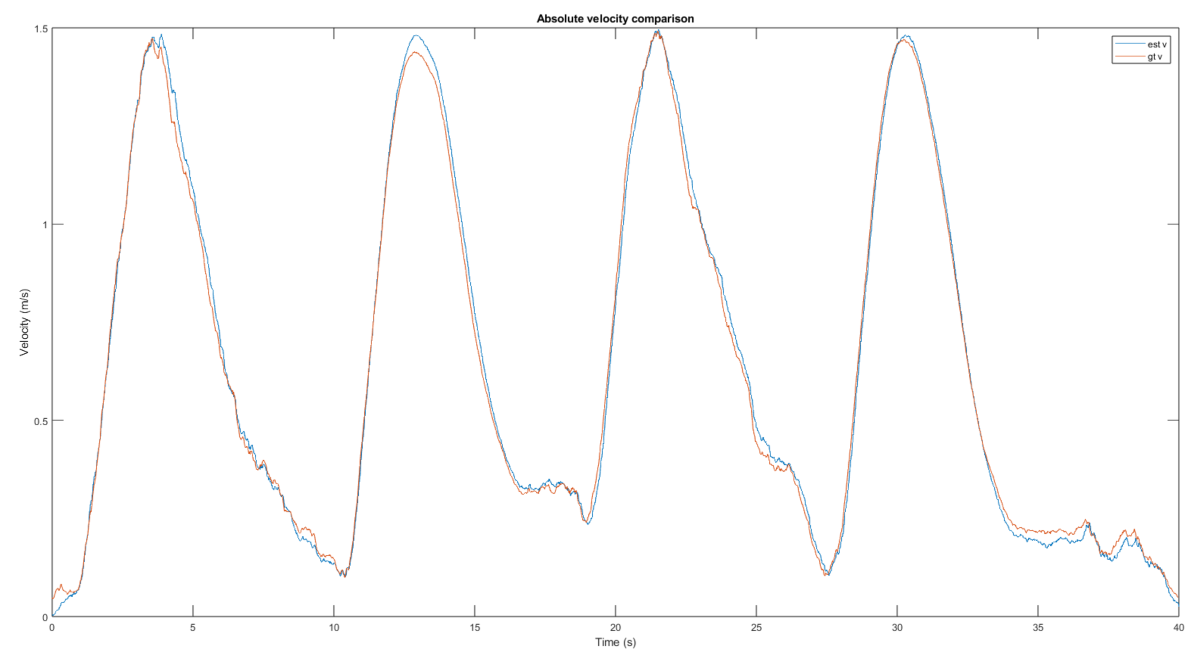

Figure 3: Estimate vs ground truth for a square pattern

Conclusions

- ● The video helped reduce drift for the estimate
- ✗ Small camera movements gave large movement in image
- ✗ Difficulties synchronizing IMU and camera in time

Thanks to all our collaborators.

LINKÖPING UNIVERSITY

SEDIT
Sensor informatics and Decision-making for the Digital Transformation

UPPSALA UNIVERSITET