Image-aided inertial navigation for an Octocopter

Baheerathan, S., Hagen, O. K.

S. Baheerathan, O. K. Hagen, "Image-aided inertial navigation for an Octocopter," Proc. SPIE 10640, Unmanned Systems Technology XX, 106400J (3 May 2018); doi: 10.1117/12.2301229

SPIE.

Event: SPIE Defense + Security, 2018, Orlando, Florida, United States

Image-aided inertial navigation for an Octocopter

S. Baheerathan* and O.K. Hagen
Norwegian Defence Research Establishment, P.O.Box 25, N-2027, Kjeller, Norway

ABSTRACT

A typical unmanned aerial system combines an Inertial Navigation System (INS) and a Global Navigation Satellite System (GNSS) for navigation. When the GNSS signal is unavailable, the INS errors grow over time and eventually become unacceptable as a navigation solution. Here we investigate an image-aided inertial navigation system to cope with GNSS failure. The system is based on tightly integrating inertial sensor data with position data of image-feature-points that corresponds to landmarks over an image sequence. The aim of this experiment is to study the challenges and the performance of the image-aided inertial navigation system in realistic flight with an Octocopter. The system demonstrated the ability to cope with the GNSS failure by reducing the position drift drastically compared to the position drift of free-inertial.

Keywords: Unmanned aerial system, inertial-navigation-system, landmark, image-feature-point

1. INTRODUCTION

A precise and robust navigation system is very important for an Unmanned Aerial Vehicle (UAV). In a typical UAV navigation system, this is fulfilled by combining data from an Inertial Navigation System (INS) and the position data from a Global Navigation Satellite System (GNSS). When the GNSS signal is unavailable, INS alone can only provide an acceptable solution for a short time before the INS error exceeds acceptable limits. Therefore, there is a need for an alternative to the GNSS-aided navigation system, to cope with GNSS failure.

One of the main research activities to reduce the dependency on GNSS-aided navigation is image-aided navigation. The Norwegian Defence Research Establishment (FFI) has been investigating an image-aided inertial navigation system to cope with GNSS failure[1-12]. All the developed systems track image-feature-points (T in Figure 1) in video streams or image sequences, from a monocular camera. In the earlier approaches[2-4] these were integrated with a flight dynamics model, and a digital terrain model (DTM) for initial estimate of the range to the landmark (L in Figure 1), in an Extended Kalman Filter (EKF). In[6] this was further developed by integrating an inertial measurement unit (IMU) in a strap-down INS implementation, where there is no longer a need for a flight dynamics model. The last system developed[9] is also based on a tight integration with an INS using an error-state EKF, but has no need for a DTM, or any other world models.

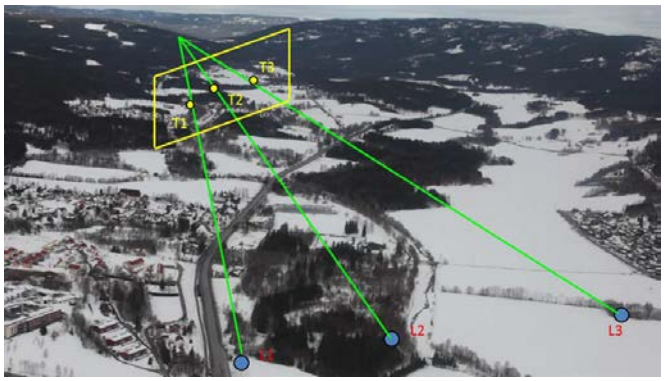

Figure 1. A landmark (L) is a static point on a surface in the scene. The image-feature-point (T) is the projection of this landmark onto the image plane via a camera model.

*baheerathan.sivalingam@ffi.no: phone 47 63 80 76 81

Unmanned Systems Technology XX, edited by Robert E. Karlsen, Douglas W. Gage,
Charles M. Shoemaker, Hoa G. Nguyen, Proc. of SPIE Vol. 10640, 106400J
 · doi: 10.1117/12.2301229

As a continuation of image-aided inertial navigation research at FFI, this work studies the performance and the challenges of the image-aided inertial navigation system in a realistic UAV flight with an Octocopter. In the previous method[9], tracking of image-feature-points was guided by the current estimates from the Kalman filter. In the present system, tracking of image-feature-points is not necessarily guided by the estimate from the Kalman filter, and is therefore able to also run independently from the Kalman filter. A major advantage with this improvement is the ability to independently filter tracks of unreliable image-feature-points before integration with the Kalman filter.

The paper is organized as follows: Related work is presented in Section 2. Section 3 describes the studied image-aided inertial navigation system in brief and Section 4 details the inertial navigation system integration. Section 5 presents the experimental setup and the performance of the studied image-aided INS is presented in Section 6. Section 7 presents the summary

2. RELATED WORK

An image-aided inertial navigation system, which integrates imaging sensors with inertial sensors, is able to cover the limitations and deficiencies of a standalone system[13-16]. In general, there are two architectures for vision and INS integration: loosely-coupled and tightly-coupled. In the loosely-coupled architecture, the position and the attitude obtained by the camera images are used to aid the INS. In the tightly-coupled architecture, the pixel coordinates of feature positions in an image, instead of the position and attitude obtained from them, are fused with inertial measurements. Performance of the tightly and loosely-coupled architecture is compared by simulation in[17].

Tight integration with image features and INS have been done by several authors. Although the name inverse depth is used in[18], the parameterization used herein is different. In[18] camera position, direction and inverse distance to landmarks, 6 states per landmark, are added to the state vector, while we only use 3 states per landmark: normalized pinhole coordinates and inverse depth along the optical axis. The parameterization we use for the landmark states is the one in[19] that used a single landmark in a local frame. In[20] the parameterization was extended to multiple landmarks in a global frame. The latter two works used a direct Unscented Kalman Filter (UKF) in the implementation of the INS. In our approach, we use an INS with an error-state (indirect) EKF.

This work also builds on[9-10]. There, the image-feature-points are tracked using feature vector computed from the circular integral of pixel values[5], and tracking is guided by the current estimates from the Kalman filter, meaning that the tracking is performed within a search region around a predicted pixel position. Here we instead use the Kanda-Lucas-Tomasi (KLT) tracker[21-23] which does not use search region. Then the tracking runs independently from the Kalman filter, so that it is possible to independently filter unreliable tracks. Another difference is that we now exploit the structure of the differential equations to solve these more efficiently, and accurately.

3. IMAGE-AIDED INERTIAL NAVIGATION SYSTEM

The concept of the image-aided inertial navigation system is illustrated in Figure 2. The integrated navigation tool NavLab[24] is used to integrate the imaging sensor with the inertial navigation sensors. Initially, ranges to the landmarks are set by the approach described in[11] (more details in Section 4).

Then, at each time step, the INS computes the estimation of the aircraft's state, and this estimate is used together with the landmark positions by the Kalman filter to predict the positions of the image-feature-points along with their uncertainties.

In[9], tracking of the image-feature-points was guided by the current estimates from the INS, by projecting the estimated uncertainties of the Kalman filter into a search region in the image pixel plane. The image processing unit then tracked the image-feature-points within these search regions, and their tracked positions were used to update the Kalman filter of the INS. Using these search regions are beneficial for real-time tracking. However, in the tightly-coupled system, we found that unreliable tracks could affect the navigation solution adversely[12].

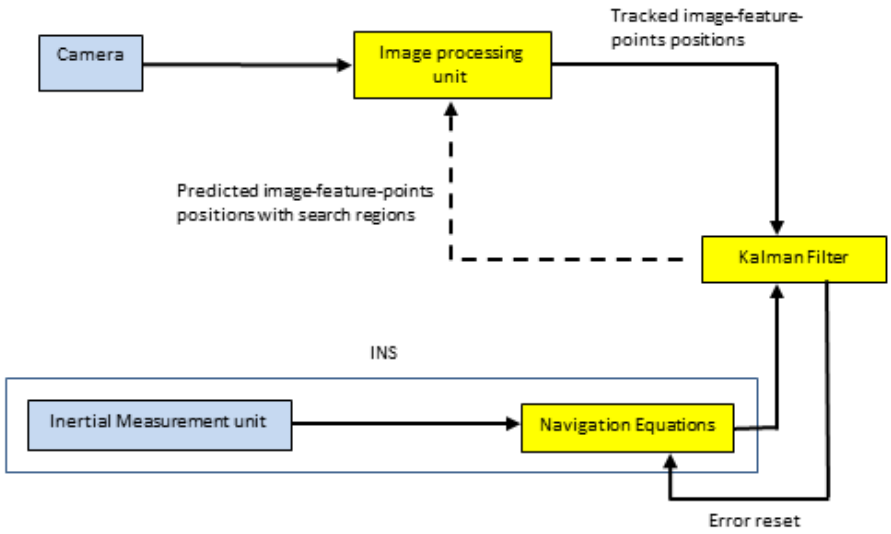

Figure 2. Image-aided inertial navigation system

In the current system, the image processing unit tracks the image-feature-points without necessarily using prediction or search regions from the Kalman filter (dashed line in Figure 2). The image processing unit runs independently from the Kalman filter. Thus, the current system is not as tightly integrated as the system in[9]. This is beneficial for the possibility of independent filtering of the unreliable tracks, and it also gives the possibility to test and develop the tracking process independently of the Kalman filter, which saves execution time in offline processing. However, the current system has the possibility to also use the search regions around the predicted positions to track the image-feature-points.

3.1 Image processing

The image processing unit performs image-feature-points selection, tracking and validation. The image-feature-points selection process has two steps. In the first step, image-feature-points are selected within a predefined region in an image using one of the standard feature detection methods such as SIFT[25], SURF[26], ORB[27], FAST[28] or Harris[29]. Here we used the ORB method. Then the image-feature-points that have high scores of goodness measure (Harris corner measure[29]), and an intra distance above a predefined distance, are selected for tracking. An algorithm for identifying image-feature-points which correspond to static points is presented in[9]. Each selected image-feature-point is represented by an image-patch of 27 x 27 pixels around its position[8].

The selected image-feature-points are tracked in the consecutive images of an image sequence using the KLT[23] tracking. KLT makes use of spatial intensity information in different levels of image pyramid to search for the position that yields the best match.

The validation process has three steps. In the first step, the image-feature-points that are identified as invalid by the KLT tracker are not accepted. The accepted points undergo step 2 and 3. In step 2, tracked image-feature-points which have pixel positions outside the predefined image border are not accepted. In the third step, template matching based on normalized cross-correlation (NCC[30]) is performed between the image-patch which represents an image-feature-point and the image-patch around its tracked pixel position. The image-feature-points which have NCC values less than a predefined threshold are not accepted. When the number of accepted tracked image-feature-points is less than a predefined minimum required number, new image-feature-points are selected and added to the KLT tracker.

4. INERTIAL NAVIGATION SYSTEM INTEGRATION

4.1 Reference systems

We introduce the following notation for the reference systems:

- I – an earth centered inertial reference frame.
- E – an earth-fixed earth centered reference frame.
- L – a local level reference frame with origin in the platform body.
- B – a body-fixed reference frame with the same origin as L.
- B_c – a camera reference frame with origin at camera focal point, x-axis outwards along optical axis, y-axis right and z-axis down.
- B_{imu} - a body-fixed reference frame with origin and axes defined by the particular IMU sensor.
- C – image plane at $x = 1$ in B_c, y- and z-axis aligned with B_c.
- P – pixel plane, origin at top-left image corner, y- and z-axis aligned with C.

By *landmark* we mean a point on a surface that is static in the *E* reference frame. A *landmark-projection-point* is the projection of a landmark on to the *C* image plane, while an *image-feature-point* is the corresponding point in the *P* pixel plane.

In the mathematical description below, we will for notational convenience only, assume that $B = B_{imu} = B_c$. The real implementation of the system described herein, however, compensates for the different lever arms and alignments between the different sensors and the platform.

4.2 Strap-down inertial navigation

A strap-down INS is based on integrating measurements from a body-fixed IMU, assumed now in *B*. The INS solves a set of differential equations, based on Newton's 2nd law with accelerometer and gyro input, to estimate velocity, position and orientation of *B*. We use a wander azimuth mechanization[31] frame *L* for these equations.

The orientation is then represented by the rotation matrix from the navigation frame to *L* to *B*, $\mathbf{R}_{LB}$. The velocity vector of *B* w.r.t *E*, decomposed in *L*, is represented by $\mathbf{v}_{EB}^{L}$, and the horizontal position is represented by the rotation matrix from *E* to *L*, $\mathbf{R}_{EL}$, while scalar z_{EB}^{L} denotes the vertical position.

The strap-down differential equations are then given by[31]

$$
\begin{aligned}
\dot{\mathbf{R}}_{LB} &= \mathbf{R}_{LB}\mathbf{S}(\boldsymbol{\omega}_{IB}^{B} - \mathbf{R}_{LB}^{T}(\mathbf{R}_{EL}^{T}\boldsymbol{\omega}_{IE}^{E} + \boldsymbol{\omega}_{EL}^{L})) \\
\dot{\mathbf{v}}_{EB}^{L} &= \mathbf{R}_{LB}\mathbf{f}_{IB}^{B} - (2\mathbf{R}_{EL}^{T}\boldsymbol{\omega}_{IE}^{E} + \boldsymbol{\omega}_{EL}^{L}) \times \mathbf{v}_{EB}^{L} + \mathbf{g}_{IB}^{L} \\
\dot{\mathbf{R}}_{EL} &= \mathbf{R}_{EL}\mathbf{S}(\boldsymbol{\omega}_{EL}^{L}) \\
\dot{z}_{EB}^{L} &= v_{EB,z}^{L}.
\end{aligned}
\tag{1}
$$

In order to solve (1), we use gyro measurements for $\boldsymbol{\omega}_{IB}^{B}$, specific force measurements for $\mathbf{f}_{IB}^{B}$, while $\boldsymbol{\omega}_{IE}^{E}$ is a model of earth-rotation, and $\mathbf{g}_{IB}^{L}$ is the local plumb-bob gravity model. The last missing term, $\boldsymbol{\omega}_{EL}^{L}$, is the turn-rate of the navigation frame that depends on the platform's velocity and the local earth curvature[31].

4.3 Error-state extended Kalman filter

To implement the INS integration, we use NavLab 4, a flexible version of NavLab described in[24] that allows for a programmable state vector. NavLab 4 is a software package in MATLAB™, developed by FFI that implements an error-state EKF[32] based on INS error equations derived from equation (1).

NavLab 4 can also integrate a variety of aiding sensors to the INS, e.g. GPS, pressure sensor, magnetometer, through measurements updates to the error-state EKF. There are then two main computation loops in NavLab 4, the navigation equations (1) solved at IMU-rate, and the error-state EKF running at the combined aiding sensors measurements rates. We denote the nominal error-states, including any bias of the usual aiding sensors of the EKF by $\delta \mathbf{x}_{INS}$.

4.4 Tight integration with inverse depth parameterization

In the Kalman filter integration of the camera measurements, each landmark i is first given by three coordinates (X_i, Y_i, Z_i) in the camera reference system, which we, for notational simplicity only, assumed to be the same as B. When we project the landmark on the image plane at $X = 1$, we get the two normalized pinhole coordinates $(y_i, z_i) \in C$ of the landmark–projection-point i. The last parameter we use to complete the parameterization is the inverse depth ρ_i of the landmark-projection-point[9, 19-20]. The inverse depth is the reciprocal of the distance X_i along the optical axis from the camera's focal point to the landmark, as shown in Figure 3, i.e. $\rho_i = \frac{1}{X_i}$.

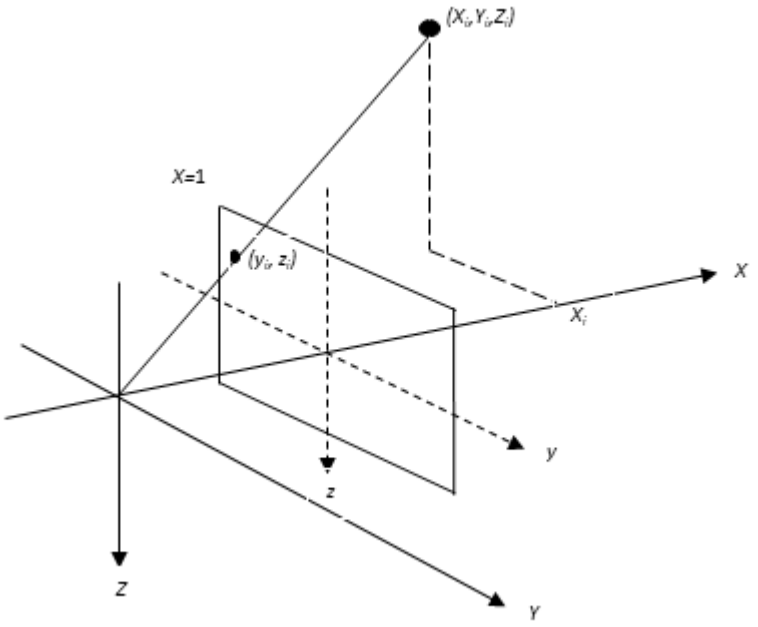

Figure 3. Inverse depth parameterization of a landmark.

In order to integrate this with the INS, we differentiate the inverse depth parameterization with time, exploiting that the landmark is static in E, and get the following differential equation for the landmark-projection-point[9-10]

$$\begin{bmatrix} \dot{y}_i \\ \dot{z}_i \\ \dot{\rho}_i \end{bmatrix} = \begin{bmatrix} y_i & -1 & 0 \\ z_i & 0 & -1 \\ \rho_i & 0 & 0 \end{bmatrix} \left(\rho_i \mathbf{R}_{LB}^T \mathbf{v}_{EB}^L + \boldsymbol{\omega}_{EB}^B \times \begin{bmatrix} 1 \\ y_i \\ z_i \end{bmatrix} \right). \quad (2)$$

Here $\boldsymbol{\omega}_{EB}^B$ denotes the angular velocity of the body with respect to E, which we find from the gyro measurements, compensating for the earth rotation and the turn-rate of the navigation frame. Since equation (2) depends on (1), but not the other way around, we can solve (2) efficiently. First we integrate the strap-down equations (1) at IMU-rate between the EKF measurements. The landmark-projection-points are then updated over this time interval by analytical triangulation[9]. The triangulation ensures that the solution from the integration of (1) and (2) satisfy the epipolar constraint[33].

The low order of non-linearity in (2) makes it feasible to integrate with the error-state EKF. The error states of the inverse depth parameterization of the landmarks in (2) are denoted by $\delta \mathbf{x}_{lm,i}$. The corresponding error-state differential equation is found to first order by a Taylor series expansion of the differential equation in (2). Since (2) is dependent on the INS solution, we recognize that the differential equations establishes correlation between $\delta \mathbf{x}_{INS}$ and $\delta \mathbf{x}_{lm,i}$, including gyro errors, but also between different landmark–projection-point error states $\delta \mathbf{x}_{lm,j}$ and $\delta \mathbf{x}_{lm,i}$.

Our complete error state then consists of nominal INS states and the landmark–projection-point states, 3 per current landmarks being tracked. When a landmark–projection-point goes out of the camera's field-of-view, it is discarded and the states are recycled for use by a new landmark.

The measurement equation of the EKF is straightforward, depending only on the camera model. For example, for a camera with effective focal length (f_y, f_z), optical axis center (c_y, c_z) and radial distortion coefficient k_0, we get

$$\begin{bmatrix} y_i \\ z_i \end{bmatrix}^P = (1 + k_0(y_i^2 + z_i^2)) \begin{bmatrix} f_y & 0 \\ 0 & f_z \end{bmatrix} \begin{bmatrix} y_i \\ z_i \end{bmatrix}^C + \begin{bmatrix} c_y \\ c_z \end{bmatrix}. \quad (3)$$

Because of the correlations established by (2), these measurement updates (3) will contribute to the estimation of the INS states, but also to the estimation of the inverse depth of the landmark, if the motion of the camera allows this to become observable[34].

One major problem is the initialization of the inverse depth for a new landmark. We adopt the heuristic approach suggested in[18], were the initial guess of ρ follows from a requirement that it should be Gaussian distributed with a 95% confidence interval from a configurable near range, $X_{\min}$, of the camera and to infinity, i.e. $[0,\ 1/X_{\min}]$.

5. EXPERIMENTAL SETUP

FFI's experimental UAV, Freefly CineStar 8 octocopter (Figure 4) is used this study. It is 1240 mm in diameter and weighs 3640 g. With payload and batteries, the weight is close to 11 kg. The onboard data collection system consists of three Grasshopper-3 cameras from PointGrey, an MTi100 MEMS IMU from Xsens, a u-blox GPS receiver, a solid state disk (SSD) with 512 GB and a Pico 880 microcomputer from Axiomtek. The microcomputer is connected to the three cameras through a USB 3.0 interface. The computer controls the camera exposure and stores the image data on the SSD. Each camera provides gray-scale images with 12 Hz frame rate and resolution of 2448 x 2040 pixels.

Figure 4. Freefly CineStar 8 Octocopter

The IMU is mounted over the middle camera. The navigation sensors are read out by a custom FPGA-based logging board which also triggers the cameras. Navigation data are time stamped and stored on an SD card together with the time stamp of the trigger pulses. The three cameras are mounted on a metal plate with a fixed angle between them such that the field-of-view within the cameras has an overlap. The whole data collection system is mounted under the octocopter (Figure 5).

Figure 5. Onboard data collection system mounted under the Octocopter

The experimental data is collected using the octocopter with the rig shown in Figure 5 in a flight in winter conditions at Eggemoen airstrip, about 90 km north-west of Oslo, Norway. Sample images from all three cameras are shown in Figure 6. The selected image-feature-points in the upper camera are shown in Figure 6(a).

(a) (b) (c)

Figure 6. Sample images from all three cameras. (a) Upper camera image with selected image-feature-points (b) Mid camera image (c) Down camera image

The purpose of having three cameras is to track image-feature-points for a long time through all three cameras. Image-feature-points that fall outside the field-of-view of a camera can be tracked in the adjacent camera's field-of-view. However, only the images from the upper camera are used in this study. The reason is that the images from the mid and down camera have no texture (snow only) in most of the images, as illustrated in the Figure 6.

6. PERFORMANCE

The data is processed and analyzed offline using the integrated navigation tool NavLab 4. The IMU data and GPS data are post-processed with the smoother in NavLab 4 and provide a reference data set for the flown position (north, east and altitude) and orientation of the Octocopter.

A GPS denied scenario (free-inertial) is simulated by disabling the GPS data after 900 seconds from the start for 430 seconds (900-1330) in NavLab 4. The Figures 7, 8 and 9, show the position drift of the free-inertial compared to the reference position in north, east and altitude respectively.

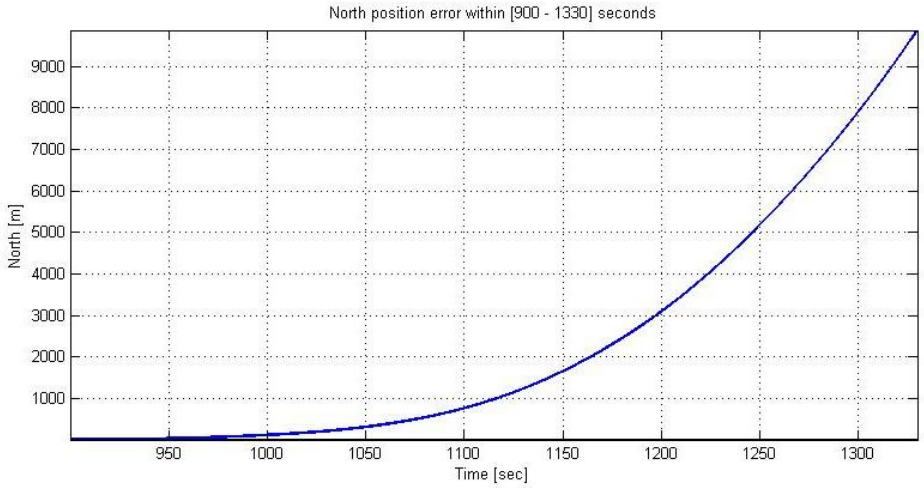

Figure 7. Position drift of free-inertial in north direction compared to the reference position.

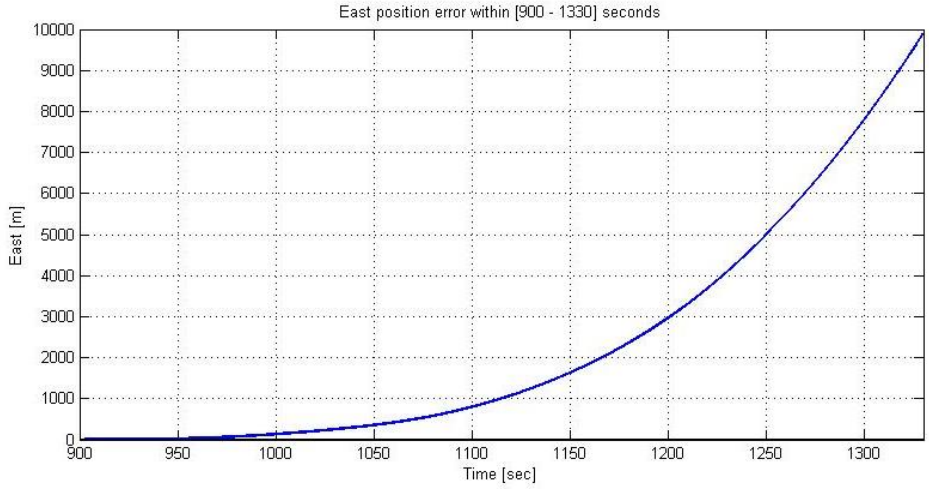

Figure 8. Position drift of free-inertial in east direction compared to the reference position.

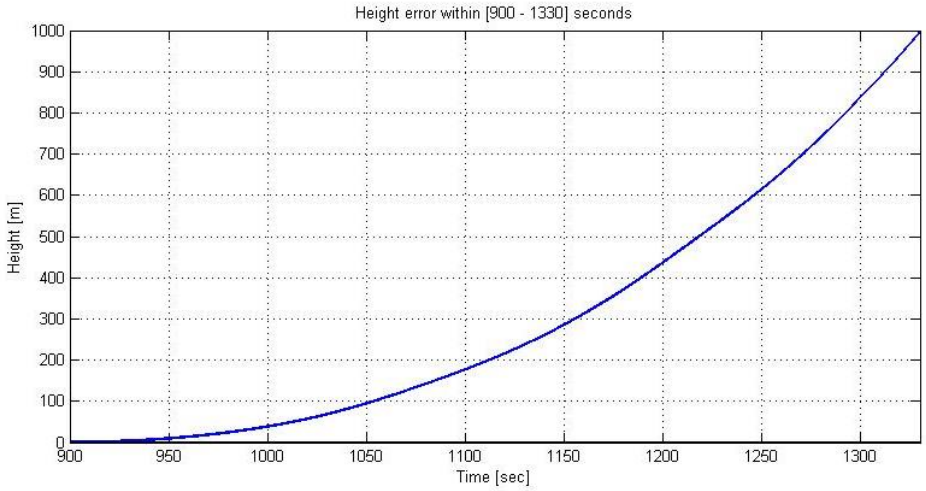

Figure 9. Altitude drift of free-inertial compared to the reference altitude.

The position drift of free-inertial in north and east direction is about 10000 m and in altitude 1000 m within 430 seconds. The position drift clearly describes the vulnerability of missing GPS for a short time, especially when using MEMS IMU.

The proposed image-aided INS is tested with image data (6 image-feature-points with 2 Hz frequency) and IMU data within the period of GPS failure (900–1330 sec.). The trajectory (Figure 10) and the position data of image-aided INS are compared with the reference data.

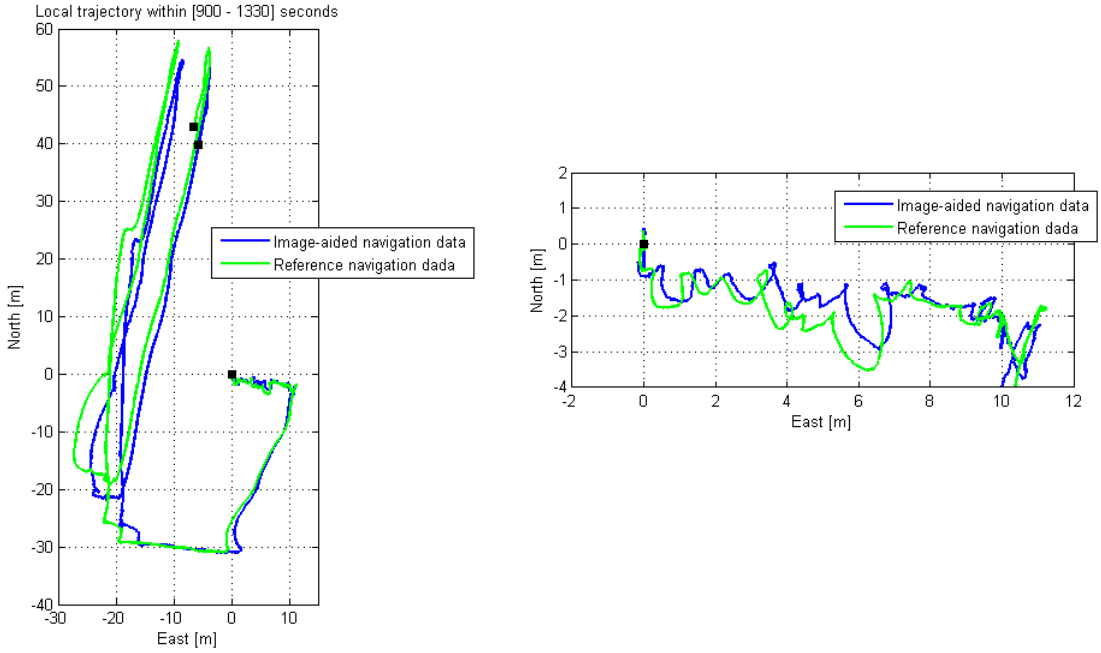

Figure 10. Comparison of flight trajectory. Left: Whole trajectory. Right: Trajectory at the beginning. Reference trajectory (green). Image-aided trajectory (blue).

Figure 10 shows that the image-aided trajectory (blue) closely follows the reference trajectory (green). Note in particular the similarities of the small curved trajectories at the beginning (Figure 10- right). However, there is an increased difference between the image-aided and the reference trajectory at the end.

Figure 11, 12 and 13 shows the position drift of the image-aided INS compared to the reference position in north, east and altitude respectively. In addition, the figures show the standard deviation (1 sigma) estimated by the EKF (blue dashed line) which depends on the models of the uncertainty of the sensors and the trajectory flown.

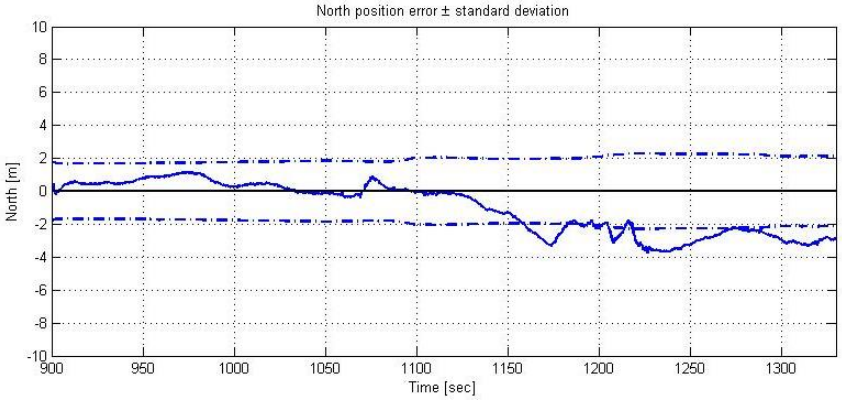

Figure 11. Position drift of the image-aided INS in north direction compared to the reference position and the estimated error (one sigma) by EKF (dashed line).

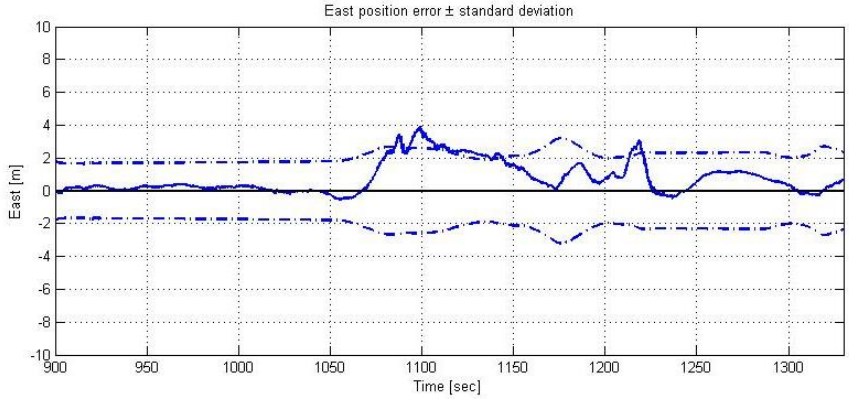

Figure 12. Position drift of the image-aided INS in east direction compared to the reference position and the estimated error (one sigma) by EKF (dashed line).

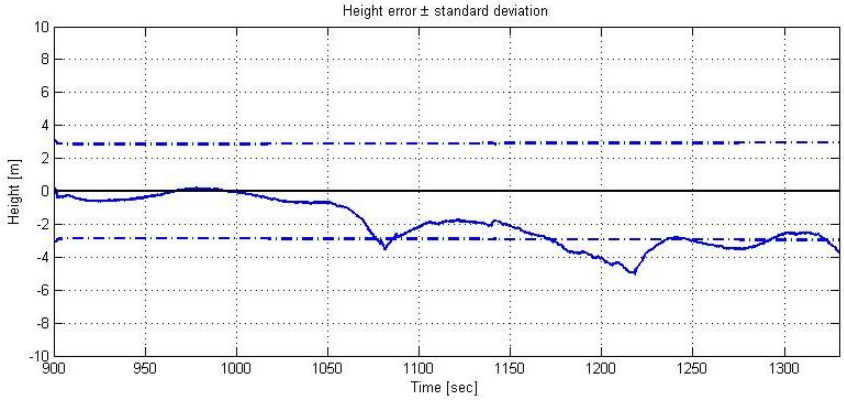

Figure 13. Altitude drift of the image-aided INS compared to the reference altitude and the estimated error (one sigma) by EKF (dashed line)

The position drift in all three directions, north, east and altitude, is within 5 m in 430 seconds. Comparing the position drift of free-inertial with position drift of image-aided INS, the image-aided INS reduces the drift drastically. The position drift in north and east direction is reduced from 10000 m to 4 m, and in altitude from 1000 m to 5 m. The drift reduction clearly demonstrates the performance of the proposed image-aided INS.

The position drift can be analyzed further by looking the root-mean-square (RMS) of the position drift in the north-east directions (horizontal direction) as shown in Figure 14 with the corresponding RMS of the standard deviations by the EKF.

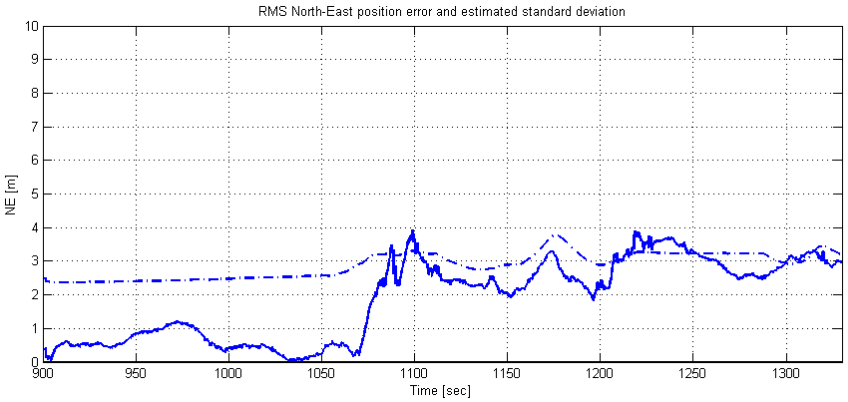

Figure 14. RMS of the position drift in horizontal direction of the image-aided INS compared to the reference position and the corresponding RMS of the standard deviations estimated by the EKF (dashed line).

Figure 14 shows that the error increased within 1070-1100 seconds and stabilized afterwards within the estimated standard deviation by the EKF. The comparison of the drift with the estimated standard deviation shows that the drift in all directions, north, east, altitude and north-east, is insignificant until 1070 seconds, and slightly over the standard deviation after that. The video data shows that the Octocopter is hovering and in slow motion until 1070 seconds, and then begin to move faster. The IMU data shows that the Octocopter made a sharp maneuver at 1070 seconds and a couple of times later in the run.

There could be a number of direct and indirect reasons for the error to increase. The image data under sharp maneuvers does not contain any tracked image-feature-points for aiding the INS, because of a total shift in the camera's field-of-view. Thus, the image-aided INS is in a free-inertial state, which causes the error to increase rapidly. This probably explains some of the variations in error throughout the flight.

The consistent drift from 1070-1100 is, however, probably due to a different reason, and we have not concluded on this yet. One hypothesis is that the EKF becomes overconfident in its estimation of the inverse depth of the image-feature-points when the Octocopter is hovering, which first leads to a bias in the estimate of the landmark positions. When it starts moving, this bias will be transformed into the EKF estimate of the velocity as a scale factor error, which again leads to drift in the position error.

The problem of losing image-feature-point tracks due to sharp maneuvering could be solved by having a rig of 3 cameras with overlapping fields of view in horizontal directions, or the aircraft could be maneuvered more gently. If the other drift is caused by overconfidence in the EKF, this could be addressed by improved modelling, possibly using a higher order filter, if the bias is caused by nonlinearities, or by ensuring that the epipolar constraint is also satisfied by the EKF measurements update.

7. SUMMARY

The proposed image-aided inertial navigation system which integrates inertial sensor data with position data of image-feature-points was tested with a realistic data set from a UAV flight with an octocopter. A rig of 3 cameras with overlapping fields-of-views in vertical direction and an onboard data collection system were used. Only the images from the upper camera were used in this experiment.

The comparison of the trajectories of the reference data with the proposed image-aided INS shows that the image-aided trajectory closely follows the reference trajectory, especially the small curve trajectory at the beginning. The image-aided INS reduced the drift drastically compared to the position drift of the free-inertial. The drift is insignificant, and well within the estimated standard deviation, for a time interval from the start when the aircraft is hovering in slow motion. After the motion begins there is a lager relative drift, but the overall drift reduction demonstrates the ability and the performance of the proposed system.

Sharp maneuvers of an aircraft are quite normal in UAV operations. These kinds of maneuvers lead the image-aided INS to a free-inertial state and increase the drift. It may be handled by having a rig of multiple cameras in horizontal and/or vertical directions with overlapping field-of-views. We have also proposed that the overconfidence in the EKF is a possible explanation for the consistent drift for a short interval of the flight and suggested various improvements in the INS integration to address this.

REFERENCES

[1] Grinaker, S., "Application of Image Analysis in Precision Guided Weapons," Proc. Of Guidance and Control of Precision Guided Weapons, Advisory Group for Aerospace Research & Development, NATO, Geilo Norway, (1988).

[2] Hagen, P.E., "BASIS Image based Navigation – method, implementation and results" (in Norwegian), FFI-Report 92/4008, (1992).

[3] Hagen, P.E. and Heyerdahl, E., "Navigation by Images", in Modeling, Identification and Control, vol. 14, no. 3, pp. 133-143, (1993).

[4] Hagen, P.E. and Heyerdahl, E., "Navigation by Optical Flow", in Pattern Recognition, vol. I, Conference A: Computer Vision and Application, Proceedings, 11th IAPR International Conf. on Pattern Recognition, pp. 700-703, (1992).

[5] Heyerdahl, E., "Circular Integrals – Simple, Noise robust, Translation and Rotation invariant Features" (in Norwegian), Proc. NOBIM, Skedsmo, Norway, pp. 43-50, (1991).

[6] Hafskjold, B.H., Jalving, B., Hagen, P.E., and Gade, K., "Integrated Camera-Based Navigation", Journal of Navigation, vol. 53, pp. 237-345, (2000).

[7] Sivalingam, B., "Robustness of Image token Retrieval based on Circular Integral", FFI-Report 2010/00339, (2010).

[8] Sivalingam, B., "Performance and Complexity analysis of Image token Retrieval based on Circular integral and Template matching", FFI-Report 2011/00124, (2011).

[9] Sivalingam, B. and Hagen, O.K., "Image-aided Inertial Navigation System", FFI-Report 2012/02019, (2012).

[10] Sivalingam, B. and Hagen, O:K., "Image-aided Inertial Navigation System based on Image Tokens", SET-168 NATO Symposium on "Navigation Sensors and Systems in GNSS denied Environments", Imzir, Turkey, (2012).

[11] Sivalingam, B. and Hagen, O.K., "Feasibility study of Tokens based Image-aided Inertial Navigation in Nano UAV", FFI-Report 13/01377, (2013).

[12] Sivalingam, B. and Hagen, O.K., "Image-aided inertial navigation in Octocopter", FFI-Report 16/01985, (2016).

[13] Mourikis, A.I. and Roumeliotis, S.I., "A Multi-State constraint Kalman Filter for Vision-aided Inertial Navigation", Proc. of IEEE Int. Conf. on Robotics and Automation, Italy, pp. 3565-3572, April (2007).

[14] Jones, E.S. and Soatto, S., "Visual-inertial navigation, mapping and localization: A scalable real-time causal approach", Int. J. Robot. Res., 30, pp. 407-430, (2011).

[15] Li, M. and Mourikis, A.I., "High-precision, consistent EKF-based visual-inertial odometry", Int. J. Robot. Res.32, pp. 690-711, (2013).

[16] Hesch, J.A., Kottas, D.G., Bowman, S.L. and Roumeliotis, S.I., "Camera-IMU-based localization: Observability analysis and consistency improvement", Int. J. Robot. Res., 33, pp. 182-201, (2014).

[17] Chu, C.C., Lie, F.A.P., Lemay, L. and Gebre-Egziabher, D., "Performance Comparision of Tight and Loose INS-Camera Integration", in Proc. of the 24th Int. Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS 2011), San Diego, CA, USA, pp. 24-26, (2011).

[18] Civera, J., Davison, A.J. and Montiel, J.M.M, "Inverse Depth Parametrization for Monocular SLAM," IEEE Transactions on Robotics, vol. 24, no. 5, pp. 932-945, Oct. (2008).
[19] Huster, A. and Rock, S.M., “Relative position sensing by fusion monocular vision and inertial rate sensors”, Proc. of ICAR 2003, Coimbra, Portugal, (2003).
[20] George, M. and Sukkarieh, S., “Inertial Navigation Aided by Monocular Camera Observations of Unknown Features”, 2007 IEEE International Conf. on Robotics and Automation, Roma, Italy, April (2007).
[21] Lucas, D.B. and Kanda, T., “Narrow-band analyzer (Thesis),” Ph.D. dissertation, Dept. Elect. Eng., Harvard Univ., Cambridge, MA, (1993).
[22] Kawasaki, N., “An iterative image registration technique with an application of stereo vision”, Proc. Of 7th Int. Conf. on Artificial Intelligence, pp. 674-679, (1981).
[23] Kalal, Z., Mikolajczyk, K. and Matas, J., “Forward-backward error. Automatic detection of tracking features,” Proc. of 20th Int. Conf. on Pattern Recognition 2010, pp. 2756-2759, (2010).
[24] Gade, K., “NAVLAB, a Generic Simulation and Post-processing Tool for Navigation” European Journal of Navigation, vol. 2, no. 4, pp. 51–59, (2004).
[25] Lowe, D.G., “Distinctive image features from scale-invariant key points” in Int. Journal of Computer Vision, 60(2), pp. 91–110, (2004).
[26] Bay, H., Ess, A., Tuytelaars, T. and Van Gool, L., “Speeded-up robust features (SURF)” in Computer Vision and Image Understanding, 110, pp. 346–359, (2008).
[27] Rublee, E. et al., “Orb: An efficient alternative to SIFT or SURF,” IEEE Int. Conf. in Computer Vision (ICCV), pp. 2564–2571, (2011).
[28] Edward. R and Tom, D., “Machine learning for high-speed corner detection,” IEEE Int. Conf. on Comp. Vision, vol. 2, pp. 1508-1511, Oct. (2005).
[29] Harris, C. and Stephnes, M., “A combined corner and edge detector,” Alvey Vision Conf., pp. 147-151, (1988).
[30] Gonzalez, R.C. and Woods, R.E., “Digital Image Processing (3rd ed.), Addison-Wesley, Reading, Massachusetts, (1992).
[31] Titterton, D.H. and Weston, J.L.,”Strapdown Inertial Navigation Technology”, AIAA and IEE, ISBN 1-56347-693-2, (2004).
[32] Gelb, A., “Applied Optimal Estimation”, MIT Press, ISBN 0262570483, (1974).
[33] Hartly, R. and Zisserman, A., “Multiple View Geometry in Computer Vision”, Cambridge University Press, ISBN 0-521-62304-0, (2000).
[34] Martinelli, A., “Vision and IMU Data Fusion: Closed-Form Solutions for Attitude, Speed, Absolute Scale, and Bias Determination”, IEEE Transactions on Robotics, vol. 28, no. 1, pp. 44-60, Feb. (2012).