**Post-Print**

This is the accepted version of:

E. Asadi, C.L. Bottasso
*Delayed Fusion for Real-Time Vision-Aided Inertial Navigation*
Journal of Real-Time Image Processing, Vol. 10, N. 4, 2015, p. 633-646
doi:10.1007/s11554-013-0376-8

**Real-Time Image Processing manuscript No.**
(will be inserted by the editor)

**Ehsan Asadi** · **Carlo L. Bottasso**

# Delayed Fusion for Real-Time Vision-Aided Inertial Navigation

Received: date / Revised: date

**Abstract** In this paper we consider the effects of delay caused by real-time image acquisition and feature tracking in a previously documented Vision-Augmented Inertial Navigation System. At first, the paper illustrates how delay caused by image processing, if not explicitly taken into account, can lead to appreciable performance degradation of the estimator. Next, three different existing methods of delayed fusion and a novel combined one are considered and compared. Simulations and Monte Carlo analyses are used to assess the estimation errors and computational effort of the various methods. Finally, a best performing formulation is identified, that properly handles the fusion of delayed measurements in the estimator without increasing the time burden of the filter.

**Keywords** Delayed fusion · Vision-Aided Inertial Navigation System · Larsen method · Delayed state EKF · Recalculation

## 1 Introduction

Navigation approaches often use vision systems, since these are among the most information-rich sensors for autonomous positioning and mapping purposes [4]. Vision-based navigation systems have been in use in numerous applications such as Autonomous Ground Vehicles (AGV) and underwater environments [11]. Recently, they have been gaining increased attention also in the field of Unmanned Aerial Vehicles (UAV) [19,30]. Vision systems provide long range, high resolution measurements with low power consumption and limited cost. On the other hand, they are usually associated with rather low sample rates, since they often require complex processing of the acquired images, and this limits and hinders their usability in fast and real-time applications [9].

Several attempts have already been documented in the design and implementation of robust visual odometry systems [21,13]. Some authors have proposed the incorporation of inertial measurements as model inputs [25] or states [23,32,20], using variants of the Kalman filtering approach to robustly estimate the vehicle motion. Other authors have used an entropy-like cost function or bundle adjustment [10, 28]. The Vision-Aided Inertial Navigation System (VA-INS) of [5] combined in a synergistic way vision-based sensors together with classical inertial navigation ones. The method made use of an Extended Kalman Filter (EKF), assuming that all measurements were available with no delay.

However, latency due to the extraction of information from images in real-time applications is one of the factors affecting accuracy and robustness of vision-based navigation systems [16,12]. In fact, visual observations are generated with delay, since image processing procedures required for tracking features between stereo images and across time steps are time consuming tasks. Because of this, real-time performance is often achieved at the expense of a reduced computational complexity and number of features [9]. If delays are small or the estimation is performed off-line, then the use of a classic filtering leads to acceptable results. Otherwise, the quality of the estimates is affected by the magnitude of the delay. Consequently, it becomes important to understand how to account for such delay in a consistent manner, without at the same time excessively increasing the computational burden of the filter.

Measurement delay has been the subject of numerous investigations in the context of systems requiring long time visual processing [22]. If the delay is rather small, a simple solution is to ignore it, but this implies that the estimates are not optimal and their quality may be affected. Another straightforward method to handle delay is to completely recalculate the filter during the delay period as measurements arrive. Usually this method cannot be used in practical ap-

E. Asadi
Department of Aerospace Science and Technology,
Politecnico di Milano, Milano, Italy
E-mail: ehsan.asadi@polimi.it

C.L. Bottasso(✉)
Department of Aerospace Science and Technology,
Politecnico di Milano, Milano, Italy, and
Wind Energy Institute, Technische Universität München,
Garching bei München, Germany
E-mail: carlo.bottasso@tum.de

plications because of its large storage cost and computational burden. Other documented methods fuse delayed measurements as they arrive through modified Kalman filters [1, 18]. These methods are effectively implemented in tracking and navigation systems for handling delays associated with the Global Positioning System (GPS). Some solutions exploit state augmentation [8,3]. The fixed-lag smoothing method [7,3] augments the state vector with all previous states throughout the interval from measurement sampling to delayed fusion; the main drawback of this approach is a possibly high computational load. Delayed state Kalman filtering [31], also known as stochastic cloning [24], is an another solution based on the augmentation of the state vector with the state at the lagged time, when measurements are sampled. The approach has been implemented in many applications in chemical and biochemical processes [14,29] and for solving localization problems [27] to optimally incorporate the delayed measurements with non-delayed ones.

The aim of this paper is to present a comprehensive study on delayed fusion approaches in a real-time tightly coupled VA-INS [5]. Tracked feature points are incorporated as delayed measurements in a multi-rate multi-sensor data fusion process using a non-linear estimator. More specifically, the paper:

- Analyzes the effects of delay caused by image processing on state estimation, when such delay is not explicitly accounted for in the estimator;
- Considers problem issues and assesses the performance of delayed state EKF alongside two other existing delayed fusion methods (Recalculation and Larsen [18]) to incorporate delayed vision-based measurements in the estimator;
- Considers improvements on the estimator performance through a combination of delayed state EKF and Larsen method, this way replacing an approximate vision-based model by an exact one;
- Assesses the quality of the various formulations and identifies the most promising one, in terms of computational burden of the filter and of the quality of its estimates, using simulation experiments and Monte Carlo analysis.

Some preliminary results on delay analysis using Recalculation and Larsen methods were previously presented in [2].

## 2 Vision-Augmented Inertial Navigation

Bottasso and Leonello (2009) proposed a VA-INS to achieve higher precision in the estimation of the vehicle motion. Their implementation used low-cost small-size stereo cameras, that can be mounted on-board small Rotorcraft Unmanned Aerial Vehicles (RUAVs). In this approach, the sensor readings of a standard inertial measurement unit (a triaxial accelerometer and gyro, a triaxial magnetometer, a GPS and a sonar altimeter) are fused within an EKF together with the outputs of so-called vision-based motion sensors. The overall architecture of the system is briefly reviewed here.

### 2.1 Kinematics

The sensor configuration and reference frames used in the kinematic modeling of the system are depicted in Fig. 1.

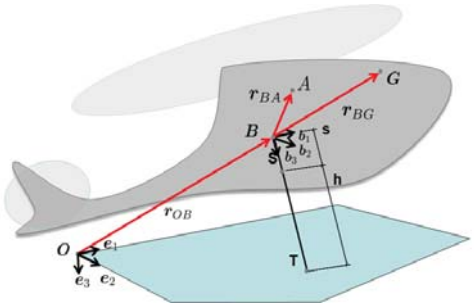



Fig. 1: Reference frames and location of sensors.

The inertial frame of reference is centered at point $O$ and denoted by a triad of unit vectors $\mathcal{E} \doteq (\boldsymbol{e}_1, \boldsymbol{e}_2, \boldsymbol{e}_3)$, pointing North, East and down (NED navigational system). A body-attached frame has origin in the generic material point $B$ of the vehicle and has a triad of unit vectors $\mathcal{B} \doteq (\boldsymbol{b}_1, \boldsymbol{b}_2, \boldsymbol{b}_3)$.

The components of the acceleration in the body-attached frame are sensed by an accelerometer located at point $A$ on the vehicle. The accelerometer yields a reading $\boldsymbol{a}_{\text{acc}}$ affected by noise $\boldsymbol{n}_{\text{acc}}$:

$$\boldsymbol{a}_{\text{acc}} = \boldsymbol{g}^{\mathcal{B}} - \boldsymbol{a}_A^{\mathcal{B}} + \boldsymbol{n}_{\text{acc}}. \tag{1}$$

In this expression, $\boldsymbol{g}^{\mathcal{B}}$ indicates the body-attached components of the acceleration of gravity, where $\boldsymbol{g}^{\mathcal{B}} = \boldsymbol{R}^T \boldsymbol{g}^{\mathcal{E}}$ with $\boldsymbol{g}^{\mathcal{E}} = (0,0,g)^T$, while $\boldsymbol{R} = \boldsymbol{R}(\boldsymbol{q})$ are the components of the rotation tensor that brings triad $\mathcal{E}$ into triad $\mathcal{B}$ and $\boldsymbol{q}$ are rotation parameters.

Gyroscopes measure the body-attached components of the angular velocity vector, yielding a reading $\boldsymbol{\omega}_{\text{gyro}}$ affected by a noise disturbance $\boldsymbol{n}_{\text{gyro}}$:

$$\boldsymbol{\omega}_{\text{gyro}} = \boldsymbol{\omega}^{\mathcal{B}} + \boldsymbol{n}_{\text{gyro}}. \tag{2}$$

The kinematic equations, describing the motion of the body-attached reference frame with respect to the inertial one, can be written as

$$\dot{\boldsymbol{v}}^{\mathcal{E}} = \boldsymbol{g}^{\mathcal{E}} - \boldsymbol{R}(\boldsymbol{a}_{\text{acc}} + \boldsymbol{\omega}^{\mathcal{B}} \times \boldsymbol{\omega}^{\mathcal{B}} \times \boldsymbol{r}_{BA}^{\mathcal{B}} + \boldsymbol{\alpha}^{\mathcal{B}} \times \boldsymbol{r}_{BA}^{\mathcal{B}}) + \boldsymbol{R}\boldsymbol{n}_{\text{acc}}, \tag{3a}$$

$$\dot{\boldsymbol{\omega}}^{\mathcal{B}} = \boldsymbol{\alpha}_h(\boldsymbol{\omega}_{\text{gyro}}, \boldsymbol{n}_{\text{gyro}}), \tag{3b}$$

$$\dot{\boldsymbol{r}}^{\mathcal{E}} = \boldsymbol{v}^{\mathcal{E}}, \tag{3c}$$

$$\dot{\boldsymbol{q}} = \boldsymbol{T}(\boldsymbol{\omega}^{\mathcal{B}})\boldsymbol{q}, \tag{3d}$$

where $\boldsymbol{v} = \boldsymbol{v}_B$ is the velocity of point $B$, $\boldsymbol{\omega}$ is the angular velocity and $\boldsymbol{\alpha}$ the angular acceleration, while $\boldsymbol{r}_{BA}$ is the position vector from point $B$ to point $A$ and $\boldsymbol{r} = \boldsymbol{r}_{OB}$ is from point $O$ to point $B$. Finally, using quaternions for the rotation parameterization, matrix $\boldsymbol{T}$ can be written as

$$\boldsymbol{T}(\boldsymbol{\omega}^{\mathcal{B}}) = \frac{1}{2}\begin{bmatrix} 0 & -\boldsymbol{\omega}^{\mathcal{B}^T} \\ \boldsymbol{\omega}^{\mathcal{B}} & -\boldsymbol{\omega}_{\times}^{\mathcal{B}} \end{bmatrix}. \tag{4}$$

Gyro measurements are used in Eq. (3b) for computing an estimate of the angular acceleration. Since this implies a differentiation of the gyro measurements, assuming a constant (or slowly varying) bias over the differentiation interval, knowledge of the bias becomes unnecessary. Hence, the angular acceleration is computed as

$$\boldsymbol{\alpha}^{\mathcal{B}} \simeq \boldsymbol{\alpha}_h(\boldsymbol{\omega}_{\text{gyro}}), \tag{5}$$

where $\boldsymbol{\alpha}_h$ is a discrete differentiation operator. The angular acceleration at time $t_k$ is computed according to the following three-point stencil formula based on a parabolic interpolation

$$\boldsymbol{\alpha}_h(t_k) = \big(3\boldsymbol{\omega}_{\text{gyro}}(t_k) - 4\boldsymbol{\omega}_{\text{gyro}}(t_{k-1}) + \boldsymbol{\omega}_{\text{gyro}}(t_{k-2})\big)/(2h), \tag{6}$$

where $h = t_k - t_{k-1} = t_{k-1} - t_{k-2}$. It is assumed that gyro outputs are available with a sufficiently high rate (e.g., 400 Hz), relative to the frequency of the inertial navigation system (e.g., 100 Hz), in order to avoid feeding the estimator with correlated measurements.

A GPS is located at point $G$ on the vehicle (see Fig. 1). The velocity and position vectors of point $G$, noted respectively $\boldsymbol{v}_G^{\mathcal{E}}$ and $\boldsymbol{r}_{OG}^{\mathcal{E}}$, can be expressed as

$$\boldsymbol{v}_G^{\mathcal{E}} = \boldsymbol{v}^{\mathcal{E}} + \boldsymbol{R}\boldsymbol{\omega}^{\mathcal{B}} \times \boldsymbol{r}_{BG}^{\mathcal{B}}, \tag{7a}$$

$$\boldsymbol{r}_{OG}^{\mathcal{E}} = \boldsymbol{r}^{\mathcal{E}} + \boldsymbol{R}\boldsymbol{r}_{BG}^{\mathcal{B}}. \tag{7b}$$

The GPS yields measurements of the position and velocity of point $G$ affected by noise, i.e.

$$\boldsymbol{v}_{\text{gps}} = \boldsymbol{v}_G^{\mathcal{E}} + \boldsymbol{n}_{v_{\text{gps}}}, \tag{8a}$$

$$\boldsymbol{r}_{\text{gps}} = \boldsymbol{r}_{OG}^{\mathcal{E}} + \boldsymbol{n}_{r_{\text{gps}}}. \tag{8b}$$

A sonar altimeter measures the distance $h$ along the body-attached vector $\boldsymbol{b}_3$, between its location at point $S$ and point $T$ on the terrain (assumed to be flat), with $\boldsymbol{r}_{OS}^{\mathcal{B}} = (0,0,s)^T$. In the body-attached frame $\mathcal{B}$, the distance vector between $S$ and $T$ has components $\boldsymbol{r}_{ST}^{\mathcal{B}} = (0,0,h)^T$, which are readily transformed into inertial components as $\boldsymbol{r}_{ST}^{\mathcal{E}} = \boldsymbol{R}\boldsymbol{r}_{ST}^{\mathcal{B}}$. Hence, we get

$$h = r_3/R_{33} - s, \tag{9}$$

where $r_3 = \boldsymbol{r}^{\mathcal{E}} \cdot \boldsymbol{e}_3$ and $\boldsymbol{R} = [R_{ij}]$, $i,j = 1,2,3$. The sonar altimeter yields a reading $h_{\text{sonar}}$ affected by noise $n_{\text{sonar}}$, i.e.

$$h_{\text{sonar}} = h + n_{\text{sonar}}. \tag{10}$$

Furthermore, we consider a magnetometer sensing the magnetic field $\boldsymbol{m}$ of the Earth in the body-attached system $\mathcal{B}$, so that

$$\boldsymbol{m}^{\mathcal{B}} = \boldsymbol{R}^T \boldsymbol{m}^{\mathcal{E}}, \tag{11}$$

where the inertial components $\boldsymbol{m}^{\mathcal{E}}$ are assumed to be known and constant in the area of operation of the vehicle. The magnetometer yields a measurement $\boldsymbol{m}_{\text{magn}}$ affected by noise $\boldsymbol{n}_{\text{magn}}$, i.e.

$$\boldsymbol{m}_{\text{magn}} = \boldsymbol{m}^{\mathcal{B}} + \boldsymbol{n}_{\text{magn}}. \tag{12}$$

Finally, considering a pair of stereo cameras located on the vehicle (see Fig. 2), a triad of unit vectors $\mathcal{C} \doteq (\boldsymbol{c}_1, \boldsymbol{c}_2, \boldsymbol{c}_3)$ has origin at the optical center $C$ of the left camera, where $\boldsymbol{c}_1$ is directed along the horizontal scanlines of the image plane, while $\boldsymbol{c}_3$ is parallel to the optical axis, pointing towards the scene. The right camera has its optical axis and scanlines parallel to those of the left camera, i.e. $\mathcal{C}' \equiv \mathcal{C}$, where we use the symbol $()'$ to indicate quantities of the right camera. Considering that $P$ is a fixed point, the vision-based observation model, discretized across two time instants $t_{k-m}$ and $t_k = t_{k-m} + \Delta t$, is

$$\begin{aligned}\boldsymbol{d}(t_k)^{\mathcal{C}_k} = &-\Delta t\, \boldsymbol{C}^T \big(\boldsymbol{R}(t_k)^T \boldsymbol{v}^{\mathcal{E}}(t_k) \\ &+ \boldsymbol{\omega}^{\mathcal{B}}(t_k) \times (\boldsymbol{c}^{\mathcal{B}} + \boldsymbol{C}\boldsymbol{d}(t_{k-m})^{\mathcal{C}_{k-m}})\big) + \boldsymbol{d}(t_{k-m})^{\mathcal{C}_{k-m}},\end{aligned} \tag{13}$$

where $m$ is the number of time samples between two consecutive captured images. $\boldsymbol{C}$ are the components of the rotation tensor that brings triad $\mathcal{B}$ into triad $\mathcal{C}$. The tracked feature point distances are noted $\boldsymbol{d}^{\mathcal{C}} = (d_1, d_2, d_3)^T$ for the left camera, and are obtained by stereo reconstruction using

$$\boldsymbol{d}^{\mathcal{C}} = \frac{b}{d}\boldsymbol{p}^{\mathcal{C}}, \tag{14}$$

where $\boldsymbol{p} = (p_1, p_2, f)^T$ is the position vector of the feature point on the image plane, $b$ is the stereo baseline and $d = p_1 - p_1{}'$ the disparity. This process yields at each time step $t_k$ an estimate $\boldsymbol{d}_{\text{vsn}}$ affected by noise $\boldsymbol{n}_{\text{vsn}}$

$$\boldsymbol{d}_{\text{vsn}} = \boldsymbol{d}(t_k)^{\mathcal{C}_k} + \boldsymbol{n}_{\text{vsn}}. \tag{15}$$

An estimate of the accuracy can be derived by considering the influence of the disparity measure error on the computed feature position. For example, differentiating the third component of Eq. (14) we get

$$\delta d_3 = \pm \frac{fb}{\omega \tilde{d}^2}\, \delta\tilde{d}, \tag{16}$$

where we have set $d = \tilde{d}w$, being $\tilde{d}$ the disparity in pixel units and $w$ the pixel width. From this equation, it is clear

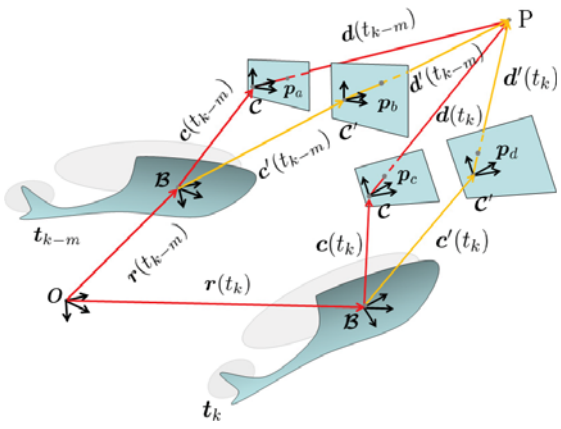



Fig. 2: Geometry for the derivation of the discrete vision-based motion sensor.

that the accuracy of a 3D measurement is directly affected by disparity error. Furthermore, accuracy is adversely affected by distance, as far points are associated with lower disparity values.

The approximate observation model of Eq. (13) uses (average) translational and rotational velocities during the measurement acquisition time interval; this way, measurements can be expressed directly via the current state of the vehicle. Therefore, this approximate model can be processed within a standard EKF.

### 2.2 Process Model and Observations

The estimator is based on the following state-space model

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}\big(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{\nu}(t)\big), \tag{17a}$$
$$\boldsymbol{y}(t_k) = \boldsymbol{h}\big(\boldsymbol{x}(t_k)\big), \tag{17b}$$
$$\boldsymbol{z}(t_k) = \boldsymbol{y}(t_k) + \boldsymbol{\mu}(t_k), \tag{17c}$$

where the state vector $\boldsymbol{x}$ is defined as

$$\boldsymbol{x} \doteq (\boldsymbol{v}^{\mathcal{E}^T}, \boldsymbol{\omega}^{\mathcal{B}^T}, \boldsymbol{r}^{\mathcal{E}^T}, \boldsymbol{q}^T)^T. \tag{18}$$

Function $\boldsymbol{f}(\cdot,\cdot,\cdot)$ in Eq. (17a) represents in compact form the rigid body kinematics expressed by Eqs. (3). The input vector $\boldsymbol{u}$ appearing in Eq. (17a) is defined in terms of the measurements provided by the accelerometers and gyros, i.e. $\boldsymbol{u} = (\boldsymbol{a}^T_{\text{acc}}, \boldsymbol{\omega}^T_{\text{gyro}})^T$, and $\boldsymbol{\nu}$ is the associated measurement noise vector.

Similarly, Eqs. (7), (9), (11) and (13) may be gathered together and written in compact form as an observation model $\boldsymbol{h}(\cdot)$ expressed by Eqs. (17b), where the vector of outputs $\boldsymbol{y}$ is defined as

$$\boldsymbol{y} = (\boldsymbol{v}_G^{\mathcal{E}^T}, \boldsymbol{r}_{OG}^{\mathcal{E}^T}, h, \boldsymbol{m}^{\mathcal{B}^T}, \ldots, \boldsymbol{d}^{\mathcal{C}^T}, \ldots)^T. \tag{19}$$

The definition of model (17) is complemented by the vector of measurements $\boldsymbol{z}$ and associated noise $\boldsymbol{\mu}$ vectors

$$\boldsymbol{z} \doteq (\boldsymbol{v}^T_{\text{gps}}, \boldsymbol{r}^T_{\text{gps}}, h_{\text{sonar}}, \boldsymbol{m}^T_{\text{magn}}, \ldots, \boldsymbol{d}^T_{\text{vsn}}, \ldots)^T, \tag{20a}$$
$$\boldsymbol{\mu} \doteq (\boldsymbol{n}^T_{v_{\text{gps}}}, \boldsymbol{n}^T_{r_{\text{gps}}}, n_{\text{sonar}}, \boldsymbol{n}^T_{\text{magn}}, \ldots, \boldsymbol{n}^T_{\text{vsn}}, \ldots)^T. \tag{20b}$$

### 2.3 Classic State Estimation using EKF

The state estimation problem expressed by Eqs. (17–20) was solved using the EKF approach, initially assuming that all measurements are available with no delay. The EKF formulation is briefly reviewed here using the time-discrete form of Eqs. (17) and assuming $\boldsymbol{\nu}$ and $\boldsymbol{\mu}$ to be white noise processes with covariance $\boldsymbol{Q}$ and $\boldsymbol{U}$, respectively. The prediction stage of states and observations is performed by using the non-linear model equations,

$$\hat{\boldsymbol{x}}^-_k = \hat{\boldsymbol{x}}_{k-1} + \boldsymbol{f}(\hat{\boldsymbol{x}}_{k-1}, \boldsymbol{u}_{k-1}, 0)\,\Delta t, \tag{21a}$$
$$\boldsymbol{y}_k = \boldsymbol{h}\big(\hat{\boldsymbol{x}}^-_k\big), \tag{21b}$$

whereas a linear approximation is used for estimating the error covariance and computing the Kalman gain matrices,

$$\boldsymbol{P}^-_k = \boldsymbol{A}_{k-1}\,\boldsymbol{P}_{k-1}\,\boldsymbol{A}_{k-1}{}^T + \boldsymbol{G}_{k-1}\,\boldsymbol{Q}_{k-1}\,\boldsymbol{G}_{k-1}{}^T, \tag{22a}$$
$$\boldsymbol{K}_k = \boldsymbol{P}^-_k\,\boldsymbol{H}_k{}^T\,(\boldsymbol{H}_k\,\boldsymbol{P}^-_k\,\boldsymbol{H}_k{}^T + \boldsymbol{U}_k)^{-1}. \tag{22b}$$

Matrices $\boldsymbol{A}_{k-1}$, $\boldsymbol{G}_{k-1}$ and $\boldsymbol{H}_k$ are computed by linearizing the non-linear model about the current estimate,

$$\boldsymbol{A}_{k-1} = \boldsymbol{I} + \Delta t \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}\Big|_{\hat{\boldsymbol{x}}_{k-1}}, \quad \boldsymbol{G}_{k-1} = \Delta t \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{\nu}}\Big|_{\hat{\boldsymbol{x}}_{k-1}}, \quad \boldsymbol{H}_k = \frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}}\Big|_{\hat{\boldsymbol{x}}^-_k}. \tag{23}$$

Finally, covariance updates and state estimates are computed as

$$\boldsymbol{P}_k = (\boldsymbol{I} - \boldsymbol{K}_k\boldsymbol{H}_k)\,\boldsymbol{P}^-_k, \tag{24a}$$
$$\hat{\boldsymbol{x}}_k = \hat{\boldsymbol{x}}^-_k + \boldsymbol{K}_k\big(\boldsymbol{z}_k - \boldsymbol{h}\big(\hat{\boldsymbol{x}}^-_k\big)\big). \tag{24b}$$

As the estimator operates at a rather high rate and hence with small time increments, the unit norm quaternion constraint was simply realized by renormalization at each prediction-correction step.

### 2.4 Image Processing and Tracking

The idea of VA-INS is based on tracking scene points between stereo images and across time steps, to express the apparent motion of the tracked points in terms of the motion of the vehicle. The vision system is designed as a multi-purpose element providing a rather large number of tracked features and a dense disparity map to support several tasks including VA-INS, 3D mapping and obstacle avoidance.

The identification and matching of feature points is begun with the acquisition of the images; then, strong corners are extracted from the left image with the feature extractor of the KLT tracker [15], and a dense disparity map is obtained. Identified feature points are encoded using the BRIEF descriptor [6]; subsequently, matches in a transformed image are found by computing distances between descriptors. According to [6], this descriptor is competitive with algorithms like SURF in terms of recognition rate; although it appears to be better for small rotations than for large ones, it is much faster in terms of generation and matching.

A real-time implementation of the system was based on an on-board PC-104 with a 1.6 GHz CPU and 512 Mb of volatile memory, with the purpose of analyzing the performance and computational effort of the feature tracking process. Images were captured by a Point Grey Bumblebee XB3 stereo vision camera, and resized images with a resolution of 640x480 were used for tracking 100 points between frames. These tests indicated the presence of a 490 millisecond latency between the instant animage is captured and the time the state estimator receives the required visual information, as shown in table 1, which reports the worst case encountered in all experiments.

## 3 Delayed Fusion in VA-INS.

Simulation analyses, presented later, show that the half a second delay of the system is significant enough not to be neglected. In other words, directly feeding this delayed vision-based measurements to the EKF estimator will affect the quality of the estimates. Since the magnitude of delays associated with the tracking system is rather high in comparison to the delays of the other sensors, these are assumed to be delay-free in this work to simplify the problem. It is however clear that the same methodology used here for treating vision-caused delays could be used for addressing delays generated by other components of the system.

The outputs of the vision-based motion sensors $\boldsymbol{d}_{vsn}(s)$ from a captured image at time $s$ will only be available at time $k = s + N$, where $N$ is the sample delay. Delayed outputs are labeled $\boldsymbol{d}^*_{vsn}(k)$. On the other hand, measurements from other sensors are not affected by such a delay, and are available at each sampling time. For the purpose of handling multi-rate estimation and delay, observations are here partitioned in two groups, one collecting multi-rate non-delayed GPS, sonar and magnetometer readings (labeled rt, for real-time), and the other collecting delayed vision-based observations (labeled vsn, for vision):

$$\boldsymbol{z}^{rt} \doteq (\boldsymbol{v}^T_{gps}, \boldsymbol{r}^T_{gps}, \boldsymbol{h}_{sonar}, \boldsymbol{m}^T_{magn})^T, \tag{25a}$$

$$\boldsymbol{z}^{vsn^*} \doteq (\boldsymbol{d}^{*T}_{vsn(1)}, \boldsymbol{d}^{*T}_{vsn(2)}, \ldots, \boldsymbol{d}^{*T}_{vsn(n)})^T. \tag{25b}$$

The state estimation process is based on using a proper EKF update for each group. The Recalculation, Larsen [18] and delayed state EKF [31] methods are surveyed here for fusing delayed tracked points in the VA-INS structure as they arrive. All methods are briefly reviewed in the following.

### 3.1 Recalculation Method

A straightforward estimate can be obtained simply by recalculating the filter throughout the delay period. As the vision-based measurements are not available in the time interval between $s$ to $k$, one may update states and covariance using only non-delayed measurements in this time interval. As soon as vision measurements originally captured at time $s$ are received with delay at time $k$, the estimation procedure begins from $s$ by repeating the update while incorporating both non-delayed measurements and lagged vision-based measurements.

Table 1: Time cost of image processing tasks.

| Process Task | Computing Time |
|---|---|
| Image acquisition | 100 ms |
| Resizing, rectification | 40 ms |
| Dense disparity mapping | 150 ms |
| Feature extraction | 130 ms |
| Feature description | 50 ms |
| Feature matching | 20 ms |
| TOTAL | 490 ms |

The computational burden of this implementation of the filter in the VA-INS is critical, because of the need of fusing a fairly large set of measurements. Therefore the approach, although rigorous and straightforward, is not a good candidate for the implementation on-board small size aerial vehicles.

### 3.2 Larsen Method

Within the VA-INS approach, the successive tracked points, their uncertainty and consequently the measurement model will be unknown until images are completely processed. Therefore, a method is needed that does not require information about $\boldsymbol{z}^{vsn^*}_k$ until new measurements arrive.

Larsen extended Alexander approach [1], by computing an optimal gain by extrapolating lagged measurements to the present ones [18]:

$$\boldsymbol{z}^{vsn(int)}_k = \boldsymbol{z}^{vsn^*}_k + \boldsymbol{H}^{vsn^*}_k \hat{\boldsymbol{x}}_k - \boldsymbol{H}^{vsn^*}_s \hat{\boldsymbol{x}}_s. \tag{26}$$

This way, a correction term is calculated based on Kalman information, accounting for measurements delay and giving

$$\boldsymbol{M}_* = \prod_{i=0}^{N-1} \left(\boldsymbol{I} - \boldsymbol{K}^{rt}_{k-i} \boldsymbol{H}^{rt}_{k-i}\right) \boldsymbol{A}_{k-i-1}. \tag{27}$$

All updates to the Kalman gain and covariance due to the lagged measurements are delayed in Larsen method and take place as the delayed measurements arrive to the estimator. When lagged measurements become available, updates are performed in a simple and fast manner as

$$\boldsymbol{K}^{vsn}_k = \boldsymbol{M}_* \boldsymbol{P}_s \boldsymbol{H}^{vsn^{*T}}_s \left(\boldsymbol{H}^{vsn^*}_s \boldsymbol{P}_s \boldsymbol{H}^{vsn^{*T}}_s + \boldsymbol{U}^{vsn^*}_k\right)^{-1}, \tag{28a}$$

$$\delta \boldsymbol{P}_k = -\boldsymbol{K}^{vsn}_k \boldsymbol{H}^{vsn^*}_s \boldsymbol{P}_s \boldsymbol{M}^T_*, \tag{28b}$$

$$\delta \hat{\boldsymbol{x}}_k = \boldsymbol{M}_* \boldsymbol{K}^{vsn}_k \left(\boldsymbol{z}^{vsn^*}_k - \boldsymbol{H}^{vsn^*}_s \hat{\boldsymbol{x}}_s\right). \tag{28c}$$

The method can be utilized for handling either constant or time varying delays.

#### *3.2.1 Flow of EKF-Larsen Processing*

Figure 3 shows an overview of the measurement processing procedures for the standard EKF and Larsen methods. The image processing routines are started at time $s$, tracking feature points in new scenes; however, there is no available vision-based measurement until time $k = s+N$.

Meanwhile, the multi-rate real-time measurements $\boldsymbol{z}^{rt}_{s+i}, 1 \leq i \leq N$ are fused through the EKF Eqs. (22,24) as they arrive, using $\boldsymbol{H}^{rt}_{s+i}$. This will produce the Kalman gain $\boldsymbol{K}^{rt}_{s+i}$, state estimates $\hat{\boldsymbol{x}}^I_{s+i}$ and covariance $\boldsymbol{P}^I_{s+i}$. Implementing Larsen approach requires only the state vector and covariance error at time $s$ to be stored and the correction term $\boldsymbol{M}^{s+i}_*$ to be calculated during the delay period as

$$\boldsymbol{M}^{s+i}_* = (\boldsymbol{I} - \boldsymbol{K}^{rt}_{s+i} \boldsymbol{H}^{rt}_{s+i}) \boldsymbol{A}_{s+i-1} \boldsymbol{M}^{s+i-1}_*. \tag{29}$$

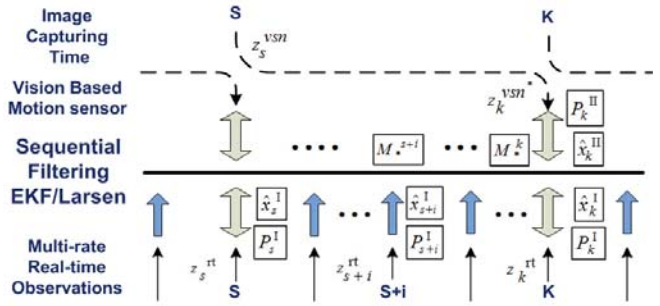



Fig. 3: Flow of sequential EKF-Larsen processing.

At time $k$, when the vision-based measurements become available, Larsen equations are used to incorporate delayed quantities $\boldsymbol{z}_k^{vsn^*}$ in the estimation procedure. The Kalman gain $\boldsymbol{K}_k^{vsn}$ is calculated using Eq. (28a). Finally, visual measurement corrections $\delta\boldsymbol{P}_k$ and $\delta\hat{\boldsymbol{x}}_k$, obtained by Eqs. (28b,28c), are added to the covariance matrix and state vector of real-time measurement updates $\boldsymbol{P}_k^I$ and $\hat{\boldsymbol{x}}_k^I$, to obtain new quantities $\boldsymbol{P}_k^{II}$ and $\hat{\boldsymbol{x}}_k^{II}$.

### 3.3 Delayed State Kalman Filtering

An alternative formulation to this problem is presented here by developing a specific delayed state Kalman filtering. In this approach, an augmented state vector provides the vehicle state at the time stereo images are captured, and it is used at the time delayed visual measurements are incorporated into the filter. The estimator is then based on an augmented state vector, indicated with the notation $\overline{\boldsymbol{x}}$, which extends the vehicle states with a copy of lagged ones. In order to prepare the cloned filter, the augmented state vector and augmented covariance matrix are reset right after the capture of a new pair of stereo images. At time step $s$, the augmented state vector is given by

$$\overline{\boldsymbol{x}}_s = (\boldsymbol{x}_s^{v^T}, \boldsymbol{x}_s^{l^T})^T, \tag{30}$$

where

$$\boldsymbol{x}_s^v \doteq \boldsymbol{x}_s = (\boldsymbol{v}^{\mathcal{E}^T}(t_s), \boldsymbol{\omega}^{\mathcal{B}^T}(t_s), \boldsymbol{r}^{\mathcal{E}^T}(t_s), \boldsymbol{q}^T(t_s))^T, \tag{31a}$$

$$\boldsymbol{x}_s^l \doteq (\boldsymbol{v}^{\mathcal{E}^T}(t_s), \boldsymbol{\omega}^{\mathcal{B}^T}(t_s), \boldsymbol{q}^T(t_s))^T, \tag{31b}$$

and the augmented covariance matrix is written as

$$\overline{\boldsymbol{P}}_s = \begin{bmatrix} \boldsymbol{P}_s^v & \boldsymbol{P}_s^{v,l} \\ \boldsymbol{P}_s^{v,l^T} & \boldsymbol{P}_s^l \end{bmatrix} = \boldsymbol{D}\,\boldsymbol{P}_s^v\,\boldsymbol{D}^T, \tag{32}$$

$$\boldsymbol{D} = \begin{bmatrix} & \mathrm{I}_{13} & \\ \mathrm{I}_6 & & 0_{6\times7} \\ 0_{4\times9} & & \mathrm{I}_4 \end{bmatrix}, \tag{33}$$

where $\boldsymbol{P}_s^{v,l}$ is the cross-correlation between the current vehicle state $\boldsymbol{x}_s^v$ and the lagged one $\boldsymbol{x}_s^l$, and $\boldsymbol{D}$ maps the covariance matrix to the corresponding augmented one at the same time step.

During the propagation steps, between the time of image capture $s$ to the time of delayed visual measurement fusion $k = s+N$, the first part of the augmented state, $\boldsymbol{x}_s^v$, is propagated forward by using new inertial measurements and the process model given by Eqs. (21a) and (22a), yielding $\boldsymbol{x}_k^{v-}$ and $\boldsymbol{P}_k^{v-}$. However, $\boldsymbol{x}_s^l$ and $\boldsymbol{P}_s^l$, which correspond to the second part of the augmented state, are kept frozen. Note that the cross-correlation term is evolved throughout the latency period. After $N$ steps of propagation, the augmented covariance matrix is given by

$$\overline{\boldsymbol{P}}_k^- = \begin{bmatrix} \boldsymbol{P}_k^{v-} & \mathcal{A}_{s|k}\boldsymbol{P}_s^{v,l} \\ \boldsymbol{P}_s^{v,l^T}\mathcal{A}_{s|k}^T & \boldsymbol{P}_s^l \end{bmatrix}, \tag{34}$$

where $\mathcal{A}_{s|k} = \prod_{i=0}^{N-1}\boldsymbol{A}_{k-i-1}$.

As the visual measurements $\boldsymbol{d}_{vsn}^*$ arrive to the estimator late, the visual output $\boldsymbol{d}(t_s)^{\mathcal{C}_s}$ and the visual observation model are rewritten according to the new augmented state vector,

$$\begin{aligned}\boldsymbol{d}(t_s)^{\mathcal{C}_s} = \boldsymbol{h}(\overline{\boldsymbol{x}}_k) = -\Delta t\,\boldsymbol{C}^T\big(\boldsymbol{R}(t_s)^T\boldsymbol{v}^{\mathcal{E}}(t_s) \\ + \boldsymbol{\omega}^{\mathcal{B}}(t_s)\times(\boldsymbol{c}^{\mathcal{B}} + \boldsymbol{C}\boldsymbol{d}(t_{s-m})^{\mathcal{C}_{s-m}})\big) + \boldsymbol{d}(t_{s-m})^{\mathcal{C}_{s-m}}.\end{aligned} \tag{35}$$

By linearizing the visual observation model about the augmented state estimate, we have

$$\overline{\boldsymbol{H}}_k^{vsn} = \frac{\partial \boldsymbol{h}}{\partial \overline{\boldsymbol{x}}_k} = [\boldsymbol{H}_k^{vsn}\ \boldsymbol{H}_s^{vsn}], \tag{36}$$

where $\boldsymbol{H}_k^{vsn}$ is equal to zero, according to Eq. (35).

The Kalman update is computed as

$$\overline{\boldsymbol{S}} = \overline{\boldsymbol{H}}_k^{vsn}\,\overline{\boldsymbol{P}}_k^-\,\overline{\boldsymbol{H}}_k^{vsn^T} + \overline{\boldsymbol{U}}_k^{vsn}, \tag{37a}$$

$$\overline{\boldsymbol{K}}_k = \overline{\boldsymbol{P}}_k^-\,\overline{\boldsymbol{H}}_k^{vsn^T}\,\overline{\boldsymbol{S}}^{-1} = \begin{bmatrix}\boldsymbol{K}_k^T & \boldsymbol{K}_s^T\end{bmatrix}^T. \tag{37b}$$

Finally, the current state of the vehicle is updated as

$$\hat{\boldsymbol{x}}_k^v = \hat{\boldsymbol{x}}_k^{v-} + \boldsymbol{K}_k\left(\boldsymbol{z}_k^{vsn^*} - \boldsymbol{H}_s^{vsn}\boldsymbol{x}_s^l\right), \tag{38a}$$

$$\boldsymbol{P}_k^v = \boldsymbol{P}_k^{v-} - \boldsymbol{K}_k\,\overline{\boldsymbol{S}}\,\boldsymbol{K}_k^T. \tag{38b}$$

The multi-rate real-time measurements $z_{s+i}^{rt}, 1 \le i \le N$ can be readily fused during the latency period as they arrive, using $\overline{\boldsymbol{H}}_{s+i}^{rt}$. However they must be fused with the current vehicle state while the lagged state remains frozen. A simple solution can be achieved through the use of the Schmidt-Kalman filter [26], whose gain is given as

$$\overline{\boldsymbol{K}}_{s+i}^{rt} = \boldsymbol{M}\,\overline{\boldsymbol{P}}_{s+i}^-\,\overline{\boldsymbol{H}}_{s+i}^{rt^T}\,\overline{\boldsymbol{S}}^{rt^{-1}}, \tag{39a}$$

$$\boldsymbol{M} = \begin{bmatrix} \mathrm{I}_{13} & 0_{13\times10} \\ 0_{10\times13} & 0_{10\times10} \end{bmatrix}, \tag{39b}$$

where $M$ indicates which states are updated.

### 3.4 Delayed Fusion in VA-INS via an Exact Model

In the previous section the motion velocity was assumed to be constant between steps, which is a simple and effective solution for sufficiently high image acquisition rates. An exact model that does not require such an hypothesis is given by the following discretization across two time instants $t_{k-m}$ and $t_k = t_{k-m} + \Delta t$:

$$\boldsymbol{d}(t_k)^{\mathcal{C}_k} = -\boldsymbol{C}^T \left( \boldsymbol{R}(t_k)^T (\boldsymbol{r}(t_k) - \boldsymbol{r}(t_{k-m})) + (\boldsymbol{I} - \boldsymbol{R}(t_k)^T \boldsymbol{R}(t_{k-m}))(\boldsymbol{c} + \boldsymbol{C}\boldsymbol{d}(t_{k-m})^{\mathcal{C}_{k-m}}) \right) + \boldsymbol{d}(t_{k-m})^{\mathcal{C}_{k-m}}. \tag{40}$$

This expression represents the position of a feature point w.r.t. the camera at time $t_k$ in terms of the position at time $t_{k-m}$ through the partial vehicle states (position and attitude) at times $t_k$ and $t_{k-m}$.

The problem of implementing this exact model in a standard estimator arises since the Kalman filter formulation requires measurements to be independent of previous filter states. To address this problem, while providing a robust estimation framework, a combination of delayed state Kalman filtering and Larsen method is proposed in this work and noted in the following as DS-EKF-Larsen. This way, by utilizing an augmented state vector that includes the current and lagged states, the exact model can be used, whereas delayed fusion is handled by Larsen method. Considering Eq. (40), the state vector is augmented with the lagged position and the lagged quaternion at the time the last measurements were obtained, yielding

$$\overline{\boldsymbol{x}}_{k-m} = (\boldsymbol{x}_{k-m}^{v^T}, \boldsymbol{x}_{k-m}^{l^T})^T, \tag{41a}$$

$$\boldsymbol{x}_{k-m}^{v} = (\boldsymbol{v}^{\mathcal{E}^T}(t_{k-m}), \boldsymbol{\omega}^{\mathcal{B}^T}(t_{k-m}), \boldsymbol{r}^{\mathcal{E}^T}(t_{k-m}), \boldsymbol{q}^T(t_{k-m}))^T, \tag{41b}$$

$$\boldsymbol{x}_{k-m}^{l} \doteq (\boldsymbol{r}^{\mathcal{E}^T}(t_{k-m}), \boldsymbol{q}^T(t_{k-m}))^T. \tag{41c}$$

### 3.5 Time Complexity

The measurement-update formulation of the standard EKF method involves the inversion of the residual covariance (innovation covariance) given by Eq. (22b), with an approximately cubic complexity of $O(N_z^{2.37})$, where $N_z$ is the dimension of the measurement vector. The computation also requires a matrix multiplication (see Eq. (24a)) of $O({N_x}^2)$, where $N_x$ is the dimension of the state vector $x^v$. In the VA-INS implementation of the filter, the most computationally expensive operation is the inversion of the residual covariance due to the fact that $N_z > N_x$, because of the incorporation of a rather large set of vision-based measurements.

In Larsen method, the computation of the correction term is accumulated throughout the delay period and it is applied at each update step using Eq. (28b). This way, the required inversion and matrix multiplication have a similar time complexity as in the EKF case. For the delayed state EKF within VA-INS, an increased cost in comparison to the EKF implementation is expected for the matrix multiplication, due to the increased state vector dimension, which is $N_x = N_{x^v} + N_{x^l}$.

## 4 Simulation Experiments

A Matlab/Simulink software environment was developed, that includes a flight mechanics model of a small Rotorcraft Unmanned Aerial Vehicle (RUAV), models of inertial navigation sensors, magnetometer, GPS and their noise models. The simulator is used in conjunction with the OGRE graphics engine [17], for rendering a virtual environment scene and simulating the image acquisition process. All sensor measurements are simulated (see table 2) as the helicopter flies in open loop at an altitude of 2 m following a rectangular path at a constant speed of 2 m/s within a small village, composed of houses and several other objects with realistic textures (see Fig. 4).

Navigation measurements are provided at a rate of 100 Hz, while stereo images at the rate of 2 Hz. The GPS, available at a rate of 1 Hz, is turned off after 10 s in the flight, to further highlight the effects of visual measurement delay. State estimates are obtained by six different data fusion processes: classic EKF with non-delayed measurements, classic EKF with delayed measurements, Recalculation, EKF-Larsen, delayed state EKF and DS-EKF-Larsen method in the presence of delay.

Table 2: Sensors and vibration noise parameters.

| Sensors | Noise Variance ($\sigma^2$) |
|---|---|
| Gyro | $50\ (deg/s)^2$ |
| Accelerometer | $0.5\ (m/s^2)^2$ |
| Magnetometer | $1 * 10^{-4}\ Gauss^2$ |
| Altimeter | $0.5\ m^2$ |
| GPS | $2\ m^2$ |

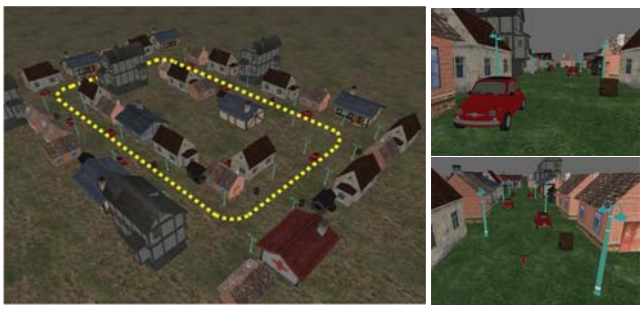
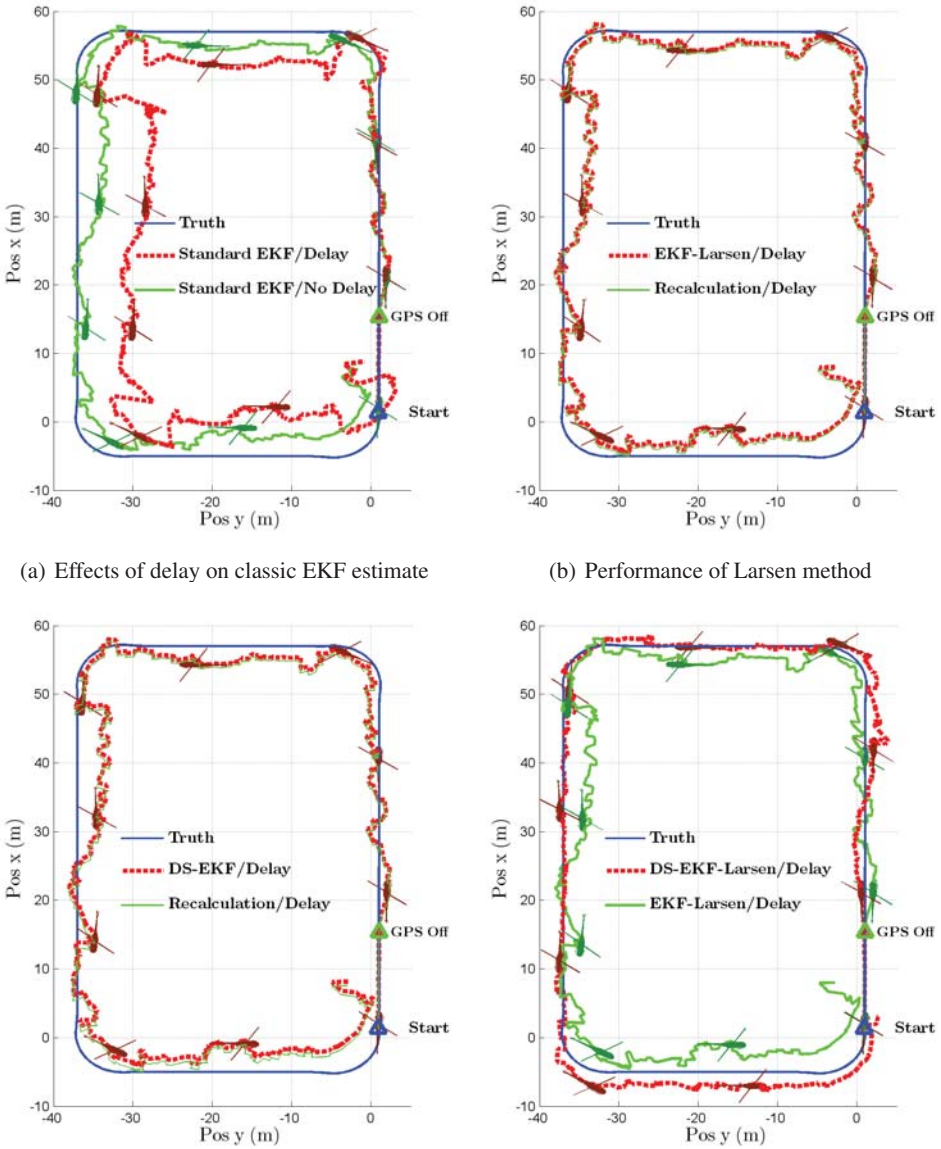


Fig. 4: View of simulated village environment and flight trajectory.

Figure 5(a) shows the effects of delay on the EKF estimates, presenting a comparison of positions obtained by classic EKF, fed with delayed and non-delayed visual measurements. Figures 5(b) and 5(c) present position estimates obtained by the two methods of EKF-Larsen and delayed state EKF, respectively, in the presence of delayed visual




(a) Effects of delay on classic EKF estimate

(b) Performance of Larsen method

(c) Performance of delayed state method

(d) Performance of DS-EKF-Larsen method

Fig. 5: Comparison of position estimates in the *x*-*y* plane. (a) EKF with (dark line) and without (light line) delay on visual measurements; (b) EKF-Larsen method and Recalculation in the presence of delay; (c) Delayed state EKF and Recalculation in the presence of delay; (d) DS-EKF-Larsen and EKF-Larsen in the presence of delay.

measurements, in comparison with the Recalculation method (with delay). Finally, Fig. 5(d) presents position estimates obtained by processing the exact vision-based model through the proposed DS-EKF-Larsen method, in the presence of delayed visual measurements.

Results clearly show the negative effects of delay on the standard EKF estimation, which are compensated with the Recalculation, the sequential filtering EKF-Larsen and also with the delayed state EKF. Moreover, the DS-EKF-Larsen method appears to provide for enhanced estimates, due to the optimal incorporation of an exact model.

### 4.1 Monte Carlo Simulation

A Monte Carlo simulation is used here for considering the effects of random variation on the performance of the approaches, as well as evaluating the computational time burden of each method. The analysis consisted of 100 runs, which is the number of simulations that were necessary in this case to bring the statistics to convergence (in the sense that adding additional simulations did not change the results). For each simulation run, measurements and stereo images are generated for the 100 s maneuver described above,

each with randomly generated errors due to noise and random walks.

The average error in the position, velocity and attitude estimates are shown in Figs. 6–8, using the six implementations of the vision-augmented data fusion procedures explained above. The ratios of the average computational effort of the Kalman update by different approaches to the standard one are depicted in Fig. 9, while the average estimation errors for each approach are reported in Fig. 10.

The EKF-Larsen and the delayed state EKF methods show a good performance, as does the Recalculation approach. In fact, the average errors of these methods is very close to the one obtained by the classic EKF with no delay on the visual measurements. However, the processing time of the filter recalculation increases twofold, as shown by Fig. 9, implying a considerable additional computational burden. On the other hand, the EKF-Larsen approach does not affect the processing time of updating the filter, and therefore conjugates high quality estimation and low computing effort. The processing time of the delayed state EKF is a little bit higher than the one of the EKF-Larsen approach, however it is significantly less than the one required by the Recalculation method. This additional computational burden is an expected effect of augmenting the state vector and covariance matrix. Finally, DS-EKF-Larsen provides better estimates, particularly for positions, in comparison to the EKF-Larsen and delayed state EKF, but with additional complexity and

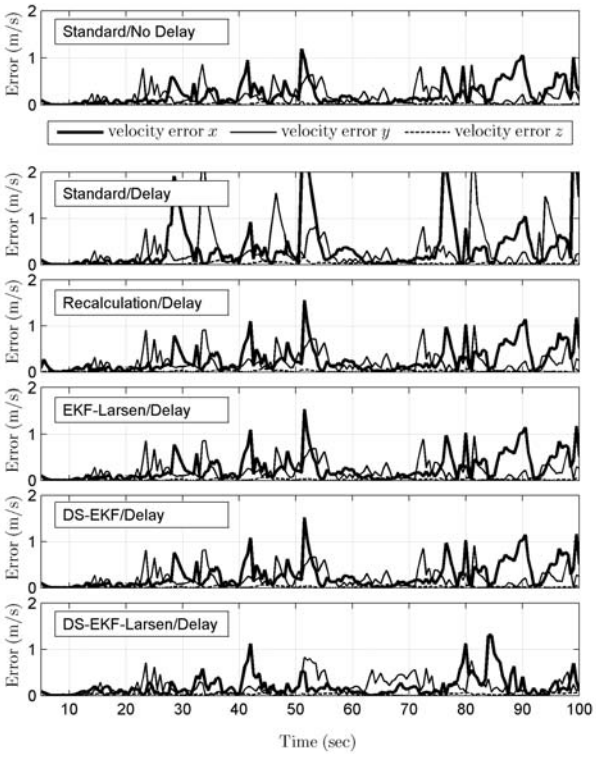



Fig. 7: Velocity estimate errors in the $x$ (thin line), $y$ (thick line) and $z$ (dashed line) directions.

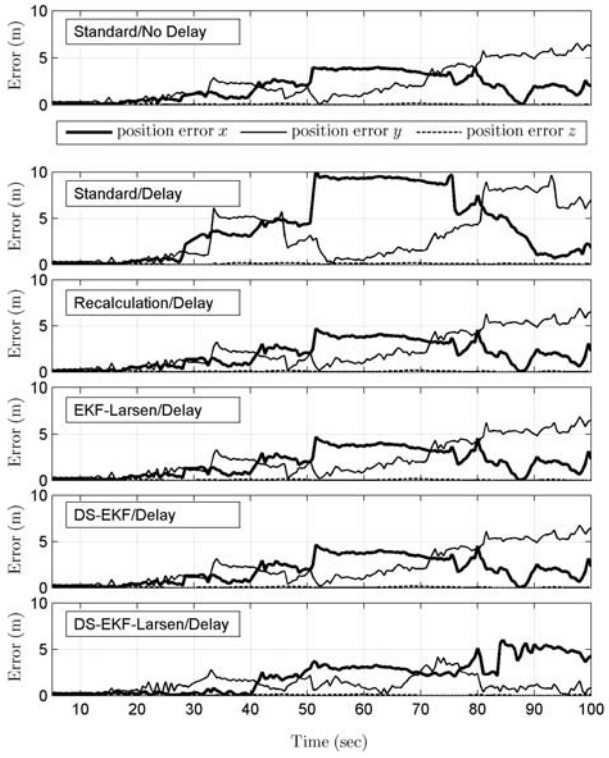



Fig. 6: Position estimate errors in the $x$ (thin line), $y$ (thick line) and $z$ (dashed line) directions.

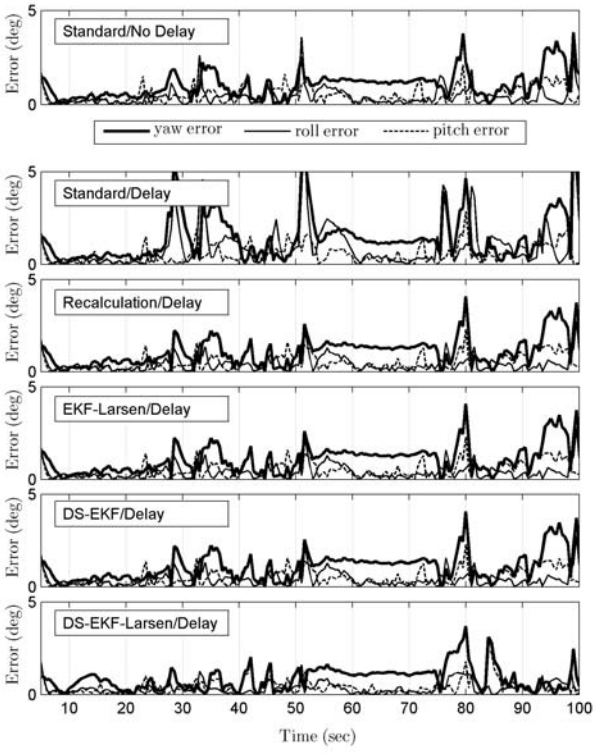



Fig. 8: Attitude estimate errors; yaw error (thin line), roll error (thick line) and pitch error (dashed line).

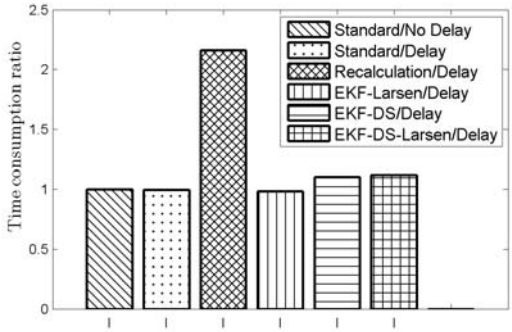



Fig. 9: Ratios of the average time cost of different approaches to the average time cost of the EKF-based implementation of VA-INS.

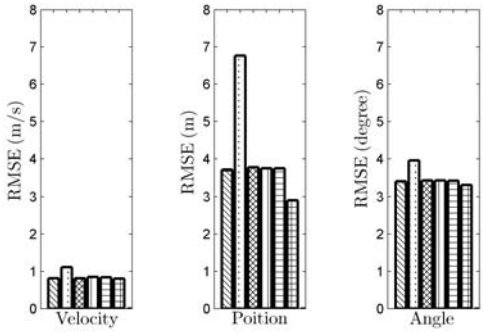



Fig. 10: Average errors of velocity, position and angle estimates.

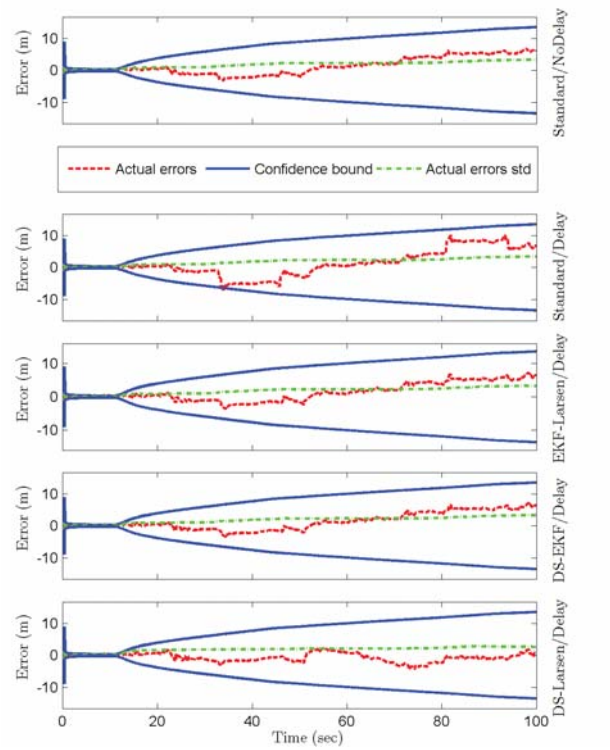



Fig. 11: Position estimate error in $x$ direction (dashed dark line) and confidence bounds (solid line).

a computational cost comparable to the delayed state EKF method. The formulations of EKF-Larsen and delayed state EKF could be also utilized for addressing delays generated by other sensors; however the computational burden of the delayed state EKF method might rise considerably due to the required additional state augmentation at several lagged steps.

Figures 11–13 report the average error in the position, velocity and quaternion estimates, together with the corresponding confidence bounds. The plots compare EKF-Larsen and delayed state EKF in the presence of delay with the standard EKF, fed with delayed and non-delayed visual measurements. As clearly shown in the figures, by feeding delayed visual measurements into the standard EKF, there is no guarantee that the errors will remain within the filter confidence bounds. On the other hand, the EKF-Larsen, delayed state EKF and DS-EKF-Larsen methods perform well in the presence of delayed visual measurements, and exhibit the same confidence bounds obtained by using a standard EKF with non-delayed visual measurements.

Figures 14(a)–14(c) report a comparison of the average error among the Recalculation method, which provides a reference optimal solution, and the other methods. Results show a similar but not identical performance of the various methods, with rather small differences.

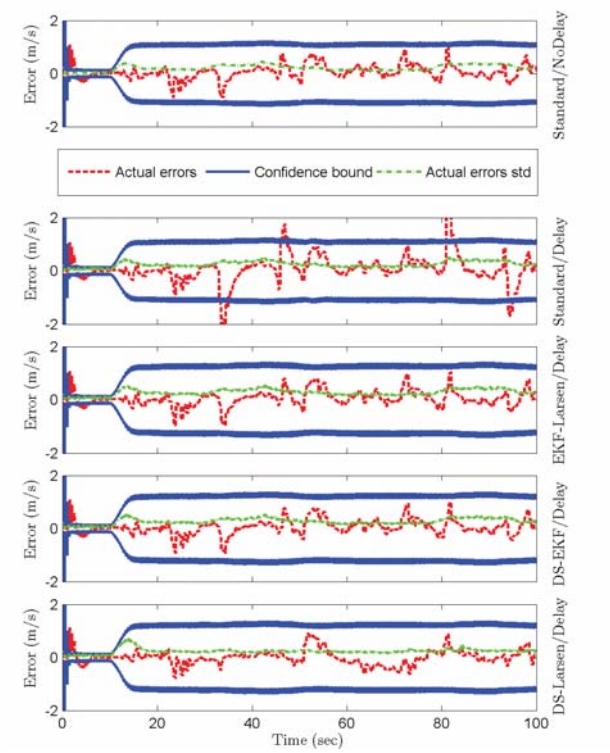



Fig. 12: Velocity estimate error in the $x$ direction (dashed dark line) and confidence bounds (solid line).

## 5 Conclusions

In this work, a previously documented VA-INS was extended by implementing various approaches to handle feature tracking delays in a multi-rate multi-sensor data fusion process. Simulation experiments were used together with Monte Carlo analyses to assess the estimation error and the computational burden of the methods.

The paper shows that delay caused by image processing, if not properly handled in the state estimator, can lead to an appreciable performance degradation. Furthermore, sequential EKF-Larsen, delayed state EKF as well as the Recalculation method restore the estimate accuracy in the presence of delay. On the other hand, the results of the paper indicate that the Recalculation approach implies a significant computational burden, while Larsen method is as expensive as the standard EKF. The delayed state EKF has a slightly higher computational cost than Larsen method, but a significantly lower one than the Recalculation method.

This study concluded that Larsen method, for the present application, provides estimates that have the same quality and computational cost of the non-delayed case. The delayed state EKF can also be a reliable solution, specifically if a few percent increase in computational burden is tolerable.

Moreover, a novel combined implementation of delayed state EKF and Larsen methods enhances the estimation accuracy through an optimal fusion by using an exact vision-based model in the presence of delay. DS-EKF-Larsen provides better estimates in comparison to the EKF-Larsen and delayed state EKF, but with additional complexity and a computational cost comparable to the delayed state EKF method.

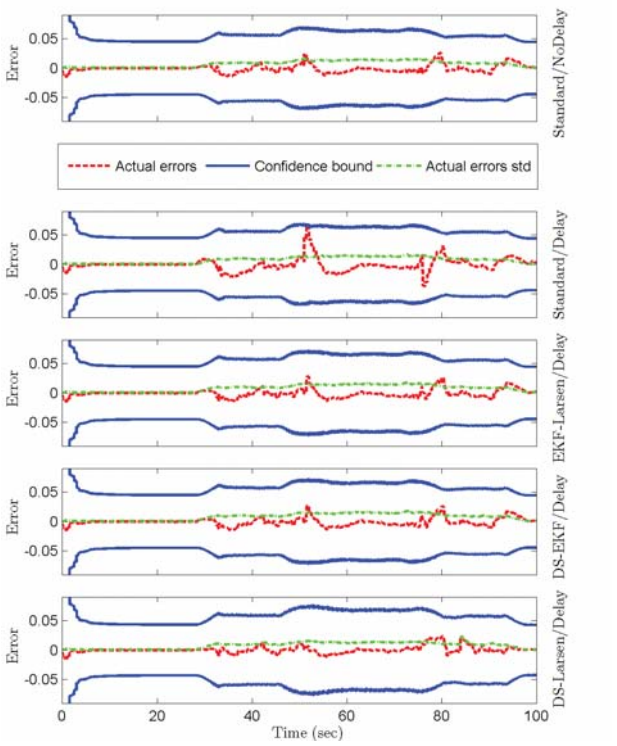
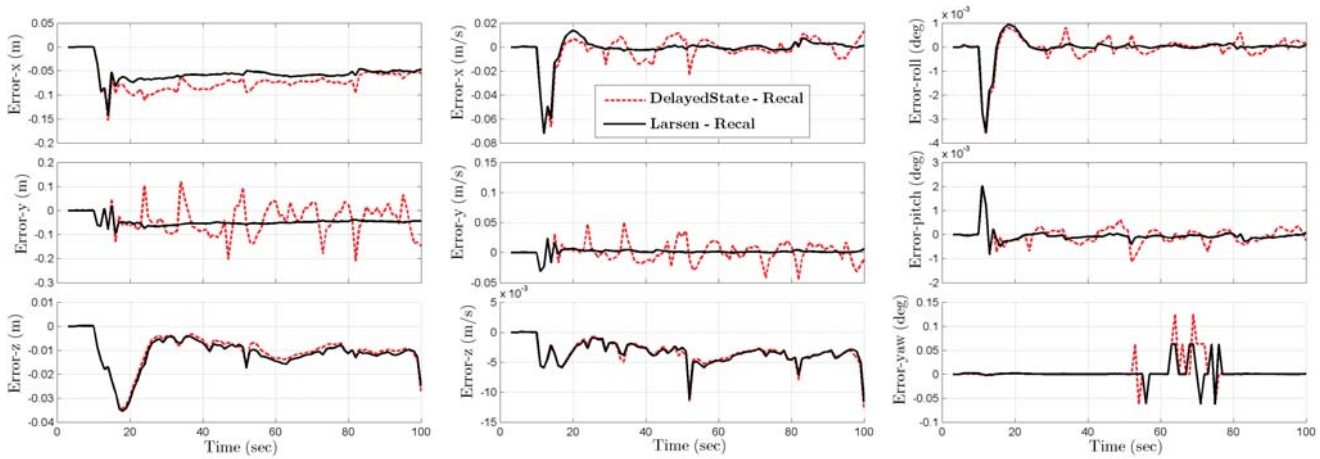



Fig. 13: Normalized quaternion estimate error (dashed dark line) and confidence bounds (solid line).

## References


1. Alexander, H.L.: State estimation for distributed systems with sensing delay. pp. 103–111. SPIE (1991). DOI 10.1117/12.44843
2. Asadi, E., Bottasso, C.L.: Handling delayed fusion in vision-augmented inertial navigation. In: Informatics in Control, Automation and Robotics, 9th International Conference on, pp. 394–401 (2012)
3. Bar-Shalom, Y., Li, X.R.: Multitarget-MultisensorTracking: Principles and Techniques. YBS Publishing (1995)
4. Bonin-Fontand, F., Ortiz, A., Oliver, G.: Visual navigation for mobile robots: A survey. Journal of Intelligent and Robotic Systems **53(3)**, 263–296 (2008)
5. Bottasso, C.L., Leonello, D.: Vision-augmented inertial navigation by sensor fusion for an autonomous rotorcraft vehicle. In: Unmanned Rotorcraft, AHS International Specialists Meeting on, pp. 324–334 (2009)
6. Calonder, M., Lepetit, V., Strecha, C., Fua, P.: Brief: Binary robust independent elementary features. In: Computer Vision, 11th European Conference on, vol. 6314(3), pp. 778–792. LNCS Springer (2010)
7. Challa, S., Evans, R.J., Wang, X., Legg, J.: A fixed-lag smoothing solution to out-of-sequence information fusion problems. Communications In Information and Systems **2**(4), 325–348 (2002)
8. Challa, S., Legg, J.A., Wang, X.: Track-to-track fusion of out-of-sequence tracks. In: Information Fusion, 2002. Fifth International Conference on, vol. 2, pp. 919–926 (2002)
9. Chun, L., Fagen, Z., Yiwei, S., Kaichang, D., Zhaoqin, L.: Stereo-image matching using a speeded up robust feature algorithm in an integrated vision navigation system. The Journal of Navigation **65**, 671–692 (2012). DOI 10.1017/S0373463312000264
10. Corato, F., Innocenti, M., Pollini, L.: Robust vision-aided inertial navigation algorithm via entropy-like relative pose estimation. Gyroscopy and Navigation **4**(1), 1–13 (2013). DOI 10.1134/S2075108713010033
11. Dalgleish, F.R., Tetlow, J.W., Allwood, R.L.: Vision-based navigation of unmanned underwater vehicles : a survey. part 2: Vision-based station-keeping and positioning. In: IMAREST Proceedings, Part B: Journal of Marine Design and Operations, vol. 8, pp. 13–19 (2005)
12. Ferreira, F.J., Lobo, J., Dias, J.: Bayesian real-time perception algorithms on GPU–real-time implementation of Bayesian models for multimodal perception using CUDA. Journal of Real-Time Image Processing **6(3)**, 171–186 (2011)
13. Goedeme, T., Nuttin, M., Tuytelaars, T., Gool, L.V.: Omnidirectional vision based topological navigation. International Journal of Computer Vision **74(3)**, 219–236 (2007)
14. Gopalakrishnan, A., Kaisare, N., Narasimhan, S.: Incorporating delayed and infrequent measurements in extended Kalman filter based nonlinear state estimation. Journal of Process Control **21(1)**, 119–129 (2011)
15. Jianbo, S., Tomasi, C.: Good features to track. In: Computer Vision and Pattern Recognition, IEEE Computer Society Conference on, pp. 593–600 (1994)
16. Jones, E.S., Soatto, S.: Visual-inertial navigation, mapping and localization: A scalable real-time causal approach. The International Journal of Robotics Research **30(4)**, 407–430 (2011)
17. Junker, G.: Pro OGRE 3D Programming. Springer-Verlag, New York (2006)
18. Larsen, T.D., Andersen, N.A., Ravn, O., Poulsen, N.: Incorporation of time delayed measurements in a discrete-time Kalman filter. In: Decision and Control, 37th IEEE Conference on, pp. 3972–3977 (1998)




(a) Position error w.r.t. Recalculation (b) Velocity error w.r.t. Recalculation (c) Attitude error w.r.t. Recalculation

Fig. 14: Performance of EKF-Larsen and delayed state EKF in comparison to the Recalculation method, assumed as an optimal reference solution in the presence of delay and an approximate model.


19. Liu, Y.C., Dai, Q.H.: Vision aided unmanned aerial vehicle autonomy : An overview. In: Image and Signal Processing, 3th International Congress on, pp. 417–421 (2010)
20. Mourikis, A.I., Roumeliotis, S.I.: A multi-state constraint Kalman filter for vision-aided inertial navigation. In: Robotics and Automation, IEEE International Conference on, pp. 3565–3572 (2007)
21. Nister, D., Naroditsky, O., Bergen, J.: Visual odometry for ground vehicle applications. Journal of Field Robotics **23(1)**, 3–20 (2006)
22. Pornsarayouth, S., Wongsaisuwan, M.: Sensor fusion of delay and non-delay signal using Kalman filter with moving covariance. In: Robotics and Biomimetics, IEEE International Conference on, pp. 2045–2049 (2009)
23. Qian, G., Chellappa, R., Zheng, Q.: Robust structure from motion estimation using inertial data. Journal of the Optical Society of America **18(12)**, 2982–2997 (2001)
24. Roumeliotis, S., Burdick, J.: Stochastic Cloning: A generalized framework for processing relative state measurements. In: Robotics and Automation, IEEE International Conference on, vol. 2, pp. 1788–1795 (2002)
25. Roumeliotis, S.I., Johnson, A.E., Montgomery, J.F.: Augmenting inertial navigation with image-based motion estimation. In: Robotics and Automation, IEEE International Conference on, pp. 4326–4333 (2002)
26. Schmidt, S.F.: Applications of state space methods to navigation problems, C. T. Leondes, advanced control systems edn. Academic Press, New York (1996)
27. Stanway, M.J.: Delayed-state sigma point Kalman filters for underwater navigation. In: Autonomous Underwater Vehicles, IEEE/OES Conference on, pp. 1–9 (2010)
28. Tardif, J.P., George, M., Laverne, M., Kelly, A., Stentz, A.: A new approach to vision-aided inertial navigation. In: Intelligent Robots and Systems (IROS), IEEE/RSJ International Conference on, pp. 4161–4168 (2010). DOI 10.1109/IROS.2010.5651059
29. Tatiraju, S., Soroush, S., Ogunnaike, B.A.: Multirate nonlinear state estimation with application to a polymerization reactor. AIChE Journal **45(4)**, 769–780 (1999)
30. Taylor, C.N.: Enabling navigation of mavs through inertial, vision, and air pressure sensor fusion. In: H. Hahn, H. Ko, S. Lee (eds.) Multisensor Fusion and Integration for Intelligent Systems, *Lecture Notes in Electrical Engineering*, vol. 35, pp. 143–158 (2009)
31. Van Der Merwe, R.: Sigma-point kalman filters for probabilistic inference in dynamic state-space models. In: Ph.D Thesis, OGI School of Science and Engineering, Oregon Health and Science University (2004)
32. Veth, M.J., Raquet, J.F., Pachter, M.: Stochastic constraints for efficient image correspondence search. Journal of IEEE Transactions on Aerospace Electronic Systems **42(3)**, 973–982 (2006)