J Intell Robot Syst (2009) 55:345–376
DOI 10.1007/s10846-008-9306-6

Robust Airborne 3D Visual Simultaneous Localization and Mapping with Observability and Consistency Analysis

Abdelkrim Nemra · Nabil Aouf

Received: 1 June 2008 / Accepted: 1 December 2008 / Published online: 24 January 2009

Abstract This paper aims to present a robust airborne 3D Visual Simultaneous Localization and Mapping (VSLAM) solution based on a stereovision system. We propose three innovative contributions to the Airborne VSLAM. The first one is the development of an alternative data fusion nonlinear $H\infty$ filtering scheme. This scheme is based on 3D vision observation model and avoids issues linked with the classical Extended Kalman Filtering (EKF) techniques such as the linearization errors, the initialization problem and noise statistics assumptions. The second contribution consists of a consistency and observability analysis for the Airborne VSLAM. The third contribution is a new approach to map management, based on the *k*-nearest landmark concept, and allowing efficient loop closure detection and map building. This approach reduces considerably the complexity of our Airborne VSLAM algorithm, which becomes independent of the map landmark number. Simulation results show the efficiency of the proposed Airborne VSLAM solution for which comparisons with other techniques are favourable.

Keywords Unmanned aerial vehicle · Simultaneous localization and mapping · Stereo vision · EKF SLAM · NH∞ SLAM · Observability · Consistency · Loop closure · Map management

1 Introduction

Today's challenge for unmanned aerial vehicles (UAV) is to reach full systems autonomy. The increase of UAV autonomy can be achieved by developing tools able

A. Nemra
Unit of Control, Polytechnic Military School, Algiers, Algeria

A. Nemra · N. Aouf (✉)
Sensors Group, Cranfield University, Cranfield, UK
e-mail: n.aouf@cranfield.ac.uk

to provide an accurate automatic localization in an accurate environment map. The simultaneous localization and map building (SLAM) problem looks at the ability of an autonomous vehicle, starting in a partially known or unknown environment, to incrementally build an environment map and simultaneously localize itself within this map. One of the popular approaches to solve the SLAM problem employs Kalman filtering techniques. Extensive research works has been reported in the literature, employing Extended Kalman Filter (EKF) to address several aspects of the SLAM problem [1–7]. Techniques based on particle filtering schemes were also investigated for the SLAM problem as in [8] and [9]. Several successful applications of SLAM algorithms have been developed for indoor robotic applications [4], outdoor robotic applications [5], underwater applications [10], underground applications [11] and aerial applications [12].

An EKF based SLAM approach estimates and stores the vehicle pose and the feature positions of the environment map in one-state vector. The uncertainties of these state estimates are stored in an error-covariance matrix, which includes cross-correlation terms representing cross-correlations between feature/vehicle pose estimates. A major and a fundamental problem with the full EKF based SLAM approach is that the computational burden becomes significantly high in the presence of a large number of features since both the state vector and the covariance matrix become large in size. Some of the recent research on EKF based SLAM approaches have identified this reported problem as the key driver for several proposed improvements [5, 7].

Current challenges and open research related to SLAM include efficient mapping of large environments, modelling of complex and dynamic environments, multi-vehicle SLAM, Airborne SLAM and Airborne 3D VSLAM. Most of these challenges require scalable representations, robust data association algorithms, consistent estimation techniques, and fusion of different sensor modalities.

Traditionally, odometers, laser-range-finder and sonar are the most used sensing means for robotics navigation. However, currently, cameras are becoming competitive alternatives due to their low cost and the rich information content they provide.

Solving SLAM with monocular or stereo vision systems is a crucial open research problem to address efficiently and naturally many autonomous systems in real life applications. Davison, in [13], proposed a vision-based SLAM approach, which used active stereo head and odometry sensing to estimate the location of a robot in planar regions. In [14], Davison has looked at the localization and mapping problem using data from a single passive camera and using an EKF filter. However, SLAM architectures based on Extended Kalman filters are very sensitive to outliers and increase the computational complexity in a square form according to the number of features. To solve these problems, Nir and Bruckstein [15], proposed a particle filter based SLAM approach to estimate camera (vehicle) poses. Thrun and Liu [16], and Hajjdiab [17], used the Sparse Extended Information filters (SEIF) and homography respectively to solve the multi-vehicle SLAM problem. To be used for dynamic real time applications, Particle filters based SLAM approaches are still under improvement, [8, 9], in order to achieve efficient algorithms that make up for their computational burden.

Another important issue, which has been neglected in a number of SLAM implementations, is the convergence analysis of the SLAM algorithm. Only very few analytical results on the convergence and essential properties of SLAM algorithms are available in the literature. Dissanayake in [1], provided some convergence properties of the Kalman Filter SLAM. Lower bounds on the absolute accuracy of the map and the vehicle location were also obtained by an EKF based SLAM algorithm [1]. These results were extended to multi-robots SLAM in [18]. Kim in [19], provided some further analyses on the asymptotic behaviour of one-dimensional Kalman Filter SLAM problem. Having said that, all the proofs presented in [18, 19], deal only with simple linear formulations of the SLAM problem in contrast to most SLAM implementation requirements in terms of process and observation model nonlinearities. In the past few years, a number of researchers have demonstrated, [20, 21], that the lower bound, for the map accuracy, presented in [1] is violated and the EKF SLAM produces inconsistent estimates due to errors introduced during the linearization process. While some explanations of these inconsistency phenomena have been reported, mainly through Monte Carlo simulations, a thorough theoretical analysis of the nonlinear SLAM problem became available only very recently [22].

To date and to the best of our knowledge, no robust and efficient solution has been proposed to solve the Airborne 3D VSLAM problem taking into account the high nonlinearity of the inertial navigation and the 3D vision observation models. We propose a new formulation of the airborne 3D VSLAM based on a stereoscopic vision system and a nonlinear H∞ (NH∞) filtering scheme. In addition, this paper provides robustness properties and consistency analysis of the Airborne VSLAM using EKF and robust NH∞ filters.

The outline of the paper is as follows: Section 2 develops the SLAM process model using the Inertial Measurements Unit (IMU) information. In Section 3, the development of the observation model, based on embedded stereo cameras, is detailed. The general architecture of the Airborne VSLAM, a theoretical explanation of the EKF VSLAM and the concept of loop closing are presented in Section 4. Section 5 details the observability analysis of the EKF VSLAM, followed, in Section 6, by the development of the robust Airborne VSLAM based on the nonlinear H∞ filter and the consistency analysis. In Section 7, we propose a new approach for landmark map management that is essential to achieve efficient and computationally attractive Airborne VSLAM. Finally, in Section 8, simulation results using realistic scenarios are presented and a comparison between the EKF Airborne VSLAM and the NH∞ Airborne VSLAM is made.

2 Process Model

The localization problem of an airborne system is formulated based on the core-sensing device, which is the inertial measurements unit (IMU). This unit measures the acceleration *(ax, ay, az)* and the rotation rates *(p, q, r)* of the airborne platform with high update rates. These quantities are then transformed and processed to provide the aerial vehicle position (X, Y, Z), velocity (U, V, W), and attitude (ϕ, θ, ψ) resulting in an Inertial Navigation System (INS).

Let us represent the INS with the following nonlinear model

$$\begin{cases} \dot{x} = f(x, u, w) \\ y = h(x, u, v) \end{cases} \tag{1}$$

where: $f(\cdot,\cdot,\cdot)$ is a non-linear state transition function which links the current state with the previous state and current control input. $h(\cdot,\cdot,\cdot)$ is a non-linear observation function, which links the observation to the current state. y is the observation vector.

x is the state vector, which contains the position in navigation frame, velocity in body frame and Euler angles, and u represents the IMU outputs (angular rates, and accelerations) as follows:

$$x = [X,\ Y,\ Z,\ U,\ V,\ W,\ \phi,\ \theta,\ \psi]^T \tag{2}$$

$$u = \left[p,\ q,\ r,\ ax,\ ay,\ az\right]^T \tag{3}$$

w and v are respectively the process and observation noises.

The navigation equations require the definition of at least two frames, one frame for the body/inertial representation (vehicle) and another frame for the navigation representation. Then, vehicle equations of motion can be given by simple integration and frame transformations as follows:

2.1 Equations of Motion

The Euler angle rates $\dot{\phi},\ \dot{\theta},\ \dot{\psi}$ can be calculated as follows:

$$\begin{bmatrix}\dot{\phi}\\ \dot{\theta}\\ \dot{\psi}\end{bmatrix} = \begin{bmatrix}1 & \sin(\phi)\tan(\theta) & \cos(\phi)\tan(\theta)\\ 0 & \cos(\phi) & -\sin(\phi)\\ 0 & \sin(\phi)\sec(\theta) & \cos(\phi)\sec(\theta)\end{bmatrix}\begin{bmatrix}p\\ q\\ r\end{bmatrix} \tag{4}$$

Assuming that the IMU is at the vehicle centre of gravity, the true vehicle acceleration in the body frame is given by $\dot{U},\ \dot{V},\ \dot{W}$ as:

$$\begin{bmatrix}\dot{U}\\ \dot{V}\\ \dot{W}\end{bmatrix} = \begin{bmatrix}ax + Vr - Wq + g\sin(\theta)\\ ay - Ur + Wp - g\cos(\theta)\sin(\phi)\\ az + Uq - Vp - g\cos(\theta)\cos(\phi)\end{bmatrix} \tag{5}$$

The resulting acceleration vector is integrated with respect to time to obtain the velocity of the vehicle in the body frame:

$$\begin{bmatrix}U\\ V\\ W\end{bmatrix} = \int \begin{bmatrix}\dot{U}\\ \dot{V}\\ \dot{W}\end{bmatrix} dt \tag{6}$$

The velocity vector is then integrated to read the position of the vehicle in the body frame. If the velocity, transformed down to the navigation frame, is integrated, we get the position vector $[X, Y, Z]^T$ in the navigation frame.

$$\begin{bmatrix}X\\ Y\\ Z\end{bmatrix} = \int C_{bn}^{T}(\phi,\theta,\psi)\begin{bmatrix}U\\ V\\ W\end{bmatrix} dt \tag{7}$$

where *Cbn* is the Direct Cosine Transform matrix that rotates a vector from the body frame to the navigation frame.

$$C_{bn} = \begin{bmatrix} \cos(\theta)\cos(\psi) & \cos(\theta)\sin(\psi) & -\sin(\theta) \\ \sin(\theta)\sin(\phi)\cos(\psi) & \sin(\psi)\sin(\theta)\sin(\phi) & \sin(\phi)\cos(\theta) \\ -\sin(\psi)\cos(\phi) & +\cos(\psi)\cos(\phi) & \\ \sin(\theta)\cos(\phi)\cos(\psi) & \sin(\phi)\sin(\theta)\cos(\phi) & \cos(\phi)\cos(\theta) \\ +\sin(\psi)\sin(\phi) & -\cos(\psi)\sin(\theta) & \end{bmatrix} \tag{8}$$

Thus the nonlinear state model of Eq. 1 is given by Eq. 9 as,

$$f(x,u) = \begin{bmatrix} \begin{bmatrix} \cos(\theta)\cos(\psi) & \cos(\theta)\sin(\psi) & -\sin(\theta) \\ \sin(\theta)\sin(\phi)\cos(\psi) & \sin(\psi)\sin(\theta)\sin(\phi) & \sin(\phi)\cos(\theta) \\ -\sin(\psi)\cos(\phi) & +\cos(\psi)\cos(\phi) & \\ \sin(\theta)\cos(\phi)\cos(\psi) & \sin(\phi)\sin(\theta)\cos(\phi) & \cos(\phi)\cos(\theta) \\ +\sin(\psi)\sin(\phi) & -\cos(\psi)\sin(\theta) & \end{bmatrix}^T \begin{bmatrix} U \\ V \\ W \end{bmatrix} \\ \begin{bmatrix} ax + Vr - Wq + g\sin(\theta) \\ ay - Ur + Wp - g\cos(\theta)\sin(\phi) \\ az + Uq - Vp - g\cos(\theta)\cos(\phi) \end{bmatrix} \\ \begin{bmatrix} 1 & \sin(\phi)\tan(\theta) & \cos(\phi)\tan(\theta) \\ 0 & \cos(\phi) & -\sin(\phi) \\ 0 & \sin(\phi)\sec(\theta) & \cos(\phi)\sec(\theta) \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \end{bmatrix} \tag{9}$$

The navigation solution provided by INS, however, drifts with time, as in most other dead reckoning systems. The drift rate of the inertial position is typically a cubic function of time, which makes the development of any inertial based localization a challenge. Even small errors in the gyros will be accumulated in angle estimates (roll and pitch), which in turn act negatively on the vehicle acceleration, thus resulting in quadratic velocity (and cubic position) errors. Therefore, the INS requires additional information to reduce these errors. This is achieved by the visual information obtained from the map building and linked to the aerial vehicle localization through the nonlinear observation model detailed below.

3 Observation Model

Computer vision becomes vital for automatic perception and recognition of the environment. Today, cameras become an important sensor for aerial robotics because of their low cost and high quality of the acquired images, which are necessary for aerial mapping.

Stereoscopic vision is broadly defined as the recovery of three-dimensional characteristics of a scene from multiple images taken from two different viewpoints. Stereo is an attractive source of information for machine perception because it leads to direct range measurements, and unlike monocular approaches, does not merely infer depth or orientation through the use of photometric and statistical assumptions.

Once the stereo images are brought into point-to-point correspondence, recovering range values is relatively straightforward. Another advantage is that stereo is a passive method. Although active ranging methods that use structured light, laser range finders, or other active sensing techniques are useful in tightly controlled domains such as industrial automation applications, they are clearly limited in use for outdoor aerial vision problems. Once the stereo images have been acquired, a number of steps are required to produce 3D data from these images.

3.1 Camera Model

The perspective camera model includes intrinsic and extrinsic parameters, which ensure the geometric transformation between camera/image and world/camera respectively Fig. 1.

3.1.1 Intrinsic Parameters (Transformation Camera/Image)

Intrinsic parameters of a camera are defined by the horizontal and vertical scale factor (k_v and k_u), the image centre coordinates ($u0$, $v0$) given in the image frame and the focal distance f as:

$$I_c = \begin{pmatrix} \alpha_u & 0 & u_0 & 0 \\ 0 & \alpha_v & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \text{ with } \begin{cases} \alpha_u = -f \times k_u \\ \alpha_v = f \times k_v \end{cases}$$

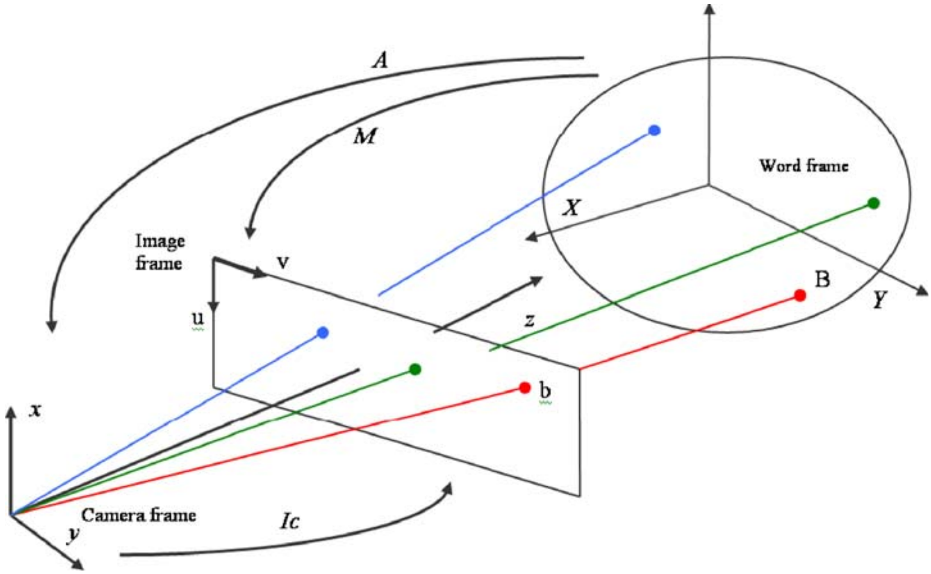


Fig. 1 Camera model

3.1.2 Extrinsic Parameters (Transformation World/Camera)

These parameters define the transformation from the world to camera frame given by the homogeneity matrix A.

$$A = \begin{pmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} R & t \\ 0 & 1 \end{pmatrix}$$

The matrix A is a combination of a rotation matrix R and a translation t from the world frame to the camera frame and obviously, the matrix A changes with the camera (UAV) displacement.

3.2 Airborne Stereo Vision

One of the important challenges for unmanned aerial vehicles is to use low cost interceptive sensors to ensure a full UAV autonomy without absolute sensor. Stereovision cameras become, nowadays, the most appropriate device to reach this objective. Airborne stereo vision is more difficult than vision (Stereo or mono) for mobile robotics because of the six degrees of freedom (DOF) requirements of the unmanned aerial vehicles. In the following, we develop an observation model using stereoscopic cameras embedded on a UAV, Fig. 2. Similar developments of camera observation models, mainly for mobile robotics applications, are presented in [5, 23], and [24].

The landmarks m_i represent specific feature points in the operation environment. These landmarks stand for features like corners or local extrema, which can be detected by feature extractors such as Harris, Harris-Laplace and SIFT [25].

The landmark m_i coordinates in the navigation frame are given by:

$$\overrightarrow{Om_i}|_n = \overrightarrow{OO_b} + \overrightarrow{O_bO_s} + \overrightarrow{O_sO_{c1}} + \overrightarrow{O_{c1}m_i}$$

$$\left.\overrightarrow{Om_i}\right|_n = \overrightarrow{OO_b} + \overrightarrow{O_bO_s} + \overrightarrow{O_sO_{c2}} + \overrightarrow{O_{c2}m_i}$$

3.2.1 Transformation Body (UAV)/Navigation Frame

$\overrightarrow{OO_b} = [X_{uav} \;\; Y_{uav} \;\; Z_{uav}]^T$ is the position of the UAV in the navigation frame. This position is given by the INS.

3.2.2 Transformation IMU/Body (UAV) Frame

$$\left.\overrightarrow{O_bO_s}\right|_n = C_b^n \times \left.\overrightarrow{O_bO_s}\right|_b = C_b^n \times \left[x_s \;\; y_s \;\; z_s\right]^T$$

where C_b^n is a rotation matrix that transforms vectors from body to navigation frame. $\left[x_s \;\; y_s \;\; z_s\right]^T$ is the position of the IMU presented in body frame. If the IMU is at the centre of gravity then $\left[x_s \;\; y_s \;\; z_s\right]^T = [0 \;\; 0 \;\; 0]^T$.

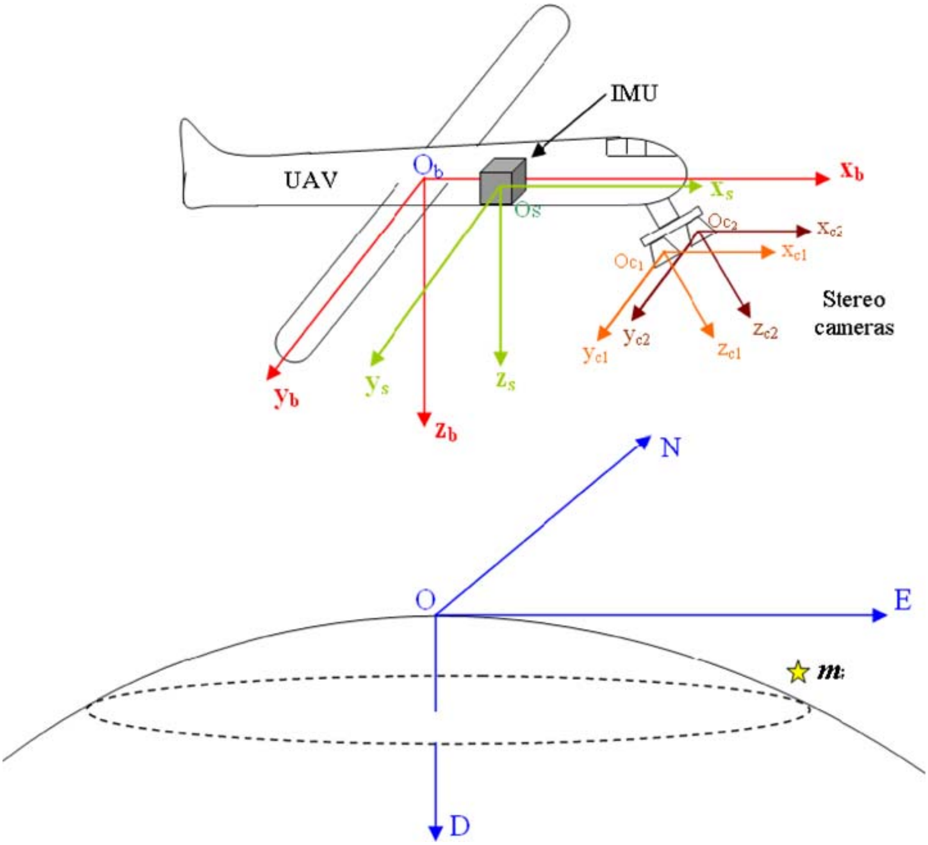


Fig. 2 UAV with IMU and stereo cameras

3.2.3 Transformation IMU/Cameras Frames

$$\left.\overrightarrow{O_s O_{c1}}\right|_n = C_b^n \times C_s^b \times \begin{bmatrix} x_{c1} & y_{c1} & z_{c1} \end{bmatrix}^T$$

$$\left.\overrightarrow{O_s O_{c2}}\right|_n = C_b^n \times C_s^b \times \begin{bmatrix} x_{c2} & y_{c2} & z_{c2} \end{bmatrix}^T$$

where C_s^b is a rotation matrix that transforms vectors from IMU to body frame. $\begin{bmatrix} x_{c1} & y_{c1} & z_{c1} \end{bmatrix}^T$ $\left(\begin{bmatrix} x_{c2} & y_{c2} & z_{c2} \end{bmatrix}^T\right)$ is the position of the camera right (left) in the IMU frame presented in IMU frame.

3.2.4 Landmark Coordinates in Cameras Frames

$$\left.\overrightarrow{O_{c1} m_i}\right|_n = C_b^n \times C_s^b \times C_{c1}^s \times \begin{bmatrix} x_{mi} & y_{mi} & z_{mi} \end{bmatrix}_{c1}^T = M_{c1}^n \times \begin{bmatrix} x_{mi} & y_{mi} & z_{mi} \end{bmatrix}_{c1}^T$$

$$\left.\overrightarrow{O_{c2} m_i}\right|_n = C_b^n \times C_s^b \times C_{c2}^s \times \begin{bmatrix} x_{mi} & y_{mi} & z_{mi} \end{bmatrix}_{c2}^T = M_{c2}^n \times \begin{bmatrix} x_{mi} & y_{mi} & z_{mi} \end{bmatrix}_{c2}^T$$

where C_{c1}^{s} $\left(C_{c2}^{s}\right)$ is a rotation matrix that transforms vectors from the right (left) camera to the IMU frame. $\left[x_{m1}\ \ y_{m1}\ \ z_{m1}\right]^{T}$ $\left(\left[x_{m2}\ \ y_{m2}\ \ z_{m2}\right]^{T}\right)$ is the position of the landmark m_i in the right (left) camera presented in right (left) camera frame.

3.2.5 Landmark Coordinates in Navigation Frame

$$\begin{aligned}\overrightarrow{Om_i}|_n &= [X_{uav}\ \ Y_{uav}\ \ Z_{uav}]_n^T + C_b^n \times \left[x_s\ \ y_s\ \ z_s\right]_b^T + C_b^n \\ &\times C_s^b \times \left[x_{c1}\ \ y_{c1}\ \ z_{c1}\right]_s^T + M_{c1}^n \times \left[x_{mi}\ \ y_{mi}\ \ z_{mi}\right]_{c1}^T \\ \overrightarrow{Om_i}\Big|_n &= [X_{uav}\ \ Y_{uav}\ \ Z_{uav}]_n^T + C_b^n \times \left[x_s\ \ y_s\ \ z_s\right]_b^T + C_b^n \\ &\times C_s^b \times \left[x_{c2}\ \ y_{c2}\ \ z_{c2}\right]_s^T + M_{c2}^n \times \left[x_{mi}\ \ y_{mi}\ \ z_{mi}\right]_{c2}^T\end{aligned}$$

If we put

$$\overrightarrow{OO_{c1}}\Big|_n = T_{c1} = [X_{uav}\ Y_{uav}\ Z_{uav}]_n^T + C_b^n \times \left[x_s\ \ y_s\ z_s\right]_b^T + C_b^n \times C_s^b \times \left[x_{c1}\ \ y_{c1}\ \ z_{c1}\right]_s^T$$

$$\overrightarrow{OO_{c2}}\Big|_n = T_{c2} = [X_{uav}\ Y_{uav}\ Z_{uav}]_n^T + C_b^n \times \left[x_s\ \ y_s\ z_s\right]_b^T + C_b^n \times C_s^b \times \left[x_{c2}\ \ y_{c2}\ \ z_{c2}\right]_s^T$$

Then

$$\overrightarrow{Omi}\Big|_n = \begin{bmatrix} M_{c1}^n & T_{c1} \\ 0 & 1 \end{bmatrix} \left[x_{mi}\ y_{mi}\ z_{mi}\ 1\right]_{c1}^T = \begin{bmatrix} M_{c2}^n & T_{c2} \\ 0 & 1 \end{bmatrix} \left[x_{mi}\ y_{mi}\ z_{mi}\ 1\right]_{c2}^T$$

$$\overrightarrow{Omi}\Big|_n = Mh_{c1}^n \times \left[x_{mi}\ y_{mi}\ z_{mi}\ 1\right]_{c1}^T = Mh_{c2}^n \times \left[x_{mi}\ y_{mi}\ z_{mi}\ 1\right]_{c2}^T$$

3.2.6 Transformation Camera/Image Frame

Using the cameras model (transformation camera/image), we can conclude:

$$\begin{aligned}[s_1 u_1\ \ s_1 v_1\ \ s_1]^T &= I_{c1} \times \left[x_{mi}\ \ y_{mi}\ \ z_{mi}\ \ 1\right]_{c1}^T \\ [s_2 u_2\ \ s_2 v_2\ \ s_2]^T &= I_{c2} \times \left[x_{mi}\ \ y_{mi}\ \ z_{mi}\ \ 1\right]_{c2}^T\end{aligned}$$

I_{c1} and I_{c2} are the matrix of intrinsic parameters of the right and the left camera respectively (Section 3.1.1)

Then

$$\begin{aligned}[s_1 u_1\ \ s_1 v_1\ \ s_1]^T &= I_{c1} \times \left(Mh_{c1}^n\right)^{-1} \times \overrightarrow{Om_i}\Big|_n \\ [s_2 u_2\ \ s_2 v_2\ \ s_2]^T &= I_{c2} \times \left(Mh_{c2}^n\right)^{-1} \times \overrightarrow{Om_i}\Big|_n\end{aligned}$$

After development, the scale factors, for camera right and left s_1 and s_2 respectively, are given by the following equation:

$$\begin{aligned}s_1 &= m_{31}^{c1} x_{mi}^n + m_{32}^{c1} y_{mi}^n + m_{33}^{c1} z_{mi}^n + m_{34}^{c1} \\ s_2 &= m_{31}^{c2} x_{mi}^n + m_{32}^{c2} y_{mi}^n + m_{33}^{c2} z_{mi}^n + m_{34}^{c2}\end{aligned}$$

Therefore, the observation model, linking the perceived visual landmarks to the SLAM state vector is given by:

$$\begin{cases} u_1 = \dfrac{m_{11}^{c1} x_{mi}^n + m_{12}^{c1} y_{mi}^n + m_{13}^{c1} z_{mi}^n + m_{14}^{c1}}{m_{31}^{c1} x_{mi}^n + m_{32}^{c1} y_{mi}^n + m_{33}^{c1} z_{mi}^n + m_{34}^{c1}} \\[2ex] v_1 = \dfrac{m_{21}^{c1} x_{mi}^n + m_{22}^{c1} y_{mi}^n + m_{23}^{c1} z_{mi}^n + m_{24}^{c1}}{m_{31}^{c1} x_{mi}^n + m_{32}^{c1} y_{mi}^n + m_{33}^{c1} z_{mi}^n + m_{34}^{c1}} \\[2ex] u_2 = \dfrac{m_{11}^{c2} x_{mi}^n + m_{12}^{c2} y_{mi}^n + m_{13}^{c2} z_{mi}^n + m_{14}^{c2}}{m_{31}^{c2} x_{mi}^n + m_{32}^{c2} y_{mi}^n + m_{33}^{c2} z_{mi}^n + m_{34}^{c2}} \\[2ex] v_2 = \dfrac{m_{21}^{c2} x_{mi}^n + m_{22}^{c2} y_{mi}^n + m_{23}^{c2} z_{mi}^n + m_{24}^{c2}}{m_{31}^{c2} x_{mi}^n + m_{32}^{c2} y_{mi}^n + m_{33}^{c2} z_{mi}^n + m_{34}^{c2}} \end{cases}$$

where $\left[x_{mi}^n \; y_{mi}^n \; z_{mi}^n \right]^T$ is the coordinate of the landmark m_i in the navigation frame (NED). m_{ij}^{c1} and m_{ij}^{c2} are the components of $I_{c1} \times \left(Mh_{c1}^n \right)^{-1}$ and $I_{c2} \times \left(Mh_{c2}^n \right)^{-1}$ respectively.

4 EKF Airborne VSLAM

Kalman Filter (KF) is an effective stochastic estimator for a large number of problems. However, as in most real applications, the process and/or observation models are nonlinear therefore the linear Kalman filter algorithm cannot be directly applied. To overcome this, a linearised Kalman filter or Extended Kalman Filter (EKF) is proposed. Based on these techniques, the navigation solution uses the current estimated states from the filter, at each time step k, as the linearization reference from which the estimation procedure can start. If the filter operates properly, the linearization error around the estimated solution can be maintained reasonably small. However, if the filter is ill-conditioned due to modelling errors, incorrect tuning of the covariance matrix, or initialisation error, then the estimation error will affect the linearization error, which in turn will affect the estimation process. This is known as a filter divergence. For this reason, the EKF requires greater care in modelling and tuning than the linear Kalman filter.

4.1 EKF Filter

The state vector of the UAV SLAM is given by:

$$\begin{aligned} x &= [x_v \; x_m] \\ x_v &= [X, \; Y, \; Z, \; U, \; V, \; W, \; \phi, \; \theta, \; \psi]^T, \\ x_m &= [m_1 \; m_2 \; m_3 \; \ldots m_N] \end{aligned}$$

x_v is the state vector of the vehicle and x_m is the state vector of the landmarks observed during the UAV flight. Landmark initialization algorithm is based on triangulation methods, which directly calculates a three-dimensional position of landmark based on stereovision measurements and the Inverse Model of Observation (IMO).

The augmented system, defined in Eq. 1, can be written as a non-linear discrete time state transition equation:

$$\begin{aligned} x_k &= f\left(x_{k-1}, u_{k-1}\right) + g\left(x_{k-1}\right) w_{k-1} \\ y_k &= h\left(x_k, v_k\right) \end{aligned} \tag{11}$$

f is the discrete version of Eq. 9 (in addition to elements of the landmarks states), g is a nonlinear function, x_k is the state at time step k, w_k is some additive process noises, y_k is the observation made at time k, v_k is some additive observation noises. We assume that w_k and v_k are uncorrelated zero mean Gaussian with known covariance Q_k and R_k. The objective of the filtering technique is, then, to estimate x_k using available observation y_k.

The non-linear vehicle and observation models may be expanded around the filtered and predicted estimates of $\hat{x}_k$ and $\hat{x}_{k-1}$ as [32]:

$$\begin{aligned} x_k = {} & f\left(\hat{x}_{k/k}, u_{k-1}\right) + \nabla f_k\left(\hat{x}_{k/k}\right)\left[x_k - \hat{x}_{k/k}\right] + \Delta_1\left(x_k - \hat{x}_{k/k}\right) \\ & + \left[g\left(\hat{x}_{k/k}\right) + \Delta_2\left(x_k - \hat{x}_{k/k}\right)\right] w_k \end{aligned} \tag{12}$$

$$y_k = h\left(\hat{x}_{k/k-1}, u_k\right) + \nabla h_k\left(\hat{x}_{k/k-1}\right)\left[x_k - \hat{x}_{k/k-1}\right] + \Delta_3\left(x_k - \hat{x}_{k/k-1}\right) + v_k \tag{13}$$

where $\nabla f_k(x)$ is the Jacobian of f evaluated at x_{k-1}, $\nabla h_k(x)$ is the Jacobian of h evaluated at x_{k-1} and Δ_i represent higher order of the Taylor series expansions.

The filter state error is defined as:

$$\tilde{x}_{k/k} = x_k - \hat{x}_{k/k} \tag{14}$$

The prediction error can be determined from subtracting the true state x_k from the prediction estimate.

$$\tilde{x}_{k/k-1} = x_k - \hat{x}_{k/k-1} \tag{15}$$

The state and observation model may be then rewritten as:

$$\begin{aligned} x_{k+1} &= F_k x_k + \Gamma_k w_k + \Omega_k + \Delta_1\left(\tilde{x}_{k/k}\right) + \Delta_2\left(\tilde{x}_{k/k}\right) w_k \\ y_k &= H_k x_k + v_k + \Psi_k + \Delta_3\left(\tilde{x}_{k/k-1}\right) \end{aligned} \tag{16}$$

where $F_k = \nabla f_k\left(\hat{x}_{k/k}\right)$, $\Gamma_k = g\left(\hat{x}_{k/k}\right)$, $H_k = \nabla h_k\left(\hat{x}_{k/k-1}\right)$, $\Omega_k = f\left(\hat{x}_{k/k}\right) - F_k\hat{x}_{k/k}$ and $\Psi_k = h\left(\hat{x}_{k/k-1}\right) - H_k\hat{x}_{k/k-1}$

Using the equations above, the final EKF formulation is written in Predictor-Corrector scheme, with the higher order terms of the Taylor series equal to zero, as follows:

$$\hat{x}_{k+1/k} = F_k x_{k/k-1} + F_k K_k\left(y_k - H_k\hat{x}_{k/k-1}\right) \tag{17}$$

$$K_k = P_{k/k-1}\left(I + H_k^T R_k^{-1} H_k P_{k/k-1}\right)^{-1} H_k^T R_k^{-1} \tag{18}$$

$$P_{k+1/k} = F_k P_{k/k-1}\left(I + H_k^T R_k^{-1} H_k P_{k/k}\right)^{-1} F_k^T + Q_k \tag{19}$$

The estimated error covariance for the system:

$$P_{k/k} = \begin{bmatrix} P_{vv}\left(k/k\right) & P_{v1}\left(k/k\right) & \dots & P_{vN}\left(k/k\right) \\ P_{1v}\left(k/k\right) & P_{11}\left(k/k\right) & \dots & P_{1N}\left(k/k\right) \\ \vdots & \vdots & \ddots & \vdots \\ P_{Nv}\left(k/k\right) & P_{N1}\left(k/k\right) & \dots & P_{NN}\left(k/k\right) \end{bmatrix} \tag{20}$$

where the sub-matrices $P_{vv}(k/k)$, $P_{vi}(k/k)$, $P_{ii}(k/k)i = 1,...N$ are the vehicle-to-vehicle, vehicle-to-landmark and landmark-to-landmark covariances respectively.

When there are large deviations of the estimated system state trajectory from the nominal trajectory, the nonlinear model in Eq. 11 is weakly approximated by the Taylor series expansion around the conditional mean. In this case, the higher order terms of the Taylor expansion become necessary. As the EKF formulation assumes the higher order terms are neglected, we propose, in the following section, other alternative filters avoiding possible problems due to the linearization. In addition to this limitation, Kalman filtering assumes that the system dynamics are not uncertain and the process and observation noises have known statistical properties (centred Gaussian). This assumption is often not realistic in cases such as navigation and visual data. Building a navigation solution for UAV's based on this kind of assumptions could lead to erroneous results with very negative consequences.

4.2 Loop Closing

As the vehicle moves in the environment, it builds a map of the landmarks and correlates this information in order to determine precisely the vehicle location. One solution to the correlation problem was introduced by Smith, Self and Cheeseman [26], and is called the Stochastic Map (SM) (the approach proposed in the previous section Eq. 21). The SM allows for the concurrent mapping of landmarks and localization of the vehicle with respect to the landmarks using Extended Kalman Filter architecture. The Stochastic Map is essentially an augmented Extended Kalman filter, where the observed landmarks states are stored in the filter state vector along with the vehicle states.

Loop closing means the capacity of recognizing previously detected landmarks to reduce the uncertainties in both the UAV and the landmark positions. Using an augmented state vector where information about the landmarks is stored, the loop closure detection becomes an automatic task.

We consider the scenario of a UAV navigating and making a map loop. The curves shown in Fig. 3 represent the evolution of uncertainty for the UAV (x-position), and one observed landmark, m. This landmark is observed at the beginning of UAV navigation and is observed again after 200s of navigation. From Fig. 3 we observe the reduction of the uncertainties, both, for the UAV (Fig. 3a) and the landmark (Fig. 3b) positions when the loop closing is detected at $t = 200$ s. At this time, the UAV detects and recognizes some landmarks observed previously at $t = 0$ s. The loop

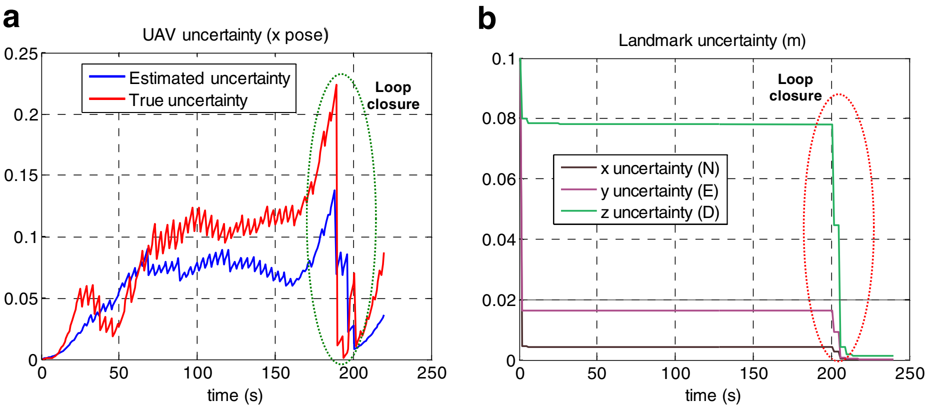


Fig. 3 Loop closure detection at $t = 200$ s; **a** UAV uncertainty, **b** Landmark uncertainty

closing detection leads to a significant decrease in the true and estimated uncertainty for localization and mapping. However, the main problem of the augmented map method (SM), which facilitates the loop closing detection, is the complexity that increases exponentially with the number of observed landmarks. This problem will be solved in Section 7, when a new approach of map management is proposed.

5 Airborne VSLAM Observability

Observability of nonlinear systems is not very easy to analyse. Many papers treat the SLAM problem without mentioning or examining the observability of the system. Observability, if it does not hold, can lead to an inconsistent SLAM solution [27, 28]. Only very few papers have tried to treat the observability of SLAM, mainly the local observability, which constitutes only a partial solution of the problem. The best way to prove the global observability of nonlinear systems, such as Airborne VSLAM, is to use Lie derivatives [22]. However, as it is very hard to develop this method in our case, where the size of state vector is larger than 12 (9 states for the UAV, 3 states for each landmark), we propose to analyse the observability problem with a reduced SLAM system assuming a planar displacement of the UAV (at a fixed altitude). The cameras are heading down vertically. This leads to a three DOF system with six model states (one landmark), rather than 12 states, as follows:

$$\dot{x} = f(x, u) = \begin{bmatrix} U\cos(\psi) - V\sin(\psi) \\ U\sin(\psi) + V\cos(\psi) \\ \psi \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad x = \left[X_{uav}\; Y_{uav}\; \psi\;\; x_{mi}\; y_{mi}\; z_{mi}\right]^T, \quad x \in \Re^6 \tag{21}$$

where $[x_{mi}, y_{mi}, z_{mi}]^T$ are the coordinates of the landmark in navigation frame. α_{ui}, α_{vi}, u_{0i}, v_{0i}, $i \in \{1, 2\}$ are the intrinsic parameters of the right and left cameras.

$$y = h(x) = \begin{bmatrix} \dfrac{\alpha_{u1}(\cos(\psi)\,x_{mi} + \sin(\psi)\,y_{mi} - \sin(\psi)\,Y_{uav} - \cos(\psi)\,X_{uav})}{z_{mi}} + u_{01} \\ \dfrac{\alpha_{v1}(-\sin(\psi)\,x_{mi} + \sin(\psi)\,X_{uav} - y_{c1} + \cos(\psi)\,y_{mi} - \cos(\psi)\,Y_{uav})}{z_{mi}} + v_{01} \\ \dfrac{\alpha_{u2}(\cos(\psi)\,x_{mi} + \sin(\psi)\,y_{mi} - \sin(\psi)\,Y_{uav} - \cos(\psi)\,X_{uav})}{z_{mi}} + u_{02} \\ \dfrac{\alpha_{v2}(-\sin(\psi)\,x_{mi} + \sin(\psi)\,X_{uav} - y_{c1} + \cos(\psi)\,y_{mi} - \cos(\psi)\,Y_{uav})}{z_{mi}} + v_{02} \end{bmatrix},$$
$$y \in \Re^4 \tag{22}$$

This system satisfies the observability rank condition if any of the observability matrices are of rank 6 (recall that $x \in \Re^6$), where the observability matrices are given by the Lie derivatives as:

$$O = \frac{\partial}{\partial x_k}\left[\, Lf_0^T \; Lf_1^T \; Lf_2^T \; Lf_3^T \; Lf_4^T \;\; Lf_5^T \right]^T \tag{23}$$

with

$$Lf_0 = h(x) = y$$
$$Lf_i = \frac{\partial Lf_{i-1}}{\partial x_k} x_{k+1} \tag{24}$$

We find that $rank(O) = 4 < 6$, which means that the SLAM is not observable.

If we introduce a known landmark $m^* = [x_{\text{known}}, y_{\text{known}}, z_{\text{known}}]^T$ to the state vector, the observation model becomes:

$$y' = h'(x) = \begin{bmatrix} \dfrac{\alpha_{u1}(\cos(\psi)\,x_{mi} + \sin(\psi)\,y_{mi} - \sin(\psi)\,Y_{uav} - \cos(\psi)\,X_{uav})}{z_{mi}} + u_{01} \\ \dfrac{\alpha_{v1}(-\sin(\psi)\,x_{mi} + \sin(\psi)\,X_{uav} - y_{c1} + \cos(\psi)\,y_{mi} - \cos(\psi)\,Y_{uav})}{z_{mi}} + v_{01} \\ \dfrac{\alpha_{u2}(\cos(\psi)\,x_{mi} + \sin(\psi)\,y_{mi} - \sin(\psi)\,Y_{uav} - \cos(\psi)\,X_{uav})}{z_{mi}} + u_{02} \\ \dfrac{\alpha_{v2}(-\sin(\psi)\,x_{mi} + \sin(\psi)\,X_{uav} - y_{c1} + \cos(\psi)\,y_{mi} - \cos(\psi)\,Y_{uav})}{z_{mi}} + v_{02} \\ \dfrac{\alpha_{u1}(\cos(\psi)\,x_{\text{known}} + \sin(\psi)\,y_{\text{known}} - \sin(\psi)\,Y_{uav} - \cos(\psi)\,X_{uav})}{z_{\text{known}}} + u_{01} \\ \dfrac{\alpha_{v1}(-\sin(\psi)\,x_{\text{known}} + \sin(\psi)\,X_{uav} - y_{c1} + \cos(\psi)\,y_{\text{known}} - \cos(\psi)\,Y_{uav})}{z_{\text{known}}} + v_{01} \\ \dfrac{\alpha_{u2}(\cos(\psi)\,x_{\text{known}} + \sin(\psi)\,y_{\text{known}} - \sin(\psi)\,Y_{uav} - \cos(\psi)\,X_{uav})}{z_{\text{known}}} + u_{02} \\ \dfrac{\alpha_{v2}(-\sin(\psi)\,x_{\text{known}} + \sin(\psi)\,X_{uav} - y_{c1} + \cos(\psi)\,y_{\text{known}} - \cos(\psi)\,Y_{uav})}{z_{\text{known}}} + v_{02} \end{bmatrix} \tag{25}$$

In this case we find that $rank(O) = 6$, which means that the SLAM problem became observable and thus consistent. However a known feature is not a practical solution especially in outdoor environment. Thus, proposing an observable and robust airborne VSLAM algorithm is our main contribution in this paper.

6 Robust Observable SLAM

6.1 EKF SLAM Consistency

As is reported in the beginning of the paper, the lower bound for the map accuracy, based on EKF SLAM, and presented in [1] is violated and produces inconsistent estimates (especially UAV position) [20].

Shoudong, in his paper [22], provides both convergence properties and a consistency analysis for some basic scenarios of the nonlinear two-dimensional EKF SLAM problem. A number of recent publications [20] indicate that the key source of EKF SLAM inconsistency is the error introduced during the linearization process. While it is clear that linearization is an approximation that can introduce errors into the estimation process it is reasonable to expect that the incorrect estimate is likely to be either too optimistic (the estimated uncertainty smaller than the true uncertainty) or too pessimistic (the estimated uncertainty larger than the true uncertainty). In the literature related to SLAM, only estimator inconsistency, as a result of optimistic estimates, is reported [22].

Furthermore, the EKF SLAM requires an accurate enough process model and known process and observation noise characteristics (centred Gaussian noises). Moreover, in all the theoretical convergence properties proved in the previous works, [1, 22], it is assumed that the Jacobians are evaluated at the true vehicle pose and the true landmark positions. For realistic SLAM scenarios, the true locations of the UAV and landmarks are not known, and the Jacobians have to be evaluated at the estimated values. This may result in overconfident (inconsistent) estimates [22].

In this section we propose a solution dealing with robustness and consistency problems for the Airborne VSLAM by using an alternative robust filtering scheme, which makes up for the negative effect of the linearization and increases the estimates consistency.

6.2 Nonlinear H∞ Filter

After a decade or so of reappraising the nature and role of Kalman filters, engineers realized that they need a new filtering scheme that could handle modelling errors and noise uncertainties while minimizing the worst-case estimation error rather than the covariance of the estimation error. State estimators that can tolerate such uncertainties are called robust. Although robust estimators based on Kalman filter theory can be designed, these approaches are somewhat ad-hoc in that they attempt to modify an already existing approach. In contrast, the H∞ filter was specifically designed for optimality and robustness [30, 32].

6.2.1 Discrete-time Linear H∞ Filter

In this section, we present the essential theory, the principle and the optimization cost function leading to the linear H∞ filter. The algorithm along with the imposed constraints are presented and then extended to the nonlinear H∞ case in Section 3.2.2:

1. Assume a linear system equations given as:

$$\begin{aligned} x_{k+1} &= F_k x_k + w_k \\ y_k &= H_k x_k + v_k \end{aligned} \tag{26}$$

   where w_k and v_k are noise terms. These noise terms may be random with possibly unknown statistics, or they may be deterministic. They may have a nonzero mean. Our goal is to estimate linear combinations of the system states. That is, we want to estimate z_k given by

$$z_k = L_k x_k$$

   where L_k is a user defined full rank matrix.

2. The cost function is given as:

$$J_1 = \frac{\sum_{k=0}^{N-1} \left\| z_k - \hat{z}_k \right\|_{S_k}^2}{\left\| x_0 - \hat{x}_0 \right\|_{P_0^{-1}}^2 + \sum_{k=0}^{N-1} \left(\|w_k\|_{Q_k^{-1}}^2 + \|v_k\|_{R_k^{-1}}^2 \right)} \tag{27}$$

   where P_0, Q_k, R_k and S_k are chosen symmetric, positive definite matrices. Our goal is to find an estimate $\hat{z}_k$ that minimizes J_1. To make this minimization tractable, we chose a performance bound θ and seek an estimation strategy leading to an estimate $\hat{z}_k$, which satisfies:

$$J_1 < \frac{1}{\theta}$$

   Rearranging this equation results in:

$$J = -\frac{1}{\theta} \left\| x_0 - \hat{x}_0 \right\|_{P_0^{-1}}^2 + \sum_{k=0}^{N-1} \left[\left\| z_k - \hat{z}_k \right\|_{S_k}^2 - \frac{1}{\theta} \left(\|w_k\|_{Q_k^{-1}}^2 + \|v_k\|_{R_k^{-1}}^2 \right) \right] < 0 \tag{28}$$

   Our estimation problem is thus a minmax problem defined as follows:

$$J^* = \min_{\hat{z}_k} \max_{w_k,\, v_k,\, x_0} J$$

   Thus, the worst-case is obtained when w_k, v_k and x_0 are chosen to maximize J then the solution is to find an estimate $\hat{z}_k$ which minimize this maximum. Since$y_k = H_k x_k + v_k$, we see that $v_k = y_k - H_k x_k$ and $\|v_k\|_{R_k^{-1}}^2 = \|y_k - H_k x_k\|_{R_k^{-1}}^2$ Since $z_k = L_k x_k$ and $\hat{z}_k = L_k \hat{x}_k$ we see that:

$$\begin{aligned} \left\| z_k - \hat{z}_k \right\|_{S_k}^2 &= \left(z_k - \hat{z}_k \right)^T S_k \left(z_k - \hat{z}_k \right) \\ &= \left(x_k - \hat{x}_k \right)^T L_k^T S_k L_k \left(x_k - \hat{x}_k \right) \\ &= \left\| x_k - \hat{x}_k \right\|_{\bar{S}_k}^2 \end{aligned} \tag{29}$$

where

$$\bar{S}_k = L_k^T S_k L_k \tag{30}$$

Then, we substitute these results in Eq. 28 to obtain:

$$J = -\frac{1}{\theta} \left\| x_0 - \hat{x}_0 \right\|^2_{P_0^{-1}} + \sum_{k=0}^{N-1} \left[\left\| x_k - \hat{x}_k \right\|^2_{\bar{S}_k} - \frac{1}{\theta} \left(\|w_k\|^2_{Q_k^{-1}} + \|y_k - H_k x_k\|^2_{R_k^{-1}} \right) \right] < 0 \tag{31}$$

The cost function J_1 can be made to be less than $1/\theta$ (a user-specified bound), which means J less than 0.

3. H∞ filter algorithm
Similarly to the Kalman filter, the H∞ filter is put in a recursive structure as follows:

$$\begin{aligned} \bar{S}_k &= L_k^T S_k L_k \\ K_k &= P_k \left[I - \theta \overline{S}_k P_k + H_k^T R_k^{-1} H_k P_k \right]^{-1} H_k^T R_k^{-1} \\ \hat{x}_{k+1} &= F_k \hat{x}_k + F_k K_k \left(y_k - H_k \hat{x}_k \right) \\ P_{k+1} &= F_k P_k \left[I - \theta \overline{S}_k P_k + H_k^T R_k^{-1} H_k P_k \right]^{-1} F_k^T + Q_k \end{aligned} \tag{32}$$

This strategy is derived from the solution of the optimization problem

$$J^* = \min_{\hat{x}_k} \max_{w_k,\, y_k,\, x_0} J \tag{33}$$

The following condition must hold at each time step k in order for the above estimator to be a solution to the problem:

$$P_k^{-1} - \theta\, S_k + H_k^T R_k^{-1} H_k > 0 \tag{34}$$

Before examining the nonlinear H∞ Filter (NH∞) with respect to the minimum H∞ entropy, we analyze, first, the H∞ linear filtering case. Two fundamental differences between the Kalman filter and the H∞ optimal filter exist. First, the H∞ filter is optimal in terms of minimizing the ∞-norm of the gain between a set of disturbance inputs, and the estimation error. Thus, the filter operates such that the worst-case gain of the estimation error is minimized. In contrast, the Kalman filter minimizes the mean square gain between the disturbances and the estimation error.
The second difference between the Kalman filter and the H∞ filter is that the minimum mean square estimate commutes with linear operations. However, the minimal ∞-norm estimate does not possess this property, and the optimal H∞ estimator depends on the plant output being estimated.

6.2.2 Nonlinear H∞ Filter

Researchers such as Shaked and Berman [29], Petersen and Savkin [30], Basar and Bernhard [31] and Einicke and White [32] have studied the nonlinear optimal *H*∞

estimator. Our algorithm uses a similar procedure to the one illustrated in Einicke and White [32]. The nonlinear $H\infty$ filter attempts to estimate the states given in Eq. 12 while satisfying the $H\infty$ performance criterion with respect to bounded in norm uncertainties Δ_i (the higher order linearization terms should be norm bounded as $\|\Delta_i\| \leq \delta_i$).

The system given in Eq. 16 can be rewritten into the following form:

$$\hat{x}_{k+1} = F_k \hat{x}_k + B_k v_k + \Omega_k + \Pi_k \tag{35}$$

$$y_k = H_k x_k + w_k + \Psi_k + \sum_k \tag{36}$$

where

$$\Pi_k = \Delta_1 \left(\tilde{x}_{k/k}\right) + \Delta_2 \left(\tilde{x}_{k/k}\right) \upsilon_k \tag{37}$$

and

$$\sum\nolimits_k = \Delta_3 \left(\tilde{x}_{k/k-1}\right) \tag{38}$$

These inputs must satisfy the following norm bounds:

$$\left\| \prod_k \right\|_2^2 \leq \delta_1^2 \left\| \tilde{x}_{k/k} \right\|_2^2 + \delta_2^2 \left\| w_k \right\|_2^2 \tag{39}$$

and

$$\left\| \sum\nolimits_k \right\|_2^2 \leq \delta_3^2 \left\| \tilde{x}_{k/k} \right\|_2^2 \tag{40}$$

Instead of solving for the non-linear vehicle/landmarks model and the observation model that contain the extra terms $\prod_k$ and Σ_k, which are not used in the EKF formulation, the following scaled $H\infty$ problem is considered:

$$\hat{x}_{k+1} = F_k \hat{x}_k + B_k c_v v_k + \Omega_k \tag{41}$$

$$y_k = H_k x_k + c_w w_k + \Psi_k \tag{42}$$

where $c_w^2 = 1 - \gamma^2 \delta_1^2 - \gamma^2 \delta_3^2$ and $c_v^2 = c_w^2 \left(1 + \delta_1^2\right)^{-1}$ and $\gamma = 1/\theta$. This final form results in the same structure as the EKF, except that the error covariance correction of the linear $H\infty$ filter is used with noise processes w_k and v_k scaled by c_w and c_v.

Notes:

- When $\delta_1 = \delta_2 = \delta_3 = 0$ the Nonlinear $H\infty$ reverts to an Extended $H\infty$ filter, similar to the EKF in the linearization procedure, while minimizing the worst case estimation error for each sampling time estimate as in the linear $H\infty$ case.
- As γ goes to ∞, the Extended $H\infty$ reverts back to the EKF.

6.3 Consistent EKF (NH∞) Airborne VSLAM

Even if the use of a robust filter, such as NH∞ filter, improves the performance of the airborne VSLAM algorithm (Optimality and Robustness) and increases the consistency, the problem of estimating the UAV states in an absolute frame using relative and uncertain data, is still posed. This problem is directly linked with the above mentioned issue of 'full observability' without known features (Section 5) [33, 34].

The following figures show an example of the EKF SLAM inconsistency based on the UAV position errors along the axes x, y and z. It is clear, from Fig. 4, that the estimated uncertainty is very small compared to the true uncertainty specifically for the UAV y and z states. At $t = 200$ s we observe the increase of the SLAM consistency because of the loop closure detection Fig. 4a–c).

As a solution, we propose the following strategy to improve the consistency of the SLAM filtering scheme whether it is the EKF filter or the NH∞ filter:

Let us assume that at $t = 0$, the UAV position is known accurately in an absolute frame. From this position, the UAV observes landmarks with their associated

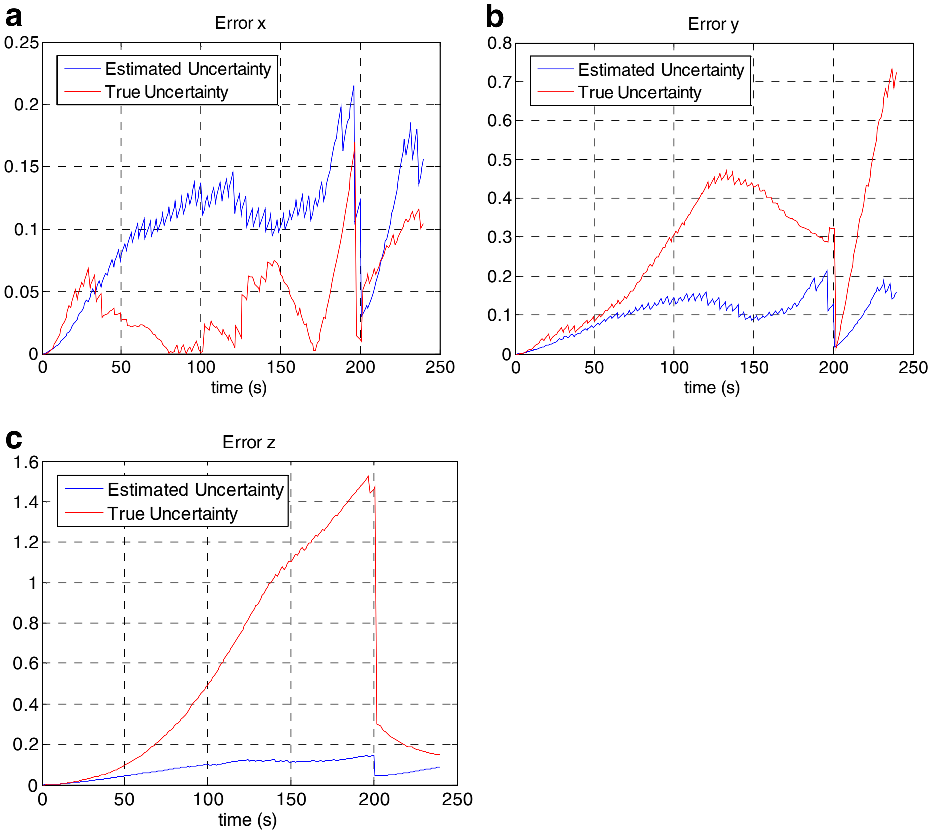


Fig. 4 Estimated and True uncertainty for the UAV position

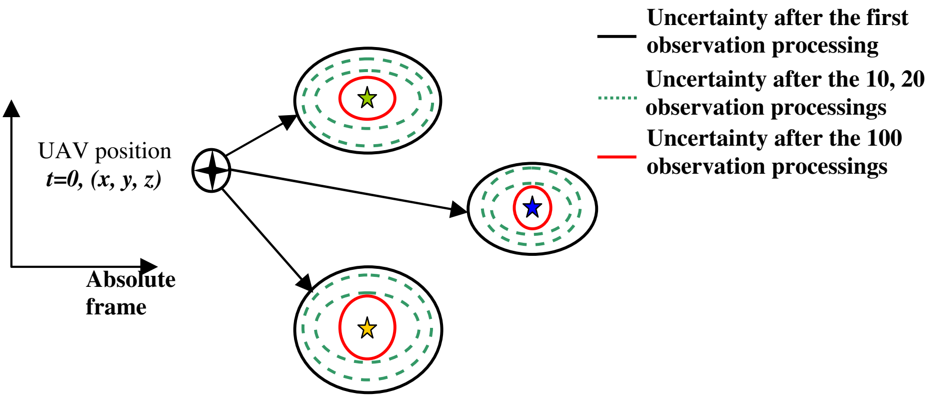

Uncertainty after the first observation processing
Uncertainty after the 10, 20 observation processings
Uncertainty after the 100 observation processings
UAV position
t=0, (x, y, z)
Absolute frame

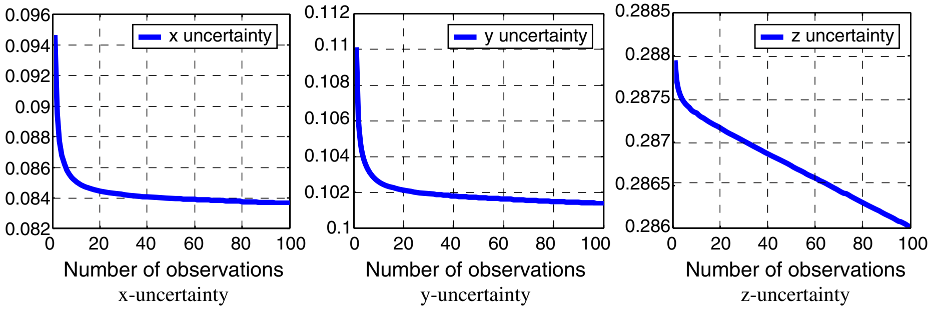
Fig. 5 Evolution of landmark uncertainties with the number of times of landmarks processed (observations)

uncertainties, which depend on the camera's uncertainties. If this observation is processed ***n*** times, through the filtering scheme (EKF), then the landmark's position uncertainties decrease to a lower bound Fig. 5, Fig. 6. Once this is achieved, these landmarks can be considered as a relative known feature, which ensure the full observability of the filter.

To evaluate the degree of SLAM consistency, we define the parameter *co* as:

$$co = \frac{\text{Estimated uncertainty}}{\text{True uncertainty}} \text{ and propose: } \begin{cases} co > 1 & \Rightarrow \text{ consistent SLAM} \\ co \leq 1 & \Rightarrow \text{ inconsistent SLAM} \end{cases}$$

This parameter, calculated at each sampling time of simulation, tells us about the instantaneous SLAM consistency. However, this is not enough to inform us about the true consistency of the SLAM filtering scheme (EKF is used here). As shown in Fig. 7a, just after the loop closure is detected at $t = 200$ s, $co = 0.18/0.08 = 2.25 > 1$, which signifies the EKF SLAM consistency. On the other hand, it is clear, from the same figure, that for $t \in [90s..200s]$, the estimated uncertainty is smaller than the



x-uncertainty y-uncertainty z-uncertainty

Fig. 6 Landmark uncertainties after 100 observations processed, x-uncertainty, y-uncertainty, z-uncertainty

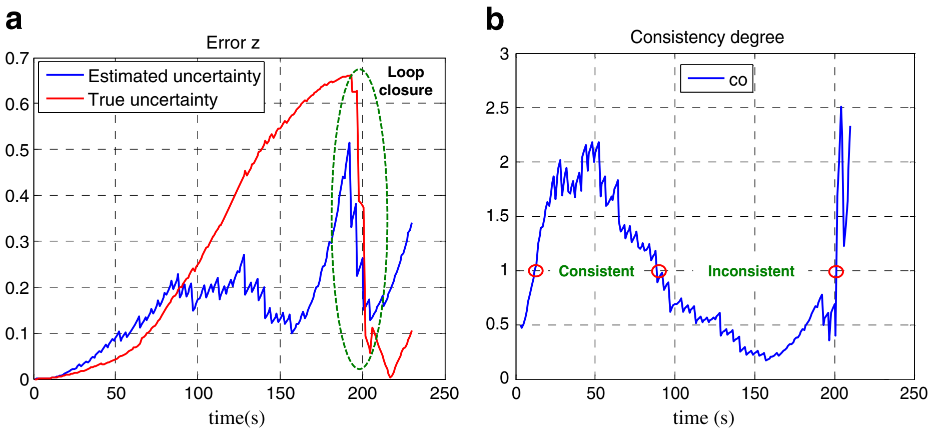


Fig. 7 Consistency analysis. **a** Estimated and True z uncertainty. **b** Evolution of the consistency degree

true uncertainty, which signifies the EKF SLAM inconsistency [22]. These results are confirmed by the consistency degree co given by Fig. 7b.

To avoid this kind of erroneous analysis, we propose to check the SLAM (based on EKF here) consistency in the worst-case situation, which is evaluated by the robust consistency parameter Rco as: $Rco = \min\left(\frac{\text{Estimated uncertainty}}{\text{True uncertainty}}\right)$. This latter is calculated at the last of simulation telling us about the whole SLAM consistency. Using this parameter for the simulation in Fig. 7b, the minimal value of co obtained at $t = 160$ s is $Rco = 0.22 < 1$, which shows clearly the EKF SLAM inconsistency.

In the following experiment, the UAV navigates and makes landmark observations. Each observed landmark is used in one (ten) observation processing (update). In other words, Eq. 17 is processed one (ten) time(s) at each filter sampling time. The following tables show the values of Rco when we used one and ten updates (observation processing) of the landmark observations.

As can be seen from the tables, the estimation accuracy increases when the number of observation processing increases. Thus, we can conclude that by observing the landmarks and processing them $\boldsymbol{n}$ times, we increase the consistency of the filter (Table 1 and Table 2). Our approach will increase, a little, the airborne VSLAM algorithm complexity. How this complexity is primarily dependent on the choice of landmarks and their initial observation uncertainties.

The solution proposed in this sub-section will be used in the later of the paper for the EKF based Airborne VSLAM and the NH∞ based Airborne VSLAM as mean of increasing the consistency of the proposed filtering schemes.

7 3D VSLAM Map Management

A critical issue in a SLAM algorithm is to decide how to represent the joint distribution over vehicle poses and feature map. In particular, the map management

Table 1 Degree of consistency with one observation processing

State	One observation processing (update)						
	Estimated uncertainty			True uncertainty			Rco
	Min	Max	Mean	Min	Max	Mean	
x	0	0.1251	0.0722	0	0.0619	0.0173	0.9033
y	0	0.1463	0.0726	0	0.2812	0.1113	0.0146
z	0	0.0995	0.0457	0	0.4908	0.1497	0.0003
U	0.0042	0.0673	0.0464	0	0.0546	0.0248	0.0161
V	0.0042	0.0767	0.0469	0	0.0854	0.0575	1.8452
W	0.0022	0.0258	0.0165	0	0.1080	0.0531	8.6046
θ	0.0004	0.0028	0.0022	0	0.0014	0.0008	7.8261
ϕ	0.0004	0.0027	0.0022	0	0.0048	0.0029	191.8137
ψ	0.0004	0.0038	0.0027	0	0.0041	0.0021	15.8661

is very important due to the fact that maps are usually represented by a high number of parameters [1, 23, 24, 34, 35] and [36].

The $o(N^3)$ complexity of the Kalman filter (N being the dimension of the state vector) does not allow large environments to be efficiently mapped since it limits the total number of landmarks that can be stored in the map [1]. Beyond this upper limit, real-time processing is no longer possible [23]. To prevent the state vector from a rapid dimension increase that would dramatically limit the mapping capacity of our SLAM system, we introduce a map management approach based on a landmark performance criterion, which allows the substitution of some elements of the state vector. The proposed approach is very efficient and is based on the following concept:

At each time step, the state vector is proposed to contain the new observed features and the best k-landmarks observed previously as explained in Fig. 8. It is clear that the size of the state vector at each time is limited to ($k+$ number of new landmarks) (in our experiment $k = 20$). The same limitation is used for the covariance matrix. In order to maintain the correlation between landmarks and UAV/landmarks, we propose to save all the observed landmarks, during the flight, with their respective covariance matrices. This will help with building a large map and to make the loop closing detection possible.

Table 2 Degree of consistency with ten observations processed

State	10 Observation processed (updates)						
	Estimated uncertainty			True uncertainty			Rco
	Min	Max	Mean	Min	Max	Mean	
x	0	0.104	0.0578	0	0.0297	0.0105	1.5416
y	0	0.128	0.0566	0	0.1097	0.0425	0.0422
z	0	0.0901	0.0406	0	0.2933	0.0724	0.0369
U	0.0042	0.0507	0.0326	0	0.0362	0.0165	0.4091
V	0.0042	0.0659	0.0382	0	0.0783	0.0532	3.8507
W	0.0022	0.0242	0.0157	0	0.0809	0.039	0.0455
ϕ	0.0004	0.0027	0.0021	0	0.0035	0.0013	9.5392
θ	0.0004	0.0025	0.0021	0	0.0043	0.0033	29.0489
ψ	0.0004	0.0034	0.0025	0	0.0027	0.0011	7.5769

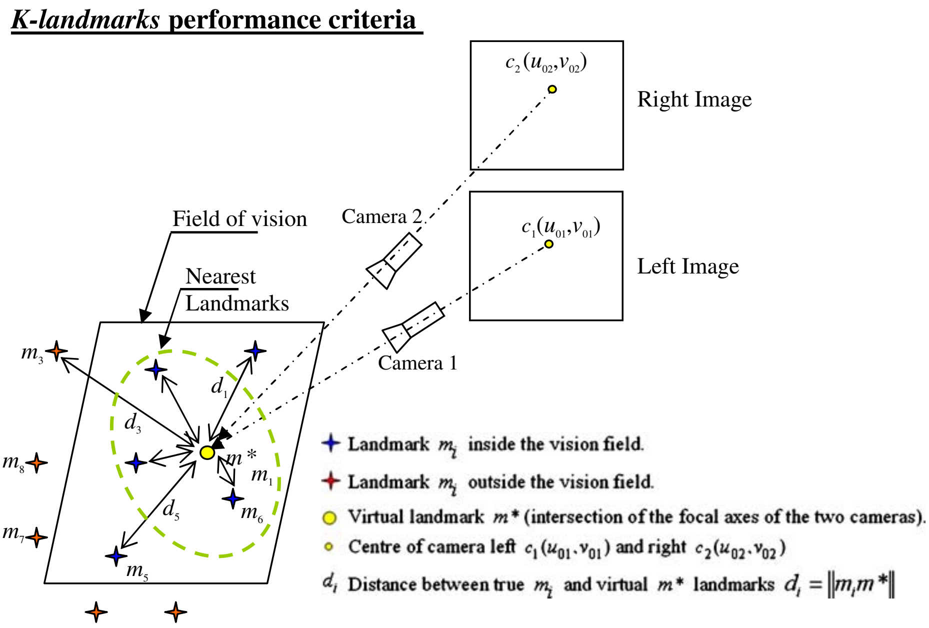

K-landmarks performance criteria
$c_2(u_{02},v_{02})$
Right Image
Camera 2
$c_1(u_{01},v_{01})$
Left Image
Field of vision
Nearest Landmarks
Camera 1
m_3
d_1
d_3
$m*$
m_8
m_1
m_6
d_5
m_7
m_5
Landmark m_i inside the vision field.
Landmark m_i outside the vision field.
Virtual landmark $m*$ (intersection of the focal axes of the two cameras).
Centre of camera left $c_1(u_{01},v_{01})$ and right $c_2(u_{02},v_{02})$
d_i Distance between true m_i and virtual $m*$ landmarks $d_i = \|m_i m*\|$

Fig. 8 Map management using the nearest k-landmarks approach

As can be seen from Fig. 8, our approach is based on the measure of Euclidean distance of observed landmarks m_i and a virtual landmark $m*$. The latter is given by the intersection of the focal axes of the two cameras making it the virtual centre point of the real image. The best $k-$landmarks are the nearest $k-$landmarks to that virtual point. Subsequently, at each time step the state vector will contain the new observed landmarks and the k, previously observed, nearest landmarks to the virtual centre point.

8 Simulation Results and Discussions

In the following simulations, we present a number of UAV flight scenarios for which NH∞ and EKF Airborne VSLAM schemes are developed and compared. The proposed map management approach is also assessed against the classical stochastic map approach.

8.1 EKF Airborne VSLAM

As can be seen from Fig. 9a–d, when the noise is a centred Gaussian, a reasonably good EKF based estimator is obtained for x, y and z, presenting much better estimated trajectory than the INS trajectory. The inaccuracy of the estimator in z, Fig. 9c, affects the performance of the UAV trajectory estimation as shown in Fig. 9d.

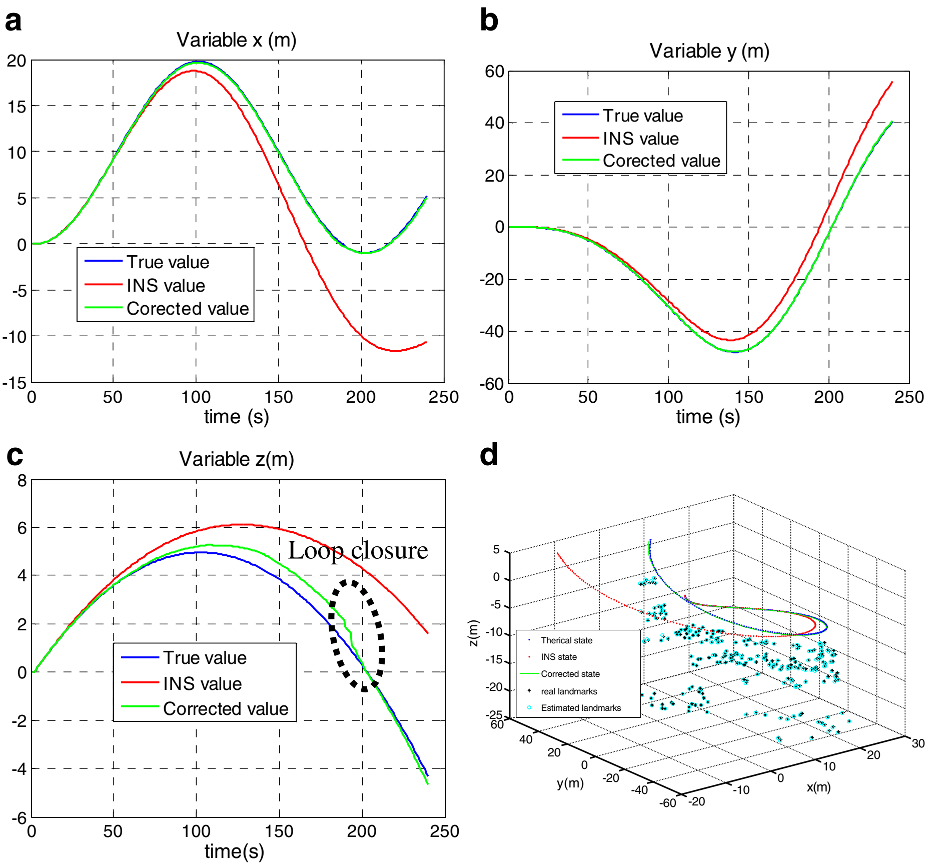


Fig. 9 True, INS and corrected UAV trajectory by the EKF SLAM in the case of centred Gaussian noise: **a** x estimation, **b** y estimation, **c** z estimation, **d** UAV trajectories with loop closure detection

This last figure shows a scenario of a UAV making a loop. Although the corrected position is much better than the INS position, it still diverges from the true position up to ($t = 200$ s). When a loop-closure is detected, at this time a significant accuracy is observed and the corrected position is a confused with the true position. The loop closing effect appears obviously in Fig. 9c (dashed ellipse).

When, the process and observation noises are non Gaussian, the performance of the EKF estimator decreases significantly as shown in Fig. 10. In this case, the corrected values of x and y are still close to their corresponding true values. This can be explained by the efficiency of the stereo cameras to observe the bearing information nevertheless the range information (z) is given with less accuracy as in Fig. 10c. As shown in this figure, the value of z diverges, as the INS information, before it starts slightly converging to the true value at $t = 150$ s. At $t = 200$ s, which corresponds to the detection of the loop closing, we observe the full convergence of the corrected values of z to their corresponding true values.

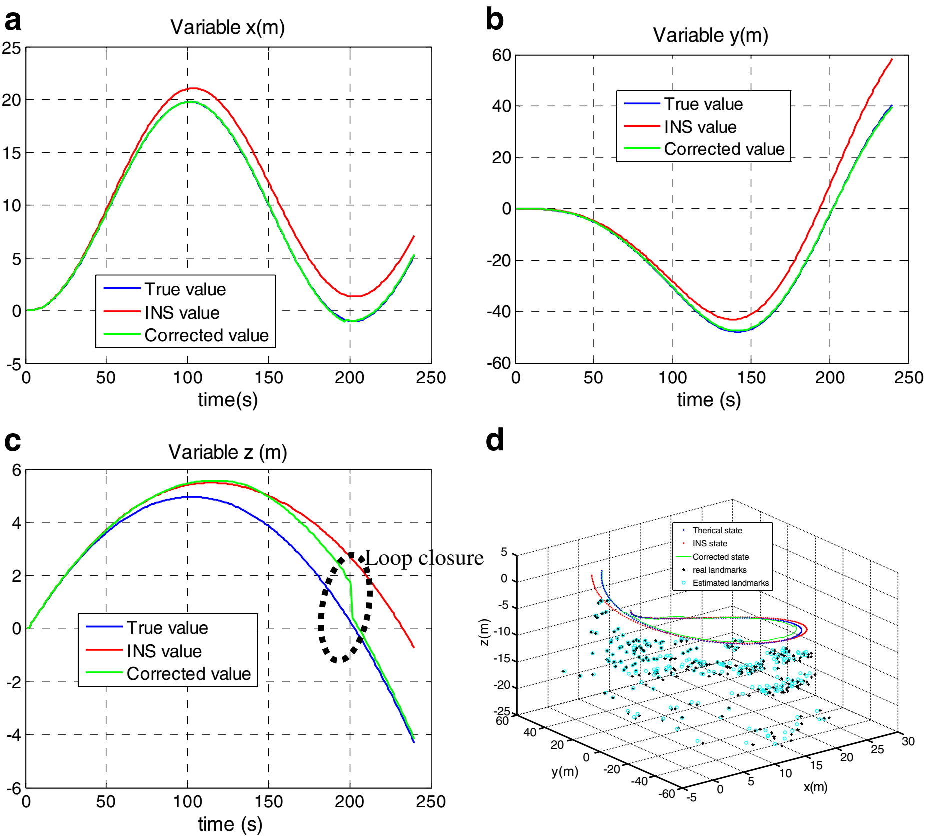


Fig. 10 True, INS and corrected UAV trajectory by the EKF SLAM in the case of centred non Gaussian noise: **a** x estimation, **b** y estimation, **c** z estimation, **d** UAV trajectories with loop closure detection

The problem of the EKF SLAM is the poor consistency caused by the linearization problem and the assumed characteristics of the process and observation noises, which should be centred Gaussian noises.

8.2 NH∞ Airborne VSLAM

As explained in Section 6, the NH∞ estimator does not require specific noise characteristics or a known error covariance matrix. From Fig. 11a–d and comparing it to Fig. 9a–d, we can conclude that if both process and observation noises are assumed centred Gaussian noises then the mean covariance estimator (EKF) could lead to similar (in many cases better) performances than the **minmax** estimator (NH∞). In contrast, the landmarks estimated by the robust NH∞ filter are more accurate than those estimated by the EKF, as can be observed from Table 3. The landmark error obtained by the NH∞ filter is very small compared with that obtained by the EKF filter. This could be explained by the linearization of the highly nonlinear observation model for which the EKF neglects the high order terms.

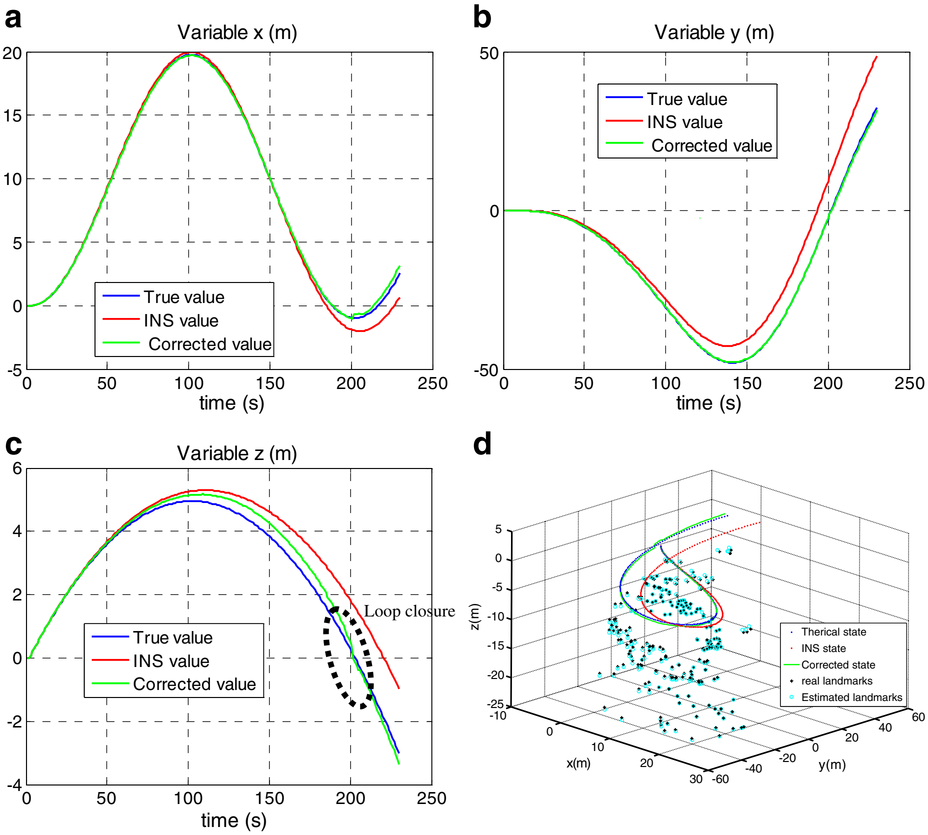


Fig. 11 True, INS and corrected UAV trajectory by the NH∞ SLAM in the case of centred Gaussian noise: **a** x estimation, **b** y estimation, **c** z estimation, **d** UAV trajectories with loop closure detection

When the assumed characteristics of noises do not hold, which is usually the case for navigation and visual data, the NH∞ filter gives much better results than the EKF. The latter conclusion can be validated by comparing Fig. 10c, d with Fig. 12c, d. From Fig. 12d, it is clear that the estimated trajectory given by the NH∞ is more accurate than that obtained by the EKF, Fig. 10d. At $t = 200$ s a loop-closure is detected and at this time a significant improvement of the accuracy is observed especially in Fig. 12c (dashed ellipse). Furthermore, the landmarks estimated by the robust NH∞ filter, for non centred Gaussian noises, are more accurate than those estimated by the EKF

Table 3 Comparison between NH∞ and EKF

Landmarks error	x-error		y-error		z-error	
	Mean	Std	Mean	Std	Mean	Std
NH∞	0.0047	0.1312	0.0059	0.2221	0.1695	0.3832
EKF	0.1861	0.2578	0.1580	0.2207	0.2929	0.3320

Landmarks accuracy estimation

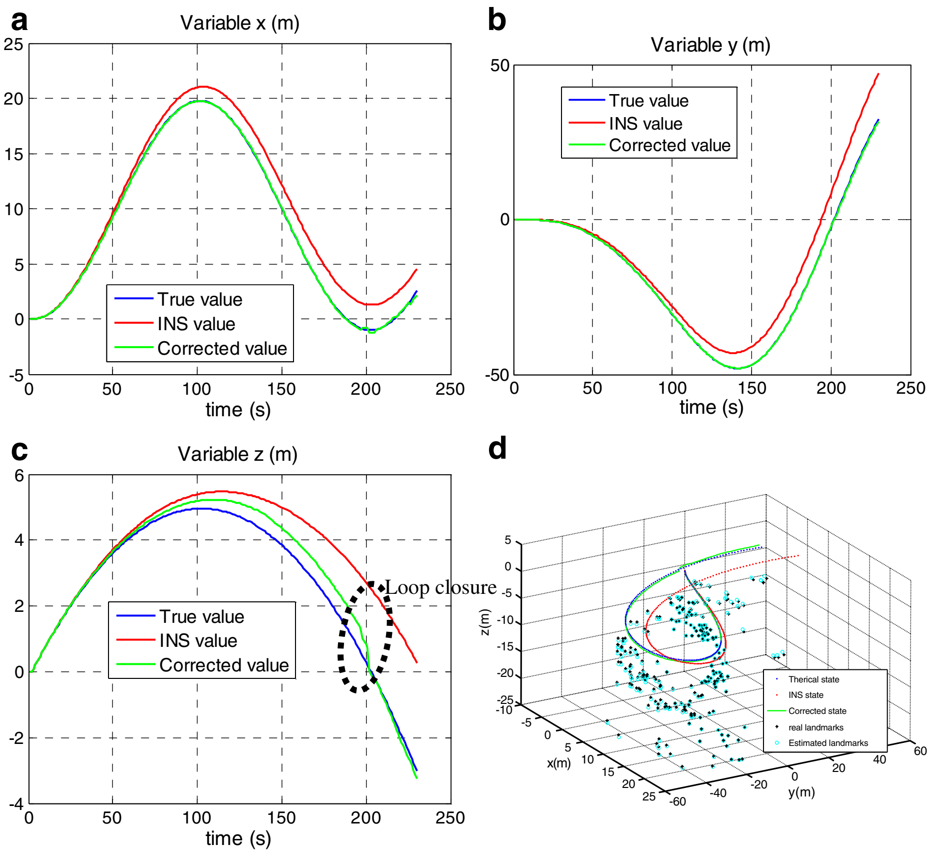


Fig. 12 True, INS and corrected UAV trajectory by the NH∞ SLAM in the case of centred non Gaussian noise: **a** x estimation, **b** y estimation, **c** z estimation, **d** UAV trajectories with loop closure detection

as shown in Table 4. As can be seen from this table, the landmark position accuracy given by the NH∞ filter decreases compared to Table 3. However, it is still much better than that obtained by the EKF filter, for which the landmark position error raise up above one meter, This is a large error taking into consideration the short navigation time (≈200 s).

Figure 13 presents the NH∞ and EKF Airborne VSLAM results for a highly nonlinear trajectory. The NH∞ filter shows more robustness and performs, as

Table 4 Comparison between NH∞ and EKF

Landmarks error	x-error		y-error		z-error	
	mean	std	mean	std	mean	std
NH∞	0.4692	0.6590	0.3832	0.6850	0.2686	0.5256
EKF	0.5584	1.1404	1.0768	1.3247	0.5998	0.7492

Landmarks accuracy estimation, with non-centred Gaussian noise

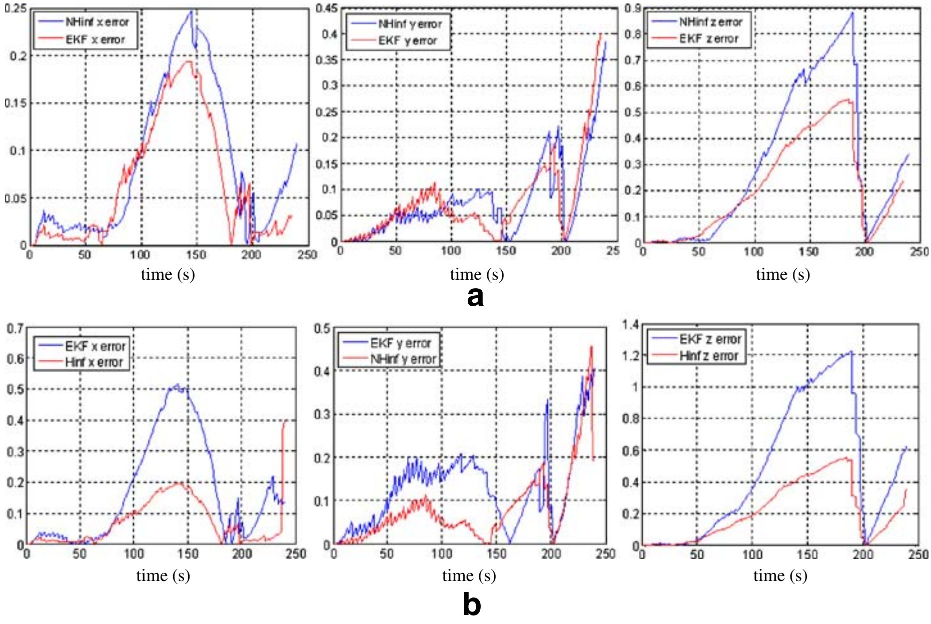
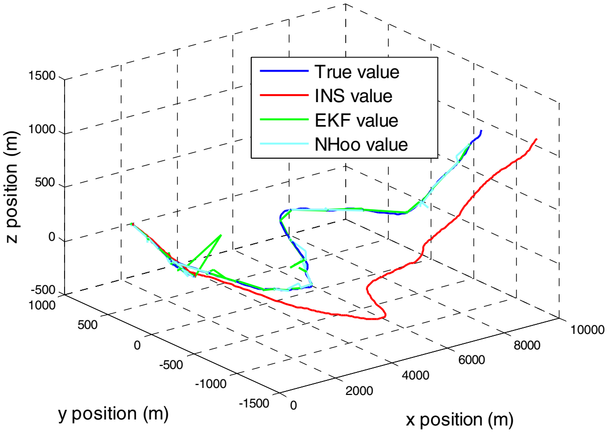


Fig. 13 Absolute error estimation by EKF and NH∞ **a** Gaussian noise case **b** Non Gaussian noise case

expected, much better than the EKF filter (Fig. 14). This can be explained by fact that the NH∞ filter takes into consideration all the high order terms of the Taylor expansion (these term should be bounded), while the EKF filter neglects these terms.

Figure 13 presents *x*, *y* and *z* UAV position errors in Gaussian and non Gaussian noise cases using the EKF and NH∞ filters. Confirming the results of the previous section, if the noise is a centred white Gaussian, the EKF may perform better than

Fig. 14 EKF and NH∞ estimator for a highly nonlinear trajectory



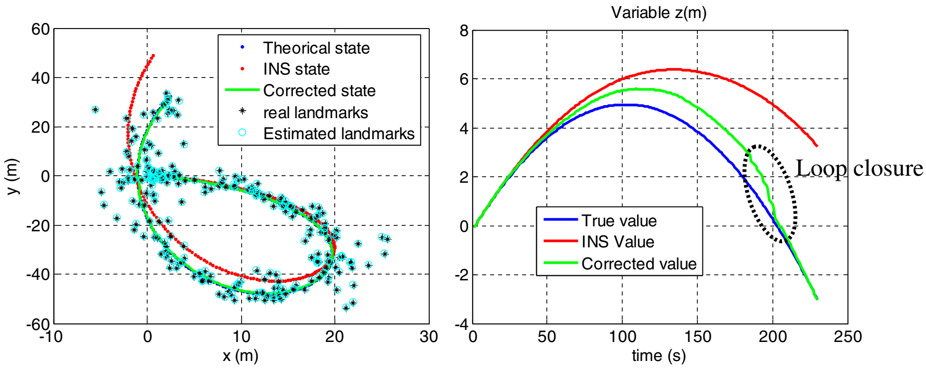


Fig. 15 3DVSLAM Map management, approach of the k-nearest landmarks. Loop closure detection despite the limited size of the state vector ($x_m \leq 24$ landmarks)

the NH∞ filter (Fig. 13a). The advantage of the NH∞ can be observed especially when the noise is non centred or non Gaussian (Fig. 13b).

8.3 Map Management Results

Figures 15–17 show the results obtained by the proposed map management approach (k – nearest landmarks). As can be seen from Fig. 15, the loop closure is correctly detected at $t = 200$ s despite the fact that the size of the state vector is limited to (k + number of new landmarks). This implies a significant decrease in the complexity of the algorithm since we choose $k = 20$ with an average of 4 new landmarks observed at each time. Figure 16 shows the results obtained by the Stochastic Map (SM) approach. From this figure, we remark that the loop closure is also detected on time but the size of the state vector increases along with the observation of new landmarks. In this case 232 landmarks were observed, which means that the size of state vector is equal to $9_{\text{uavstates}} + 232 \times 3_{\text{landmarkstates}} = 705$ states after only 4 min of navigation. This is very heavy and impractical compared to our map management

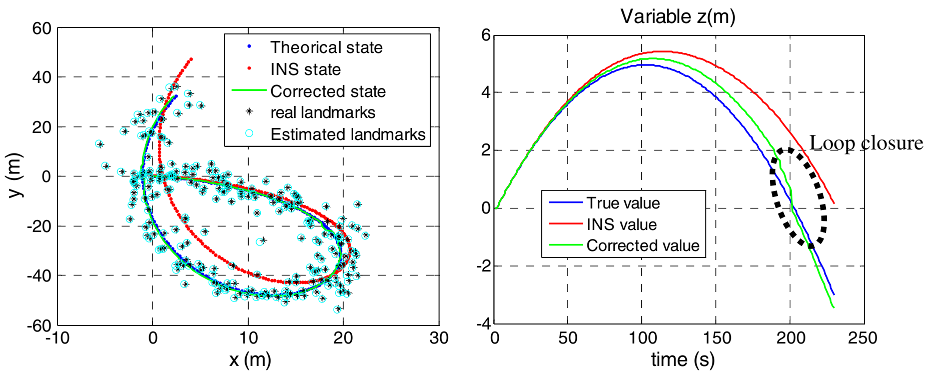


Fig. 16 3DVSLAM Map management, augmented map. Loop closure detection, (232 landmarks)

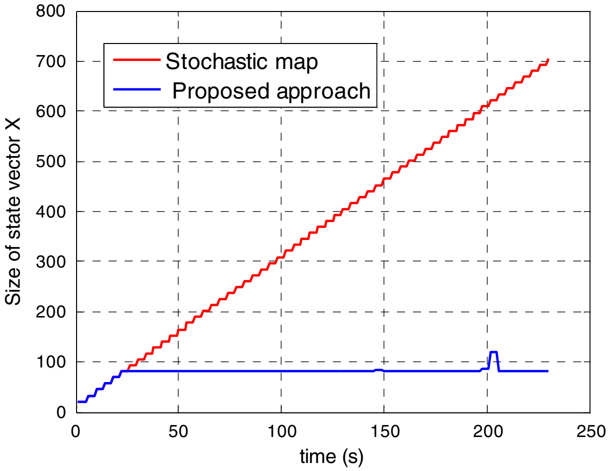


Fig. 17 Complexity comparison between classical stochastic map and the proposed *k*-nearest landmarks map

solution where the size of state vector is equal to $9_{\text{uavstates}} + 24 \times 3_{\text{landmarkstates}} = 81$ states only. Furthermore, this state vector size stays small and fixed independently of the navigation time. This improvement can be clearly seen in Fig. 17 where the size of the state vector is shown increasing linearly with time in the SM approach while it keeps quasi constant in our approach.

9 Conclusion

In this paper, we proposed a new approach to solve the Airborne VSLAM problem. It is based on the development of a full observation model of an IMU/Stereo cameras embedded on a UAV. Contributions in this paper range from the observability and consistency analyses of the EKF Airborne VSLAM to the proposition of a robust version of an Airborne VSLAM using a new map management approach (nearest $k-$landmarks) reducing algorithm complexity and based on the Nonlinear H∞ filter. The latter is compared favourably with the EKF VSLAM and accurate UAV positions are estimated when no assumptions are made either on the process model or on the noise characteristics.

References

1. Dissanayake, M.W.M.G., Newman, P., Clark, S., Durrant-Whyte, H.F.: A solution to the simultaneous localization and map building (SLAM) problem. IEEE Trans. Robot. Autom. **17**(3), 229–241 (2001), Jun
2. Smith, R., Cheeseman, P.: On the representation and estimation of spatial uncertainty. Int. J. Rob. Res. **5**(4), 56–68 (1986)
3. Moutarlier, P., Chatila, R.: Stochastic multisensory data fusion for mobile robot location and environment modeling. In: Proc. 5th Int Symp. Robot. Research. Tokyo, Japan (1989)
4. Davison, A.J.: Mobile robot navigation using active vision. Ph.D. dissertation, Univ. Oxford, Oxford, U.K. (1998)
5. Bailey, T.: Mobile robot localization and mapping in extensive outdoor environments. Ph.D. dissertation, Univ. Sydney, Sydney, NSW, Australia (2002)

6. Davison, A.J., Murray, D.W.: Simultaneous localization and map building using active vision. IEEE Trans. Pattern Anal. Mach. Intel. **24**(7), 865–880 (2002), Jul
7. Guivant, J., Nebot, E.: Optimization of the simultaneous localization and map-building algorithm and real-time implementation. IEEE Trans. Robot. Autom. **17**(3), 242–257 (2001), Jun
8. Montemerlo, M., Thrun, S., Koller, D., Wegbreit, B.: FastSLAM 2.0: an improved particle filtering algorithm for simultaneous localization and mapping that provably converges. In: Proc. 18th Int. Joint Conf. Artif. Intell. Acapulco, Mexico (2003)
9. Grisetti, G., Stachniss, C., Burgard, W.: Improving grid-based SLAM with Rao-Blackwellized particle filters by adaptive proposals and selective resampling. In: Proc. IEEE Int. Conf. Robot. Automat., pp. 2443–2448. Barcelona, Spain (2005)
10. Williams, S., Dissanayake, G., Durrant-Whyte, H.F.: Towards terrain-aided navigation for underwater robotics. Adv. Robot. **15**(5), 533–550 (2001)
11. Thrun, S., Ahnel, D.H., Ferguson, D., Montemerlo, M., Triebel, R., Burgard, W., Baker, C., Omohundro, Z., Thayer, S., Whittaker, W.: A system for volumetric robotic mapping of abandoned mines. In: Proc. IEEE Int. Conf. Robot. Automat, pp. 4270–4275. Taipei, Taiwan, May (2003)
12. Bryson, M., Sukkarieh, S.: Building a robust implementation of bearing-only inertial SLAM for a UAV. ARC Report Centre of Excellence in Autonomous Systems Australian Centre for Field Robotics. University of Sydney, NSW, Australia (2006)
13. Davison, A.J., Murray, D.W.: Simultaneous localization and map building using active vision. IEEE Trans. Pattern Anal. Mach. Intell. **24**(7), 865–880 (2002)
14. Davison, A.J.: Real-time simultaneous localisation and mapping with a single camera. In: Proceedings of the 9th International Conference on Computer Vision, pp. 1403–1410 (2003)
15. Nir, T., Bruckstein, A.M.: Causal camera motion estimation by condensation and robust statistics distance measures. In: Proceedings of the 8th European Conference on Computer Vision (2004)
16. Thrun, S., Liu, Y.: Multi-robot SLAM with sparse extended information filters. In: Proceedings of the 11th International Symposium of Robotics Research (2003)
17. Hajjdiab, H., Laganiere, R.: Vision-based Multi-Robot Simultaneous Localization and Mapping. In: Proceedings of the First Canadian Conference on Computer and Robot Vision, pp. 155–162 (2004)
18. Fenwick, J.W., Newman, P.M., Leonard, J.J.: Cooperative concurrent mapping and localization. In: Proc. IEEE Conf. Robot. Automat., pp. 1810–1817. Washington, DC, May (2002)
19. Kim, S.J.: Efficient simultaneous localization and mapping algorithms using submap networks. Ph.D. Dissertation, Dept. Ocean. Eng., Mass. Inst. Technol., Cambridge, May (2004)
20. Julier, S.J., Uhlmann, J.K.: A counter example for the theory of simultaneous localization and map building. In: Proc. IEEE Conf. Robot. Automat., pp. 4238–4243. Seoul, Korea, May 21–26 (2001)
21. Castellanos, J.A., Neira, J., Tardos, J.D.: Limits to the consistency of EKF-based SLAM. In: 5th IFAC Symp. Intell. Autonom. Veh., IAV'04. Lisbon, Portugal, Jul. (2004)
22. Huang, S., Dissanayake, G.: Convergence and consistency analysis for extended Kalman filter based SLAM. IEEE Trans. Robot. **23**(5), 1036–1049 (2007)
23. Benmessaoud, M.L., Lamrani, A., Nemra, K., Souici, A.K.: Single-Camera EKF-vSLAM. In: Proceedings of World Academy of Science, Engineering and Technology PWASET Volume, ISSN 1307-6884, 30 July (2008)
24. Ortega, J.S.: Towards visual localization, mapping and moving objects tracking by a mobile robot: a geometric and probabilistic approach. PhD Dissertation, National Polytechnic Institute of Toulouse (INPT), February (2007)
25. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vis. **60**(2), 91–110 (2004)
26. Smith, R., Self, M., Cheeseman, P.: Estimating uncertain spatial relationships in robotics. In: Cox, I.J., Wilfon, G.T. (eds.) Autonomous Robot Vehicles, pp. 167–193. Springer, New York (1990)
27. Bryson, M., Sukkarieh, S.: Observability analysis and active control for airborne SLAM. IEEE Trans. Aerosp. Electron. Syst. **44**(1), 261–280 (2008)
28. Vidal-Calleja, T., Bryson, M., Sukkarieh, S., Sanfeliu, A., Andrade-Cetto, J.: On the Observability of Bearing-only SLAM. In: IEEE Conference on Robotics and Automation (2007)
29. Shaked, U., Berman, N.: H∞ nonlinear filtering of discrete-time process. IEEE Trans. Signal Process. **43**, 2205–2209 (1995)

30. Petersen, I., Savkin, A.: Robust Kalman Filtering for Signals and Systems with Large Uncertainteis. Birkhauser, Boston (1999)
31. Basar, T., Baernard, P.: H∞ Optimal Control and Related Minimax Design Problems. A Dynamic Game Approach. Systems and Control: Foundations and Applications. Birkhauser (1991)
32. Einicke, G., White, L.: Robust extended kalman filtering. IEEE Trans. Signal Process. **47**(9), 2596–2599 (1999)
33. Castellanos, J.A., Martinez-Cantin, R., Tardos, J.D., Neira, J.: Robocentric map joining: improving the consistency of EKF-SLAM. Robot. Auton. Syst. **55**, 21–29 (2007)
34. Bailey, T., Nieto, J., Guivant, J., Stevens, M., Nebot, E.: Consistency of the EKF-SLAM Algorithm. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (2006)
35. Dissanayake, G., Williams, S., Durrant-Whyte, H.F., Bailey, T.: Map management for efficient Simultaneous Localization and Mapping (SLAM). Auton. Robots **12**, 267–286 (2002), Kluwer Academic Publishers
36. Jensfelt, P., Kragic, D., Folkesson, J., Bjorkman, M.: A framework for vision based bearing only 3D SLAM. In: IEEE International Conference on Robotics and Automation (ICRA) (2006)