AAS 17-073

REDUCED-DYNAMICS POSE ESTIMATION FOR NON-COOPERATIVE SPACECRAFT RENDEZVOUS USING MONOCULAR VISION

Sumant Sharma,[*] Simone D'Amico[†]

This work addresses the design and validation of a monocular vision based reduced-dynamics pose estimation architecture for close-range on-orbit-servicing and formation-flying applications. The aim is to estimate the pose of a passive space resident object using its known three-dimensional wireframe model and low-resolution two-dimensional images collected on-board the servicer spacecraft. In contrast to previous works, the proposed architecture has the potential to be on-board executable and capable to estimate the pose of the target spacecraft without the use of fiducial markers and without any a-priori range measurements or state information. An innovative pose initialization subsystem based on the filtering of weak image gradients and a Newton-Raphson method is used to identify edges of the target spacecraft and compute an initial pose estimate, even in the presence of the Earth in the background. A novel navigation filter utilizing a multiplicative extended Kalman filter for pose estimation is formulated, validated, and stress-tested. Based on high-fidelity numerical simulations of reference trajectories, the filter is shown to provide pose estimates with centimeter-level relative position error (3D rms) and sub-degree level relative attitude error (3D rms).

INTRODUCTION

Recent advancements have been made to utilize monocular vision navigation as an enabling technology for formation-flying and on-orbit servicing missions (e.g., PROBA-3 by ESA,[1] ANGELS by US Air Force,[2] PRISMA by OHB Sweden[3]). Monocular navigation on such missions relies on a robust and real-time estimation of the position and orientation, together denoted as the pose, of the space resident object with respect to the camera. Prior demonstrations of close-range pose estimation have utilized known visual markers[4] and a-priori knowledge of the pose.[5–7]However, the presence of such markers and initial pose knowledge cannot be relied on during on-orbit servicing missions as the space resident object might be non-functional and freely tumbling with lost attitude control. To overcome these challenges, traditional computer vision algorithms, typically used in terrestrial applications, coupled with an Extended Kalman Filter (EKF) have been proposed.[8] However, the performance of the estimation filter and the convergence of the pose estimates are highly sensitive to the tuning of the filter and the image processing parameters.[9] Additionally, image processing algorithms typically used in terrestrial applications[10,11] require significant computational effort making them unsuitable for spaceborne hardware. This paper presents an architecture for the

[*]Doctoral Candidate, Department of Aeronautics & Astronautics, 496 Lomita Mall, Stanford University, Stanford, California, USA.

[†]Assistant Professor, Department of Aeronautics & Astronautics, 496 Lomita Mall, Stanford University, Stanford, California, USA.

robust pose estimation and tracking of a space resident object that has the potential to be run in real-time on-board a servicer spacecraft. Edge features detected from the 2D images from a single monocular camera, in conjunction with a 3D computer model, are leveraged within a novel navigation filter which utilizes a Multiplicative Extended Kalman Filter (MEKF) for pose estimation. The architecture does not require any training phase or a-priori knowledge of the pose. The pose initialization subsystem improves upon the state-of-the-art[9,12–14] by introducing a hybrid approach to image processing by fusing the weak gradient elimination technique with the Sobel operator followed by Hough transform[15] to detect both small and large features of the target spacecraft. The navigation filter improves the state-of-the-art[16–18] by utilizing the Relative Orbital Elements (ROE) parametrization of the state which allows the use of an elegant state-transition-matrix to propagate the relative position dynamics for a servicer orbit of an arbitrary eccentricity. The filter also relies on the pseudo-measurements of a normal vector generated from the pixel locations of the target spacecraft's edges, thereby making the filter robust to the detection of partial edges in the image.

Edge features, in contrast to traditionally used point features, are selected as they are less sensitive to illumination changes and can easily distinguish the boundaries of spacecraft geometry from the background in an image. Moreover, edges contain structural information which allows for drastic search space reduction when determining the feature correspondences required for calculating the initial pose estimate.[19] The pose initialization subsystem, which provides an initial pose estimate to the navigation filter, is validated using actual images collected in the frame of the PRISMA mission at about 700 [km] altitude and 10 [m] inter-spacecraft separation.[20] Accuracy of the initial pose estimates is quantified by comparisons with independent flight dynamics operational products. A detailed analysis of the stability and sensitivity of the navigation filter to measurement noise, measurement update interval, and initial pose estimate, is also provided. A reduced-dynamics model capturing the evolution of the object's relative position and velocity with respect to the camera is used for the time update of the state. This allows for the architecture to provide a reliable pose solution even at low image acquisition rates or during short data blackouts (e.g., if the camera is temporarily blinded by the Sun, or in eclipse). The remainder of the paper begins by introducing an innovative subsystem for pose initialization which is followed by the development of the navigation filter. Both the pose initialization subsystem and the navigation filter are validated and stress-tested in the subsequent section. Finally, the paper discusses the lessons learned and the contributions to the state-of-the-art, as well as a proposal of the way forward.

POSE INITIALIZATION

Navigation filters based on noncooperative monocular vision rely on finding an estimate of the initial pose, i.e., the attitude and position of the target spacecraft with respect to the camera, based on a minimum number of features from a 3D wire-frame model of the target spacecraft and a single two dimensional image. Estimation of the initial pose is especially challenging as there is no a-priori information about the attitude and position of the target. Aside from a 3D wireframe model, no assumption on the relative translational or rotational information is made. This is quite conservative as usually a-priori information is available from prior phases of the flight, however, the "lost-in-space" capability that this subsystem provides is helpful in increasing robustness of a relative navigation system.

In particular, the pose initialization problem consists of determining the position of the target's center of mass, ${}^{\mathcal{C}}\boldsymbol{T}$, and the orientation, $\boldsymbol{R}_{\mathcal{TC}}$, of the target's principal axes, with respect to the camera frame $\mathcal{C}$, given a single monocular image. The 3D model of the target is defined in the

body-fixed coordinate system, $\mathcal{T}$, which is defined to be aligned with the target spacecraft's principal axes. $\boldsymbol{R}_{\mathcal{TC}}$ is the direction cosine matrix rotating the frame $\mathcal{T}$ to $\mathcal{C}$. Let ${}^{\mathcal{T}}\boldsymbol{P}$ be a point of the 3D model expressed in the coordinate system $\mathcal{T}$. By employing the standard pinhole camera model,[21] the corresponding point $\boldsymbol{p} = [u\ v]^T$ in the rectified image can be obtained using the 3D-2D true perspective projection equation

$$ {}^{\mathcal{C}}\boldsymbol{r} = \left[{}^{\mathcal{C}}x\ {}^{\mathcal{C}}y\ {}^{\mathcal{C}}z\right]^T = \boldsymbol{R}_{\mathcal{TC}}\,{}^{\mathcal{T}}\boldsymbol{P} + {}^{\mathcal{C}}\boldsymbol{T} \tag{1} $$

$$ \boldsymbol{p} = \left[\frac{{}^{\mathcal{C}}x}{{}^{\mathcal{C}}z}f_x + C_x,\ \frac{{}^{\mathcal{C}}y}{{}^{\mathcal{C}}z}f_y + C_y\right] \tag{2} $$

where ${}^{\mathcal{C}}\boldsymbol{r}$ represents a point on the target spacecraft expressed in the camera frame $\mathcal{C}$ according to the current pose ${}^{\mathcal{C}}\boldsymbol{T}$ and $\boldsymbol{R}_{\mathcal{TC}}$. The focal lengths of the camera are denoted by f_x and f_y while (C_x, C_y) is the principal point of the image. Without loss of generality, it is assumed that the direction $\mathcal{C}_3$ is pointed along the boresight of the camera and that the directions $\mathcal{C}_1$ and $\mathcal{C}_2$ of the camera frame, $\mathcal{C}$, are aligned with the directions $\mathcal{P}_1$ and $\mathcal{P}_2$ of the image frame, $\mathcal{P}$, respectively. The system of perspective projection equations given by Eq. (1) and Eq. (2) has six unknown coefficients due to the three components of the target's position ${}^{\mathcal{C}}\boldsymbol{T}$ and the three parameters that define the rotation matrix $\boldsymbol{R}^{\mathcal{TC}}$ of the target orientation. To solve the perspective projection equations, at least three image points and corresponding model points are required. However, at least six correspondences between image and model points are required to obtain a unique solution with a general configuration of points.[22]

In order to estimate the initial pose, the architecture proposed by Sharma et. al.[23] is used in this paper. This architecture is illustrated in Figure 2. The key feature of this architecture is that the initial pose is uniquely determined by solving the perspective equations using the most significant features detected in the input monocular image and the corresponding significant features present in the 3D model of the same vehicle. Accordingly, the pose initialization subsystem consists of two main routines:

1. Image processing: As shown in Figure 2, this routine accepts as input a single 2D image. The goal of the image processing is to detect the most significant features of the target spacecraft in the input image by extracting edges in the image. With respect to the off-the-shelf feature detection techniques, the key innovation of this routine is the fusion of a weak gradient elimination technique in conjunction with the state-of-the-art edge detection techniques.[15,24] The resulting routine not only finds the edges corresponding to the large features of the spacecraft but it also detects edges corresponding to smaller features such as antennae. Unlike previous work,[9] this allows the determination of a pose for a satellite with highly symmetric visual features. The detected edges are then grouped into local geometric constellations to dramatically reduce the search space required for determining the correspondence of pixel points with points available on the 3D model.

2. Pose determination: As shown in Figure 2, this routine accepts as input the feature groups detected in the image processing routine and the 3D model. It begins by pairing the 2D and 3D geometric groups to create multiple correspondence hypotheses. For each hypothesis, the end-points of the line segments forming the geometric groups are used in solving the perspective equations. Of the multiple resulting pose solutions, the five solutions with the least reprojection error are iteratively refined using the Newton Raphson Method and the best is selected using the reprojection error as the metric.[25]

NAVIGATION FILTER

Theoretically, the pose initialization subsystem can be run for each image to produce a pose estimate, however, it is not feasible to run the constituent computationally expensive routines at high bandwidth on processors typical in spaceborne applications. Therefore, it is imperative that the following simple yet accurate navigation filter be used to produce pose estimates at a high frequency. The state of our system is represented by the position, orientation, translational and angular velocity of the target spacecraft with respect to the servicer spacecraft. An elementary state parametrization includes the relative position and velocity of the target spacecraft defined in a rotating reference frame located at the center of mass of the servicer spacecraft. In this frame, the x-axis is aligned with the zenith (R) direction, the z-axis is aligned with the angular momentum of the servicer's orbit (N) direction, and the y-axis is aligned with the servicer's flight (T) direction. The rectilinear state defined in this right-handed RTN triad, $\delta\boldsymbol{x}_{\text{RTN}}$, is given by

$$\delta\boldsymbol{x}_{\text{RTN}} = (\rho_{\text{RTN}}\ \dot{\rho}_{\text{RTN}})^T = (\delta x\ \delta y\ \delta z\ \delta\dot{x}\ \delta\dot{y}\ \delta\dot{z})^T \tag{3}$$

where δx, δy, and δz, denote the R, T, and N components of the relative position vector. Instead of using relative position and velocity, the state can also be parametrized as a function of the absolute orbital elements of the spacecraft. This paper uses the Relative Orbital Elements (ROE) state defined by D'Amico,[26] which is given by the following difference of quasi-nonsingular orbital elements

$$\delta\boldsymbol{x}_{\text{ROE}} = \begin{bmatrix}\delta a\\ \delta\lambda\\ \delta\boldsymbol{e}\\ \delta\boldsymbol{i}\end{bmatrix}^T = \begin{bmatrix}\delta a\\ \delta\lambda\\ \delta e_x\\ \delta e_i\\ \delta i_x\\ \delta i_y\end{bmatrix}^T = \begin{bmatrix}(a_t - a_s)/a_s\\ (M_t - M_s) + (\omega_t - \omega_s) + c_{i_s}(\Omega_t - \Omega_s)\\ e_t c_{\omega_t} - e_s c_{\omega_s}\\ e_t s_{\omega_t} - e_s s_{\omega_s}\\ i_t - i_s\\ s_{i_s}(\Omega_t - \Omega_s)\end{bmatrix}^T \tag{4}$$

where a, e, i, Ω, ω, and M are the classical Keplerian orbital elements. In the ROE parametrization, δa is the relative semi-major axis, $\delta\lambda$ is the relative mean longitude, δe is the relative eccentricity vector, and δi is the relative inclination vector. In comparison to the rectilinear RTN state, it is preferable to use the ROE state as it allows for an improved modelling of dynamics in closed-form. Moreover, five of the six elements in the ROE state are static for the case of unperturbed motion and only slowly varying in the case of perturbed motion. To represent the relative orientation, this paper uses the 1×4 unit quaternion, $\boldsymbol{q}$. More precisely, $\boldsymbol{q}$ represents the relative orientation of the target spacecraft's principal axes with respect to the servicer spacecraft's principal axes. Finally, the angular velocity is represented in the state as the 1×3 vector $\boldsymbol{\omega}$, which is the rate of change of the relative orientation of the target's principal axes with respect to the servicer's principal axes. Therefore, the state of the system is the following 13×1 vector

$$\boldsymbol{x} = [\delta\boldsymbol{x}_{\text{ROE}}\ \boldsymbol{q}\ \boldsymbol{\omega}]^T \tag{5}$$

The objective of the close-range navigation problem is the estimation of the motion of the target with respect to the servicer spacecraft from a set of line correspondences between the 3D model of the target spacecraft and the measured line segments in the image. A rigorous formulation of this problem requires a dynamics model and a measurement model. The dynamics model describes how the state evolves over time. This paper only considers a linear dynamics model due to their simplicity and suitability for on-board implementation on spacecraft. Assuming that the dynamics of

the linear relative position and velocity are independent of the relative attitude and angular velocity, we can derive and combine their respective State Transition Matrices (STM) to get a single STM to propagate the state, $\boldsymbol{x}$. The dynamic models considered are of the form

$$\boldsymbol{x}(t+\tau) = \boldsymbol{\Phi}(t,\tau)\boldsymbol{x}(t) \tag{6}$$

where $\boldsymbol{x}(t)$ is the state at time t, which is related to the state at an arbitrary time $t+\tau$ by the STM, $\boldsymbol{\Phi}(t,\tau)$. For the propagation of the relative orbital elements in unperturbed Keplerian orbits of arbitrary eccentricity, the STM is given by D'Amico[26] as

$$\boldsymbol{\Phi}_{\text{ROE}}(t,\tau) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ -1.5n\tau & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{7}$$

where n is the mean motion of the servicer spacecraft. In comparison to the state-of-the-art in relative translational state propagation (for example, the Yamanaka-Ankersen STM[27]), the STM given by Eq. (7) is superior as the linearization only occurs for one of the six elements of the state rather than the entire state. The assumption of unperturbed motion is justified as this paper considers close-range rendezvous applications with short measurement update intervals. For the relative attitude dynamics, the filter uses a constant relative angular velocity model. The model used for attitude dynamics assumes unperturbed rotational motion as well as small angular velocities.

$$\dot{\boldsymbol{q}} = \frac{1}{2}\boldsymbol{\Omega}(\omega)\boldsymbol{q} = \frac{1}{2}\begin{bmatrix}\omega \\ 0\end{bmatrix} \otimes \boldsymbol{q} \tag{8}$$

$$\dot{\boldsymbol{\omega}} = \mathbf{0} \tag{9}$$

However, there are problems with using the unit quaternion as a state parametrization in the navigation filter since the unit quaternion has four parameters, one constraint equation, and three degrees of freedom. Therefore, one of the elements of the unit quaternion is deterministic instead of stochastic resulting in a rank-deficient covariance matrix of the quaternion due to a null eigenvalue.[28] In the case of numerical error, this could lead to the divergence of the navigation filter as the entire covariance matrix may become non-positive definite. As a remedy to this problem, Lefferts et al.[29] suggests using the Multiplicative Extended Kalman Filter (MEKF). The underlying principle is to account for the relative attitude by keeping track of the full reference unit quaternion $\boldsymbol{q}_{\text{ref}}$ and a three-element error vector $\boldsymbol{a}_{\text{p}}$. This paper follows the approach of using the Modified Rodrigues Parameters (MRP) for $\boldsymbol{a}_{\text{p}}$ as suggested by Tweddle et al.[17] The error vector, $\boldsymbol{a}_{\text{p}}$, is used in propagation and measurement update step followed by a final reset step to "correct" the value of $\boldsymbol{q}_{\text{ref}}$ with the posterior value of $\boldsymbol{a}_{\text{p}}$. This "corrected" value of $\boldsymbol{q}_{\text{ref}}$ is reported as $\boldsymbol{q}$ at each time step. The value of $\boldsymbol{a}_{\text{p}}$ is reset to zero for the next iteration. Due to this reset step after the propagation and measurement update step, the value of $\boldsymbol{a}_{\text{p}}$ is maintained far away from the singularity of parameterizing a 360 degree rotation using the MRP. Therefore, the state vector parametrization is modified inside the navigation filter as

$$\tilde{\boldsymbol{x}} = \begin{bmatrix}\delta\boldsymbol{x}_{\text{ROE}} & \boldsymbol{a}_{\text{p}} & \boldsymbol{\omega}\end{bmatrix}^T \tag{10}$$

To obtain the STM for $\boldsymbol{a}_{\text{p}}$, the time derivative $\dot{\boldsymbol{a}}_{\text{p}}$ must be found in terms of $\boldsymbol{a}_{\text{p}}$ and ω. Note that $\boldsymbol{a}_{\text{p}}$ can be calculated using the corresponding unit quaternion

$$\boldsymbol{a}_{\text{p}} = \frac{4}{1+q_4}\bar{\boldsymbol{q}} \tag{11}$$

where $\bar{\boldsymbol{q}}$ is the vector portion of the unit quaternion and q_0 is the scalar component. Therefore, the derivative of $\boldsymbol{a}_\mathrm{p}$ can be written as

$$\dot{\boldsymbol{a}}_\mathrm{p} = \frac{4}{1+q_0}\dot{\bar{\boldsymbol{q}}} - \frac{4}{(1+q_0)^2}\dot{q_0}\bar{\boldsymbol{q}} \tag{12}$$

$$= \left(-\frac{1}{2}[\omega\times] + \frac{1}{8}\omega\cdot\boldsymbol{a}_\mathrm{p}\right)\boldsymbol{a}_\mathrm{p} + \left(1 - \frac{1}{16}\boldsymbol{a}_\mathrm{p}^T\boldsymbol{a}_\mathrm{p}\right)\omega \tag{13}$$

$$= -\frac{1}{2}[\omega\times]\boldsymbol{a}_\mathrm{p} + \omega \tag{14}$$

The attitude dynamics can now be written as

$$\begin{bmatrix}\dot{\boldsymbol{a}}_\mathrm{p}\\ \dot{\omega}\end{bmatrix} = \begin{bmatrix}-\frac{1}{2}[\omega\times] & \boldsymbol{I}_{3\times3}\\ 0_{3\times3} & 0_{3\times3}\end{bmatrix}\begin{bmatrix}\boldsymbol{a}_\mathrm{p}\\ \omega\end{bmatrix} \tag{15}$$

Now the STM for the propagation of $\boldsymbol{a}_\mathrm{p}$ can be derived as

$$\boldsymbol{\Phi}_\mathrm{att}(t,\tau) = \begin{bmatrix}-\boldsymbol{e}^{-\frac{1}{2}[\omega\times]\tau} & \int_0^\tau \boldsymbol{e}^{-\frac{1}{2}[\omega\times]\tau'}\mathrm{d}\tau'\\ 0_{3\times3} & \boldsymbol{I}_{3\times3}\end{bmatrix} \tag{16}$$

The integral in the preceding equation can be computed numerically using the matrix exponential at each iteration as shown by Tweddle et al.[17] The STM for the complete 12×1 state used in the filter $\tilde{\boldsymbol{x}}$ can now be written as

$$\boldsymbol{\Phi}(t,\tau) = \begin{bmatrix}\boldsymbol{\Phi}_\mathrm{ROE}(t,\tau) & 0_{6\times6}\\ 0_{6\times6} & \boldsymbol{\Phi}_\mathrm{att}(t,\tau)\end{bmatrix} \tag{17}$$

The relative position of the target spacecraft, and thus the modeled line segments in the image of the target spacecraft, can be computed from the state using simple transformations and the perspective equations (Eq. (1) and Eq. (2)). To obtain the measurements for the navigation from the image, consider a pin-hole camera model where the intrinsic camera parameters are assumed to be known. The line segments in the 3D model of the target are known with respect to the target reference framework, $\mathcal{T}$. Without loss of generality, $\mathcal{T}$ is assumed to originate at the center of mass of the target spacecraft with its directions aligned with the principal axes of the spacecraft. Let L_i be a line segment represented with the Cartesian coordinates of its two end-points P_1^i and P_2^i. These points can be expressed in the camera reference frame, $\mathcal{C}$. For instance, ${}^{\mathcal{C}}P_1^i$ can be written as

$${}^{\mathcal{C}}P_1^i = \boldsymbol{q}_{\mathcal{T}/\mathcal{C}}P_1^i\boldsymbol{q}_{\mathcal{T}/\mathcal{C}}^{-1} + {}^{\mathcal{C}}T_{\mathcal{T}/\mathcal{C}} \tag{18}$$

$${}^{\mathcal{C}}P_2^i = \boldsymbol{q}_{\mathcal{T}/\mathcal{C}}P_2^i\boldsymbol{q}_{\mathcal{T}/\mathcal{C}}^{-1} + {}^{\mathcal{C}}T_{\mathcal{T}/\mathcal{C}} \tag{19}$$

where the unit quaternion $\boldsymbol{q}_{\mathcal{T}/\mathcal{C}}$ is used to transform vectors in $\mathcal{T}$ to $\mathcal{C}$. ${}^{\mathcal{C}}T_{\mathcal{T}/\mathcal{C}}$ is the vector from the origin of $\mathcal{T}$ to the origin of $\mathcal{C}$, expressed in $\mathcal{C}$. Further, $\boldsymbol{q}_{\mathcal{T}/\mathcal{C}}$ and ${}^{\mathcal{C}}T_{\mathcal{T}/\mathcal{C}}$ can re-written in terms of the pose of the target with respect to the servicer reference frame, $\mathcal{S}$. The origin of $\mathcal{S}$ is assumed to be at the center of mass of the servicer spacecraft and its directions are aligned with the principal axes of the servicer spacecraft.

$${}^{\mathcal{C}}P_1^i = \boldsymbol{q}_{\mathcal{S}/\mathcal{C}}\boldsymbol{q}P_1^i\left(\boldsymbol{q}_{\mathcal{S}/\mathcal{C}}\boldsymbol{q}\right)^{-1} + {}^{\mathcal{C}}T_{\mathcal{T}/\mathcal{S}} + {}^{\mathcal{C}}T_{\mathcal{S}/\mathcal{C}} \tag{20}$$

$$^{\mathcal{C}}P_2^i = \boldsymbol{q}_{\mathcal{S}/\mathcal{C}}\boldsymbol{q}P_2^i\left(\boldsymbol{q}_{\mathcal{S}/\mathcal{C}}\boldsymbol{q}\right)^{-1} +^{\mathcal{C}} T_{\mathcal{T}/\mathcal{S}} +^{\mathcal{C}} T_{\mathcal{S}/\mathcal{C}} \tag{21}$$

where the unit quaternion $\boldsymbol{q}_{\mathcal{S}/\mathcal{C}}$ is used to transform vectors in $\mathcal{S}$ to $\mathcal{C}$ and the unit quaternion $\boldsymbol{q}$ (part of the state vector, $\boldsymbol{x}$) transforms the vectors from $\mathcal{T}$ to $\mathcal{S}$. $^{\mathcal{C}}T_{\mathcal{T}/\mathcal{S}}$ is the vector from the origin of $\mathcal{T}$ to the origin of $\mathcal{S}$ while $^{\mathcal{C}}T_{\mathcal{S}/\mathcal{C}}$ is the vector from the origin of $\mathcal{S}$ to the origin of $\mathcal{C}$. Finally, $^{\mathcal{C}}T_{\mathcal{T}/\mathcal{S}}$ can be re-written as a function of the ROE state, $\delta\boldsymbol{x}_{\text{ROE}}$, using a linear map, $\boldsymbol{T}(t)$ (see Appendix A), between the ROE state and the rectilinear state defined by the RTN triad.[26] Note that this mapping is valid for orbits of arbitrary eccentricity. The points defining L_i as a function of our state parameters can now be written as

$$^{\mathcal{C}}P_1^i = \boldsymbol{q}_{\mathcal{S}/\mathcal{C}}\boldsymbol{q}P_1^i\left(\boldsymbol{q}_{\mathcal{S}/\mathcal{C}}\boldsymbol{q}\right)^{-1} + \boldsymbol{q}_{\text{RTN}/\mathcal{C}}\boldsymbol{T}(t)\delta\boldsymbol{x}_{\text{ROE}}\boldsymbol{q}_{\text{RTN}/\mathcal{C}}^{-1} +^{\mathcal{C}} T_{\mathcal{S}/\mathcal{C}} \tag{22}$$

$$^{\mathcal{C}}P_2^i = \boldsymbol{q}_{\mathcal{S}/\mathcal{C}}\boldsymbol{q}P_2^i\left(\boldsymbol{q}_{\mathcal{S}/\mathcal{C}}\boldsymbol{q}\right)^{-1} + \boldsymbol{q}_{\text{RTN}/\mathcal{C}}\boldsymbol{T}(t)\delta\boldsymbol{x}_{\text{ROE}}\boldsymbol{q}_{\text{RTN}/\mathcal{C}}^{-1} +^{\mathcal{C}} T_{\mathcal{S}/\mathcal{C}} \tag{23}$$

Since the goal is to estimate the pose using edges detected in the image, a constraint equation can be formed which relates the state vector to the line segments of the 3D model and the corresponding edges detected in the image.[30] As seen in Figure 5, the points $^{\mathcal{C}}P_1^i$, $^{\mathcal{C}}P_2^i$, and the center of projection O are coplanar. Therefore, the unit vector, $^{\mathcal{C}}N_i$ is normal to this projection plane. $^{\mathcal{C}}N_i$ can be expressed as

$$^{\mathcal{C}}N_i = \frac{^{\mathcal{C}}P_1^i \times^{\mathcal{C}} P_2^i}{\left\|^{\mathcal{C}}P_1^i \times^{\mathcal{C}} P_2^i\right\|} \tag{24}$$

Further, a measurement, $^{\mathcal{C}}n_i$, of the normal vector, $^{\mathcal{C}}N_i$, can be obtained from the image. An edge detected in the image which is matched with L_i also belongs to the same projection plane. Let this detected edge be represented by its two end-points $m_1^i = \begin{bmatrix}u_1^i & v_1^i & 1\end{bmatrix}^T$ and $m_2^i = \begin{bmatrix}u_2^i & v_2^i & 1\end{bmatrix}^T$ (defined using homogeneous coordinates in the image plane). These vectors can be rewritten in the camera frame, $\mathcal{C}$ as

$$^{\mathcal{C}}m_1^i = \boldsymbol{K}^{-1}m_1^i \tag{25}$$

$$^{\mathcal{C}}m_2^i = \boldsymbol{K}^{-1}m_2^i \tag{26}$$

where the matrix $\boldsymbol{K}$ contains the internal camera calibration parameters and is assumed to be known. Therefore, $^{\mathcal{C}}n_i$ can be written as

$$^{\mathcal{C}}n_i = \frac{^{\mathcal{C}}m_1^i \times^{\mathcal{C}} m_2^i}{\left\|^{\mathcal{C}}m_1^i \times^{\mathcal{C}} m_2^i\right\|} \tag{27}$$

Note that $^{\mathcal{C}}n_i$ could be formed with any two points on the detected edge and these two points do not have to correspond with the points $^{\mathcal{C}}P_1^i$ and $^{\mathcal{C}}P_2^i$. Hence, this innovative use of edges makes this navigation filter robust to the detection of partial edges during the image processing routine. Combining Eq. (24) and Eq. (27), a measurement equation can be written for each corresponding pair of a line segment in the 3D model and an edge in the image

$$\boldsymbol{z}(t) = h(\boldsymbol{x}(t)) + \boldsymbol{\nu}(t) \tag{28}$$

where $\boldsymbol{z}(t)$ is the pseudo-measurement written by concatenating all instances of $^{\mathcal{C}}n_i$ and $h(\boldsymbol{x}(t))$ is the modelled pseudo-measurement written by concatenating all instances of $^{\mathcal{C}}N_i$. The corresponding noise in the pseudo-measurements based on the measurement covariance matrix $\boldsymbol{R}(t)$ is represented by $\boldsymbol{\nu}(t)$. The Jacobian matrix, $H(t)$, of the function $h(\boldsymbol{x}(t))$ can be computed by differentiating Eq. (24) with respect to each of the elements of the vector, $\tilde{\boldsymbol{x}}(t)$. The Jacobian matrix can be populated at each time step and can be used to write

$$\boldsymbol{z}(t) = \boldsymbol{H}(t)\boldsymbol{x}(t) + \boldsymbol{\nu}(t) \tag{29}$$

Instead of using a computationally expensive matrix measurement update, a sequential scalar measurement update is performed by reducing the vector measurement to a sequence of scalar measurements. The values of $\boldsymbol{\Phi}(t,\tau)$ and $\boldsymbol{H}(t)$ are used to run the navigation filter at each time step and determine the values of propagated state, $\hat{\tilde{\boldsymbol{x}}}(t)^-$, the a-posteriori state, $\hat{\tilde{\boldsymbol{x}}}(t)^+$, the propagated covariance $\boldsymbol{P}(t)^-$, and the a-posteriori covariance, $\boldsymbol{P}(t)^+$, based on the pseudo-measurements $\boldsymbol{z}(t)$. Note that only four 3D-2D line correspondences are needed to estimate the state $\tilde{\boldsymbol{x}}(t)$ but more may be included to increase accuracy. At the end of each measurement update, the a-posteriori state, $\hat{\tilde{\boldsymbol{x}}}(t)^+$, is used to calculate the a-posteriori state, $\hat{\boldsymbol{x}}^+$ (defined in Eq. (5)) by "correcting" the reference quaternion $\boldsymbol{q}_{\text{ref}}$ using the a-posteriori estimate of $\boldsymbol{a}_{\text{p}}$ using quaternion multiplication.

RESULTS

The performance of the pose initialization is assessed on the imagery collected during the PRISMA mission[20] while the performance of the navigation filter is assessed using numerical simulations of three reference trajectories of the Tango spacecraft from the PRISMA mission[20] in Low Earth Orbit (LEO). The numerical simulations allow the authors to thoroughly stress-test the navigation filter by varying measurement noise, measurement update interval, and the error in the a-priori pose estimate. The 3D wire-frame model of the Tango spacecraft used for the performance assessment is illustrated in Figure 6, it was derived from a high fidelity CAD model of the Tango spacecraft. The model consists of a polygon representing the solar panel (560×750 [mm]), a convex polyhedron representing the spacecraft body ($560 \times 550 \times 300$ [mm]). Five additional segments (204 [mm]) represent the radio-frequency antennae. The model is input in MATLAB as a stereo-lithographic file from which information about surfaces, edges, and feature groups is generated.

Pose Initialization

Flight dynamics products from the PRISMA mission[31] have been used for comparison and performance evaluation. Specifically, on-ground precise relative orbit determination based on GPS (accurate to about 2 cm 3D rms) is used as the "true" relative position. The on-board attitude estimate provided by star-trackers aboard the Mango spacecraft and a combination of sun-sensors and magnetometers aboard the Tango spacecraft (accurate to about 3^o 3D rms)[9] are used to calculate the "true" relative attitude in the camera frame. The accuracy in the estimated relative position is evaluated by the following translation error

$$\boldsymbol{E_T} = {}^{\mathcal{C}}\boldsymbol{T}_{\mathcal{T}/\mathcal{C},\text{true}} - {}^{\mathcal{C}}\boldsymbol{T}_{\mathcal{T}/\mathcal{C},\text{est}} \tag{30}$$

which represents the element-wise difference between the relative position ${}^{\mathcal{C}}\boldsymbol{T}_{\mathcal{T}/\mathcal{C},\text{true}}$ of the target spacecraft obtained from the flight dynamics products and the position ${}^{\mathcal{C}}\boldsymbol{T}_{\mathcal{T}/\mathcal{C},\text{est}}$ provided by the pose initialization. Similarly, the accuracy of the estimated attitude of the target vehicle is evaluated through the Euler angle representation of the rotational error

$$\boldsymbol{E_R}(\boldsymbol{R}_{\text{diff}}) = (\alpha,\ \beta,\ \gamma) \tag{31}$$

$$\boldsymbol{R}_{\text{diff}} = \boldsymbol{R}_{\mathcal{TC},\text{est}}\left(\boldsymbol{R}_{\mathcal{TC},\text{true}}\right)^T \tag{32}$$

where $\boldsymbol{R}_{\text{diff}}$ is a direction cosine matrix representing the relative rotation between the true and the estimate value of $\boldsymbol{R}_{\mathcal{TC}}$. Note that geometrically α, β, and γ represent the errors in the estimated attitude about the directions $\mathcal{C}_1$, $\mathcal{C}_2$, and $\mathcal{C}_3$, respectively. As shown in Figure 1, the direction $\mathcal{C}_3$ is pointed along the bore-sight while $\mathcal{C}_1$ and $\mathcal{C}_2$ are aligned with the image frame, $\mathcal{P}$.

Figure 7 shows the output initial pose estimates after applying the pose initialization subsystem to four images spanning 110 [min.] of the PRISMA mission. These images were selected as test cases since they span the different poses and illumination conditions during an orbit. The experiment was run on a 2.4 GHz Intel Core i5-4570T processor and made use of vectorized implementation of the pose estimation subsystem in MATLAB. The four test cases took an average of 8.2163 [s] (wall clock time measured using the `tic` and `toc` commands in MATLAB). However, since the ultimate objective of this work is to enable close-range navigation on satellites, it is imperative to note the limited computational power available from the microprocessors on these satellites. For example, the clock frequency on the LEON3-FT fault tolerant processor is 400 MHz,[32] whereas the presented results are from tests on a processor with a clock frequency of 2.4 GHz. This implies that we can expect an order of magnitude slower performance on a current state-of-the-art space hardened processor, approximately. However, the current code base is running in MATLAB which is shown to be 9-11 times slower than the best C++ executable version of the same.[33] Table 1 presents the accuracy of the pose initialization for the four test cases. Notably, the method provides better accuracy in the $\mathcal{C}_1$ and $\mathcal{C}_2$ axes for both relative position and attitude since these are aligned with the image plane while $\mathcal{C}_3$ is the bore-sight direction of the camera. The accuracy results for pose initialization from these four test cases are used to inform the initial condition for the navigation filter in the following numerical simulations.

Table 1: Accuracy of the pose initialization for the four test cases.

Test Case	$\boldsymbol{E_R}$ [deg]	$\boldsymbol{E_T}$ [m]
1	$[0.43,\ -0.95,\ 7.04]$	$[0.05,\ -0.003,\ 0.13]$
2	$[-0.97,\ -0.72,\ 5.12]$	$[0.04,\ -0.06,\ 0.23]$
3	$[1.60,\ -1.10,\ 10.61]$	$[-0.05,\ 0.04,\ 0.18]$
4	$[-1.45,\ -1.78,\ 7.17]$	$[-0.06,\ -0.04,\ 0.17]$

Navigation Filter

The performance of the proposed navigation filter is assessed by generating reference relative trajectories which emulate a close proximity mission. The servicer absolute orbit conditions are selected based on the orbit of the Mango spacecraft[20] and are presented in Table 2 along with the initial conditions for the reference trajectories. ROE 1 is purely an along-track separation describing a standard v-bar hold point. ROE 2 describes a mid-range hold point whereas ROE 3 introduces a small relative semi-major axis, allowing the servicer to approach the target spacecraft and initiate docking. The three reference trajectories are visualized in Figure 8. The initial relative attitude of the target spacecraft principal axes with respect to the principal axes of the servicer, $\boldsymbol{q}(0)$, is $[0.707\ 0.707\ 0\ 0]$. The initial relative angular velocity $\boldsymbol{\omega}$ of the target is selected as $[3\ 0\ 0]$ deg/s. The magnitude of $\boldsymbol{\omega}$ was selected to represent the estimated angular velocity of the Envisat spacecraft.[34] The servicer frame $\mathcal{S}$ is assumed to be aligned with the camera frame $\mathcal{C}$ and the servicer spacecraft is initially oriented such that $\mathcal{C}_3$ (camera bore-sight direction) is aligned with the along-track direction. The angular velocity of the servicer is selected to always maintain this alignment. The "ground truth" for the relative position and velocity of each of the spacecraft are numerically propagated for one orbit with a time step of one second using rigorous force models including GRACE's GGM01S geopotential of order and degree 40,[35] a cannonball model for atmospheric drag,[36] analytical models for third-body gravity from the Sun and the Moon,[37] and solar radiation

Table 2: Initial servicer orbital conditions and reference relative trajectories parametrized in the ROE space.

Servicer Orbit	a = 7078.1 [km]	e = 0.001	i = 98.2°	Ω = 189.9°	ω = 0 °	M_0 = 0°
Initial ROE	$a\delta a$ [m]	$a\delta\lambda$ [m]	$a\delta e_x$ [m]	$a\delta e_y$ [m]	$a\delta i_x$ [m]	$a\delta i_y$ [m]
ROE 1	0	-10	0	0	0	0
ROE 2	0	-5	0.3	0	-0.3	0
ROE 3	-0.15	-3.75	0.225	0	-0.225	0

pressure including a conical Earth shadow model. For the same time step and duration, the "ground truth" for the relative attitude is obtained through the numerical integration of Euler's rotation equations for each spacecraft. From the simulated trajectories and attitudes, line segment measurements are obtained at each time step through the true perspective projection (Eq. (1)- (2)) of four randomly selected visible line segments at that time step. A pinhole camera model is adopted to model the close-range vision camera embarked on the Mango spacecraft of the PRISMA mission. The effective focal length is $20,187 \cdot 10^{-6}$ meters for both axes of the image sensor which produces images 752 [px] $\times$ 580 [px] in size. Gaussian noise is added to the pixel positions of each of the line segment end-points according to Table 3 and corresponding pseudo-measurements (${}^{\mathcal{C}}n_i$) are generated using Eq. (27). Similarly, modelled pseudo-measurements are generated using Eq. (24). For each of the filter test cases, the initial conditions for the state, $\boldsymbol{x}(0)$, are perturbed from the ground truth values according to Table 3. The initial conditions are selected based on the aforementioned performance results of the pose initialization subsystem. In Table 3, $\boldsymbol{v}(0) - \boldsymbol{v}_{\text{true}}(0)$ and $\boldsymbol{\omega}(0) - \boldsymbol{\omega}_{\text{true}}$ are the initial errors in the relative translational and angular velocities of the target spacecraft with respect to the servicer, as supplied to the navigation filter, respectively. These two quantities are expressed in the camera frame, $\mathcal{C}$.

Table 3: Simulation noise and initial conditions for the three filter test cases.

Pixel Location [px]	$\boldsymbol{E_T}(0)$ [m]	$\boldsymbol{E_R}(0)$ [deg]	$\boldsymbol{v}(0) - \boldsymbol{v}_{\text{true}}(0)$ [m/s]	$\boldsymbol{\omega}(0) - \boldsymbol{\omega}_{\text{true}}$ [deg/s]
$\mathcal{N}(0, 5)$	[0.10, 0.10, 0.10]	[1, 1, 10]	[1, 1, 1]	[0.6, 0.6, 0.6]

The error metrics, $\boldsymbol{E_R}$ and $\boldsymbol{E_T}$, are computed at each time step for the three ROE test cases and are presented in Figures 9 - 14 along with the standard deviation in the relative position and attitude as reported by the covariance matrix of the navigation filter. In Table 4, the mean and standard deviation of the error metrics $\boldsymbol{E_R}$ and $\boldsymbol{E_T}$ are computed for the last 1000 seconds of the simulation. These mean and standard deviation values represent the "quality" of the state estimate at the steady state. The steady state position error is centimeter-level while the attitude errors are sub-degree level for all three ROE test cases. For all ROE test cases, the $\mathcal{C}_3$ direction has the largest error in relative position. This is to be expected since the $\mathcal{C}_3$ direction is aligned with the boresight direction and estimating the range is especially challenging from 2D imagery. This effect can also be seen in the case of ROE 1 where there is a steady-state error in relative position in the $\mathcal{C}_3$ direction due to a constant along-track separation. The relative position error in $\mathcal{C}_3$ is higher in ROE 3 than ROE 2 as the mean along-track separation is constantly changing. The magnitude of the position errors in

the $\mathcal{C}_1$ and $\mathcal{C}_2$ direction appear to be proportional to the range of target spacecraft from the servicer spacecraft.

Table 4: Steady state errors in the relative position and attitude for the three ROE test cases.

Relative Position [m]			
Initial ROE	$\boldsymbol{E_{t,\mathcal{C}_1}}$ [m]	$\boldsymbol{E_{t,\mathcal{C}_2}}$ [m]	$\boldsymbol{E_{t,\mathcal{C}_3}}$ [m]
ROE 1	0.018 ± 0.004	0.004 ± 0.004	0.106 ± 0.007
ROE 2	0.017 ± 0.007	-0.010 ± 0.006	-0.048 ± 0.008
ROE 3	-0.001 ± 0.003	-0.0006 ± 0.0002	-0.067 ± 0.017
Relative Attitude [deg]			
Initial ROE	$\boldsymbol{E_{R,\mathcal{C}_1}}$ [deg]	$\boldsymbol{E_{R,\mathcal{C}_2}}$ [deg]	$\boldsymbol{E_{R,\mathcal{C}_3}}$ [deg]
ROE 1	-0.53 ± 0.44	0.36 ± 0.57	-0.16 ± 0.53
ROE 2	0.03 ± 0.57	-0.52 ± 0.46	0.10 ± 0.37
ROE 3	0.01 ± 0.31	-0.12 ± 0.93	-0.02 ± 0.74

Finally, the navigation filter is stress-tested by varying the measurement noise, measurement update interval, and the a-priori position estimate. For all stress-tests, the ROE 3 servicer and target spacecraft orbital conditions were used (see Table 2) and a complete orbit was numerically simulated in the manner described above. For each level of the stress-test variable, the mean and standard deviation of the error metrics, $\boldsymbol{E_R}$ and $\boldsymbol{E_T}$, are computed for the last 1000 seconds of the simulation and reported in Appendix B. The measurement noise was varied by increasing the standard deviation of the zero mean Gaussian noise being added to the pixel locations of the line segment end-points. The impact of increasing the measurement noise is the highest in the relative position in the $\mathcal{C}_3$ direction, where the error increases linearly with the standard deviation of the measurement noise. The magnitude of the mean relative position error in the $\mathcal{C}_1$ and $\mathcal{C}_2$ direction as well as the attitude error in each direction stays approximately constant with increasing measurement noise. The standard deviation of the relative position and attitude increases with increasing levels of measurement noise. The time interval between successive measurement updates in the navigation filter was varied to assess the applicability of the navigation filter in processors with low computational resources. There seemed to be no significant increase in the relative attitude error, however, the relative position error in the $\mathcal{C}_3$ direction increases drastically for a measurement update interval greater than 3 seconds. This increase in error is due to the modelling errors in the system which are more apparent at larger time steps since the ground truth of the relative position is propagated using full-force models whereas the filter's system dynamics only consider unperturbed motion. However, note that the ROE parametrization used in the navigation filter allows for easy inclusion of perturbations in the system dynamics to increase modelling accuracy at the expense of computational complexity. The error in the a-priori estimate of the relative position is varied by adding a bias value to the ground truth of the relative position in each axis of the camera frame. This assesses the robustness of the navigation filter to a poor pose initialization. Performance in the $\mathcal{C}_2$ an $\mathcal{C}_3$ directions worsened considerably for an error greater than 0.5 m in the a-priori relative position estimate. The navigation filter fails to improve upon the a-priori estimate in the $\mathcal{C}_2$ direction when initialized with an error greater than 0.8 [m] in the initial relative position. This failure in convergence is expected due to the linearization of the dynamics and measurement models in an EKF. However, the

initial errors corresponding to poor performance are far greater than the errors present in the initial pose estimates provided by the pose initialization subsystem (shown to provide estimates with less than 0.3 [m] of error).

CONCLUSION

This paper has described an architecture for monocular vision based reduced-dynamics pose estimation to enable autonomous proximity operations in on-orbit-servicing and formation flying applications. The description of the applied techniques in the pose initialization subsystem is accompanied with a detailed formulation of a novel navigation filter comprising of an EKF and an MEKF. The pose initialization subsystem is validated on actual space imagery from the PRISMA mission and by comparing the pose estimates with independent flight dynamics operational products. The accuracy of the initial pose estimates has been shown to be at the level of 3.0375° (3D rms error) and 0.0786 [m] (3D rms error). The navigation filter is validated and stress-tested with representative relative trajectory test cases to exhibit its applicability and accuracy potential. Comparisons with the numerically simulated ground truth of the three relative trajectory test cases show the accuracy of the pose estimates to be at the level of 0.4376° (3D rms error) and 0.0755 [m] when initialized with a pose solution with a relative position error of 0.1732 [m] (3D rms) and a relative attitude error of 10.100° (3D rms). Notably, the errors in the translation and attitude in and around the camera bore-sight direction were the largest. The pose initialization subsystem improves upon the state-of-the-art by introducing a hybrid approach to image processing by fusing the weak gradient elimination technique with the Sobel operator followed by Hough transform to detect both small and large features of the target spacecraft. The navigation filter improves the state-of-the-art by utilizing the ROE parametrization of the state which allows the use of an elegant state-transition-matrix to propagate the relative position dynamics for a servicer orbit of an arbitrary eccentricity. In addition, the ROE model conveniently decouples the range from the remaining state elements. The filter also relies on the pseudo-measurements of a normal vector generated from the pixel locations of the target spacecraft's edges, thereby making the filter robust to the detection of partial edges in the image. The navigation filter's robustness is also exhibited by stress-testing it with varying levels of pose initialization errors, measurement noise, and measurement update intervals. Notably, with an increase of the measurement noise of the pixel locations, the accuracy only worsened for the relative position in the bore-sight direction while only increasing the uncertainty in other directions. The navigation filter is shown to provide precise and accurate pose estimates with an initial position error of up to 0.8 [m]. Future work will include the modeling of control efforts in the system dynamics to allow the full validation of the navigation filter with actual space imagery from the PRISMA mission. This will pave the path for the performance assessment of the entire pose estimation architecture through hardware-in-the-loop simulations of a wide spectrum of orbital and illumination conditions using Space Rendezvous Laboratory's robotic testbed at Stanford University. This testbed incorporates a 7-degree-of-freedom robotic manipulator and 10 custom-engineered illumination panels which can simulate relative orbital trajectories typical in close-proximity operations in space, with high fidelity illumination conditions.

REFERENCES

[1] L. T. Castellani, "PROBA-3 mission," *International Journal of Space Science and Engineering*, Vol. 1, No. 4, 2013.

[2] A. F. R. Laboratory, "Fact sheet: Automated navigation and guidance experiment for local space (ANGELS)," 2013, 10.2762/41007.

[3] S. D'Amico, P. Bodin, M. Delpech, and R. Noteborn, "PRISMA," *Distributed Space Missions for Earth System Monitoring* (M. D'Errico, ed.), pp. 599–637, 2013, 10.1007/978-1-4614-4541-8.

[4] M. R. Leinz, C.-T. Chen, M. W. Beaven, T. P. Weismuller, D. L. Caballero, W. B. Gaumer, P. W. Sabasteanski, P. A. Scott, and M. A. Lundgren, "Orbital Express Autonomous Rendezvous and Capture Sensor System (ARCSS) flight test results," *Proceedings of SPIE*, Vol. 6958, 2008, pp. 69580A–69580A–13, 10.1117/12.779595.

[5] A. Petit, E. Marchand, and K. Kanani, "Vision-based space autonomous rendezvous: A case study," *IEEE International Conference on Intelligent Robots and Systems*, 2011, pp. 619–624, 10.1109/IROS.2011.6048176.

[6] A. Cropp and P. Palmer, "Pose Estimation and Relative Orbit Determination of a Nearby Target Microsatellite using Passive Imagery," *5th Cranfield Conference on Dynamics and Control of Systems and Structures in Space 2002*, 2002, pp. 389–395.

[7] S. Zhang and X. Cao, "Closedform solution of monocular visionbased relative pose determination for RVD spacecrafts," *Aircraft Engineering and Aerospace Technology*, Vol. 77, No. 3, 2005, pp. 192–198, 10.1108/00022660510597214.

[8] J. Ventura, A. Fleischner, and U. Walter, "Pose Tracking of a Noncooperative Spacecraft During Docking Maneuvers Using a Time-of-Flight Sensor," No. January, 2016, pp. 1–16, 10.2514/6.2016-0875.

[9] S. D'Amico, M. Benn, and J. L. Jørgensen, "Pose estimation of an uncooperative spacecraft from actual space imagery," *International Journal of Space Science and Engineering*, Vol. 2, No. 2, 2014, p. 171, 10.1504/IJSPACESE.2014.060600.

[10] H. Bay and A. Ess, "Speeded-Up Robust Features (SURF)," Vol. 110, No. September, 2008, pp. 346–359, 10.1016/j.cviu.2007.09.014.

[11] D. G. Lowe, "Distinctive image features from scale invariant keypoints," *Int'l Journal of Computer Vision*, Vol. 60, 2004, pp. 91–11020042, http://dx.doi.org/10.1023/B:VISI.0000029664.99615.94.

[12] A. Petit, K. Kanani, and E. Marchand, "Vision-based Detection and Tracking for Space Navigation in a Rendezvous Context," *Int. Symp. on Artificial Intelligence, Robotics and Automation in Space, i-SAIRAS˜2012*, 2012.

[13] H. Benninghoff, T. Boge, and F. Rems, "Autonomous Navigation for On-Orbit Servicing," *KI - Künstliche Intelligenz*, Vol. 28, No. 2, 2014, pp. 77–83, 10.1007/s13218-014-0297-0.

[14] A. A. Grompone, *Vision-Based 3D Motion Estimation for On-Orbit Proximity Satellite Tracking and Navigation*. PhD thesis, Naval Postgraduate School, 2015.

[15] R. O. Duda and P. E. Hart, "Use of the Hough transform to detect lines and cures in pictures," *Communications of the Association Computing Machinery*, Vol. 15, No. 1, 1972.

[16] F. Ababsa, "A New 3D Model-Based Tracking Technique for Robust Camera Pose Estimation," *International Journal of Advanced Computer Science and Applications(IJACSA)*, Vol. 3, No. 4, 2012, pp. 31–37.

[17] B. E. Tweddle and A. Saenz-Otero, "Relative Computer Vision-Based Navigation for Small Inspection Spacecraft," *Journal of Guidance, Control, and Dynamics*, Vol. 38, No. 5, 2015, pp. 969–978, 10.2514/1.G000687.

[18] F. M. Mirzaei and S. I. Roumeliotis, "Globally Optimal Pose Estimation from Line-Plane Correspondences," *Science*, No. 612, 2010, pp. 5581–5588, 10.1109/ICRA.2011.5980272.

[19] D. G. Lowe, "Three-dimensional object recognition from single two-dimensional images," *Artificial Intelligence*, Vol. 31, No. 3, 1987, pp. 355–395, 10.1016/0004-3702(87)90070-1.

[20] P. Bodin, R. Noteborn, R. Larsson, T. Karlsson, S. D'Amico, J.-S. Ardaens, M. Delpech, and J. C. Berges, "The prisma formation flying demonstrator: Overview and conclusions from the nominal mission," *Advances in the Astronautical Sciences*, Vol. 144, 2012, pp. 441–460.

[21] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge University Press, second ed., 2004.

[22] M. A. Fischler and R. C. Bolles, "Random Sample Consensus: A Paradigm for Model Fitting with Applicatlons to Image Analysis and Automated Cartography," *Communications of the ACM*, Vol. 24, No. 6, 1981, pp. 381 – 395, 10.1145/358669.358692.

[23] S. Sharma and J. Ventura, "Robust Model-Based Monocular Pose Estimation for Noncooperative Spacecraft Rendezvous," *Journal of Spacecraft and Rockets*, 2017, pp. 1–35.

[24] J. Canny, "A Computational Approach to Edge Detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-8, No. 6, 1986, pp. 679–698, 10.1109/TPAMI.1986.4767851.

[25] S. Sharma and S. D'Amico, "Comparative assessment of techniques for initial pose estimation using monocular vision," *Acta Astronautica*, 2015, 10.1016/j.actaastro.2015.12.032.

[26] S. D'Amico, *Autonomous Formation Flying in Low Earth Orbit*. PhD thesis, Technisce Universiteit Delft, 2005.

[27] K. Yamanaka and F. Ankersen, "New State Transition Matrix for Relative Motion on an Arbitrary Elliptical Orbit," *Journal of Guidance, Control, and Dynamics*, Vol. 25, No. 1, 2002, pp. 60–66, 10.2514/2.4875.

[28] F. L. Markley, "Multiplicative vs . Additive Filtering for Spacecraft Attitude Determination Quaternion estimation," *Journal of Guidance, Control, and Dynamics*, Vol. 26, No. 2, 2003, pp. 311–317.

[29] E. J. Lefferts, F. L. Markley, and M. D. Shuster, "Kalman Filtering for Spacecraft Attitude Estimation," *Journal of Guidance, Control, and Dynamics*, Vol. 5, No. 5, 1982, pp. 417–429, 10.2514/3.56190.

[30] F. Ababsa, "Robust Extended Kalman Filtering for camera pose tracking using 2D to 3D lines correspondences," *2009 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, 2009, pp. 1834–1838, 10.1109/AIM.2009.5229789.

[31] J. S. Ardaens, S. D'Amico, and O. Montenbruck, "Final commissioning of the prisma gps navigation system," *Journal of Aerospace Engineering, Sciences and Applications*, Vol. 4, No. 3, 2012, pp. 104–118, 10.7446/jaesa.0403.10.

[32] J. Gaisler, "A portable and fault-tolerant microprocessor based on the SPARC V8 architecture," *Proceedings of the 2002 International Conference on Dependable Systems and Networks*, 2002, pp. 409–415, 10.1109/DSN.2002.1028926.

[33] S. Aruoba and J. Fernández-Villaverde, "A Comparison of Programming Languages in Economics," *NBER Working Paper Series*, Vol. 3, 2014, pp. 0–16, 10.3386/w20263.

[34] S. Lemmens, H. Krag, J. Rosebrock, and I. Carnelli, "Radar mappings for attitude analysis of objects in orbit," *6th European Conference on Space Debris*, Vol. 2013, No. April, 2013, pp. 22–25.

[35] T. Grace and A. Notes, "GRACE Gravity Model GGM02," tech. rep., 2000.

[36] L. G. Jacchia, "Revised Static Models of the Thermosphere and Exosphere with Empirical Temperature Profiles," tech. rep., Smithsonian Astrophysical Observatory, 1971.

[37] O. Montenbruck and E. Gill, *Satellite Orbits: Models, Methods, and Applications*. Springer-Verlag, 2001.

APPENDIX A: MAPPING BETWEEN ROE STATE AND RTN POSITION

$$\rho_{\text{RTN}} = \boldsymbol{T}(t)\delta x_{\text{ROE}} \tag{33}$$
$$= r(f)\Psi_1\Psi_2\delta x_{\text{ROE}} \tag{34}$$

where

$$\Psi_1 = \begin{bmatrix} 1 & 0 & -kc_f & -ks_f & 0 & 0 \\ 0 & 1 & (k+1)s_f & -(k+1)c_f + \frac{e}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & s_\theta & -c_\theta \end{bmatrix} \tag{35}$$

$$\Psi_2 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & \xi & -\frac{(1-\xi)s_\omega}{e} & \frac{(1-\xi)c_\omega}{e} & 0 & \cot i \\ 0 & 0 & \frac{c_\omega}{\eta^2} & \frac{s_\omega}{\eta^2} & 0 & \frac{e\cot i}{\eta^3} \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{36}$$

Here $r(f)$ denotes the servicer orbital radius at the true anomaly, f, $k = 1 + ec_f$, $\theta = f + \omega$, and the auxiliary eccentricity factor, ξ, is defined as

$$\xi = \frac{1 + \frac{1}{2}e^2}{\eta^3} \tag{37}$$

APPENDIX B: FIGURES

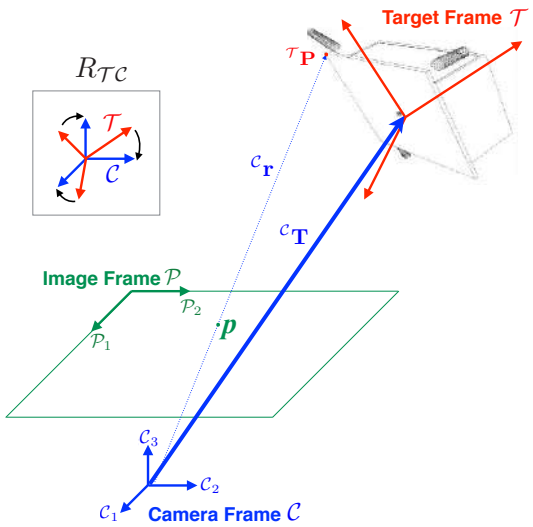

Figure 1: Schematic representation of the pose initialization problem using a monocular image.

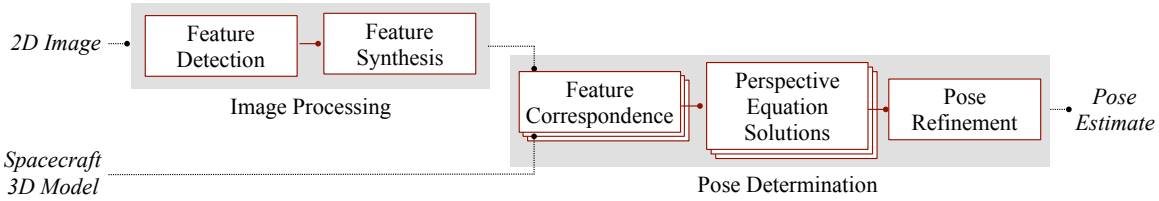

Figure 2: Proposed pose initialization subsystem with inputs of a single 2D image and a 3D model and an output of the pose estimate.

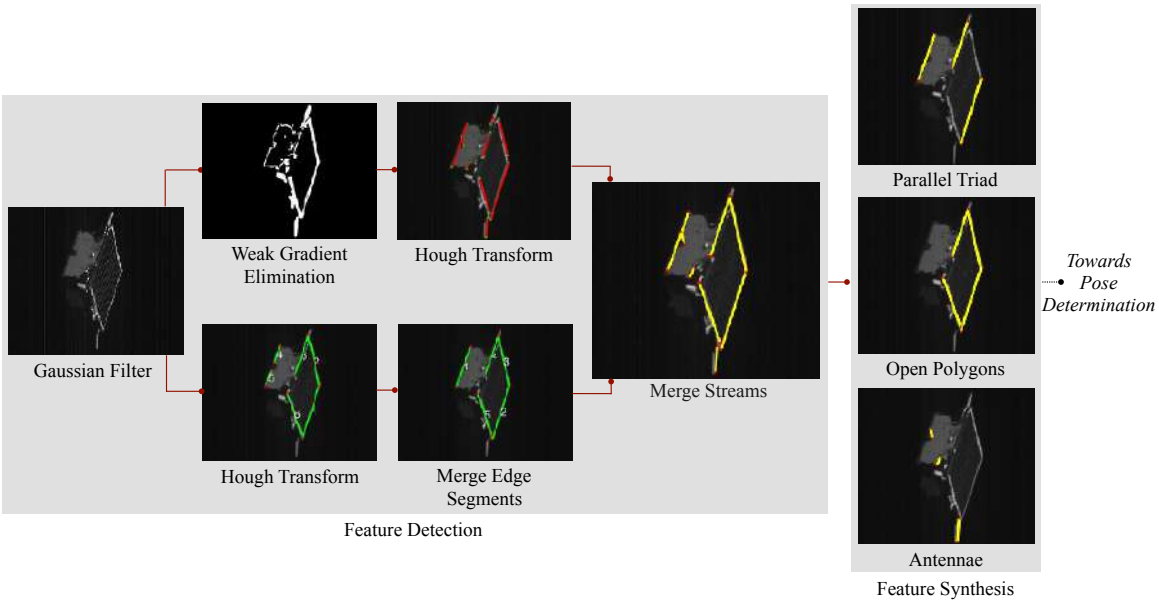

Figure 3: Main steps of the image processing subsystem with a single 2D image as the input and feature groups as the output.

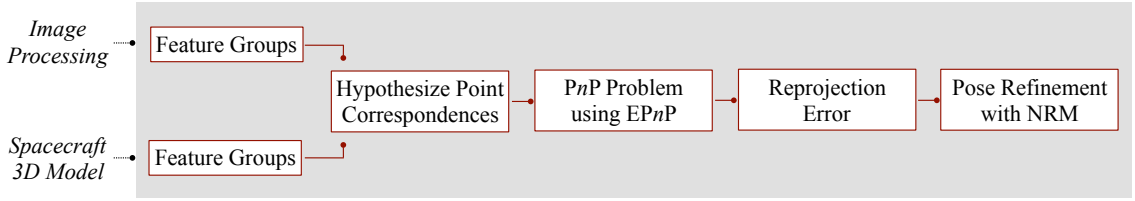

Figure 4: Main steps of the pose determination subsystem with feature groups from the image and the 3D model as input and the pose estimate as the output.

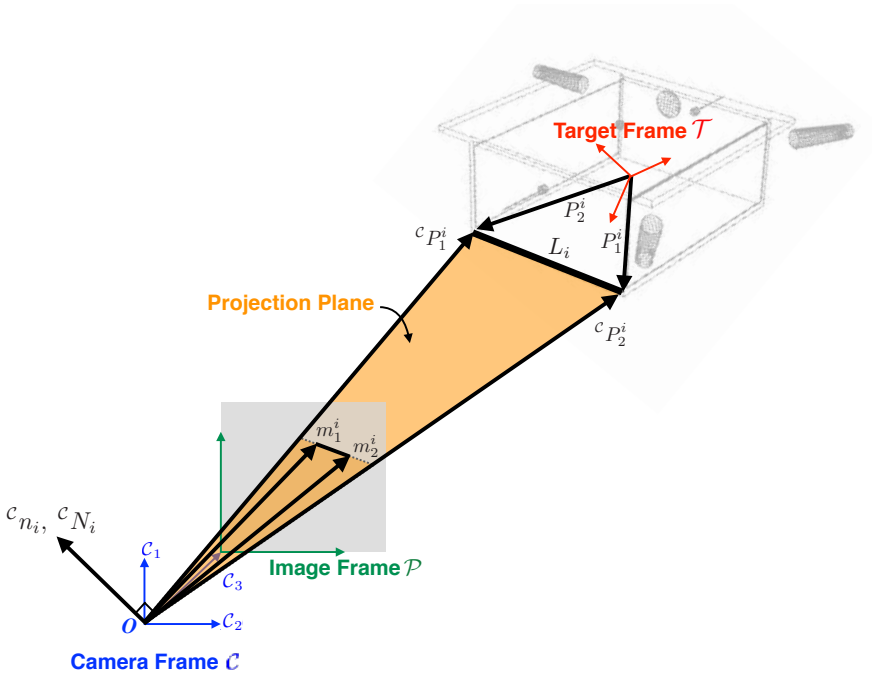

Figure 5: Geometric representation of the measured unit vector, $^{\mathcal{C}}n_i$, and the modelled unit vector, $^{\mathcal{C}}N_i$.

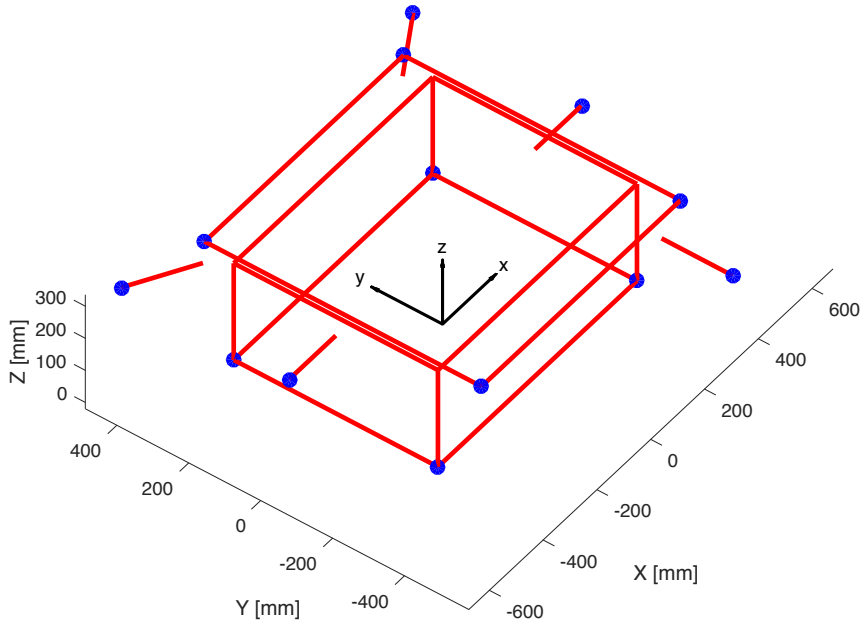

Figure 6: 3D wireframe model of the TANGO satellite

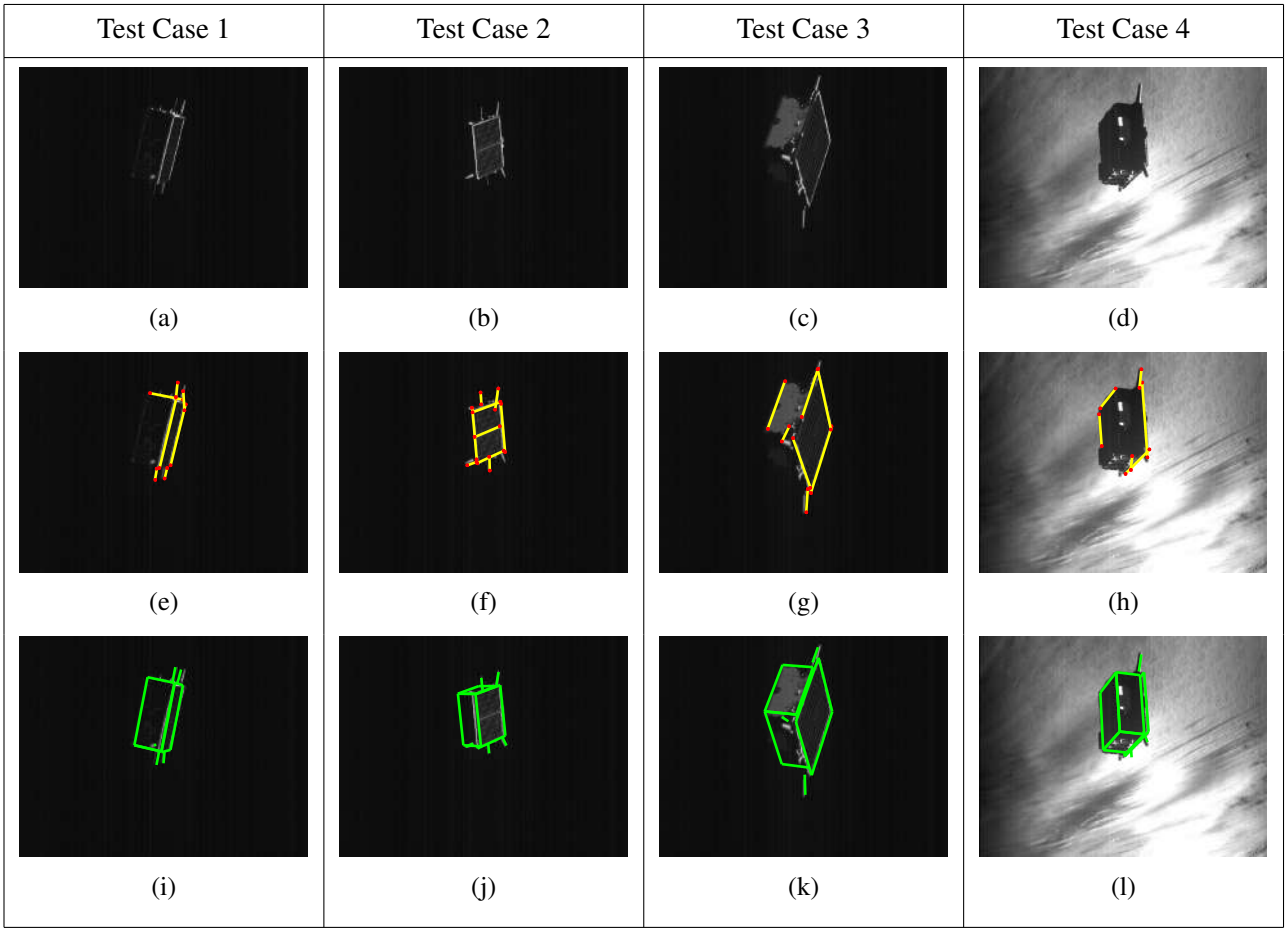

Figure 7: Results from the application of our architecture to the four test cases. Sub-figures (a–d) show the respective input images, sub-figures (e–h) show the respective image processing outputs, and sub-figures (i–l) show the respective final pose estimates after pose refinement.

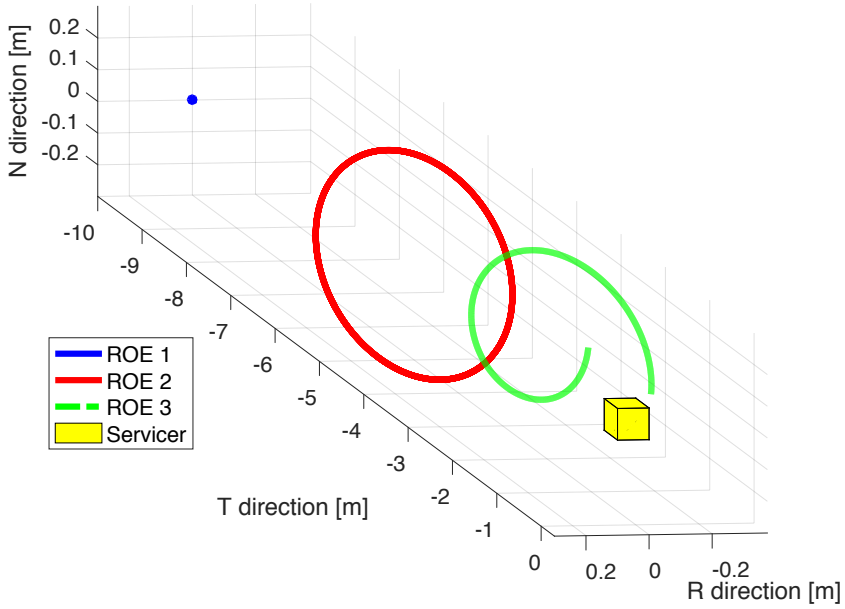

Figure 8: Visualization of the three reference relative trajectories used in the performance assessment of the navigation filter.

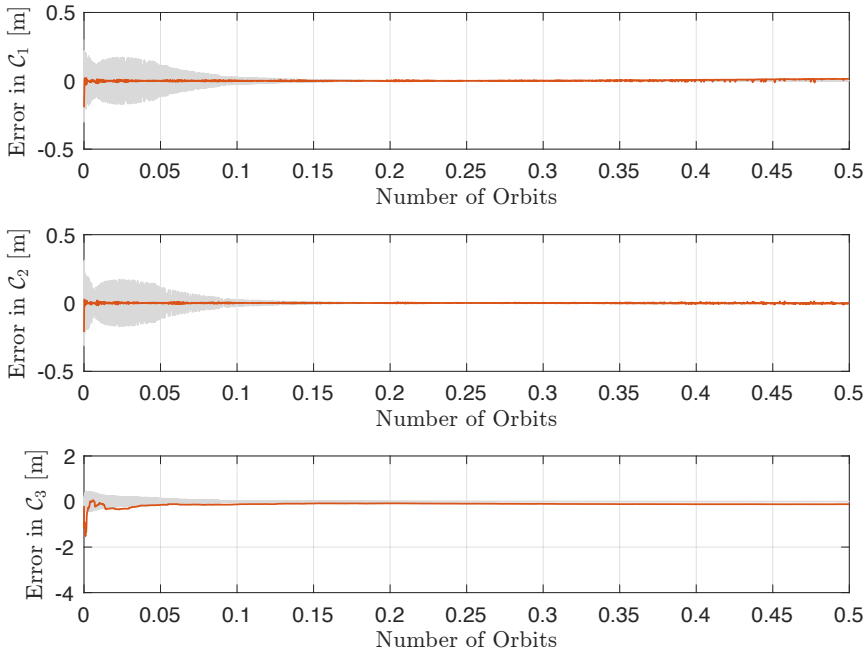

Figure 9: Evolution of the relative position error in the camera frame for the ROE 1 test case.

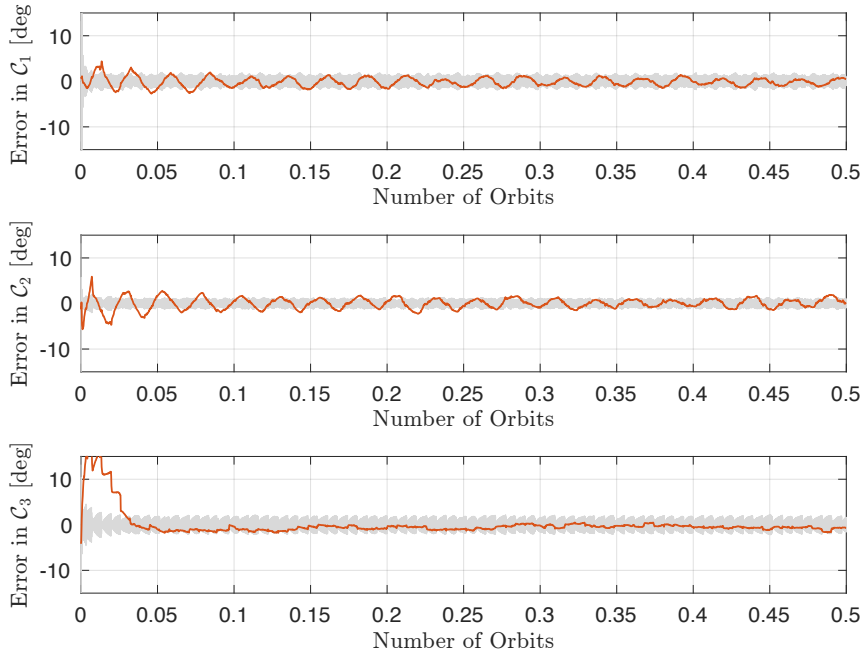

Figure 10: Evolution of the relative attitude error in the camera frame for the ROE 1 test case.

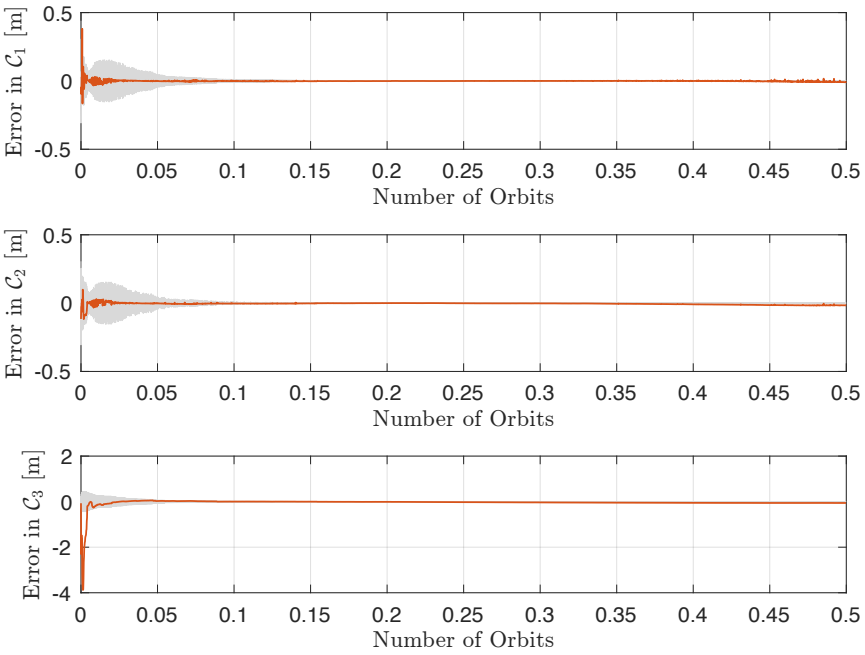

Figure 11: Evolution of the relative position error in the camera frame for the ROE 2 test case.

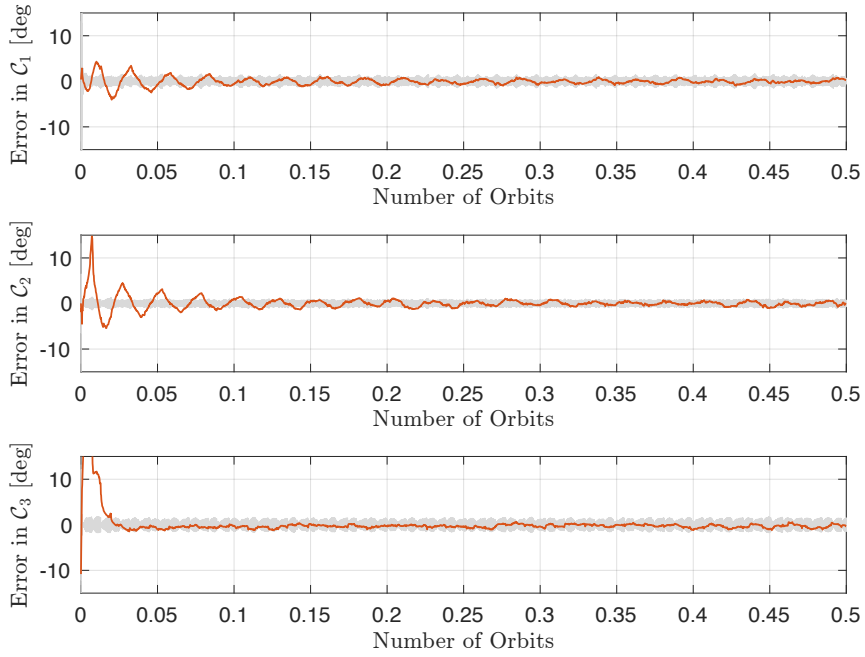

Figure 12: Evolution of the relative attitude error in the camera frame for the ROE 2 test case.

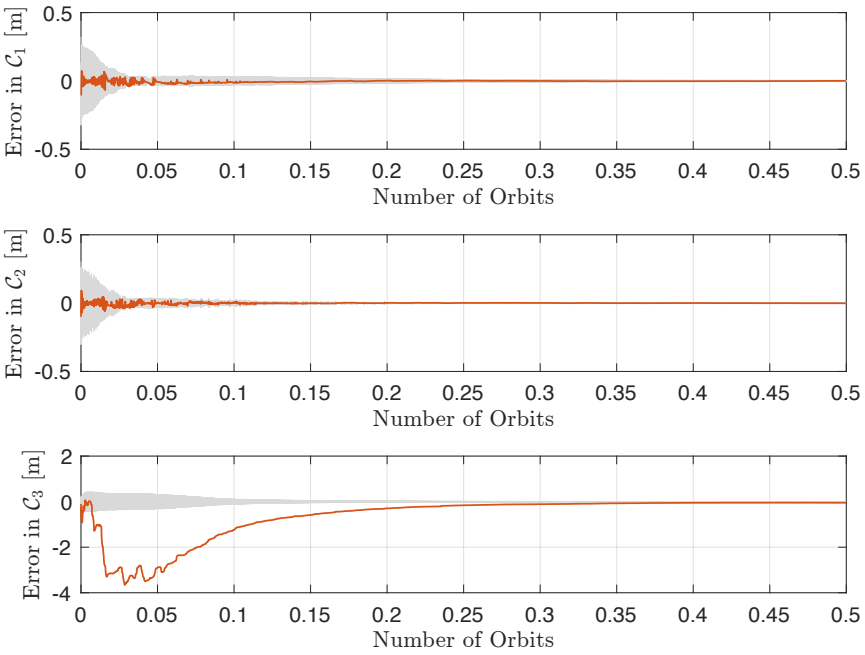

Figure 13: Evolution of the relative position error in the camera frame for the ROE 3 test case.

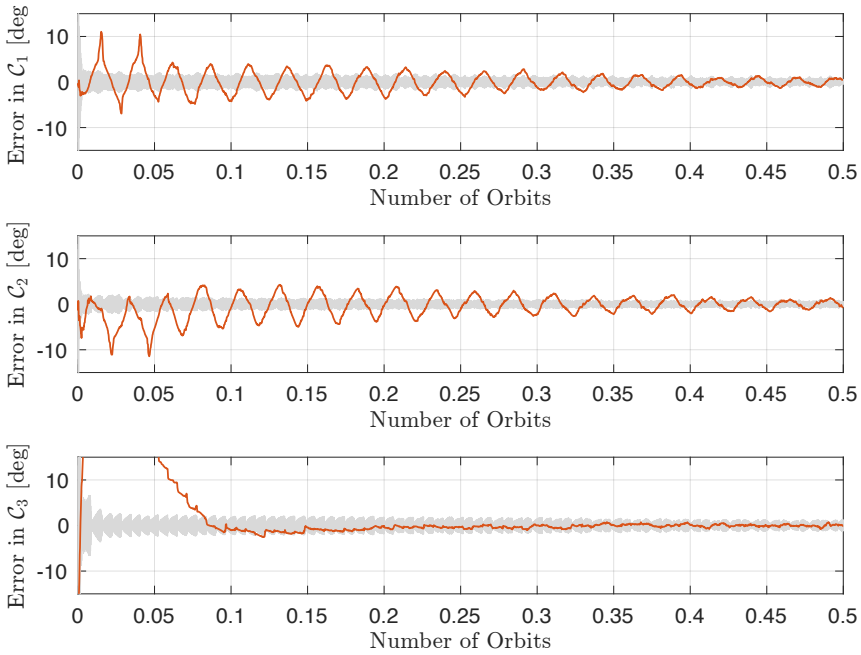

Figure 14: Evolution of the relative attitude error in the camera frame for the ROE 3 test case.

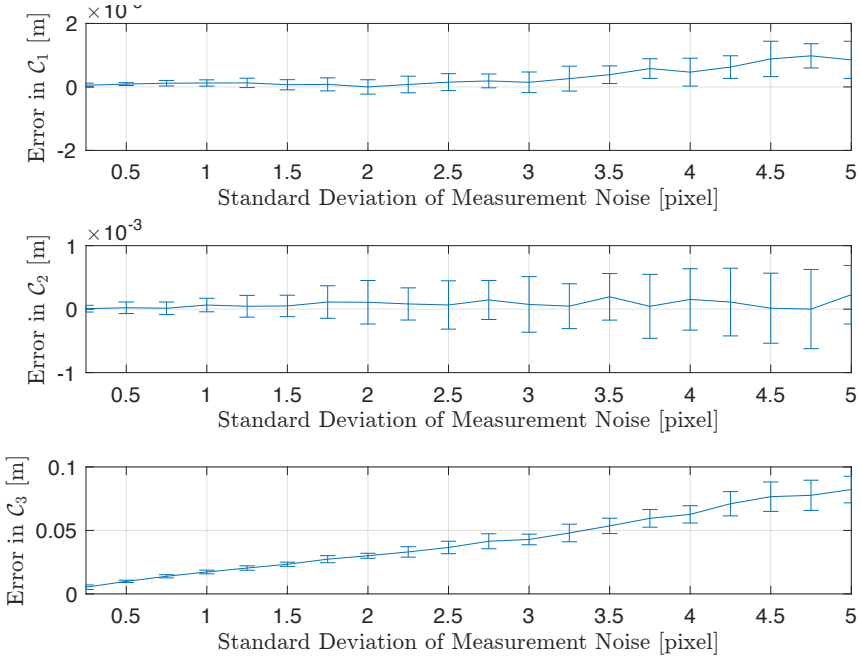

Figure 15: Effect of varying the Gaussian noise in the measured pixel locations on the relative position error.

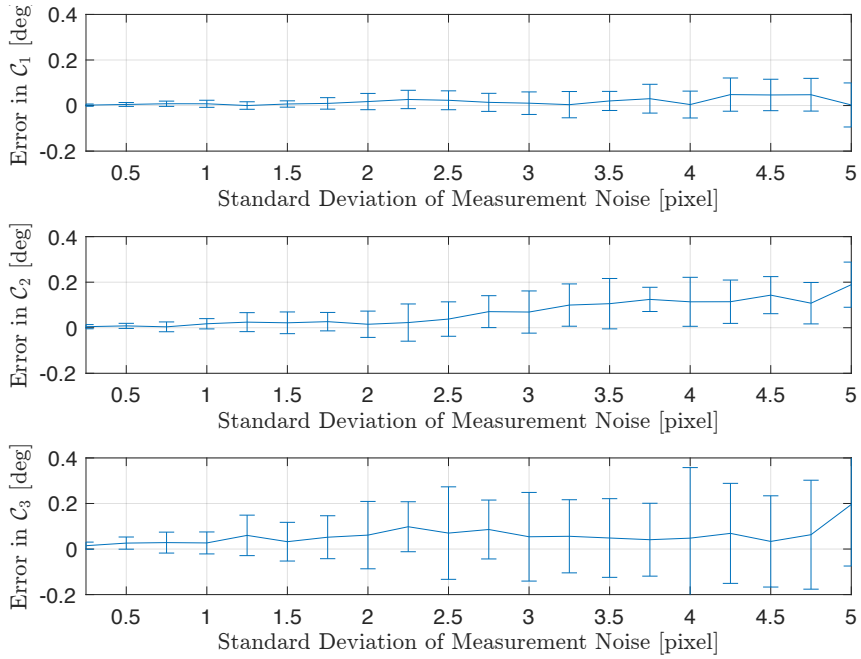

Figure 16: Effect of varying the Gaussian noise in the measured pixel locations on the relative attitude error.

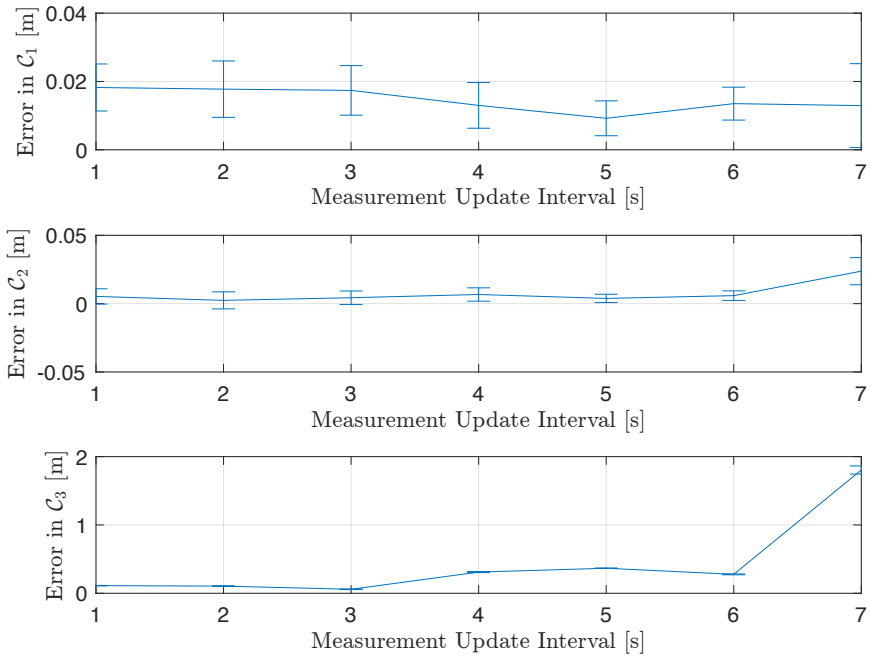

Figure 17: Effect of varying the measurement update interval on the relative position error.

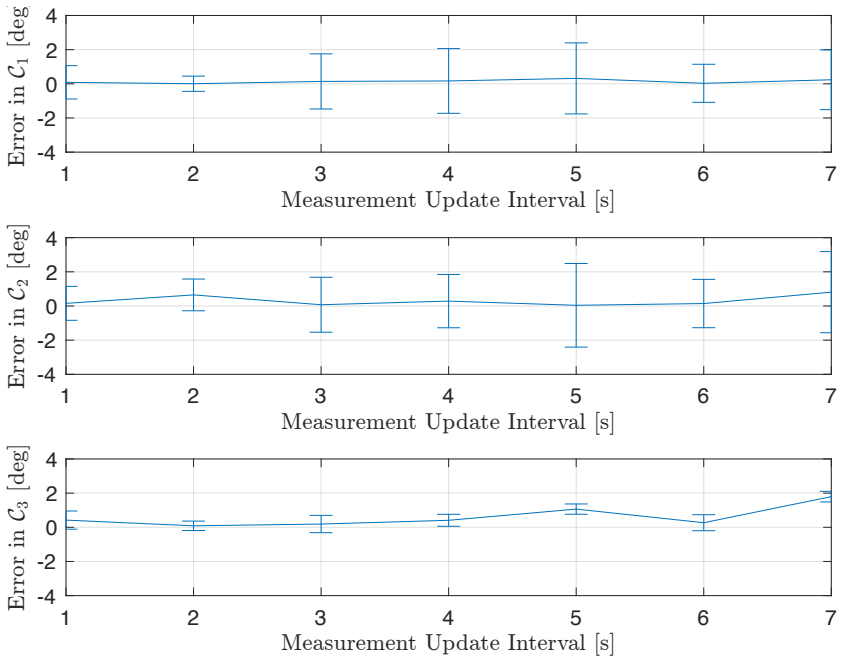

Figure 18: Effect of varying the measurement update interval on the relative attitude error.

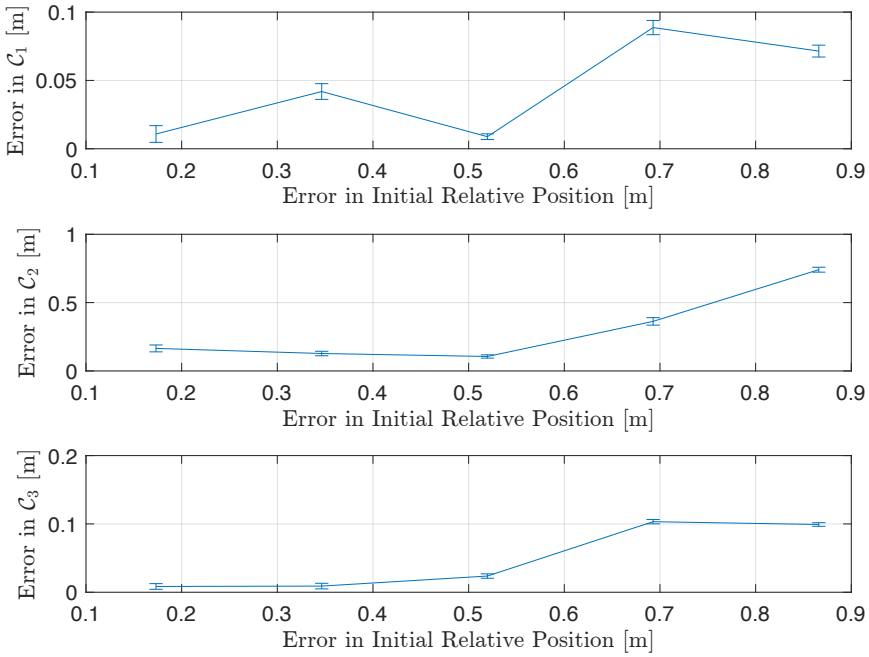

Figure 19: Effect of varying error in the a-priori position estimate on the relative position error.