69th International Astronautical Congress (IAC), Bremen, Germany, 1-5 October 2018.

https://iafastro.directory/iac/paper/id/46161/summary/

IAC-18- C1.IP-20-x46161

Space-oriented Navigation Solutions with Integrated Sensor-Suite: the I3DS H2020 Project

Antonio F. Scannapieco[a]*, Luke M. Feetham[a], Mónica Estébanez Camarena[a], Nabil Aouf[a]

[a] *Centre for Electronic Warfare, Information and Cyber, Cranfield University, Defence Academy of the United Kingdom, SN6 8LA Shrivenham, Wiltshire, United Kingdom*
* Corresponding Author, a.scannapieco@cranfield.ac.uk

Abstract

In all orbital applications, such as on-orbit servicing and repair, rendezvous and docking, active debris removal (ADR), and planetary applications, such as exploration of unknown environments for scientific purposes by means of rovers, GPS-denied navigation aspects have a very large impact on the successful outcome of missions. Having a sensor suite, and hence several different sensors, also requires, at the same time, a suite of navigation algorithms able to deal with different kinds of inputs. Some of them, however, can be shared between multiple sensors, after thorough pre-processing of the raw data. Additionally, the same kind of sensor can require two different navigation algorithms depending on the scenario. The work described in this paper aims to present and critically discuss the approach to precise relative navigation solutions with a complete suite of sensors and their performance in different space-oriented application scenarios.

Standalone navigation filters are examined. In the case of a high-resolution camera for an orbital scenario, the pose of a target, with respect to a chaser, can be thoroughly obtained with the aid of fiducial markers. Stereo camera-based navigation is also addressed with visual odometry. In the case of a stereo camera the problem of scale estimation during odometry is solved by means of triangulation. Since the outputs of the sensor-suite are also dense 3D point clouds, Iterative Closest Point and Histogram of Distances (HoD) with Kalman filter approaches are analyzed, paying attention to the provision of correct sensor characterization. The results for each filter are exhaustively examined, highlighting their strengths and the points where some improvements can be achieved.
Keywords: pose estimation; relative navigation; camera; point clouds.

1. Introduction

The determination of relative position and attitude of an active spacecraft with respect to a target has been widely investigated in recent years for various mission scenarios that involve autonomous manoeuvres in close-range proximity, such as active debris removal (ADR) [1],[2] or on-orbit servicing (OOS) [3],[4]. A comprehensive review of the techniques for cooperative and uncooperative targets for close-proximity operations has been presented in [5].

At same time, space robotic systems are increasing in complexity and versatility in order to tackle more advanced tasks in orbit. Robots in space reduce costs related to life support systems, but their level of autonomy must be constantly improved to reach the capabilities of human skills and dexterity. It is clear that pose determination is a complex task requiring ad-hoc solutions in terms of algorithms and technology. In this regard, the H2020 project, Integrated 3D Sensors (I3DS), aims to develop a modular Inspector Sensors Suite (INSES), which will be a smart collection of building blocks with a common set of various sensors. The INSES should provide suitable and accurate pose estimates to the control algorithms to be exploited by diverse target scenarios, such as interplanetary missions, formation flying missions, cooperative and non-cooperative rendezvous, and planetary exploration.

The architecture of I3DS enables easy and low-cost configurations and reconfigurations of a robotic platform for any mission using the modular sensors, and allows vision-based and other exteroceptive sensors to be part of future exploration satellite platform's standard Guidance, Navigation and Control (GNC) units. It enables computing navigation solutions with on-board computers to be available for post-2020 missions with sophisticated autonomy, thus requiring only minimal intervention from Ground Control.

This paper mainly focuses on navigation solutions that can be enabled with the set of sensors available in the I3DS sensor suite. The structure of the paper is as follows: Section 2 describes the requirements and the framework of operations of Cranfield University within the I3DS project. Section 3 presents the theory behind the navigation algorithms, and finally Section 4 shows and discusses results obtained in a simulated environment.

2. The operation framework

The I3DS project aims to produce a sensor suite that is able to cope with many different scenarios. However, the most demanding one is represented by spacecraft uncooperative orbital rendezvous operations. In the case of close range observation at around 20m it is expected that the sensor suite can provide the relative distance and attitude of the target with an accuracy of (0.2m, 2.5°) in

attitude/position and of (0.01m/s, 0.1°/s) in speed/angular rate. For the final rendezvous, the aim is to maintain a "vicinity point" within the following boundaries on the position/attitude error of the servicer bus: (0.03m, 2.5°). The requirements are also listed in Table 1 and in [6].

Table 1. Basic requirements for the I3DS project

Mission Phase	Distance Range	Position Error	Angular Error
Flyaround and Inspection	20m	0.2m	2.5°
Close-range rendezvous	20m – 3m	0.03m	2.5°

In this paper only a subset of the entire sensor suite is analyzed for this purpose, namely the High Resolution (HR) camera, the stereo camera, and the LIDAR (see Fig. 1).

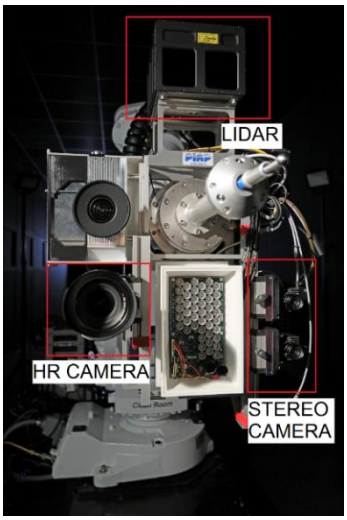

Fig. 1. Sensor suite mounted on chaser robotic arm with chaser. Courtesy of Thales Alenia Space France.

Fig. 2 shows the processing steps that are carried out on the raw sensor data, leading up to, and including the final relative pose of the target with respect to the chaser. Cranfield University is mainly responsible for pre-processing of the raw data, since navigation filters require data without distortions or outliers. The HR Camera and Stereo Camera produce, as input for navigation filters, both images and point clouds, in the case of HR in combination with a pattern projector [7],[8]. The LIDAR, on the contrary, just gives point clouds. Point clouds undergo further processing to remove outliers and to reduce size in order to gain time.

It is worth noting that, due to the computational load, some pre-processing algorithms, as well as navigation algorithms, may run on the on-board computer (OBC), or in the testing phase on the Electrical Ground Support Equipment (EGSE), whereas the other pre-processing solutions run on the Interface Control Unit (ICU).

3. The Navigation Algorithms

This section presents the navigation algorithms tested through the course of the I3DS project. The inputs for each relative-navigation solution are the pre-processed data, as briefly explained in Section 2 and detailed in [9].

3.1 Stereo Navigation

The algorithm is initialized when the first pair of stereo-images is received. A Kanade-Lucas-Tomasi (KLT) feature detector [10],[11] is applied to the left-camera image. The selection of left-camera image as reference frame for the pose estimation does not lead to a lack of generality, providing all successive computation steps are relative to it. The location of detected features is then refined to sub-pixel precision. It is worth noting that the sub-pixel accuracy refinement is done in a window around the feature location. Hence, the size of the window can affect the process if not properly tuned for the entire range of distances envisaged in the operations.

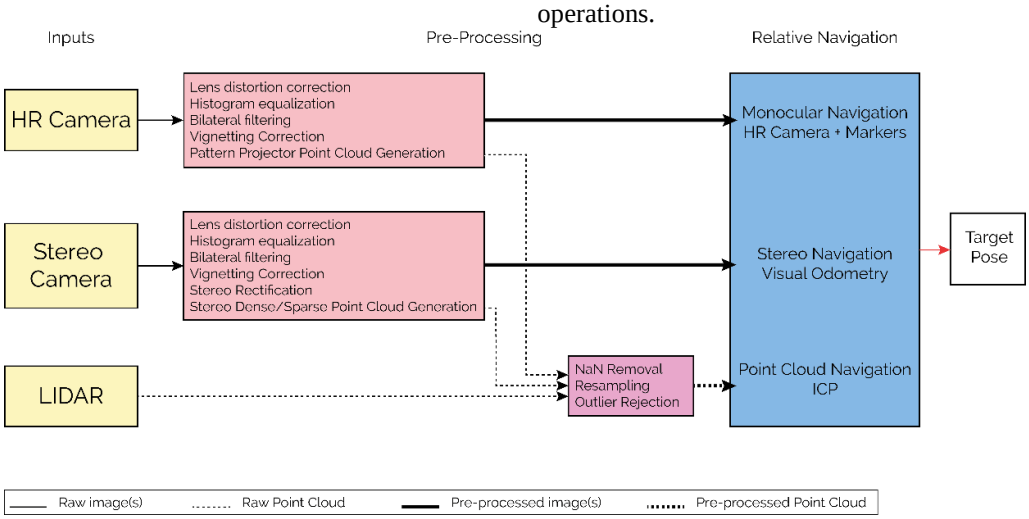

Fig. 2. Operations in the framework of pre-processing and navigation.

The sub-pixel refined positions of each detected feature in the left-camera image and the entire right-camera image are fed into a KLT tracker to obtain stereo matching of features. That is, given the location of feature in the left-camera image and a given window size, the tracker searches in the right-camera image for the corresponding point, whose surroundings within the window maximize the probability of a match. Even in the case of stereo matching with the KLT tracker, the window size has great importance, since it has to be sufficient to cover the entire range of disparities that can be encountered in the mission profile.

Since the images fed to the navigation filter are already undistorted and rectified, the matching of features in the two images has to be only along corresponding epipolar lines. Additionally, all the disparity values should have the same sign to be compliant with the stereo constraint. Therefore, after the stereo-matching procedure, a preliminary outlier removal is performed and pairs of features that do not respect the stereo constraint or that do not lie in the same epipolar line are discarded. Calling *t* the discrete time of the initialization step, once a new stereo pair is available, at time *t+1*, features in the left-camera image at time *t* are tracked with the KLT tracker in the left-camera image at time *t+1*. Simultaneously, features in the right-camera image at time *t* are tracked with the KLT tracker in the right-camera image at time *t+1*. This operation returns two sets of features, one in the left-camera image and one in the right-camera image, at time *t+1*. Stereo matching between these sets of features is obtained by verifying stereo constraint and matching features on the same epipolar line. Since the features are tracked from time *t* to time *t+1*, the number of correspondences *M* at the two time steps is

$$M_{t+1} \leq M_t \quad (1)$$

Therefore, the stereo-matched pairs at time *t* are further refined, by discarding the ones that have no match at time *t+1*.

Once the correspondences are obtained, and given the camera calibration parameters, it is possible to obtain a non-linear mapping of the point correspondence *h* between image at time *t* and image at time *t+1* via the trifocal constraint. If the time between the two pair of images is known, then, the motion is retrieved by applying an Extended Kalman Filter strategy [12]. Under the assumptions that the noise is zero-mean white noise and that the system noise and measurement noise are uncorrelated, the instantaneous ego-motion state

$$e_m = (V_X, V_Y, V_Z, \omega_X, \omega_Y, \omega_Z) \quad (2)$$

where *V* is the velocity and ω is the angular velocity, are integrated and filtered with the measurements *h*, to retrieve the pose, i.e. rotation *R* and translation *T*.

The entire procedure to obtain stereo visual odometry is also summarized in Fig. 3. The final output of the stereo-navigation is the relative translation along the three principal components and the relative orientation, as a unit quaternion, of the target with respect to the first left-camera frame. Therefore, if distance and orientation between camera, and hence chaser, and target is known at an initial time, it is possible to obtain at each time a relative pose.

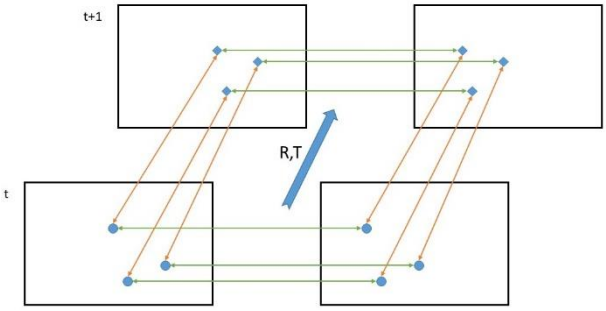

Fig. 3. Scheme for stereo pose estimation. The matched features at time *t* are tracked at time *t+1* and Pose Estimation in terms of rotation and translation is obtained.

3.2 Monocular Navigation with Fiducial Markers

The I3DS project deals with solutions for cooperative and uncooperative targets. When dealing with cooperative targets, an efficient strategy involves the detection of fiducial markers, whose three-dimensional position in the target reference frame is known. Fiducial markers are patterns that are robustly detectable with a clearly defined centre point, e.g. a 2x2 checkerboard, and placed at known locations. An example of fiducial markers is shown in Fig. 4.

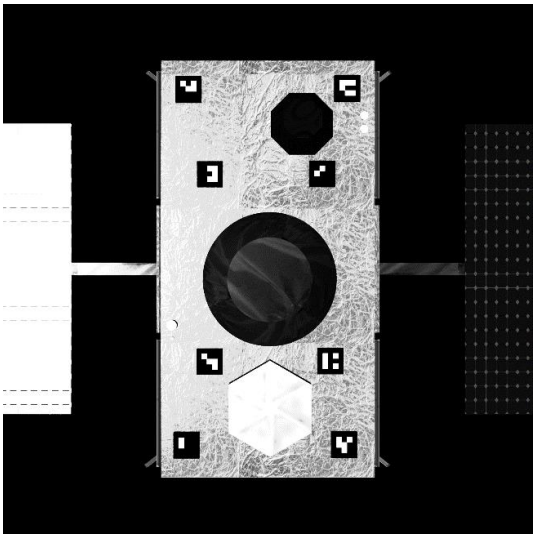

Fig. 4. Example of simulated target with custom ArUco markers.

The proposed algorithm makes use of two custom dictionaries of fiducial markers, based on ArUco libraries

[13]. One dictionary is for medium-range operations while the other is for close-range operations. The core of the navigation solution is the Perspective-n-Point (PnP) algorithm [14]. The general formulation of the PnP problem requires finding the transformation T_k that minimizes the image re-projection error

$$\min_{T_k} \sum_{i=1}^{N} \left\| p_k^i - \hat{p}_{k-1}^i \right\|^2 \qquad (3)$$

where p_k^i is the position of feature i in the image k, $\hat{p}_{k-1}^i$ is the reprojection of the 3-D point X_{k-1}^i into image I_k according to the transformation T_k . The minimal case involves four 3D-to-2D correspondences to provide an unambiguous solution.

The algorithm is initialized when an image is received from the monocular camera. A scale-invariant and rotation-invariant template matching finds the fiducial markers from the first dictionary in the image and establishes a set of Regions of Interest (ROIs) around them [15],[16]. Once the ROIs are set, the centre of each marker is detected at sub-pixel accuracy. If at least four markers have been detected, their 2D position in the image is matched with the corresponding 3D position and the PnP algorithm provides the pose estimate.

Then the navigation filter searches for fiducial markers from the second dictionary in the same fashion. It is worth noting that if no markers from the first dictionary are found, then relative pose is computed with the second set of fiducial markers. If, on the contrary, pose is already detected with first set of markers and second-dictionary markers are detected, the already available pose is refined. The entire workflow is depicted in Fig. 5.

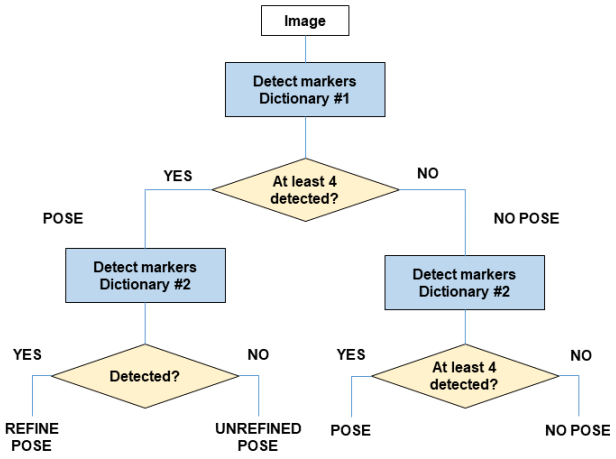

Fig. 5. Workflow of HR-camera navigation algorithm.

Since there is direct correspondence between 2D and 3D points and the camera parameters are known, the output of the monocular navigation with fiducial markers is the relative position and orientation of the target reference frame with respect to the camera reference frame. Hence, further known translation and rotation should be applied for control purposes.

3.3 Point Cloud-based Navigation

As seen in section 2, LIDAR as well as cameras are able to produce clouds of points. The relative-pose estimation can be obtained with the presence of dense 3D point clouds by means of point cloud registration. The proposed methods are the Iterative Closest Point (ICP) and the Histogram of Descriptors (HoD) + Kalman Filter (KF).

The ICP algorithm is well-known and traditionally established for point cloud registration [17],[18]. The general flow of the ICP algorithm is shown in Fig. 6. It iteratively searches for the best transformation between two point clouds until stopping criteria are met. The three different stopping criteria hereby considered are:

(i) maximum number of iterations; (ii) difference value between the previous transformation and the current estimated transformation; (iii) threshold for the Euclidean squared errors.

The first step of the algorithm involves establishing correspondences between points in the two point clouds. The correspondence is obtained by finding the nearest neighbour of a point in the first point cloud within the second point cloud, and it is calculated via a k-d tree.

In the second step the algorithm searches for the best transformation matrix between the corresponding points in the two point clouds by means of Singular Value Decomposition (SVD). Indeed, given two point clouds, $P = \{p_1, \dots, p_n\}$ the target point cloud, and $Q = \{q_1, \dots, q_n\}$ the source point cloud, the translation T and the rotation R are computed by minimizing the mapping error E:

$$E(R,T) = \sum_{I=1}^{N_p} \sum_{j=1}^{N_q} w_{i,j} \left\| p_i - \left(R q_j + T\right) \right\|^2 \qquad (4)$$

where N_p and N_q are the number of points in the two point clouds. The values $w_{i,j}$ represent the weights of the correspondence between the point p_i in the target point cloud and the point q_j in the source point cloud. If the correspondence is verified, then $w_{i,j} = 1$, otherwise $w_{i,j} = 0$.

Finally, the transformation matrix is applied to the source point cloud. The algorithm iteratively calculates the transformation matrix between the last estimated point cloud and the target point cloud until one of the stopping criterion is met.

It has to be noted that the advantage of the ICP algorithm is its simplicity and its optimization in terms of computational time, due to extended literature interest. However, because of its minimization process, it can result in a local minimum, which will not correspond to the global minimum.

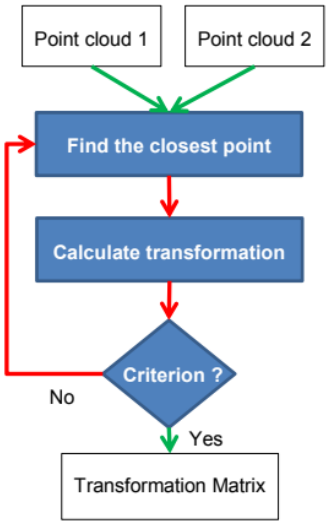

Fig. 6. General flow of ICP algorithm.

The second possible method involves assigning HoD descriptors [19] to all the points in the point cloud at discrete time t and to all the point in the point cloud at time $t+1$. Then, correspondences between the descriptors are sought and thus point-cloud registration is performed.

The point cloud registration is finally achieved by means of a Kalman Filter [20]. The KF process is divided in two main parts: the prediction and the correction. In our registration situation, we want to find the rotation matrix R and the translation vector T to minimize the error (4). In order to include the rotation matrix and translation vector into the KF process, the state vector is defined as the concatenation vector of the rotation values and the translation values as

$$s^T = \begin{bmatrix} (R_{11}, R_{12}, R_{13})^T \\ (R_{21}, R_{22}, R_{23})^T \\ (R_{31}, R_{32}, R_{33})^T \\ (T_x, T_y, T_z)^T \\ (\dot{T}_x, \dot{T}_y, \dot{T}_z)^T \\ (\ddot{T}_x, \ddot{T}_y, \ddot{T}_z)^T \end{bmatrix} \quad (5)$$

In addition, linear velocity and acceleration are added to increase accuracy.

The prediction state equation has no input control and the observation matrix is in the form of

$$H = \begin{bmatrix} m & 0 & 0 & 1 & 0 & 0 & \Delta t & 0 & 0\Delta t^2 & 0 & 0 \\ 0 & m & 0 & 0 & 1 & 0 & 0 & \Delta t & 0\ 0 & \Delta t^2 & 0 \\ 0 & 0 & m & 0 & 0 & 1 & 0 & 0 & \Delta t\ 0 & 0 & \Delta t^2 \end{bmatrix} \quad (6)$$

where m is the measurement and Δt is the time between two measurements. The KF process can be started to obtain the transformation matrix from the final state vector s.

The flow for HoD+Kalman algorithm is represented in Fig. 7.

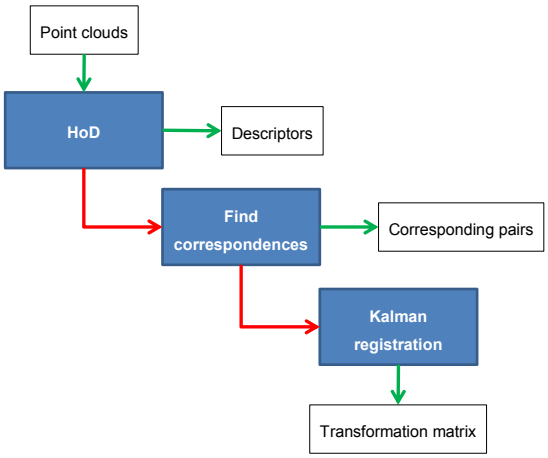

Fig. **7**. General flow of HoD+Kalman algorithm.

4. Results and discussion

Preliminary results for the navigation solutions within I3DS have been obtained by making use of images simulated with SPICAM software in Thales Alenia Space France, the coordinator of the project. The navigation filters, developed in the C++ language, have not yet been tested within the robotic facility for I3DS. The analyses of results hereby shown are mainly of preliminary Straight-Line Approach trajectories for algorithm development purposes.

The characteristics of the simulated High-Resolution monocular camera and stereo-camera are listed in Table 2.

Table 2. Parameters of HR and stereo cameras

	Stereo	HR
Focal length (*mm*)	12	21
Pixel pitch (μm)	5.5	5.5
Principal point x (*pixel*)	1024	1024
Principal point y (*pixel*)	1024	1024
Baseline (*cm*)	15	

4.1 Stereo Navigation

The first trajectory is a straight-line from 20m to 5m with the chaser approaching the target at a speed of 5 cm/s. The motion is along the camera z-axis and the target z-axis is perfectly aligned with the camera z-axis, therefore no yaw, pitch and roll are present.

For the case of stereo-based navigation, despite the solution being able to run at speeds greater than 10 Hz, a framerate of 10 Hz led to erroneous results. This is because at a speed of 5cm/s and framerate of 10 Hz, the inter-frame motion would be 5mm, which is too small for the proposed stereo-camera to track the feature motion properly. Indeed, the inter-frame disparity is too small and consequently the signal to noise ratio of the tracked features is insufficient to allow for filter convergence.

Therefore, different framerates have been tested, namely 1Hz and 2Hz.

Fig. 8 and Fig. 9 show the estimated trajectory and the estimated attitude (in blue), respectively, together with the ground truth (in red). For this analysis it is more intuitive to express the attitude using the yaw, pitch, roll convention, instead of using quaternions as this will give a more thorough understanding of the results. Conversion to quaternions for control purposes is however straightforward. Overall, the estimated pose follows the ground truth trajectory, even though errors on the final position are present. It is therefore necessary to analyse the errors to have a better indication of the current performance and eventually improve the tuning.

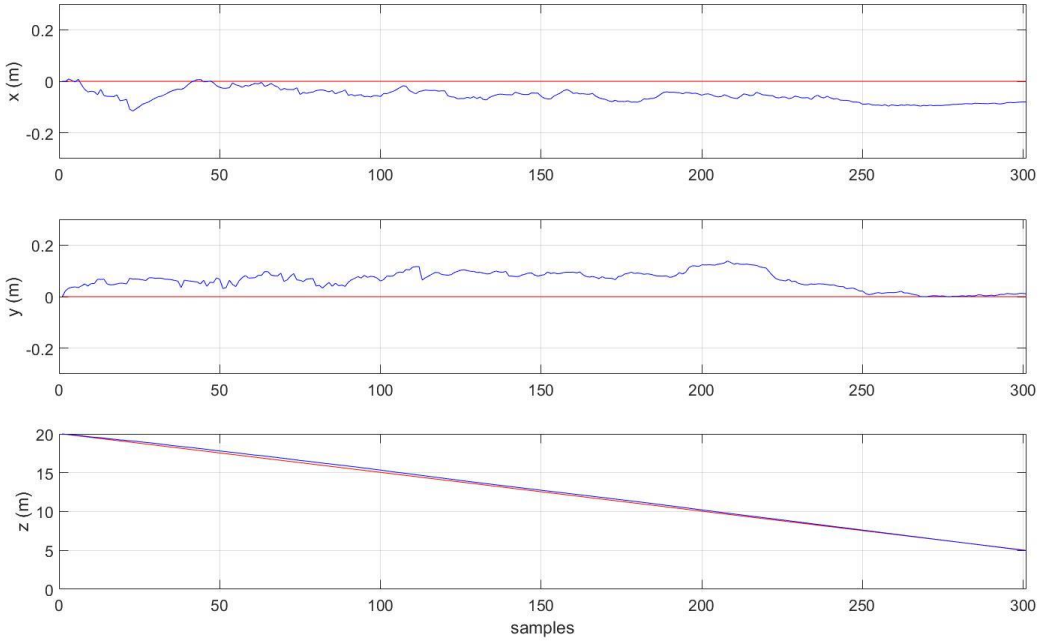

Fig. 8. Estimated trajectory (red) and ground truth (blue) for straight-line approach with stereo navigation filter.

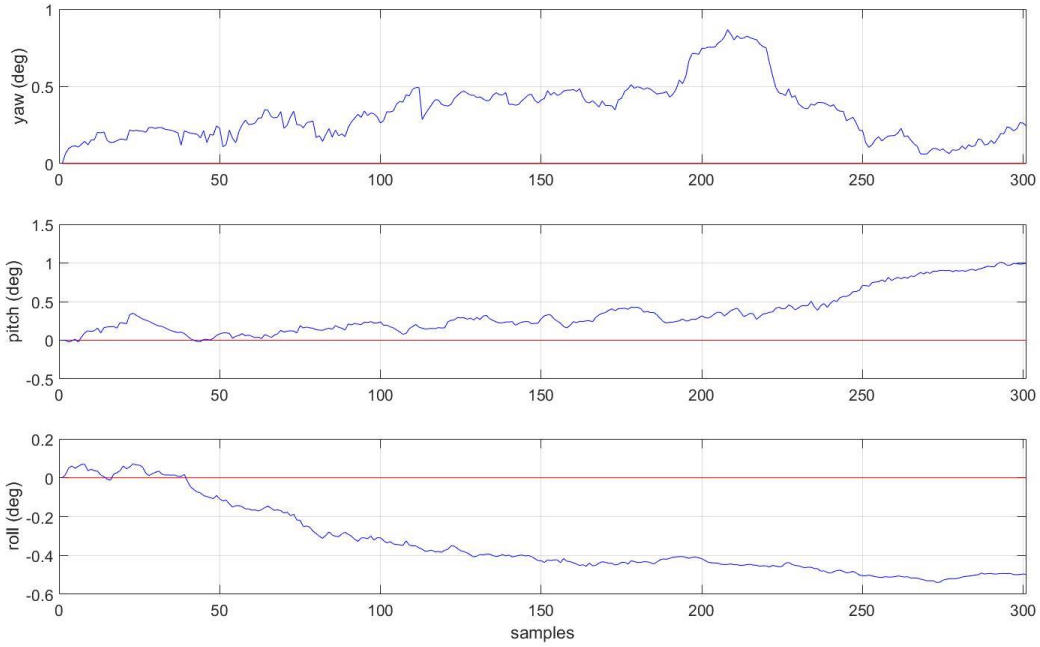

Fig. 9. Estimated attitude (red) and ground truth (blue) for straight-line approach with stereo navigation filter.

The inter-frame motion errors are analysed first. The inter-frame motion error allows an understanding of the performance on a frame-by-frame level. The plot in Fig. 10 shows the inter-frame errors for the trajectory. Larger inter-frame errors are present at the beginning of the trajectory, when the platform is further away. This can be explained with the fact that the disparity, i.e. the difference in pixels along the baseline direction between corresponding points in left and right images, is very small and hinders the process. Indeed, since depth is inversely proportional to disparity, very slight errors in disparity due to image noise, when the overall disparity is very small, can lead to very large errors in depth. Additionally, also the difference in motion of the features between two successive frames at large distance is minimal and thus may affect the accuracy of the results. When the target gets closer, the algorithm tracks the attitude much better and the inter-frame error decreases to the sub-centimetre level. The plot in Fig. 11 shows the inter-frame errors for attitude. In general, they show an almost constant trend, fluctuating around zero value.

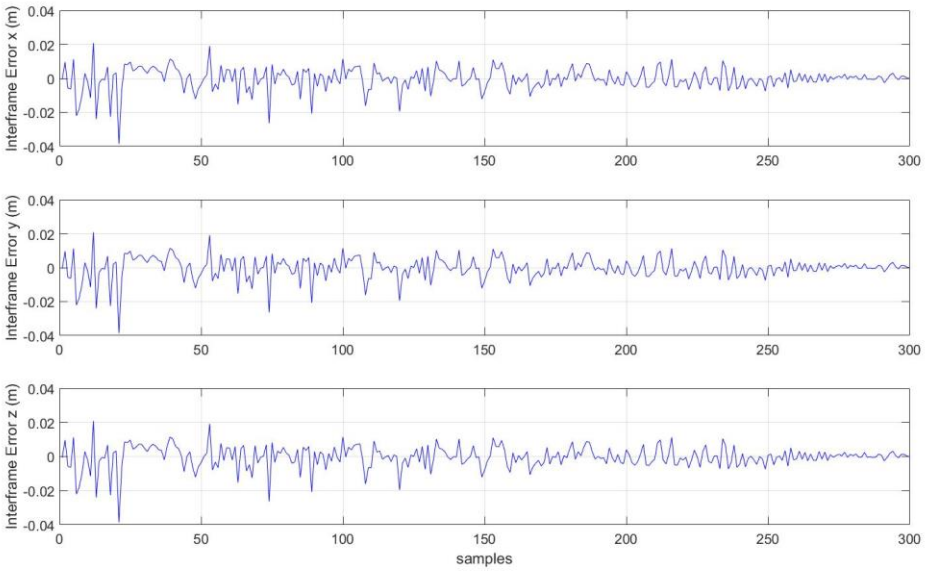

Fig. 10. Inter-frame errors for trajectory in straight-line approach with stereo navigation filter.

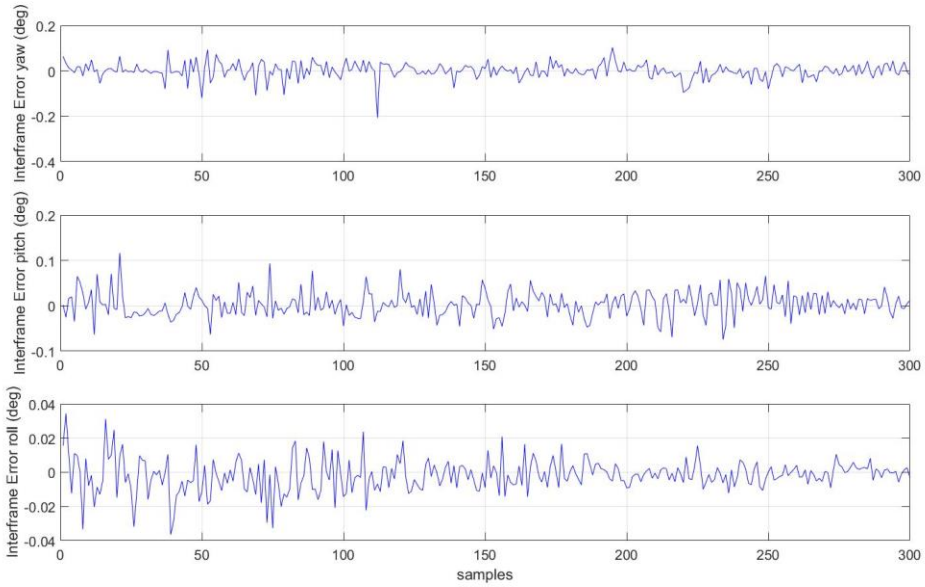

Fig. 11. Inter-frame errors for attitude in straight-line approach with stereo navigation filter.

The sum of inter-frame motion errors gives the cumulative errors. Cumulative errors are plotted in Fig. 12 for trajectory and Fig. 13 for attitude. The final error in the trajectory is smaller than 2% of the entire path. However, the inter-frame errors accumulate and give fluctuations. Hence, further refinements should be tested to reduce these errors and perhaps also the effects of using different baselines.

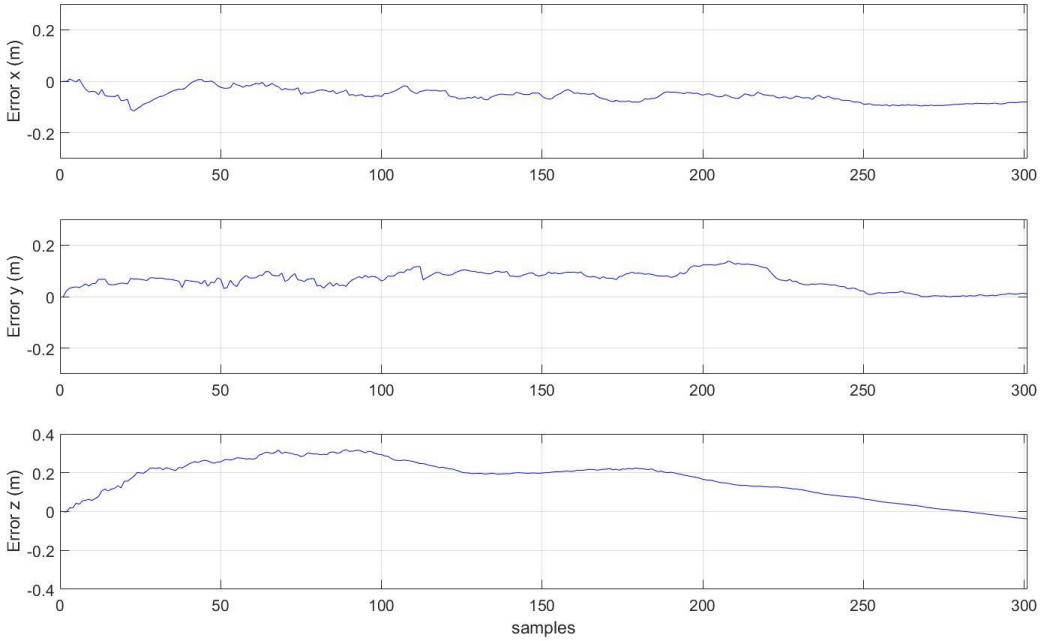

Fig. 12. Cumulative errors for trajectory in straight-line approach with stereo navigation filter.

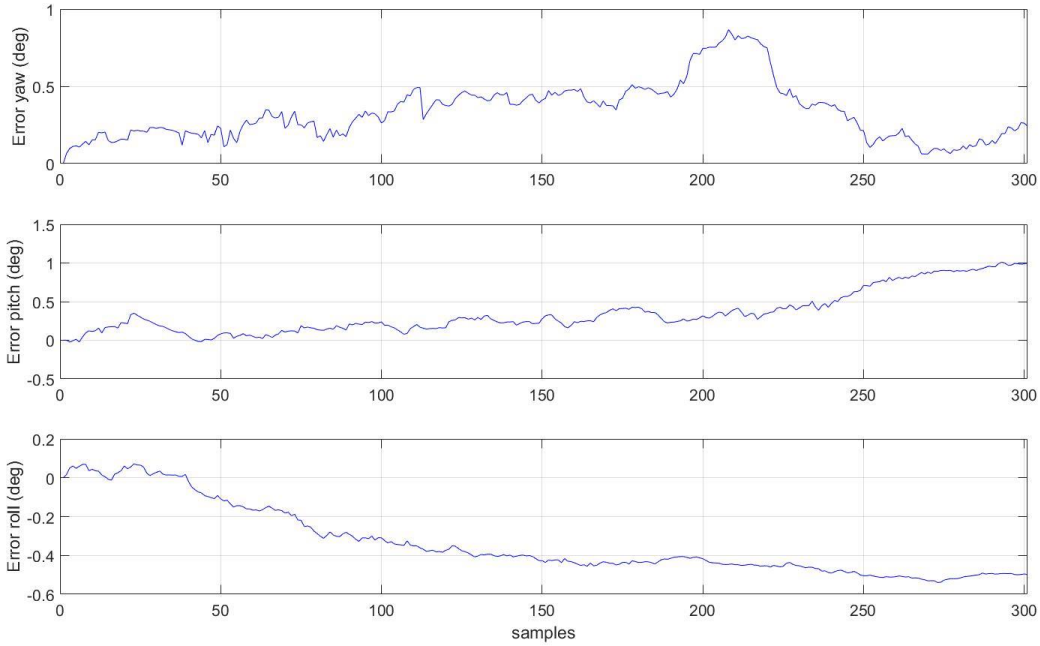

Fig. 13. Cumulative errors for attitude in straight-line approach with stereo navigation filter.

The second test, with 2 Hz framerate has been analyzed with a shorter travelled distance, i.e. from 12.5m to 5m. The estimated trajectory and attitude are represented in Fig. 14 and .Fig. 15 Despite the shorter distance, the errors appear larger than previous case. This is related to the smaller inter-frame step, which in this case is 2.5 cm.

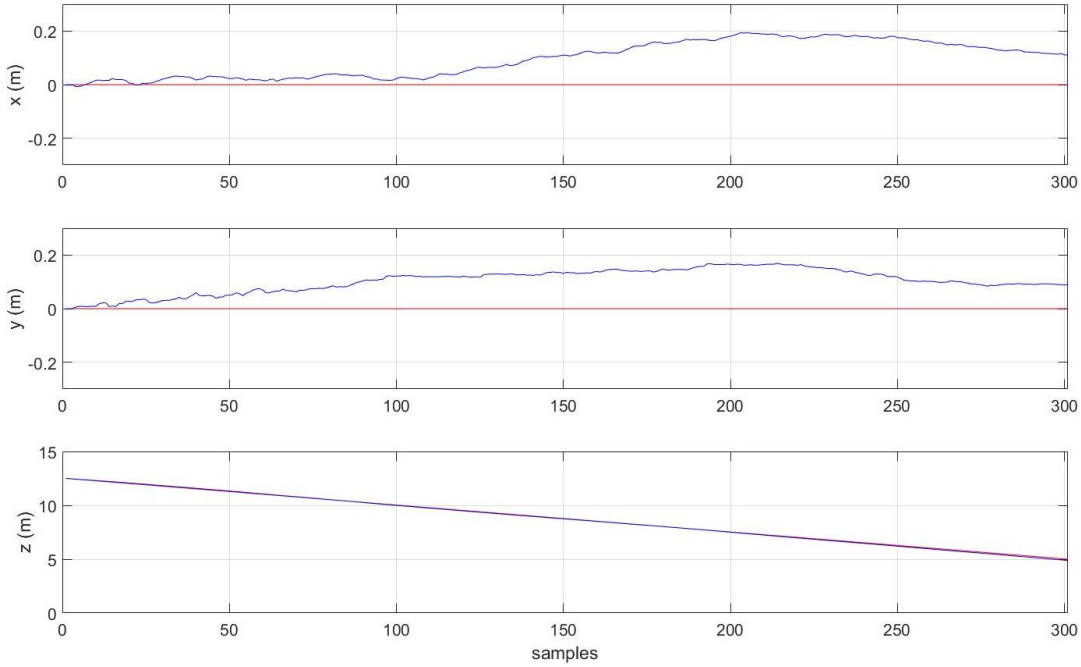

Fig. 14. Estimated trajectory (blue) and ground truth (red) for straight-line approach with stereo navigation filter. Second test.

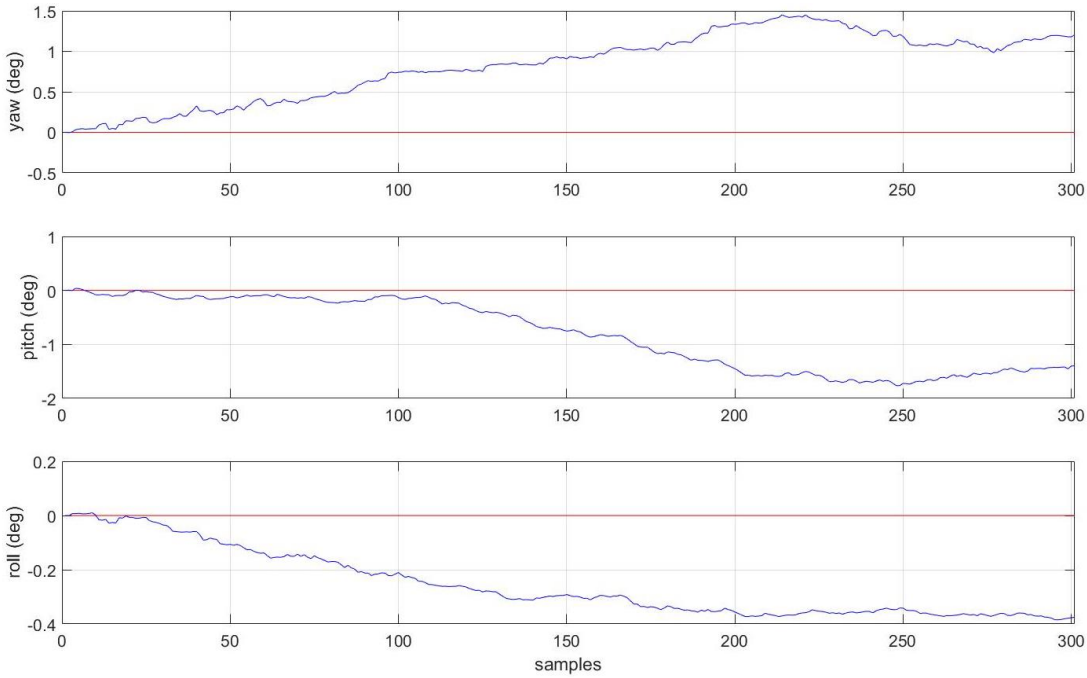

Fig. 15 Estimated attitude (blue) and ground truth (red) for straight-line approach with stereo navigation filter. Second test.

The inter-frame motion errors are then analysed. The plot in Fig. 16 shows the inter-frame errors for trajectory. The errors fluctuate around zero value and are small but since the inter-frame step is smaller as well, their significance has increased. The plot in Fig. 17 shows the inter-frame errors for attitude.

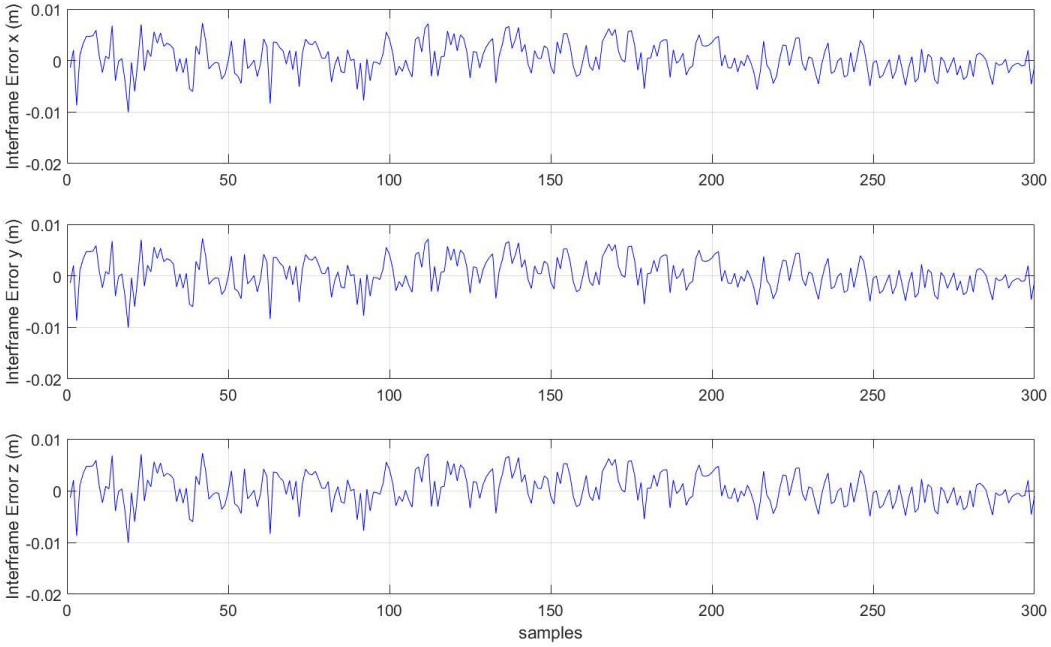

Fig. 16. Inter-frame errors for trajectory in straight-line approach with stereo navigation filter. Second test.

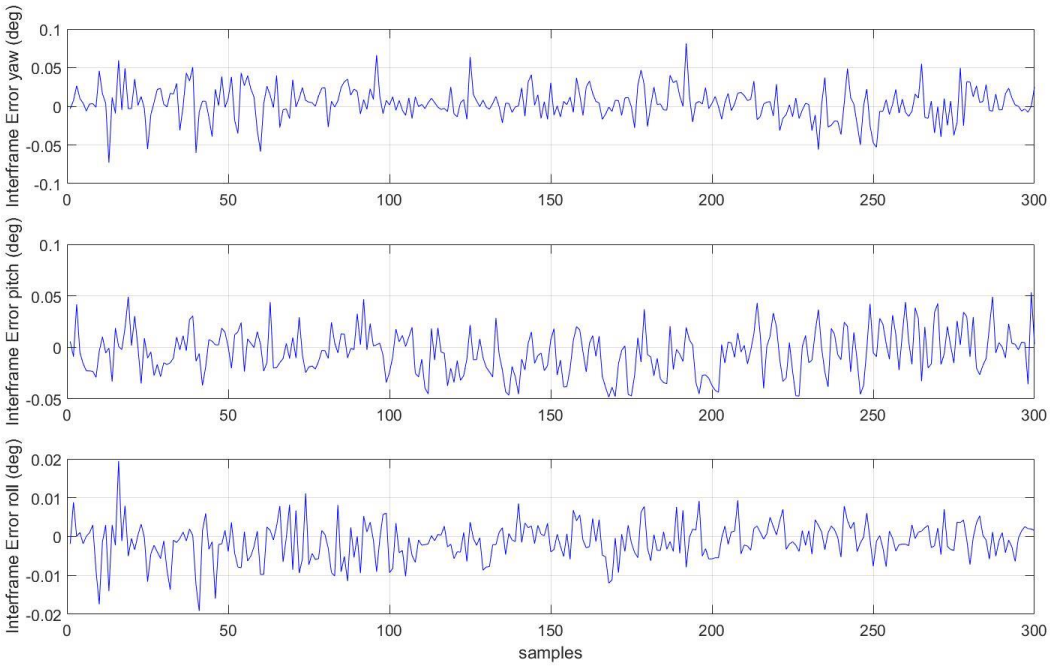

Fig. 17. Inter-frame errors for attitude in straight-line approach with stereo navigation filter. Second test.

Fig. 18 and Fig. 19 show the final errors for the trajectory and attitude, respectively. Despite the errors being small, since the inter-frame motion is smaller it was expected to result in a larger error with respect to the previous case due to the reduced capability in discrimination of very close motion. These results should serve as a way to improve the tuning.

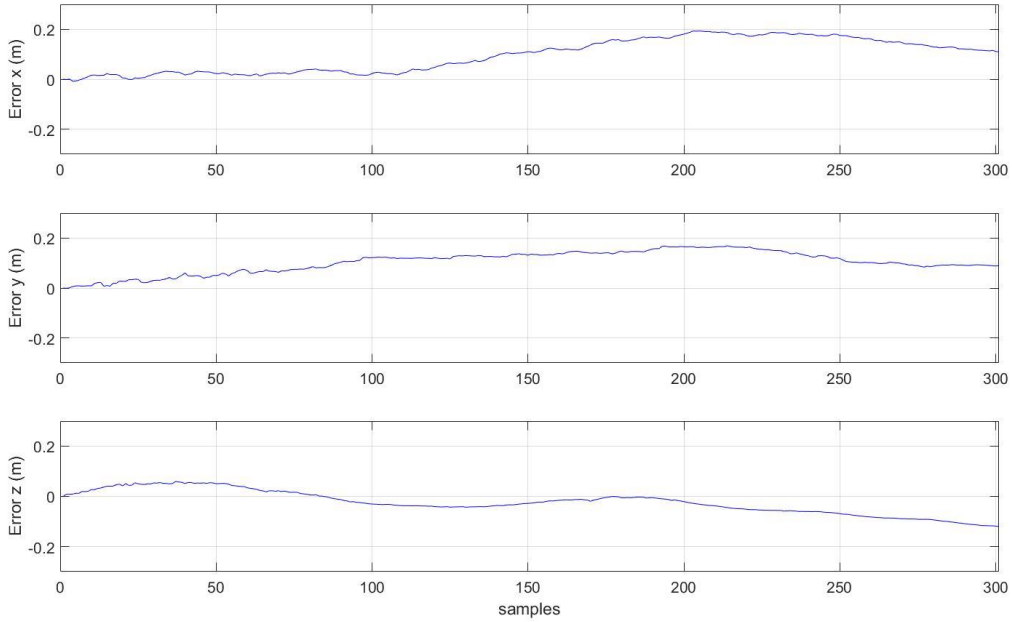

Fig. 18. Cumulative errors for trajectory in straight-line approach with stereo navigation filter. Second test.

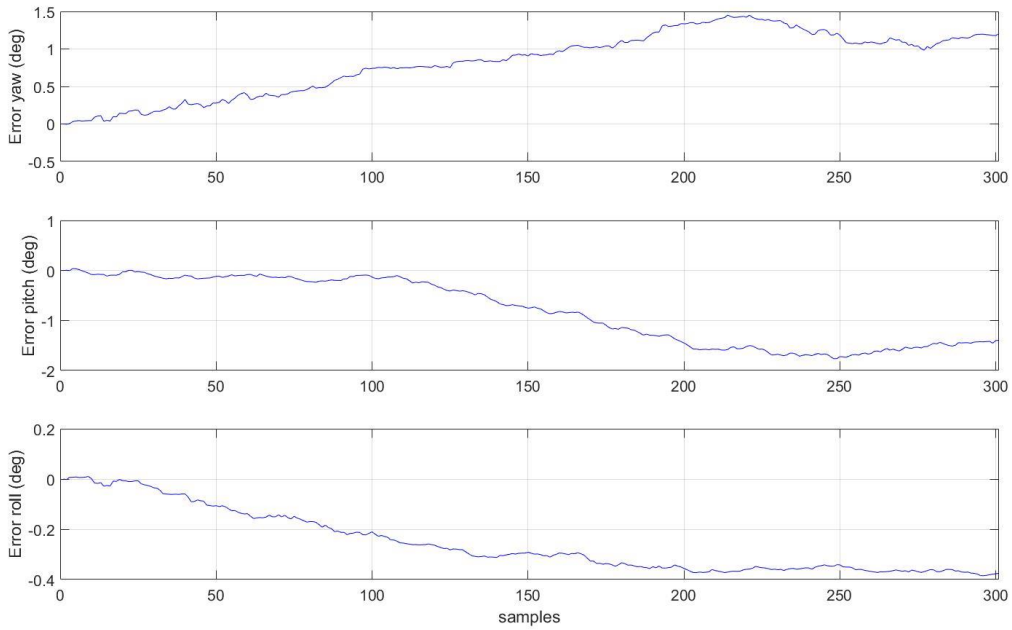

Fig. 19. Cumulative errors for attitude in straight-line approach with stereo navigation filter. Second test.

The second analyzed case is again a straight line trajectory at a slower framerate and with the chaser keeping the distance and attitude stable in the last part. The mission profile is shown in Fig. 20 and Fig. 21. The framerate is about 0.13 Hz.

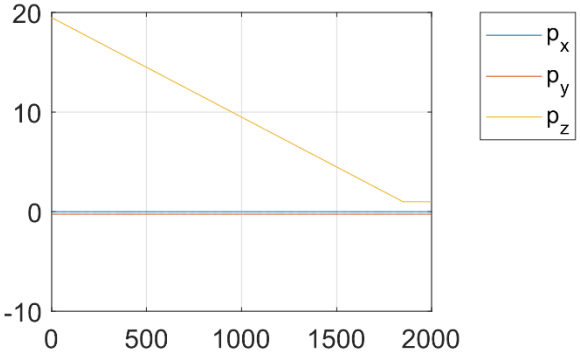

Fig. 20. Translation profile mission.

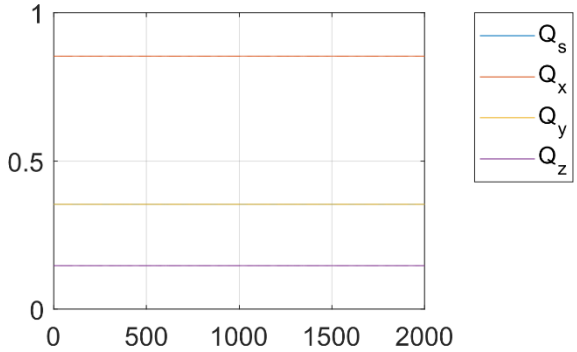

Fig. 21. Attitude profile mission.

The estimated trajectory, as shown in Fig. 22 and Fig. 23, follows properly the ground truth tracks. A detailed analysis of the translation error as a function of covered distance (see Fig. 24) shows a final error smaller than 1%.

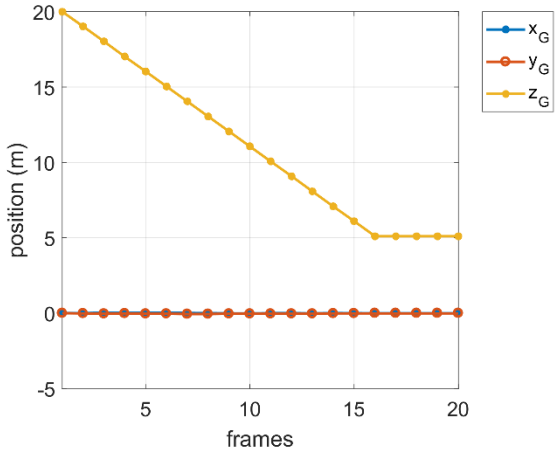

Fig. 22. Reconstructed translation for the second navigation case.

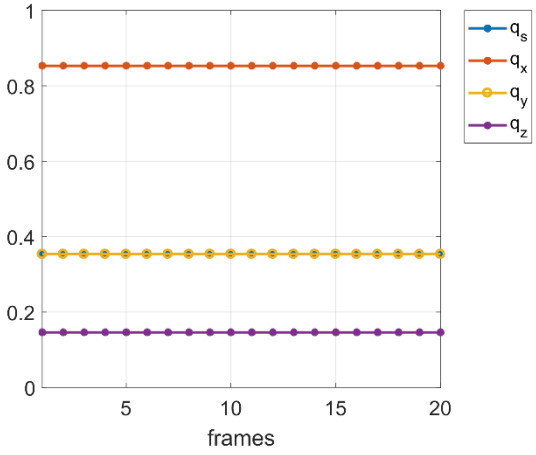

Fig. 23. Reconstructed quaternion of orientation for the second navigation case.

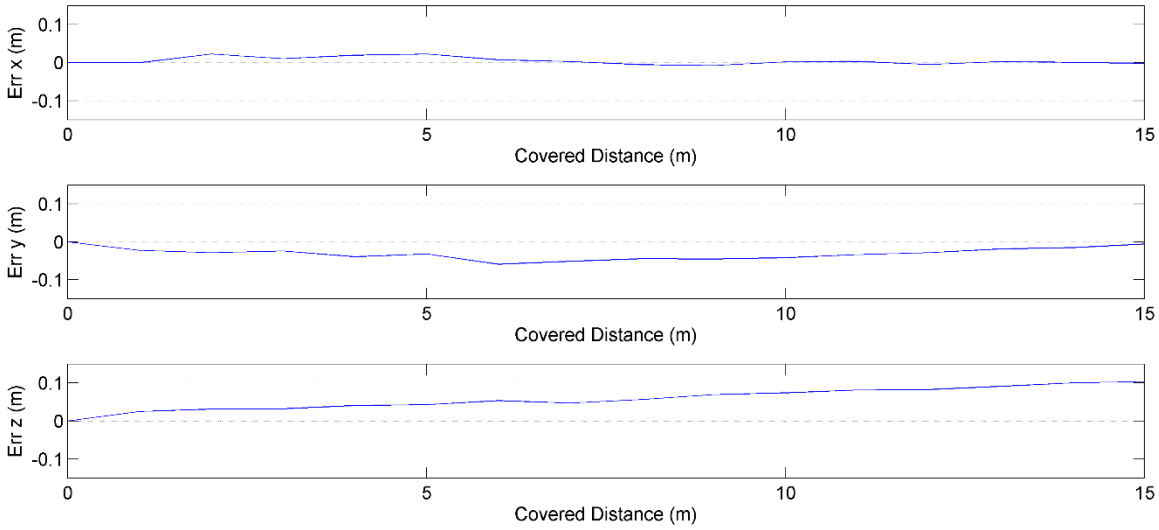

Fig. 24. Translation error as function of covered distance for stereo navigation.

4.2 Navigation with Fiducial Markers

The first trajectory is a straight-line from 20m to 5m with the chaser approaching the target at a speed of 5 cm/s. The motion is along the camera z-axis and the target z-axis is perfectly aligned with the camera z-axis, therefore no yaw, pitch and roll are present. Since for this kind of navigation filter the errors do not sum up, the plots shown will involve exclusively the errors for each frame.

A detailed error analysis is presented in Fig. 25 for the trajectory and Fig. 26 for the attitude. Throughout the entire mission the error in position of the target is almost constant and below 2 cm. At about 6m, the 4 markers leave the FOV, causing a slight variation in the attitude estimate and position. This is natural, since the PnP is an optimization method and the more points detected and associated in the target, the better the estimate. The major effect of the missing markers is in the roll in this case. It is worth noting that misplacement of the 2D position of the centre of the markers in the image affected mainly the roll and pitch angles, as shown in Fig. 26. In general, it clearly depends on the position of the undetected markers.

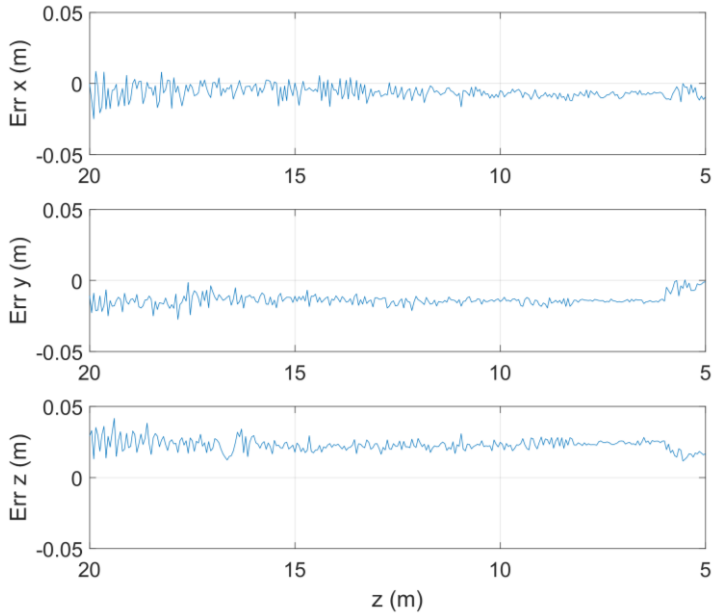

Fig. 25. Trajectory errors for straight-line approach with markers-only filter.

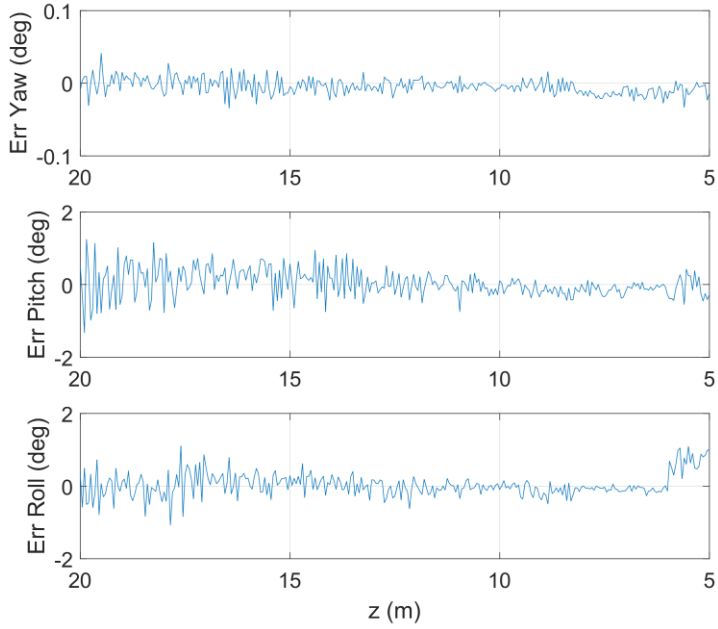

Fig. 26. Attitude errors for straight-line approach with markers-only filter.

The second analyzed case is again a straight line trajectory at a slower frame-rate and with the chaser keeping the distance and attitude stable in the last part. The mission profile is shown in Fig. 20 and Fig. 21. The frame-rate is about 0.13 Hz.

The estimated trajectory, as shown in Fig. 27 and Fig. 28, properly follows the ground truth tracks. It is worth noting that reconstruction of the trajectory stops when the fiducial markers are not in view anymore.

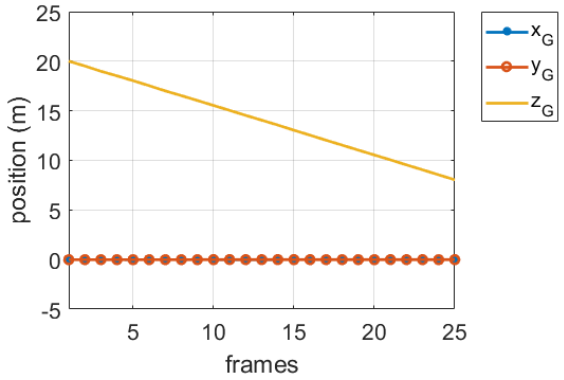

Fig. 27. Reconstructed translation for the second navigation case.

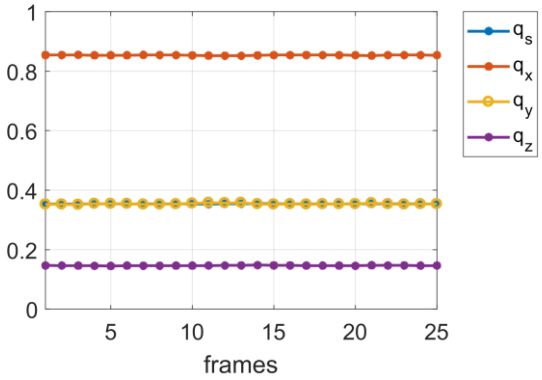

Fig. 28. Reconstructed quaternion of orientation for the second navigation case.

4.3 Point Cloud-based Navigation

The simulated case for point cloud navigation is with LIDAR. A straight-line approach trajectory is presented even in this case, as shown in Fig. 29, together with a Generalized ICP reconstruction.

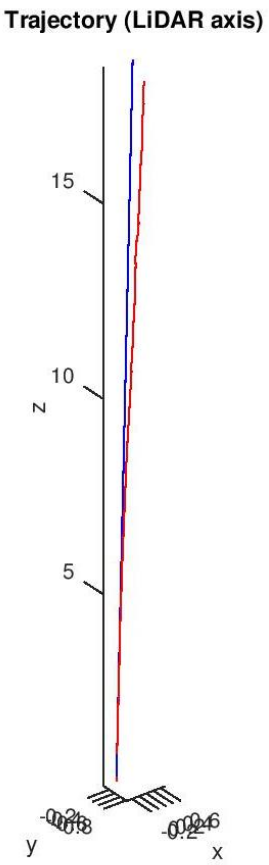

Fig. 29. Ground truth (blue) and LIDAR-reconstructed trajectory (red) for straight-line approach mission.

The error in the trajectory is shown in Fig. 30 and the percentage error is shown in Fig. 31. It can be seen that overall the error is kept below 5%, but due to the very small movement along the x and y axes, the inter-frame errors are accumulated. In the direction of motion, however, the error was kept low.

Conclusions

The work has presented the main results of the navigation algorithms proposed for the H2020 I3DS project. Some sensors in the sensor suite have been selected to assess the relative-navigation performance, namely the stereo camera, monocular HR camera, and LIDAR. The proposed solutions can keep errors small, even though in some case it might be necessary to further refine the tuning to reduce the difference between the ground truth and the estimated pose even further, thus being fully compliant with the desired requirements of I3DS. One main outcome of the research is that in some cases, when errors accumulate, as per stereo-navigation and point-cloud based navigation, it may be necessary to relax the constraints on frequency of execution to avoid the system noise adding up to an unacceptable extent.

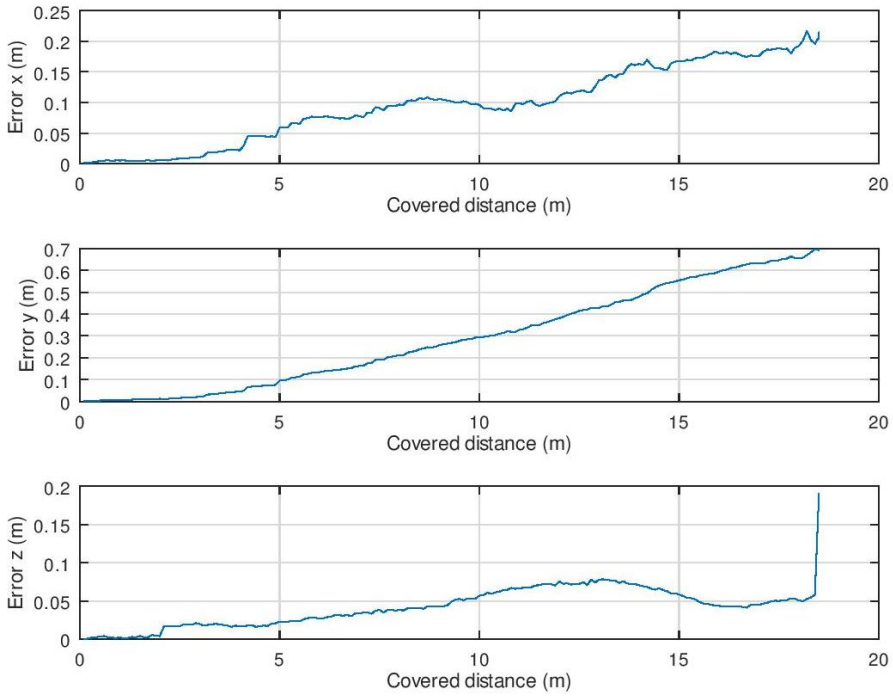

Fig. 30. Reconstructed quaternion of orientation for the second navigation case.

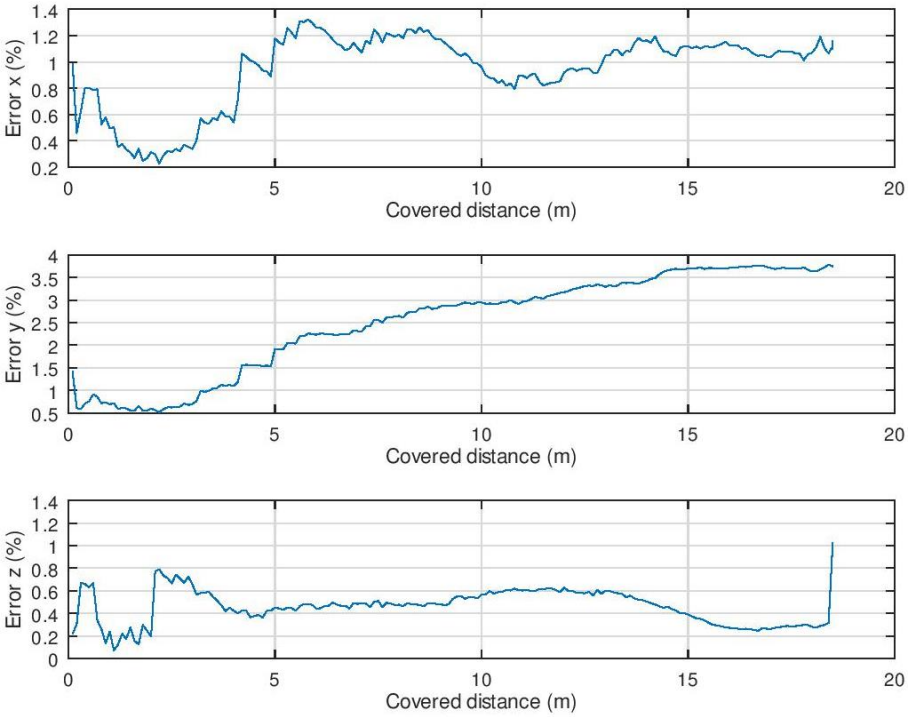

Fig. 31. Reconstructed quaternion of orientation for the second navigation case.

Acknowledgements

This project brings together the following companies throughout Europe: THALES ALENIA SPACE - France / Italy / UK / Spain, SINTEF (Norway), TERMA (Denmark), COSINE (Netherlands), PIAP Space (Poland), HERTZ Systems (Poland), and Cranfield University(UK).

I3DS is co-funded under Horizon 2020 EU research and development program and part of the Strategic Research Cluster on Space Robotics Technologies as the Operational Grant n°4 among 6. It has received funding

from the European Union's Horizon 2020 research and innovation program under grant agreement N°730118.

References

[1] A. Tatsch, N. Fitz-Coy, S. Gladun, On-orbit servicing: a brief survey, Proceedings of the 2006 Performance Metrics for Intelligent Systems Workshop, Gaithersburg, MD, USA, Aug. 2006.

[2] A. Flores-Abad, O. Ma, K. Pham, S. Ulrich, A review of space robotics technologies for on-orbit servicing, Prog. Aerosp. Sci. 68 (2014) 1–26.

[3] X. Clerc, I. Retat, Astrium vision on space debris removal, Proceedings of the 63rd International Astronautical Congress, Naples, Italy, 1–5 Oct. 2012.

[4] C. Bonnal, J.M. Ruault, M.C. Desjean, Active debris removal: recent progress and current trends, Acta Astronaut. 85 (2013) 51–60.

[5] R. Opromolla, G. Fasano, G. Rufino, M. Grassi, A review of cooperative and uncooperative spacecraft pose determination techniques for close-proximity operations, Progress in Aerospace Sciences 93 (2017), 53-72.

[6] D1.2: Use-case Identification, I3DS "Integrated 3D Sensors suite", 2017, COMPET-4-2016: SRC - Space Robotics Technologies – "Inspection Sensor Suite". Grant Agreement n° 730118. http://i3ds-h2020.eu/images/OG4_I3DS_D1.2-Use_case_Identification_PublicRelease.pdf (accessed 24/09/2018).

[7] G. Sansoni, M. Carocci, R. Rodella, Three-dimensional vision based on a combination of gray-code and phase-shift light projection: analysis and compensation of the systematic errors, Appl Opt (1999) 6565–6573.

[8] S. Zhang, High-Speed 3D Imaging with Digital fringe Projection Techniques, CRC Press, 2016.

[9] M. Estébanez Camarena, L. M. Feetham, A.F. Scannapieco, N. Aouf, FPGA-based multi-sensor relative navigation in space: Preliminary analysis in the framework of the I3DS H2020 project, IAC-18-B2.IP-20-x46175, 69th International Astronautical Congress, Bremen, Germany, 2018, 1 – 5 October..

[10] C. Tomasi, T. Kanade, Detection and Tracking of Point Features, International Journal of Computer Vision (1991).

[11] J. Shi, C. Tomasi, Good features to track, Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 1994.

[12] B. Kitt, A. Geiger, H. Lategahn, Visual odometry based on stereo image sequences with RANSAC-based outlier rejection scheme, 2010 IEEE Intelligent Vehicles Symposium, San Diego, CA, 2010.

[13] S. Garrido-Jurado, R. Muñoz-Salinas, F.J. Madrid-Cuevas, M.J. Marín-Jiménez, Automatic generation and detection of highly reliable fiducial markers under occlusion, Pattern Recognition (2014) 2280-2292.

[14] R. Hartley, A. Zisserman, Multiple View Geometry in Computer Vision, second ed., Cambridge University Press, 2003.

[15] D. G. Lowe, Object recognition from local scale-invariant features, Comput. Vis. 1999 Proc. Seventh IEEE Int. Conf. On (1999) 1150–1157.

[16] D. G. Lowe, Distinctive Image Features from Scale-Invariant Keypoints, Int J Comput Vis. (2004) 91–110.

[17] A. Segal, D. Haehnel, S. Thrun, Generalized ICP, Robotics: Science and Systems V (2009) .

[18] R. Opromolla, G. Fasano, G. Rufino, M. Grassi, Uncooperative pose estimation with a LIDAR-based system, Acta Astronautica (2015).

[19] O. Kechagias-Stamatis, N. Aouf, Histogram of distances for local surface description, 2016 IEEE International Conference on Robotics and Automation (ICRA), Stockholm, 2016.

[20] A. Amamra, N. Aouf, D. Stuart, M. Richardson, A recursive robust filtering approach for 3D registration, SIViP (2016) 835-842.