Acta Astronautica 140 (2017) 1–9

Contents lists available at ScienceDirect

Acta Astronautica

journal homepage: www.elsevier.com/locate/actaastro

Self-supervised learning as an enabling technology for future space exploration robots: ISS experiments on monocular distance learning

Kevin van Hecke [a,*], Guido C.H.E. de Croon [a,**], Daniel Hennes [b], Timothy P. Setterfield [c], Alvar Saenz-Otero [c], Dario Izzo [b]

[a] MAV-lab, Control and Simulation, Faculty of Aerospace Engineering, TUDelft, The Netherlands
[b] Advanced Concepts Team of the European Space Agency, ESTEC, Noordwijk, The Netherlands
[c] Space Systems Lab, Massachusetts Institute of Technology, Cambridge, MA, USA

ARTICLE INFO

Keywords:
Persistent self-supervised learning
Stereo vision
Monocular depth estimation
Space robotics

ABSTRACT

Although machine learning holds an enormous promise for autonomous space robots, it is currently not employed because of the inherent uncertain outcome of learning processes. In this article we investigate a learning mechanism, Self-Supervised Learning (SSL), which is very reliable and hence an important candidate for real-world deployment even on safety-critical systems such as space robots. To demonstrate this reliability, we introduce a novel SSL setup that allows a stereo vision equipped robot to cope with the failure of one of its cameras. The setup learns to estimate average depth using a monocular image, by using the stereo vision depths from the past as trusted ground truth. We present preliminary results from an experiment on the International Space Station (ISS) performed with the MIT/NASA SPHERES VERTIGO satellite. The presented experiments were performed on October 8th, 2015 on board the ISS. The main goals were (1) data gathering, and (2) navigation based on stereo vision. First the astronaut Kimiya Yui moved the satellite around the Japanese Experiment Module to gather stereo vision data for learning. Subsequently, the satellite freely explored the space in the module based on its (trusted) stereo vision system and a pre-programmed exploration behavior, while simultaneously performing the self-supervised learning of monocular depth estimation on board. The two main goals were successfully achieved, representing the first online learning robotic experiments in space. These results lay the groundwork for a follow-up experiment in which the satellite will use the learned single-camera depth estimation for autonomous exploration in the ISS, and are an advancement towards future space robots that continuously improve their navigation capabilities over time, even in harsh and completely unknown space environments.

1. Introduction

Future space missions will increasingly rely on the help of robotic systems and in some cases even be performed purely by fully autonomous robots. Currently, space robots are either tele-operated remotely [1–3] or perform autonomous tasks such as driving short stretches in a pre-programmed manner [4–7]. An illustrative example of this is the Curiosity Mars rover, which uses its HazCams and computer vision algorithms to do autonomous traversing. However, its navigational goal is set manually by a human operator and the traversal algorithms are fixed or pre-programmed [8]. Current space robots do not have the ability to learn from their environment.

Learning can be very advantageous, since it decreases pre-deployment engineering effort and allows the robot to adapt to specific properties of its potentially unknown environment. This last advantage is especially relevant for planetary explorers that are by definition sent to areas of which little is known. Despite these advantages, machine learning is not yet used in space robotics. There have been proposals to use learning for instance for visual pose recognition of known satellites [9,10], but this learning would take place before deployment and would not allow the involved robot to learn from unforeseen circumstances. A major reason that space robots are not equipped with algorithms to learn in their unknown space environment is that it introduces extra risk for missions with extremely small margins for error.

A potential suitable candidate for learning on space robots is Self-Supervised Learning (SSL), since it is a reliable learning method that

* Corresponding author.
** Corresponding author.
E-mail addresses: k.g.vanhecke@tudelft.nl (K. van Hecke), g.c.h.e.decroon@tudelft.nl (G.C.H.E. de Croon).

http://dx.doi.org/10.1016/j.actaastro.2017.07.038
Received 27 February 2017; Received in revised form 22 June 2017; Accepted 27 July 2017
Available online 29 July 2017

Acronyms/abbreviations

SSL	Self-Supervised Learning
JEM	Japanese Experimentation Module
SPHERES	Synchronized Position Hold Engage and Reorient Experimental Satellite
VERTIGO	Visual Estimation for Relative Tracking and Inspection of Generic Objects
ISS	International Space Station
ROC	curve Receiver Operating Characteristic curve
TPR/FPR	True Positive Rate/False Positive Rate

allows robots to adapt to their environment. In an SSL setup, a robot extends its capabilities by using a trusted, primary sensor cue to train a still unknown secondary sensor cue. The most well-known example of SSL is its use on autonomous driving cars such as Stanley [11–15] for the recognition of drivable terrain. Stanley used a laser scanner to detect drivable terrain close by. It then used the outputs from the laser scanner as classification targets for a supervised learning process. This process learned a function that maps colors in the camera image to the classes of "drivable" and "non-drivable". Since the camera has a much longer range than the laser scanner, Stanley could see where the road was going for much further distances, allowing it to move more quickly and win the grand DARPA challenge. However, SSL can be more broadly applied to other modalities and for other purposes [16–18].

We recently introduced a novel SSL setup that allows a robot equipped with stereo vision to cope with a potential failure of one of its cameras [19]. The idea is that the robot learns how to see distances in a single, still image while operating in its environment. To this end, it will learn a function that maps textures in an image to the distances obtained with its stereo vision system. Our proposed setup will require the robot to learn a model that persists over time when the primary cue fails, which is why we term it *persistent* SSL. We have previously successfully tested persistent SSL for the navigation of autonomous drones on Earth [19], which showed its potential in terms of reliability for application in space.

In this article, we will present preliminary results from an experiment on the International Space Station (ISS) performed with the MIT/NASA SPHERES VERTIGO satellite, which is equipped with a stereo vision system that allows it to perceive depth and navigate by itself. The presented preliminary experiments, prepared by a mixed team from TU Delft, the European Space Agency (ESA) and the Massachusetts Institute of Technology (MIT), were performed on October 8th, 2015 on board the ISS. The main goals were (1) data gathering, and (2) navigation based on stereo vision. Going beyond the goals of the experiments, the SPHERES VERTIGO satellite also performed the learning during operation, making it – to the best of the authors' knowledge – the first learning robot in space. Please remark that this paper is an extension of [20], where we now elaborate on the methods and algorithms, and show the results of the preparatory glass-table experiments.

The remainder of the article is structured as follows. In Section 2 we explain the materials and methods used in the experiment. Subsequently, in Section 3 we present the experimental results, which are discussed in Section 4. Finally, we draw conclusions in Section 5.

We have made an explainer video about the experiment in Ref. [21].

2. Algorithms

In recent work, we proposed a persistent Self Supervised Learning (SSL) method [19]. Persistent SSL is able to use on board available training data to train another algorithm online, which may use a different sensor. Since ground truth is available during most of the run-time, performance guarantees can be made on the outcome of the learned estimate. Moreover, the learning algorithm is trained in exactly the same environment as the deployment environment, while providing large amounts of training data (since the labelling is done automatically by a trusted algorithm based on a trusted sensor cue available on the drone). Advantages of the learned cue include possible extrapolation to results better than the trusted cue (since the cue on which the learned algorithm was trained may be better in some situations), while possibly being less CPU intensive, and most importantly for this work, providing another adaptive and redundant cue to become more resistant against sensor failure etc. In this paper, we again use the persistent-SSL method to solve the proof of concept problem of learning monocular distance estimation on board a robot.

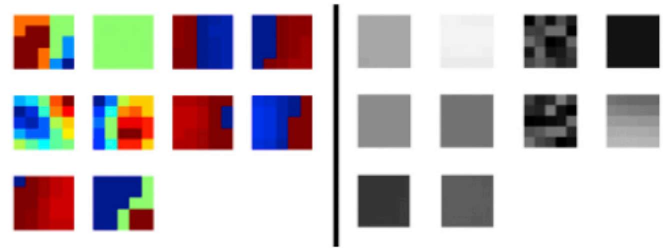

Fig. 1. The Texton library used in the experiments. The right set of Textons are based on pixel intensities, the left set contains (artificially colored) gradient Textons (i.e. Textons based on gradient images).

The general idea behind the monocular depth estimation method in our application is that the distribution of textures in an image varies when the camera gets closer to obstacles. To capture this phenomenon we employed a Texton based Visual Bag of Words (VBoW) method fitted with a kNN ($k = 5$) regressor to do the monocular distance estimation (as in our work in Ref. [19]). Although more complex methods are available that could lead to a better performance (e.g., deep neural networks [22]), the computational resources on the SPHERES are very restricted, as is common for space systems (e.g., [23]). To make the learning as efficient as possible, the learning happened in batches of 1 min.

In VBoW a fixed number of image features are treated as words, in our case textons are words. A texton is a filter of size $w \times h$ pixels, which is convoluted with the input space. A total of m patches are extracted from each input image of size (?? × ??), and each of the m patches is convoluted with each of the n textons. For each image, a texton occurrence histogram of size n is calculated using a Euclidean distance metric. The closest texton for each patch is found using this distance and the corresponding bin is incremented by one. To increase computational efficiency, we apply a grid wise subsampling such that ??≪(?? × ??). This may introduce some sampling noise, which in turn is partly countered by smoothing the estimator output. We also optimized using SIMD processor instructions for the Euclidian distance calculations.

Compiling the Texton library (Fig. 1) was done with Kohonen clustering of textures using a data set gathered from an office cubicle environment, as proper ISS data was not available beforehand. Two types of textons are used: normal textons calculated from (offline) Kohonen clustered textures and gradient textons obtained similarly but based upon the gradient of the images. Gradient textures have been shown in

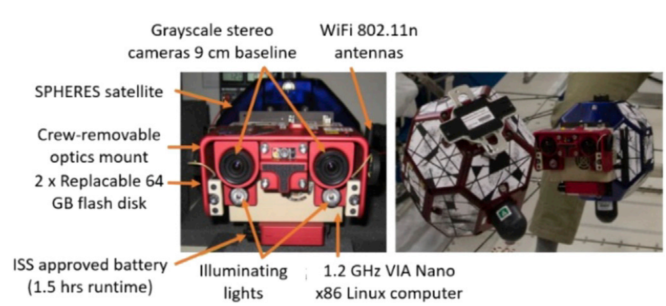

Fig. 2. SPHERES VERTIGO details.

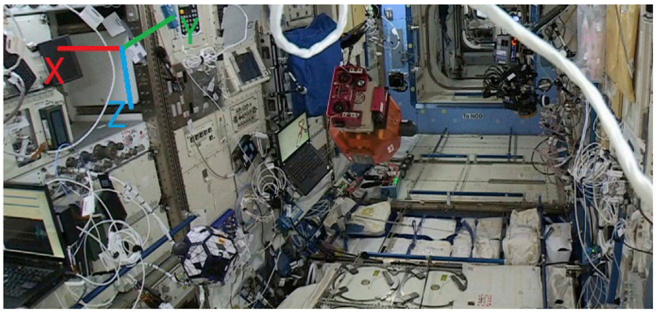

Fig. 3. The orange SPHERES satellite with VERTIGO in the JEM of the ISS. The JEM has two open ends (one is visible here, the other is behind the camera) and 6 walls enclosing the SPHERES test volume of approximate dimensions: x: 1.5 m, y: 1.8 m, z: 1.5 m. In the lower left corner the astronaut laptop is visible, showing the graph of Fig. 12 to the astronaut. Coincidentally one can see another SPHERES satellite (a little bit right to the laptop) floating in the test volume, which could be an obstacle the orange satellite needed to avoid. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Ref. [24] to be an important depth cue.

The stereovision algorithm used comes from LibElas [25] as this is the standard algorithm used with VERTIGO. The disparity map generated by this library was averaged to an average disparity scalar (disparity $\sim$ 1/distance). Fig. 4 shows how these algorithms are connected using persistent SSL. Stereo camera images are passed into the stereo processor, the average disparity is used as ground truth to train the monocular average disparity estimator, which receives only one of the images from the stereo pair as its input. The error rate of the monocular estimator is determined by means of a Receiver Operator Characteristic (ROC) curve analysis. To do so, both the trusted stereo disparity and the regression output of the monocular disparity estimator are binary classified to obstacle/no obstacle signals. An input image is classified as containing an obstacle when the average disparity becomes higher than a threshold. In practice a wall is approached straight up to about 0.3 m distance before being classified as an obstacle. Smaller obstacles may be approached closer because the averaging of the disparity map filters out smaller objects. We use disparity instead of depth to compensate for this effect; disparity counts close by pixels relatively heavy towards the total. The obstacle threshold on the stereo average disparity ground truth is tuned by hand once so that collisions are avoided always. By varying the threshold on the monocular estimator, a ROC curve is created.

Using the ROC curve to determine the optimal threshold level on the monocular estimator, and subsequently the resulting optimal True Positive Rate (TPR, rate of correctly classified obstacle image) versus the False Positive Rate (FPR, rate of no-obstacle images that were classified as obstacle images), the quality of the learned estimator is evaluated. An additional threshold on this TPR/FPR ratio is applied to determine whether the monocular estimator is ready for use by the behavior routines, i.e. whether SSL succeeded to learn enough so that the monocular estimator output can be trusted.

The behavior routines are kept as simple as possible: 1) move into a straight line forward (in the direction of the camera) until an obstacle is detected (e.g. one of the ISS walls), 2) stop and rotate into another random direction, 3) accelerate towards that direction in a straight line again and repeat. This results in an autonomous exploration behavior that was also shown in Ref. [19].

3. Experiments setup

To perform the experiment we utilized the SPHERES test bed system, which was launched to the ISS in 2006. This system is meant as a test bed platform, to give engineers a chance to test algorithms on real spacecraft in a microgravity environment before deploying them on multi-million dollar satellites. A stereo vision upgrade (VERTIGO) was installed in 2013 [26,27]. Fig. 2 shows a satellite of the SPHERES platform with the VERTIGO upgrade. A SPHERES satellite contains 12 CO_2 thrusters to provide full 6-DOF control in the micro gravity environment of the ISS. The VERTIGO system consists out of a monochrome 640 $\times$ 480 resolution stereo camera, and a 1.2 GHz VIA $\times$86 embedded computer running Ubuntu. Using this platform, we tested our persistent-supervised learning method first in the 3-DOF SPHERES preparation facility of the Space Systems Lab at MIT and subsequently on board the ISS, inside the Japanese Experiment Module (JEM).

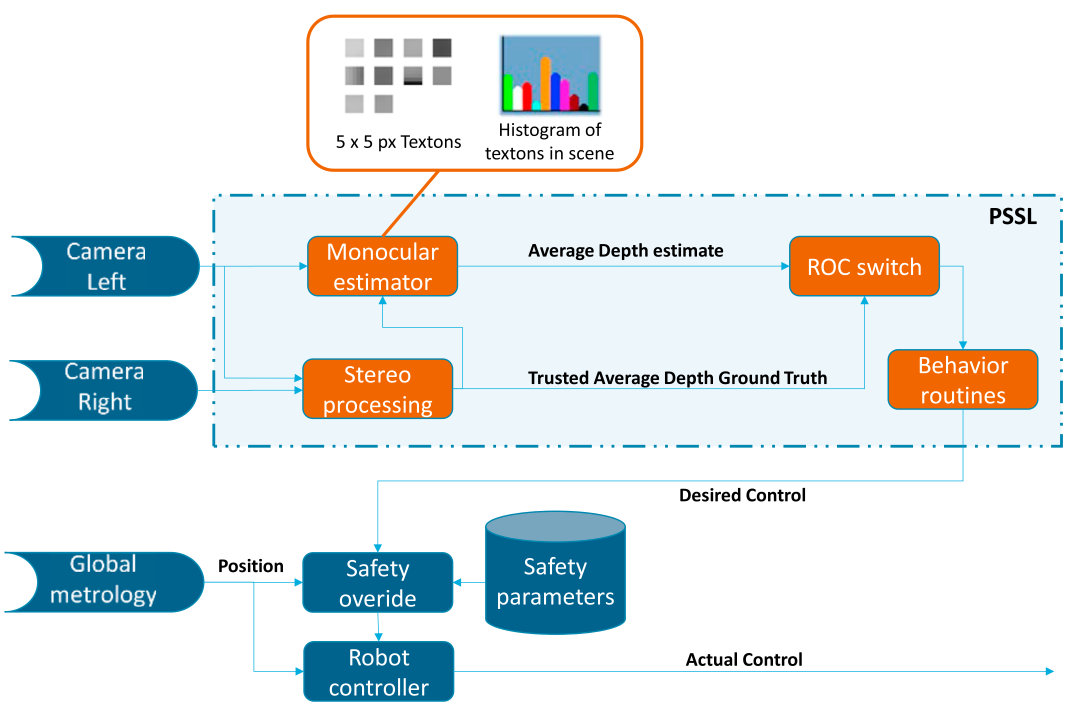

Fig. 4. System overview.

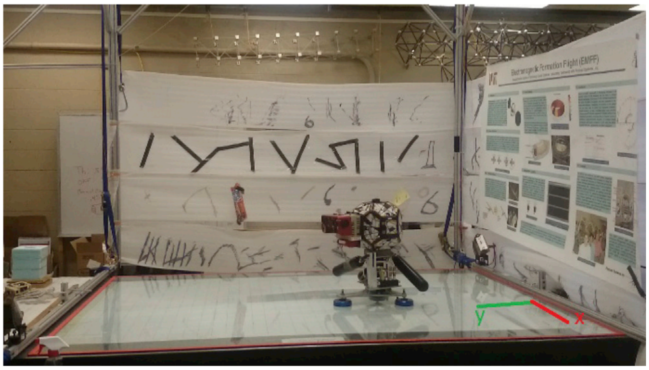

Fig. 5. Glass table test. The SPHERES satellite can move freely over the table enclosed by the two 'walls' and the two open ends. The glass table has x,y dimensions of [1.5, 2.0] m.

Table 1
Glass Table test results overview.

Test	Duration	Images[a]	Distance	Notes
Manual	128 s	1281/370	5.3 m	Success
Auto	606 s	3642/2057	20.9 m	Success

[a] Acquired/processed and learned on-board.

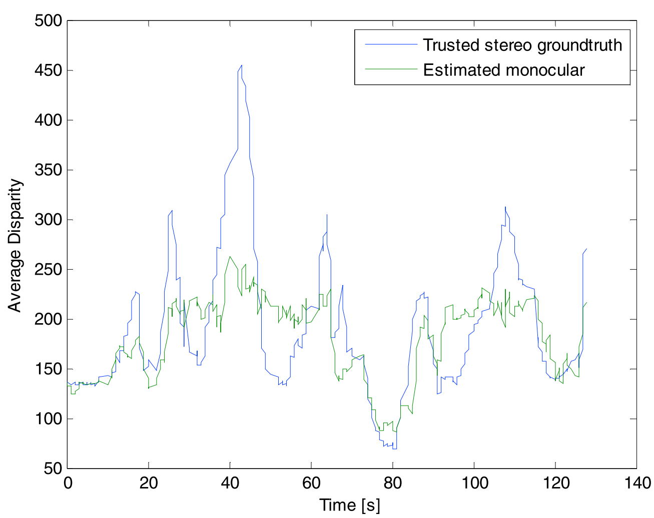

Fig. 6. Test set results of the manual glass table test.

3.1. ISS experiments

We have conducted two experiments on the ISS. The first experiment was a short manual experiment (2 min). As the autonomous behavior and specifically tuned control algorithms had not been previously tested in the ISS, the first experiment mainly served as a contingency and data collection experiment. Therefore, more emphasis was placed on acquiring data at the cost of processing them. The frame rate of acquired images was set to 10 fps, while the algorithms executed at about 2.5 fps during the tests on the ISS. The satellite was manually moved through the JEM volume by the astronaut in a way that simulated the autonomous behavior as closely as possible, keeping the untested 6-DOF control parts out of the loop. The astronaut was provided haptic feedback to move the satellite differently by means of short bursts of the thrusters. If the thrusters fired, the astronaut moved too quickly. This was calculated by evaluating the mean square error between the learned estimate and the trusted average disparity. A manual and quick briefing before the test was given to the astronaut on how to move the satellite.

The second, autonomous, test was set to take 10 min. The satellite starts by using its stereo vision to explore the SPHERES-volume inside the JEM, by moving in a straight line until an obstacle (JEM wall) is detected. After detection, the satellite rotates in a random direction, checks if there is no obstacle in that direction, and then proceeds in another straight line towards that direction. In the meantime, persistent-SSL is training the estimation of a single (average) depth based on the stereo vision depth information and the monocular image data from one of the cameras of the stereo vision system. After 7 min, or earlier if the algorithm achieved a low error rate on its learned results, the monocular vision was set to be enabled in the behavior loop.

Since the SPHERES test volume consists of two open ends, the stereo vision will not detect an obstacle in those regions, but the satellite is not allowed to venture out of these regions. In those cases, the SPHERES Global Metrology (sonar based pseudo-GPS) is used to determine if the satellite is within the volume. In case the satellite is leaving the volume, the satellite stops, rotates randomly to a direction inwards to the volume, and starts a straight line towards that new direction. A safety override control maneuver is activated when the satellite moves close to the outer ranges of the expected range of the global metrology system, in which case attitude control of the satellite is disabled and full thruster power is used for position control to return to the test volume. The global metrology system is also used to log the exact position of the satellite for later analysis. The SPHERES test volume in which the global metrology system is working has the following approximate dimensions: x: 1.5 m, y: 1.8 m, z: 1.5 m. Fig. 3 shows the orange satellite during one of the experiments in the ISS through the overhead camera. During the autonomous tests, emphasis was placed on results at the cost of sole data gathering. Images were captured at 6 fps instead of 10 fps during the manual test. The algorithms executed at about 4.7 fps, meaning about 1.3 fps were not used during operation itself.

3.2. Preliminary experiments

Before the ISS experiment, tests here on earth were conducted to test both the learning algorithm, control, and the resulting behavior on the same hardware as in the ISS. At the Space Systems Laboratory in MIT, copies of the SPHERES are available to perform such tests. The satellites are mounted on an air-cushion undercarriage, that can float freely on a glass table. This means the satellite can move in 3-DOF using its normal CO^2 thrusters, as can be seen in Fig. 5. Movement in the Z direction, and rotations around the X, Y axes are not possible. We present an experiment on the glass table with exactly the same algorithms as the tests that were to be run in the ISS. The main goal is to demonstrate the viability of the *persistent* SSL method on the SPHERES system. To make the experiments as comparable as possible with the ISS experiments, first the satellite is manually moved over the table for 2 min. Then, a follow up autonomous experiment runs the satellite for 10 min through the confined space of the glass table. Somewhat similar to the ISS, the glass table was surrounded with two (artificially textured) walls and two open ends. The glass table is equipped with the same Global Metrology system that is placed in the ISS, and the same override algorithm is used (and tested) on the open end of the table. The Global Metrology system covers the whole glass table, which means the satellite cannot accidentally leave the covered volume like in the ISS. This means that safety override control maneuver, in which attitude control is disabled in favor of the position control, could not be tested on the glass table. The vision algorithms cope with the walled parts of the glass table. The random rotation selected after detecting the walls are restricted to the 3-DOF space of the glass table.

4. Glass table results

Several engineering tests on the glass table were conducted, of which one is selected for presentation in this section. The full test can be viewed online in the video playlist at [28]. Both the rendered results as well as the video of an external camera can be seen. A summary of the data acquired as well as the distance travelled during these tests is presented

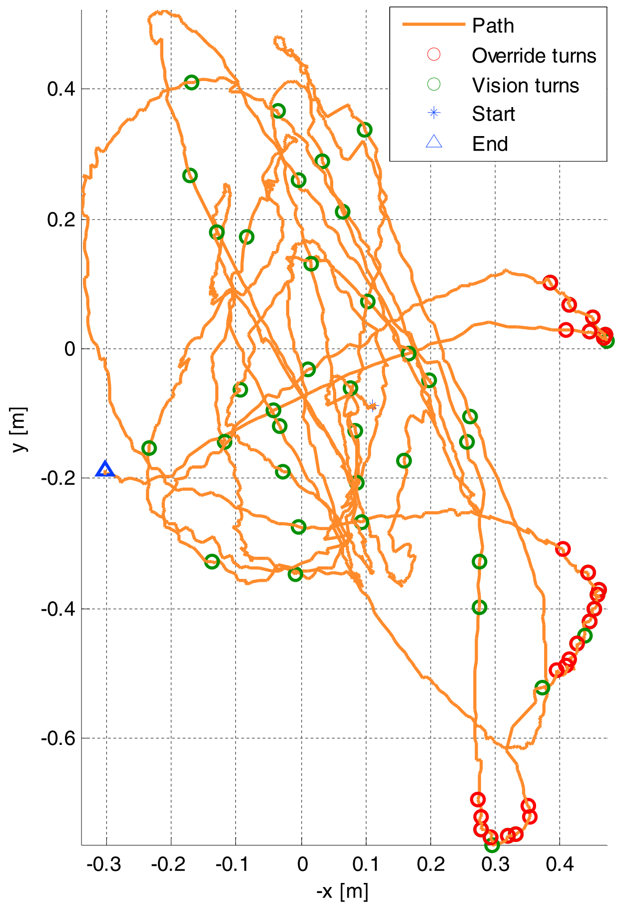

Fig. 7. Trajectory of the autonomous glass table test.

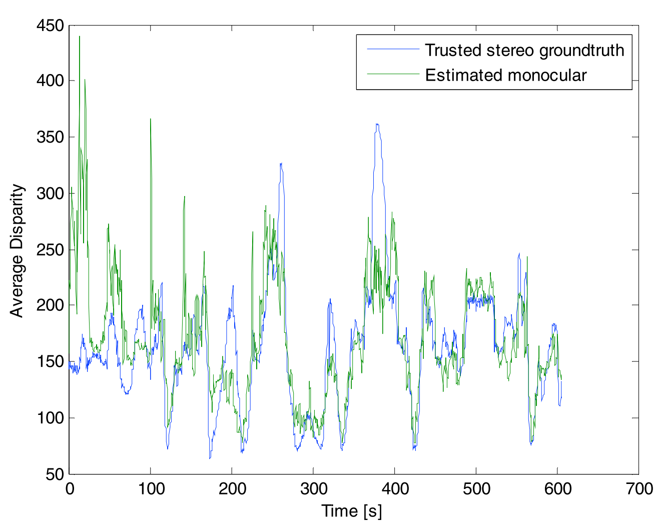

Fig. 8. Test set results of the autonomous glass table test.

Table 2
ISS test results overview.

Test	Duration	Distance	Images[a]	Notes
T4.1	~50 s	2.2 m	596/144	IR overload reset
T4.2	120 s	6.5 m	1131/267	Success
T5.1	~30 s	0.5 m	261/123	Battery empty
T5.2	210 s	8.1 m	1293/605	IR overload reset

[a] Acquired/processed and learned on-board.

in Table 1.

4.1. Manual test

The incrementally learned results of the manual test are visualized in Fig. 6. As expected for the initially untrained algorithm, the estimated monocular depth shows a large difference compared to the stereo depth ground truth, however in the video some visual correlation is appearing rapidly while the experiment progresses. After the 2-min test, the optimal TPR/FPR ratio is 0.99/0.23 on the training dataset.

An exact plot of the trajectory cannot be given, as the manual operation of the satellite disturbed the global metrology system at the glass table.

4.2. Autonomous test

A trajectory of the autonomous test is plotted in Fig. 7. The TPR/FPR ratio determined above based on the training data from the manual test is below the manually tuned threshold of 0.95/0.1. This means the algorithm decides to start its explorative behavior based on its trusted stereo ground truth depth. However, at 0:34 in the video (62 s in real time), the TPR/FPR ratio tips over the threshold, which results in the satellite using its learned monocular depth estimator as its primary cue. If at any point the TPR/FPR ratio drops below the threshold again, e.g. due to increased uncertainty because of entering a yet unknown area, the algorithm switches back to stereo. In this case however, the error stays low enough for the remainder of the experiment to justify continuous exploration based on the learned monocular depth cue. As can be seen from the trajectory, the 2D space is explored randomly, as intended, to fully test the algorithms.

The incrementally learned results are shown in Fig. 8. In the beginning the algorithm is trained on a relatively small training set, which is why the correlation between the estimate and the ground truth is low. However, as time proceeds and the algorithm is trained on more data, the correlation visually increases as the regression error goes down. More importantly, TPR/FPR ratio on the training set drop down to 0.96/0.07 in the final moments. This result builds confidence that the test on the ISS may work well, although the data is applicable only to a point due to the restricted 3-DOF space.

5. ISS results

SPHERES Test Session 74B was conducted on October 8th, 2015, with astronaut Kimiya. The test session contained six tests; this paper describes the results of test four and test five in particular. The other tests were unrelated experiments from other scientists. Both tests four and five were done twice due to hardware failures. A summary of the results of these tests is given in Table 2. Videos of each test and their raw data can be viewed online in Ref. [29].

5.1. Manual tests #1 (T4.1 & T4.2)

The objective of the manual test, was for the astronaut to move the SPHERES satellite through the volume, in order to obtain images of the surroundings. Kimiya picked up the instructions well and attempted to execute some useful trajectories. However, after 50 s, a satellite reset occurred. Offline analysis shows the reason for the reset as a disturbance on the global metrology infrared detector on the satellite, which caused a high priority interrupt to continuously fire, which caused a system reset by means of a watchdog.

The 3D path of T4.1, as undertaken by Kimiya before the reset, is depicted in Fig. 9.

In the time before the unexpected reset, 596 stereo images have been acquired of which 144 were directly used for training.

Due to the reset in T4.1, a second run of T4 was attempted and successfully finished. The learned data from T4.1 was automatically

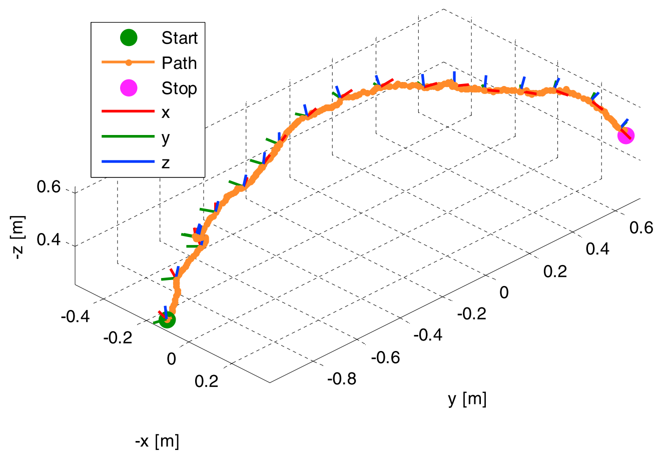

Fig. 9. Satellite flight path of T4.1.

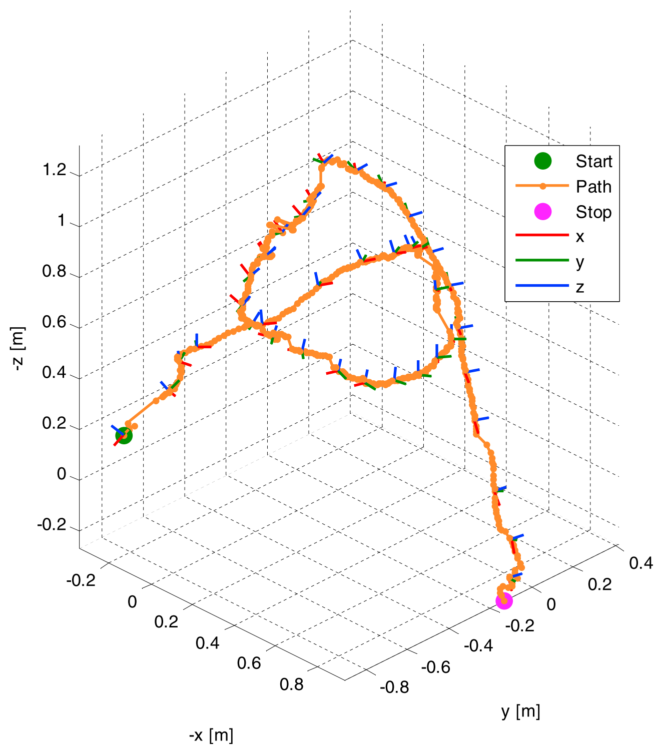

Fig. 10. Satellite flight path of T4.2.

concatenated to T4.2. During T4.2 an additional 1131images were acquired, of which 267 were directly used for online training. A 3D plot of the trajectory of the satellite is shown in Fig. 10.

The stereo images acquired during T4 were analyzed in real-time and as ground truth the average disparity for each stereo pair was calculated. The results of this can be seen in Fig. 11. The estimated disparity is based on the self-supervised learned algorithm trained on the 144 images from T4.1 during T4.2. As can be seen, some correlation is visible, but the training set was too small to properly estimate the disparity at this point. The final optimal TPR/FPR was 1/0.33 on training data, unsurprisingly below the threshold to already use the monocular estimator as the primary cue.

5.2. Autonomous tests (T5.1 & T5.2)

The autonomous test was also attempted twice. The first time, T5.1, a

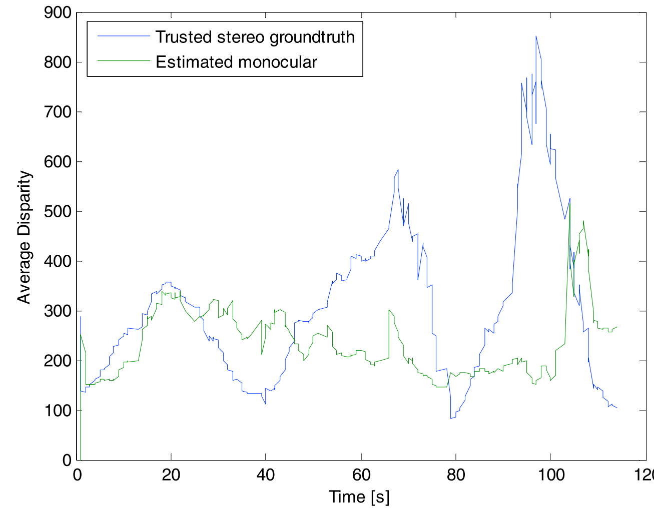

Fig. 11. Test set results of T4.2.

battery empty failure occurred, which caused the VERTIGO system to automatically shut down after 30 s, and loss of control happened after 11 s. The 3D path is depicted in Fig. 13. During its first and only trajectory, the satellite managed to evade another stationary satellite (that was placed there mistakenly) using its trusted stereo algorithm. This evasion maneuver is shown as the first green circle in the plot. The subsequent vision turns were commanded by the algorithm, but not executed due to the battery problem. During T5.1 an additional 261 images were acquired, of which 123 were analyzed directly for training.

As T5.1 failed due to power loss, another attempt was done. Unfortunately, the second attempt failed after 210s. Offline analysis showed the cause to again be the IR disturbance problem that also ended T4.1. Before that time, the satellite could make several crossings through the volume exploring and learning simultaneously. Also, another 1293 images were acquired of which 605 analyzed during the test. The 3D flight path of T5.2 is depicted in Fig. 14 and Fig. 15.

The first turn was initiated by the global metrology based safety override, as the satellite ventured out of the open end of the volume. The safety override has therefore been proven to work as expected. The second and third turn were both stereo vision avoidance commands, avoiding collision with the JEM wall. The fourth turn was caused by the safety override, but happened at too high a speed. This caused satellite to drift too far out of the allowed volume causing the emergency safety override to become active. The satellite diverted all control power to position control, disabling attitude control, in order to move back into the volume as quickly as possible. In a best-case scenario, this would not have been necessary. For future missions we plan to reduce the top speed of the satellite such that this will not happen again. The last two turns are caused by stereo avoidance commands, again attempting to avoid the JEM wall. However, before the satellite had a chance to execute the avoidance maneuver, the IR reset failure interfered.

Other, previous experiments on the SPHERES VERTIGO were much less affected than ours by the IR reset failure. The current hypothesis is that it has to do with the more complete exploration as performed by the autonomous stereo vision based behavior. This brings the satellite closer to interfering electronics in the module. Various solutions are under evaluation to prevent the problem in future experiments.

A video of the results calculated on the satellite during the experiment in the ISS can be viewed online in [30]. A still image from this video can be seen in Fig. 12, this video was also visible for Kimiya during the tests by means of a Wi-Fi stream. The image contains the raw input images from the left camera (top left in the figure) and right camera (top, middle), the disparity map (top right – red is close, blue is far), and a graph of the average disparity over time (bottom time line plot). The average disparity from the trusted stereo vision algorithm is shown in blue, the

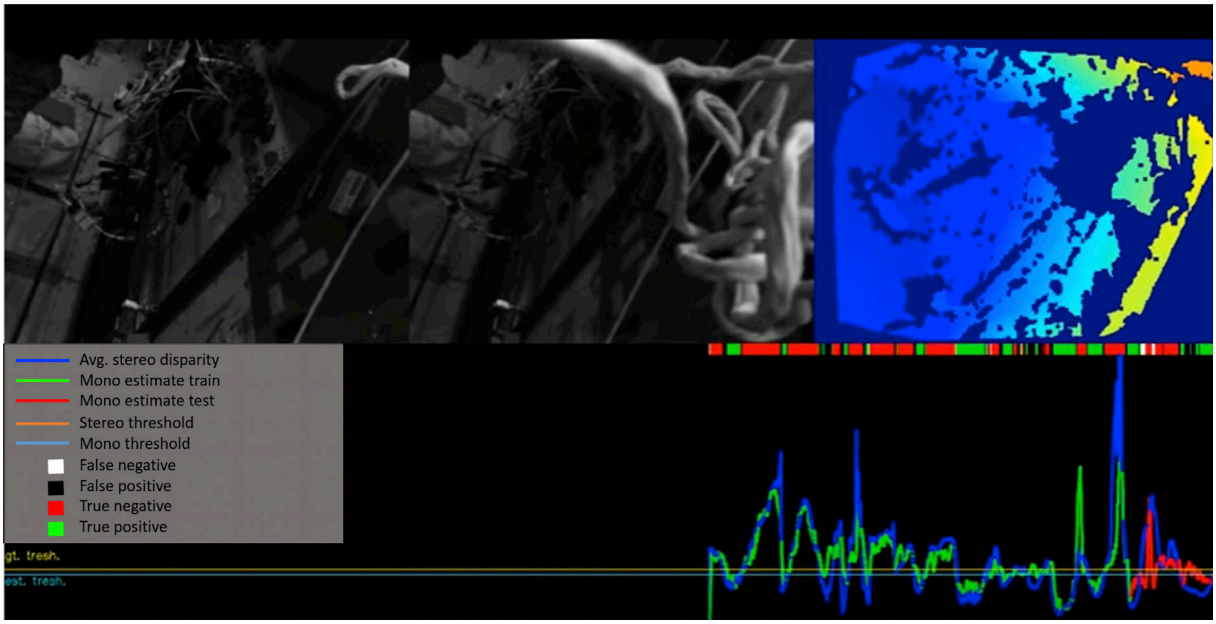

Fig. 12. Results from T5.2 overview. This is a still taken from the video stream shown to the astronaut.

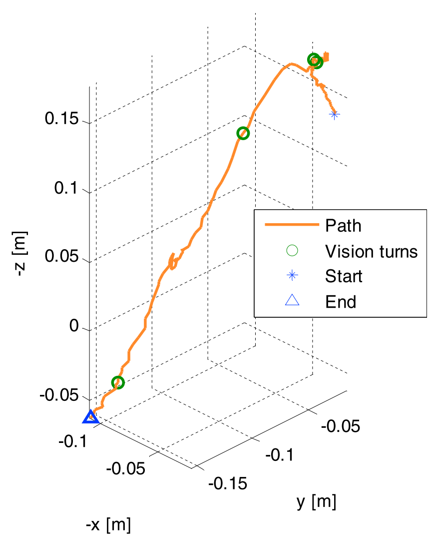

Fig. 13. Satellite flight path of T5.1.

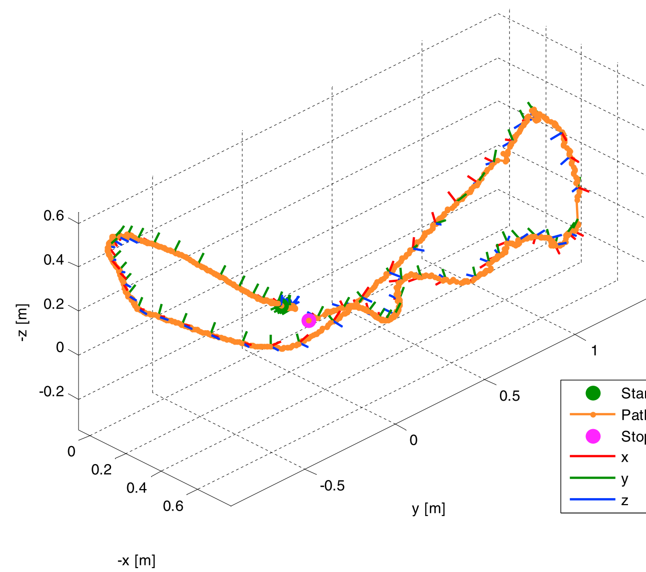

Fig. 14. Satellite flight path of T5.2 with vision results.

learned estimates are shown in green, while predictions of the robot of previously unseen cases are shown in red. When the trusted disparity supersedes the threshold, an avoidance maneuver is commanded.

The learned results can be seen in Fig. 16. We zoom in on two moments in time during the learning on board the satellite. Fig. 17 and 18 show the learned disparities (green) and predicted disparities (red) just before $t = 141$ s and after $t = 141$ s. The predictions before $t = 141$ s do not seem to correlate much with the "ground-truth" stereo vision estimates. However, after learning on these samples, the predictions on new, unseen samples do correlate well with the ground-truth. The predictions can be evaluated objectively with respect to the stereo vision threshold – resulting in a classification problem setting. Before learning at $t = 141$ s, the True Positive Rate (TPR) of the predictions is 0.2 while the False Positive Rate (FPR) is 0.6. After learning that part, the predictions are indeed objectively better, with a TPR of 0.7 and FPR of 0.3. These results correspond to results obtained on the glass table with fewer degrees of freedom. They show that the learning is successful, but that the amount of gathered data is not yet enough to cover the entire environment.

6. Conclusions

We have presented preliminary results from a Self-Supervised Learning (SSL) experiment on the International Space Station (ISS) performed with the MIT/NASA SPHERES VERTIGO satellite. The main goals of the experiment were (1) data gathering, and (2) navigation based on stereo vision. Both goals were successfully achieved, although the experiments were hampered by automatic resets triggered by an interference of the IR detector of the SPHERES satellite. During both parts of the experiment, the satellite was learning online to map the appearance of the environment to the distance estimates from its stereo vision system. Despite the extremely limited training time, some successful generalization of the learned mapping to unseen images can be observed.

These first robotic learning experiments in space hold a promise for follow-up experiments in which the satellite will use the learned mapping to navigate with only a single camera and further proving the reliability of SSL. For a follow-up experiment, the following main insight from the

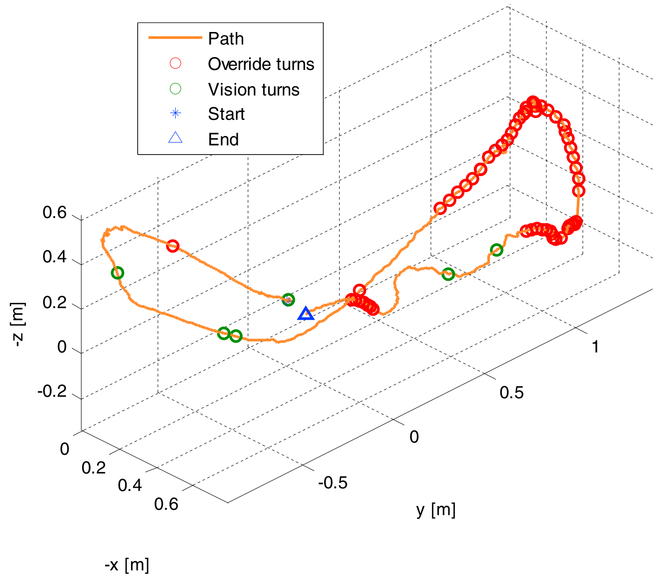

Fig. 15. Satellite flight path of T5.2 with orientation.

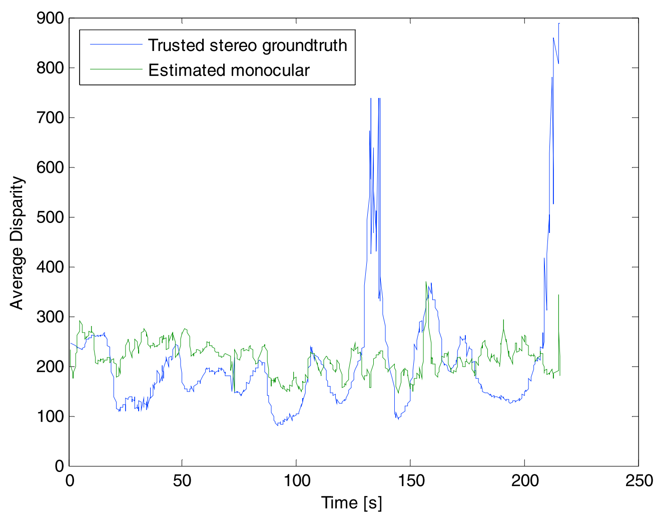

Fig. 16. Test set results of T5.2.

current experiment should be considered. The combination of a limited experiment time with a relatively slow speed of the robot satellite implies that it will be difficult to learn the entire available space in the module.

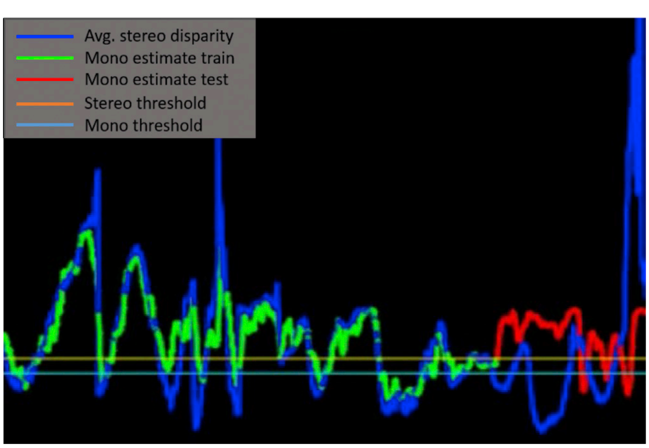

Fig. 17. Estimated disparity before learning the particular scene at t = 141 s.

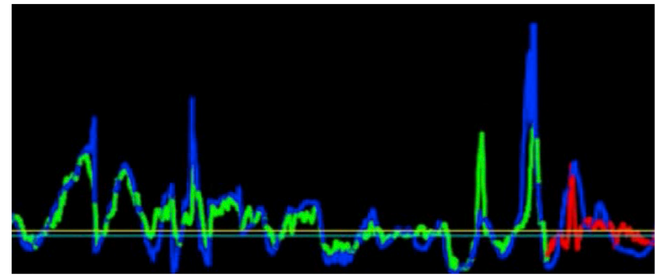

Fig. 18. Estimated disparity at t = 212s, after learning the particular scene up to t = 141s.

This is especially true given the fact that the robot in space can move with 6 DOF, leading to a high variety in environment appearance. Given a limited experiment time, the movement space of the satellite should also be more limited, e.g., by not having it travel to the open ends of the module. As a by-effect, this will avoid the problems we experienced when our emergency override system disabled attitude control to enter back into the central space. This will hopefully allow learning and switching to monocular vision control in a single test session.

References

[1] S.S. Heikkilä, A. Halme, A. Schiele, 'Affordance-based indirect task communication for astronaut-robot cooperation', J. Field Robot. 29 (4) (2012) 576–600.
[2] G. Hirzinger, B. Brunner, J. Dietrich, J. Heindl, Sensor-based space robotics-ROTEX and its telerobotic features, IEEE Trans. Robot. Autom. 9 (5) (1993) 649–663.
[3] G. Hirzinger, K. Landzettel, B. Brunner, M. Fischer, C. Preusche, D. Reintsema, B.M. Steinmetz, DLR's robotics technologies for on-orbit servicing, Adv. Robot. 18 (2) (2004) 139–174.
[4] Larry Matthies, et al., Computer vision on mars, Int. J. Comput. Vis. 75 (1) (2007) 67–92.
[5] M.W. Maimone, P.C. Leger, J.J. Biesiadecki, Overview of the mars exploration rovers' autonomous mobility and vision capabilities, in: IEEE International Conference on Robotics and Automation (ICRA) Space Robotics Workshop, 2007, April.
[6] N.W. Oumer, G. Panin, Q. Mülbauer, et al., Vision-based localization for on-orbit servicing of a partially cooperative satellite, Acta Astronaut. 117 (2015) 19–37.
[7] Z. Wen, et al., Robust, fast and accurate vision-based localization of a cooperative target used for space robotic arm, Acta Astronaut. 136 (2017) 101–114.
[8] J.N. Maki, et al., Mars exploration rover engineering cameras, J. Geophys. Res. Planets 108 (2003) E12.
[9] H. Zhang, Z. Jiang, A. Elgammal, Vision-based pose estimation for cooperative space objects, Acta Astronaut. 91 (10) (2013) 115–122.
[10] Hao peng Zhang, Zhi guo Jiang, Multi-view space object recognition and pose estimation based on kernel regression, Chin. J. Aeronaut. 27 (5) (2014) 1233–1241.
[11] S. Thrun, M. Montemerlo, H. Dahlkamp, D. Stavens, A. Aron, J. Diebel, P. Fong, J. Gale, M. Halpenny, G. Hoffmann, et al., Stanley: the robot that won the darpa grand challenge, in: The 2005 DARPA Grand Challenge, Springer, 2007, pp. 1–43.
[12] D. Lieb, A. Lookingbill, S. Thrun, Adaptive road following using self-supervised learning and reverse optical flow, Robot. Sci. Syst. (2005) 273–280.
[13] A. Lookingbill, J. Rogers, D. Lieb, J. Curry, S. Thrun, Reverse optical flow for self-supervised adaptive autonomous robot navigation, Int. J. Comput. Vis. 74 (no. 3) (2007) 287–302.
[14] R. Hadsell, P. Sermanet, J. Ben, A. Erkan, M. Scoffier, K. Kavukcuoglu, U. Muller, Y. LeCun, Learning long-range vision for autonomous off-road driving, J. Field Robot. 26 (no. 2) (2009) 120–144 [Online].
[15] U.A. Muller, L.D. Jackel, Y. LeCun, B. Flepp, Real-time adaptive off-road vehicle navigation and terrain classification, in: SPIE DefenseSecurity, and Sensing. International Society for Optics and Photonics, 2013, 87 410A.
[16] J. Baleia, P. Santana, J. Barata, On exploiting haptic cues for self-supervised learning of depth-based robot navigation affordances, J. Intell. Robot. Syst. (2015) 1–20.
[17] H. Ho, C. De Wagter, B. Remes, G. de Croon, Optical flow for self supervised learning of obstacle appearance, in: Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on. IEEE, 2015, pp. 3098–3104.
[18] K. Otsu, M. Ono, T.J. Fuchs, I. Baldwin, T. Kubota, Autonomous terrain classification with Co-and self-training approach, IEEE Robot. Autom. Lett. 1 (2) (2016) 814–819.
[19] K. van Hecke, G. de Croon, L. van der Maaten, D. Hennes, D. Izzo, Persistent Self-supervised Learning Principle: from Stereo to Monocular Vision for Obstacle Avoidance, 2016 arXiv preprint arXiv:1603.080.
[20] Kevin van Hecke, et al., Self-supervised Learning as an Enabling Technology for Future Space Exploration Robots: ISS Experiments, 2016.
[21] https://youtu.be/GZHE0E6AsSE.
[22] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, in: Advances in neural information processing systems, 2012, pp. 1097–1105.

[23] M. Yang, H. Wang, Y. Zheng, et al., Graph-tree-based software control flow checking for COTS processors on pico-satellites, Chin. J. Aeronaut. 26 (2) (2013) 413–422.
[24] B. Wu, T.L. Ooi, Z.J. He, Perceiving distance accurately by a directional process of integrating ground information, Nature 428 (no. 6978) (2004) 73–77.
[25] A. Geiger, M. Roser, R. Urtasun, Efficient large-scale stereo matching, Comput. Vis. ACCV (2010) 2011.
[26] B.E. Tweddle, T.P. Setterfield, A. Saenz-Otero, D.W. Miller, An open research facility for vision-based navigation onboard the international space station, J. Field Robot. 33 (2015) 157–186.
[27] B.E. Tweddle, D.W. Miller, Computer Vision-based Localization and Mapping of an Unknown, Uncooperative and Spinning Target for Spacecraft Proximity Operations, Massachusetts Institute of Technology, 2013.
[28] https://www.youtube.com/watch?v=svZJ2jfUDZM&index=1&list=PL_KSX9GOn2P82eg_PE5eGC_EmLUC9daap.
[29] https://www.youtube.com/playlist?list=PL_KSX9GOn2P-NpLU8DlS-PmgqlfB3a_-A.
[30] https://www.youtube.com/watch?v=6mk0om2TPTo&index=8&list=PL_KSX9GOn2P-NpLU8DlS-PmgqlfB3a_-A.