

Flight Results of Vision-Based Navigation for Autonomous Spacecraft Inspection of Unknown Objects

Dehann Fourie,^{*} Brent E. Tweddle,[†] Steve Ulrich,[‡] and Alvar Saenz-Otero[§]
Massachusetts Institute of Technology, Cambridge, Massachusetts 02139

DOI: 10.2514/1.A32813

This paper describes a vision-based relative navigation and control strategy for inspecting an unknown, noncooperative, and possibly spinning object in space using a visual-inertial system that is designed to minimize the computational requirements while maintaining a safe relative distance. The proposed spacecraft inspection system relies solely on a calibrated stereo camera and a three-axis gyroscope to maintain a safe inspection distance while following a circular trajectory around the object. The navigation system is based on image processing algorithms, which extract the relative position and velocity between the inspector and the object, and a simple control approach is used to ensure that the desired range and bearing are maintained throughout the inspection maneuver. The hardware implementation details of the system are provided. Computer simulation results and experiments conducted aboard the International Space Station during Expedition 34 are reported to demonstrate the performance and applicability of the proposed hardware, and related navigation and control systems to inspect an unknown spacecraft.

I. Introduction

SPACECRAFT robotic inspection is an important technological capability that enables a large number of different types of spacecraft proximity operations, including the International Space Station (ISS) supply, and on-orbit servicing of spacecraft [1]. However, because of the very nature of on-orbit servicing missions, the object spacecraft might not be fully cooperative. Additionally, it may be freely tumbling due to its intrinsic design, or to various mechanical failures or fuel depletion [2]. To address these problems, it is required that the servicer spacecraft relies on an accurate navigation and control system to autonomously determine and control its relative states with respect to the object. Recently, a number of theoretical research studies have addressed the problem of spacecraft navigation for proximity operations. In particular, Hablani [3] proposed a relative navigation technique for autonomous rendezvous and docking using an integrated sensor suite onboard an active chaser satellite, which comprises an imaging sensor, a laser range finder, the space-integrated GPS and inertial navigation system, and a star tracker. However, it is well known that GPS and GPS-like signals are susceptible to interference and jamming.

For this reason, GPS-less navigation systems for autonomous vehicles are currently an active area of research. Particularly, vision-based relative navigation systems are expected to play an important role in future missions because, compared to other navigation technologies (e.g., light detection and ranging (LIDAR), laser range finder, GPS, ground-based measurements), computer vision offers several advantages, such as low power and volume requirements [4]. For example, Linares et al. [5] developed an approach for the relative

attitude determination between two autonomous vehicles based on line-of-sight measurements between both vehicles and a third cooperative object. References [6] and [7] report the on-the-ground laboratory experimentations of vision-based relative navigation between a chaser spacecraft and a target object equipped with different fiducial markers (light-emitting diodes in the previous study, and four point correspondences in the latter work). In the past, vision-based navigation systems using fiducial markers have been successfully validated in different space missions. Jointly developed by the Canadian Space Agency and Neptec, the Space Vision System used cameras to monitor a pattern of special target dots placed on the object to be tracked. As the object moved, the system tracked the changing position of the dots, calculated the location and orientation of the object, and presented this information to the operator in the form of both graphical and textual cues. This information was namely used to position and orient a payload using Canadarm2, or to join two objects as in the case of the space station assembly [8]. On the Defense Advanced Research Projects Agency's Orbital Express mission, the advanced video guidance sensor was used for docking between two spacecraft. A laser diode was used to illuminate a retroreflective visual target that was processed by machine vision algorithms to determine the relative position and orientation between the two spacecraft [9].

However, when the target object has an unknown geometry and no fiducials, such vision systems cannot be employed, and more advanced technologies are required. Therefore, to address the problem of autonomous vision-based relative navigation and control to perform an inspection maneuver of an unknown object, the Massachusetts Institute of Technology (MIT) Space Systems Laboratory developed the Visual Estimation for Relative Tracking and Inspection of Generic Objects (VERTIGO) ISS-based research experiment. This space robotics experimental test bed enables the research and development of computer-vision-based navigation and mapping algorithms capable of achieving two main objectives: 1) performing relative navigation by relying solely on camera hardware and inertial sensors, and 2) building a three-dimensional (3-D) map of another unknown, noncooperative object, and performing relative navigation solely by reference to this 3-D model. Building upon the experience gained under the U.S. Naval Research Laboratory's Low Impact Inspection Vehicle program [10,11], the VERTIGO program designed, built, test, and launch to the ISS a hardware upgrade to the existing Synchronized Position Hold, Engage, Reorient, Experimental Satellites (SPHERES) laboratory facility to test these algorithms.

This paper addresses the first phase of the VERTIGO program (i.e., the problem of performing an autonomous robotic vision-based circumnavigation maneuver about an unknown object by relying solely on computer vision and inertial sensors). To achieve this,

Presented as Paper 2013-4759 at the AIAA Guidance, Navigation and Control Conference, Boston, MA, 19–22 August 2013; received 22 August 2013; revision received 31 January 2014; accepted for publication 1 February 2014; published online 8 May 2014. Copyright © 2014 by the American Institute of Aeronautics and Astronautics, Inc. All rights reserved. Copies of this paper may be made for personal or internal use, on condition that the copier pay the \$10.00 per-copy fee to the Copyright Clearance Center, Inc., 222 Rosewood Drive, Danvers, MA 01923; include the code 1533-6794/14 and \$10.00 in correspondence with the CCC.

^{*}Visiting Ph.D. Student, Department of Aeronautics and Astronautics, 77 Massachusetts Avenue.

[†]Ph.D. Candidate, Department of Aeronautics and Astronautics, 77 Massachusetts Avenue; currently Postdoctoral Associate, Member AIAA.

[‡]Postdoctoral Associate, Department of Aeronautics and Astronautics, 77 Massachusetts Avenue; currently Assistant Professor, Department of Mechanical and Aerospace Engineering, Carleton University, Ottawa, Ontario, Canada. Member AIAA.

[§]Research Scientist, Department of Aeronautics and Astronautics, 77 Massachusetts Avenue. Senior Member AIAA.

the proposed relative navigation and control scheme uses relative-position measurements provided by calibrated stereo monochrome cameras, along with angular velocities from a three-axis gyroscope mounted on the inspecting spacecraft, to ensure that it maintains a body-fixed orientation with respect to the object. It is important to note that this approach assumes that a safe keep-out distance is known a priori. The feedback control system presented in this paper directly regulates this keep-out distance to ensure that there is no contact between the inspector and its target object.

The primary contribution of this relative navigation and control system is its inherent simplicity. The main benefit of this simplicity is that, if implemented correctly, this algorithm can have very low computational cost. As a result, it can run as a higher-frequency background control process, ensuring safe relative positioning, while more complicated mapping and planning algorithms (e.g., those found in [12–17]) are being run ahead of close proximity and docking operations. Alternatively, this relative navigation and control system can be used with a very low power embedded computer (possibly on a very small satellite) to perform an inspection mission at a safe keep-out distance.

This paper is organized as follows: Sec. II provides an overview of the relative navigation and control approach. Section III describes the existing SPHERES facility along with the stereo cameras and associated processor unit (referred to as the VERTIGO Goggles) required to execute the vision-based navigation algorithms. Section IV outlines the software modifications required to accommodate the c.m. offset location due to the addition of the VERTIGO Goggles to the SPHERES satellite. In Sec. V, the image processing strategy employed to provide relative measurements is described, and the computation time of the overall algorithm is discussed. Section VI presents and discusses the numerical simulation results, and Sec. VII presents the on-orbit results of experiments conducted aboard the ISS during Expedition 34 by Commander Kevin Ford. To the best of our knowledge, this was the first time a fully autonomous robotics vision-based navigation strategy was demonstrated in space with a noncooperative spacecraft.

II. Circumnavigation Control Approach

The proposed relative navigation and control approach intends to accomplish an autonomous circumnavigation trajectory about an unknown object based only on visual–inertial information. This proposed control approach is illustrated in Fig. 1. Throughout the inspection trajectory, the inspector spacecraft must always point its body-frame x axis toward the object to maintain the object within its field of view. To do so, the inspector satellite control input torque is obtained with a simple linear attitude control law of the form:

$$\boldsymbol{\tau} = \mathbf{k}_\omega \boldsymbol{\omega}_e \quad (1)$$

in which $\boldsymbol{\tau}$ and \mathbf{k}_ω denote the control input-torque vector and the angular velocity control gain matrix, respectively, and in which the angular velocity control error, denoted by $\boldsymbol{\omega}_e$, is defined as

$$\boldsymbol{\omega}_e \triangleq \boldsymbol{\omega}_d - \boldsymbol{\omega}_m \quad (2)$$

with $\boldsymbol{\omega}_m = [\omega_{mx} \ \omega_{my} \ \omega_{mz}]^T$ denoting the inspector angular velocity, as measured directly by its body-mounted gyroscopes, and with $\boldsymbol{\omega}_d$ given by

$$\boldsymbol{\omega}_d = [0 \ 0 \ \omega_{dz}]^T \quad (3)$$

such that the angular velocity about the z -body axis (i.e., the axis normal to the x - y plane of inspection) is controlled to a constant value, while the other two angular velocity components (x and y) are regulated to zero.

As the spacecraft rotates with a constant angular velocity, the position of the target object will drift to one side in the field of view of the camera system. To ensure that the target is maintained in the center of the camera field of view, the inspector spacecraft is translated by nonrotational forces until the pan error distances have

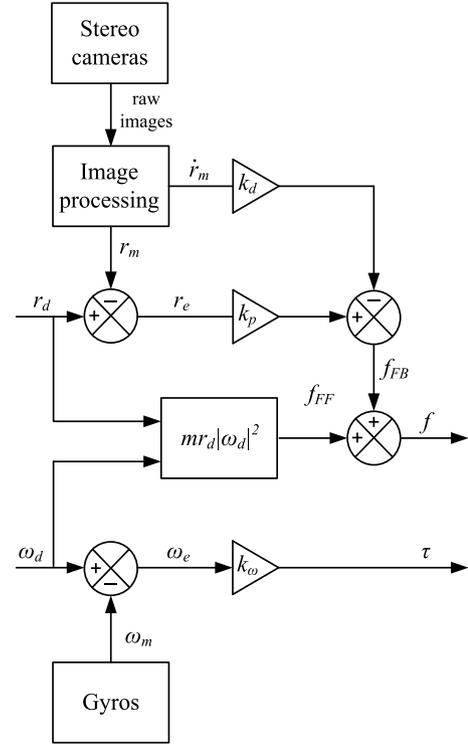


Fig. 1 Vision-based control strategy block-scheme implementation diagram.

been regulated to zero. Simultaneously, the relative position to the target spacecraft, which is obtained from the camera system and through image processing algorithms (see Sec. V), is controlled to a desired value along the x axis of the body frame, to follow an inspection trajectory with a constant radius. The control input force is given by the following proportional–derivative translational feedback control law:

$$\mathbf{f}_{\text{FB}} = \mathbf{k}_p \mathbf{r}_e + \mathbf{k}_d \dot{\mathbf{r}}_e \quad (4)$$

in which \mathbf{f}_{FB} , and \mathbf{k}_p and \mathbf{k}_d denote the feedback control input force vector, and the proportional and derivative control gain matrix, respectively, and in which the Cartesian-position control error, denoted by \mathbf{r}_e , is defined as

$$\mathbf{r}_e \triangleq \mathbf{r}_d - \mathbf{r}_m \quad (5)$$

in which $\mathbf{r}_m = [x_m \ y_m \ z_m]^T$ denotes the relative Cartesian position, which is measured by the camera system, and in which $\mathbf{r}_d = [x_d \ y_d \ z_d]^T$ denotes the desired relative Cartesian position, as commanded by the user. Because it is desired to maintain a constant, safe distance along the line of sight of the cameras between both spacecraft as the inspector follows a circumnavigation trajectory, \mathbf{r}_d and $\dot{\mathbf{r}}_d$ were selected as

$$\mathbf{r}_d = [x_d \ 0 \ 0]^T \quad \text{and} \quad \dot{\mathbf{r}}_d = [0 \ 0 \ 0]^T \quad (6)$$

In other words, the visual navigation system measures the panning offset as the inspector satellite rotates off target. The translational controllers then react by nulling the pan offset of the target in the field of view.

Once the inspector satellite achieves a steady-state rotation rate, that is, with the attitude control law in Eq. (1), and a constant tangential velocity with the control law given by Eq. (4), a feedforward force term, denoted by f_{FF} , is used to compensate for the centripetal acceleration, as follows:

$$\mathbf{f}_{\text{FF}} = m \mathbf{r}_d |\boldsymbol{\omega}_d|^2 \quad (7)$$

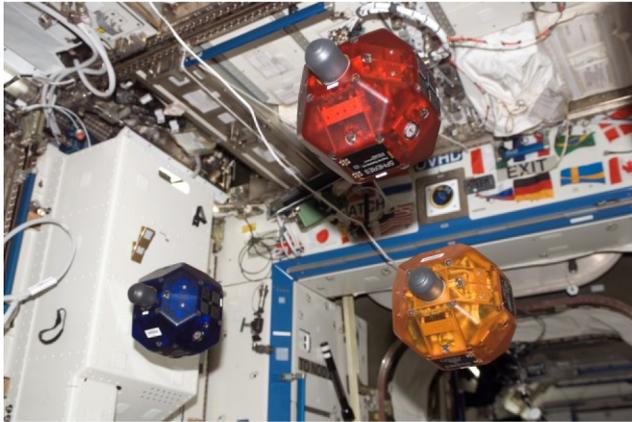


Fig. 2 Picture of three SPHERES satellites performing a test onboard the ISS.

in which m denotes the satellite mass. The resulting control input force is then given by

$$\mathbf{f} = \mathbf{f}_{\text{FB}} + \mathbf{f}_{\text{FF}} \quad (8)$$

III. Hardware Design and Description

The existing SPHERES research facility and the recently developed VERTIGO Goggles that enable vision-based navigation research are summarized here for completeness. For more details on SPHERES and VERTIGO Goggles, the reader is referred to the work of Mohan et al. [18] and Tweddle [12], respectively.

A. SPHERES

From 1999 to 2000, as part of a capstone design course at the MIT Space Systems Laboratory, the SPHERES nanosatellite test bed was developed to provide researchers with an experimental laboratory to validate autonomous guidance, navigation, and control (GN&C) algorithms for advanced space mission scenarios, such as formation flying, and rendezvous and docking. The SPHERES facility consists of three nanosatellites, which can autonomously maneuver in a 6-degree-of-freedom (DOF) environment. Figure 2 shows a picture of three SPHERES satellites onboard the ISS. The test bed includes both a ground segment and a space segment, each able to operate independently of one another. The test bed is primarily designed to operate inside the ISS microgravity environment to represent the realistic dynamics of spacecraft during close proximity operations, but it can also operate onboard NASA's reduced-gravity aircraft as well as in a 3-DOF environment on a large flat floor or on a laboratory air table. The algorithms are first validated in a high-fidelity MATLAB® simulation environment. After experimental testing with the SPHERES hardware in the 3-DOF air-bearing facility at the MIT, the GN&C software is packaged and sent to NASA for testing on the ISS. The software (written in C) consists of a series of modules, each accomplishing a specific task (navigation, control, thruster management, and maneuver sequencing). The modularity of the code allows flexibility in designing an experiment through the selection of different combinations of modules. Inheritance between experiments and reuse of previously validated software modules increase the robustness of the code and enable incrementally mature GN&C technologies.

Each of the three SPHERES satellites is self-contained, and is equipped with three-axis gyroscopes and accelerometers, 12 cold gas thrusters and a CO₂ tank for propulsion, AA batteries for power, as well as two low-power RF links to communicate with the other satellites and with an ISS laptop throughout a test session (for real-time monitoring and postprocessing analysis). Its onboard computer is a Texas Instruments C6701 floating-point digital signal processor (DSP) that has a clock speed of 167 MHz and 16 MB of RAM. The navigation system employs time-of-flight data of ultrasonic signals emitted from external transmitters and received by microphones on

the satellites. In other words, the baseline measurement is the time that the ultrasonic signal takes to travel from a given transmitter mounted at a known location in the test volume to a given microphone (receiver) located on the satellite. Given that there are five beacons mounted on the walls and 24 microphones on each satellite, there is a potential of 120 measurements per satellite per measurement cycle. The time-of-flight data can be converted to range data using the known speed of sound in the ISS. At a 5 Hz frequency, these data are then merged with the three-axis gyroscope measurements into an extended Kalman filter to compute the 6-DOF state vector (position, velocity, quaternion, and angular velocity). The resulting precision on the estimates is a few millimeters in position, and approximately 1 deg in attitude in most of the testing volume [19].

B. VERTIGO Goggles

The VERTIGO Goggles, depicted in Fig. 3, are an upgrade to the SPHERES satellites that enable computer vision based navigation algorithms to be developed, implemented, and tested in a microgravity environment aboard the ISS. They include a pair of monochrome stereo cameras, an embedded Ubuntu Linux computer, a Wi-Fi card, lithium batteries, and the associated electromechanical hardware to operate safely in the ISS environment. Two VERTIGO Goggles were launched to the ISS in October of 2012, and were successfully used by astronaut Thomas Marshburn for the first time in February 2013. A checkerboard camera-calibration target was launched with the Goggles, and a recalibration using OpenCV was successfully performed while on orbit. Table 1 provides the specifications of the VERTIGO Goggles hardware.

The Goggles can be expanded through a single 1 Gbps Ethernet port, two universal serial bus 2.0 ports, and a 802.11n Wi-Fi connection, or by removing the optics mount (four thumbscrews) and making use of the 50-pin power and data connector.

IV. Actuation and Reconfigurable Thrusting

The original SPHERES satellites were designed to have the c.m. and geometric center of the satellite coincide. While this is an expected design choice, the addition of extra mass to the satellite with the VERTIGO extension moves the c.m. away from the geometric center of the satellite. Furthermore, the addition of asymmetric mass to the system complicates the rotational and translational axes of the satellite in a different way. The original control software for SPHERES relied on the assumption of symmetrical physical properties. The addition of the Goggles hardware has required several more significant changes to the control software. To illustrate this, consider the numerical values pertaining to the physical hardware used and reported in Table 2.

The Goggles hardware is only around 27% of the total system mass, yet it contributes more than 63% of the inertia. The c.m. of the total system is moved by 25 mm from the original satellite geometric center. More specifically, the c.m. location offset vector in the satellite body frame is



Fig. 3 SPHERES satellite equipped with the VERTIGO Goggles operating onboard the ISS.

Table 1 VERTIGO Goggles hardware specifications

Characteristic	Description
Cameras	Two monochrome 640 × 480 complementary metal–oxide–semiconductor (CMOS) cameras
Lenses	Wide-angle 2.8 mm f/1.3 closed-circuit television (CCTV) lens with manual iris and focus
Lighting	Two 617 nm, 137 lm exposure synchronized light-emitting diodes
Computer	VIA Pico-ITX EPIA-P830: 1.2 GHz × 86 VIA Nano Processor 4 GB RAM, 1 MB L2 cache, SSE3
Operating system	Linux Ubuntu 10.04 server with OpenCV 2.3.1 and Intel Integrated Performance Primitives preinstalled
Storage	Two 64 GB flash serial advanced technology attachment drives
Wi-Fi	802.11n Universal serial bus
Power consumption	17 W (idle), 23 W (typical), 30 W (maximum)
Battery	Nikon EN-EL4a 11.1 V, 2500 mAh (approximately 90 min run time)
Mass	1.750 (with battery), 1.588 kg (without battery)

Table 2 SPHERES and VERTIGO physical parameters

Parameter	Unit	SPHERES	VERTIGO
<i>Mass</i>			
M	kg	4.25	1.75
<i>Inertia</i>			
I_{xx}	kg · m ²	0.0241	0.006
I_{yy}	kg · m ²	0.0234	0.038
I_{zz}	kg · m ²	0.0201	0.038

$$\mathbf{R}_{GC \rightarrow CM} = [0.025 \quad 0 \quad 0.003] \text{ m} \quad (9)$$

This translates into a c.m. disturbance of around 11% the original satellite’s longest dimension. This change is separated from the satellite control laws through the use of a reconfigurable thruster-mapping strategy, which is discussed as follows. The disturbance to the rotational dynamics of the vehicle is affected to a much larger extent. The addition of the 1.75 kg VERTIGO Goggles component at a lever-arm distance of 0.168 m (satellite geometric center to VERTIGO Goggles c.m.) increases the original rotational inertia by more than 170% along the y and z body reference frame axes. The x axis is not perturbed as significantly, only 27%. The drastic and asymmetric perturbation to the rotational inertial properties complicates the control requirement of the SPHERES satellites. The problem was further worsened in that the original SPHERES control software architecture has been developed over a period of nearly a decade with mostly symmetrical inertia properties.

A. Actuation Strategy

As mentioned earlier, the SPHERES satellites control their rotational and translational motions by the use of 12 × CO₂ gas on-off thrusters evenly distributed around the satellite body. The thrusters are driven from a common pressurized gas tank through a regulator. However, the use of thrusters as the only actuators to steer the spacecraft leads to dynamic coupling between rotation and translation. As a result, the designed continuous control forces and torques described in Sec. II have to be mapped into the thrusters’ forces. Because of the thrusters’ on-off nature, the output of the thruster-mapping algorithm is the input of the pulse-width modulator that modulates a continuous thrust in a pulsed thrust aimed at minimizing the difference between the actual pulsed control and the designed continuous control [20–22].

Typical thruster-mapping strategies are based on simplex methods [23] or more advanced approaches, such as the Lyapunov-based methods that minimize the number of thrusters activated [24]. More specifically, in the present work, control forces and torques are combined together by a simple thruster-mapping function that calculates, at a frequency of 1 Hz, the thruster force of each of the 12 individual thrusters, denoted by $u_n, \forall n = 1, \dots, 12$. The mapping between thruster forces and body forces and torques is described as follows:

$$\begin{bmatrix} f \\ \tau \end{bmatrix} = \mathbf{A} \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_{12} \end{bmatrix} \quad (10)$$

in which \mathbf{A} is a 6 × 12 matrix that specifies the relation between individual thruster firing times and the forces and torques defined with respect to the c.m. of the satellite. The mapping is generally hard coded, as the inertia tensor and mass properties of the satellite were expected to remain constant over the course of the SPHERES test session. Indeed, thus far, the SPHERES satellites have operated under the constant c.m. assumption.

The addition of the VERTIGO Goggles hardware moves the c.m. position of a SPHERES satellite, as described by Eq. (9). Figure 4 conceptually shows how the translational thrusters of the SPHERES satellite need to fire to maintain rotation free translation actuation, allowing a stable circumnavigation around the target object. Notice how the mass asymmetry translates into asymmetric thruster firings.

Future experiments related to SPHERES and VERTIGO are expected to include robotic manipulation tasks, which involve various c.m. variations during flight operations of the satellite. A dynamic reconfigurable thrusting capability will allow future experiments to vary the c.m. location of the system, as the satellite interacts with or manipulates its environment in real time. This capability — together with visual-based navigation and autonomous volumetric object reconstruction — makes robotic operations, such as maintenance or assembly of objects, more achievable.

B. Real-Time Reconfigurable Thrust Mapping

Building in part upon a reconfigurable thruster-mapping technique from a previous work at the MIT Space Systems Laboratory [25], a real-time reconfigurable thrusting-computation capability was developed and integrated for the VERTIGO experiments. The developed

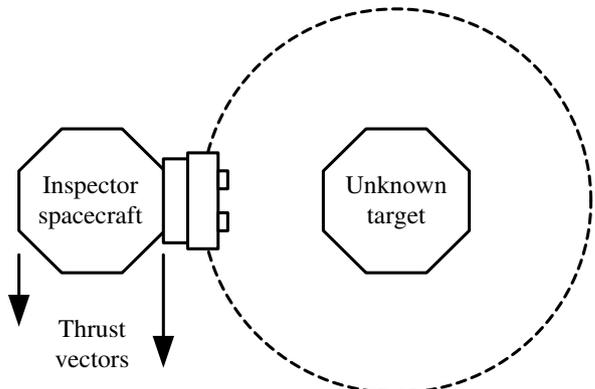


Fig. 4 Different thrust requirements to circumnavigate around a target, considering an offset c.m.

thruster-mapping algorithm dynamically adjusts the firing time asymmetry for each thruster to accommodate changes in the c.m. of the satellite. Although a fully dynamic reconfigurable mixing strategy is not required for VERTIGO–SPHERES operations, the capability has been nevertheless established in the current work. Such a strategy will be most useful for future spacecraft robotic operations involving dexterous manipulation arms. The matrix A , from Eq. (10), is dynamically modified according to the varying system parameters. The 3-D force impulse (newton per second) of each of the 12 thrusters is hard coded in a 3×12 input-force matrix \hat{M}_b . Thereby, the directionality of the input force is described in the row-wise unit direction vectors in the \hat{M}_b matrix, as follows:

$$\hat{M}_b = \begin{bmatrix} f_{x1} & \cdots & f_{x12} \\ f_{y1} & \cdots & f_{y12} \\ f_{z1} & \cdots & f_{z12} \end{bmatrix} \quad (11)$$

Note that thruster variation and misalignments are accommodated through the proper definition of \hat{M}_b . The 3-D relative position of the thrusters is described in a similar matrix, R_0 . The translational vector for each thruster is described in terms of a particular reference point inside the satellite. Conventionally, this has been the geometric center of the SPHERES satellite, which was assumed to coincide with the c.m. of the satellite. A second offset vector, geometric center location to the c.m. location, is now defined to compensate for changes in the c.m. position, and is denoted by $R_{GC \rightarrow CM}$. The new thruster position matrix is dynamically computed:

$$R_{CM} = R_0 - R_{GC \rightarrow CM} \quad (12)$$

The torque about the c.m. of the satellite, denoted τ_{CM} , is computed, column by column, for each thruster through a standard cross product:

$$\tau_{CM}^T = R_{CM}^T \times \hat{M}_b^T \quad (13)$$

The thruster-force matrix \hat{M}_b and the computed torque mappings are stacked to form the new A matrix:

$$A = \begin{bmatrix} \hat{M}_b \\ \tau_{CM} \end{bmatrix} \quad (14)$$

Because the thrusters allow a full 6-DOF actuation, A is full rank, and the pseudoinverse of A can be used to solve for the inverse of Eq. (10):

$$A' = A^T A \quad (15)$$

The Cholesky decomposition [26] is used to solve for the inverse of A' as part of the satellite control cycle computations to obtain the reconfigured force per thruster:

$$\begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_{12} \end{bmatrix} = (A')^{-1} A^T \begin{bmatrix} f \\ \tau \end{bmatrix} \quad (16)$$

Finally, the control software cycle opens and closes the solenoid-valve-actuated thrusters according to their respective firing times, which are calculated by a pulse-width modulation scheme that divides the required force with the known thruster impulse per time thrust value. The maximum firing time is set to 400 ms per 1 Hz cycle. The SPHERES satellites restrict the minimum and maximum firing times for each thruster. In the event of a thruster firing time exceeding the maximum allotted time, all thrust firing times are scaled down, such that the maximum firing time is within the allowable firing time limit.

V. Image Processing for Relative Navigation

In order for a spacecraft to inspect a target object, it must use an estimation algorithm that is capable of determining the relative pose of the inspecting vehicle, so that its planning and control system can maintain a safe distance while keeping the target object in view. A typical method to achieve this would be to track feature points on the object, and use a visual odometry algorithm [27–30]. Visual odometry algorithms would solve for the relative position and orientation between the target and the inspector, thereby estimating a total of 6 DOF. One problem with this approach occurs if the target is tumbling or spinning. A visual odometry algorithm would assume that the target is static, and estimate a circular trajectory for the motion of the inspector even if the inspector spacecraft was stationary in inertial space. This type of estimate would be problematic if no other reference to an inertial reference frame was considered, due to the fact that a command to station keep with the target object would actually have the inspector spacecraft follow a circular trajectory and expending considerable fuel in the process. This is the primary reason why a visual odometry approach was not considered in this work.

Alternatively, the ability to estimate the target object's c.m. would provide a measurement that does not vary as the target object tumbles or rotates. However, the c.m. of the target object is unobservable from a pair of stereo images taken at a single instant of time. Instead, an estimate of the center of geometry of the visible portion of the target object is used as an approximation of the c.m. Clearly, the center of geometry will be an inherently biased estimate of the c.m., and may, in fact, vary as the target object spins. However, if a lower-accuracy navigation and control system is acceptable for an application, this type of approach would be largely independent of tumbling and spinning motions. The image processing strategy developed is described in the following subsection.

A. Image Processing

The first step is to capture a set of raw stereo images, shown in Fig. 5. It is important to note that, when observing a tumbling or spinning object, both stereo cameras must be precisely synchronized to capture images at the same time. Additionally, the exposure time



Fig. 5 Raw stereo images before undistortion.

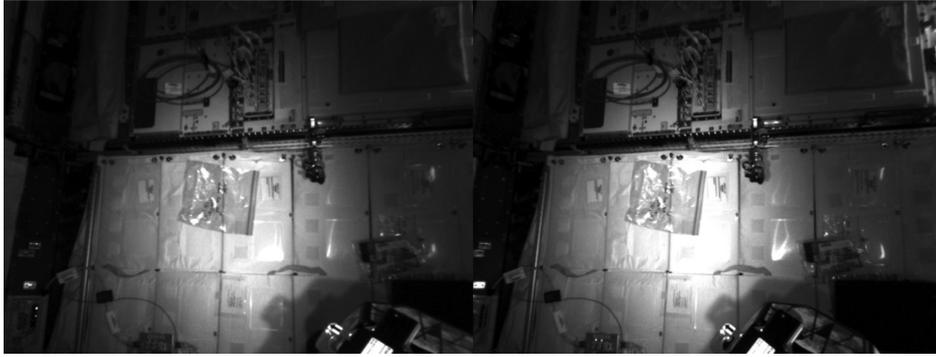


Fig. 6 Stereo images after undistortion.

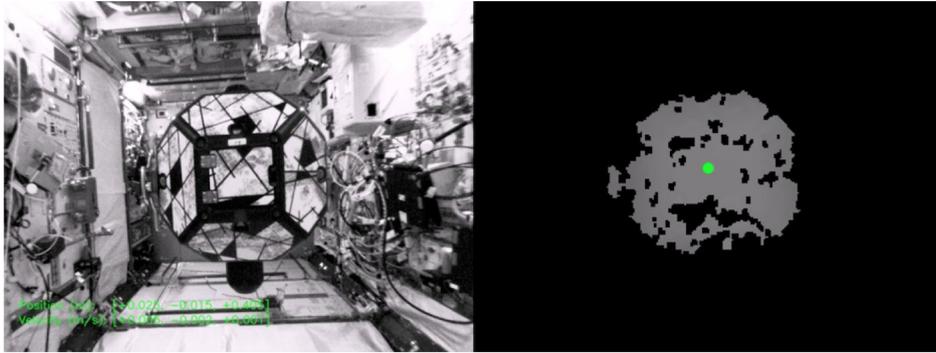


Fig. 7 Stereo disparity map.

must be set low enough that there is very little motion blur. Note that an object would need to be moving at 32 cm/s (or rotating at 30 rpm if it had a 10 cm diameter) before it caused a blur across three or more pixels using the Goggles when they are configured for a 10 ms exposure. Both of these requirements are met by the VERTIGO Goggles hardware and experimental configuration.

The second step is to remove any lens distortions and to adjust for any misalignment between the image sensors and lenses, as well as between the two image sensors themselves. This is achieved using the publically available (i.e., open source) OpenCV library [31]. The result of this rectification and undistortion step is shown in Fig. 6.

The third step is to compute a stereo disparity map. Because it is likely that there will be other objects and clutter in the background when the tests are performed aboard the ISS, it is useful to filter out parts of the stereo disparity map points that are too far away. The OpenCV's stereo block-matching algorithm was originally considered as a method to compute the disparity map, due to its speed and simplicity. However, because block matching is a local disparity algorithm, the resulting disparity map was often sparse and the field of view was restricted. Instead, the algorithm by Geiger et al. [32] was used. This algorithm uses belief propagation with a prior obtained using the Delaunay triangulation of matched Sobel features to compute a global disparity map with a relatively low computational cost. Once the disparity map is computed, the remaining points are triangulated to compute a 3-D point cloud. The mean value of this point cloud is calculated and used as the estimate for the geometric center of the target object, which is an inherently biased estimate of the c.m. The velocity is computed using a first-order difference.

In Fig. 7, the left-hand image shows the original image from the left camera, whereas the right-hand image shows the thresholded stereo disparity map, where a green dot is overlaid at the location of the geometric center, and the position and velocity of this point, in meters and meters per second, respectively, are shown as green text overlaid on the left-hand image. Note that these measurements represent the relative position and velocity between the geometric center of the object and the origin of a reference frame centered at the focal point of the left camera.

Additionally, ground experiments revealed that, although there can be a sizable rotation that occurred between the two images (in the case

of a rotating target), the position and velocity estimates overlaid in the two images show that the estimates only changed by a few millimeters and millimeters per second. This supports the conclusion that, although this type of estimator is inherently biased, it has a high repeatability that will not cause the inspector spacecraft to expend an overly large amount of fuel due to tracking sensor noise. Additionally, it is helpful to note that a tape measure was used in a ground experiment to measure the geometric center/rotation of a target object, and it showed that the vision-based estimate was accurate to within 1 or 2 cm for this particular case.

B. Computational Performance

A breakdown of the computational time is shown in Table 3. The average times, in terms of milliseconds, are listed for the main algorithmic elements specified in the first column. The third column identifies whether the 167 MHz SPHERES DSP or the 1.2 GHz Goggles \times 86 computer performed the computation.

The first algorithmic element is pixel-wise preprocessing, which mainly includes the histogram equalization of both of the stereo images. The second element, which involved computing and thresholding the stereo disparity map [32], was the most computationally intensive. This high level of computation is expected in image processing algorithms that have a two-dimensional locality due to the fact that these characteristics lead to an abnormally high number of cache misses in modern computer architectures [4].

The third and fourth algorithmic elements involve computing the 3-D centroid and its velocity, filtering this estimate, logging data, and

Table 3 Computation time

Algorithmic element	Average time, ms	Computer
Pixel-wise preprocessing	6.7	Goggles
Compute disparity map	294.8	Goggles
3-D Centroid	4.0	Goggles
Centroid velocity, finite-impulse-response filter, data logging and communications	34.8	Goggles
control law computation	5.9	SPHERES

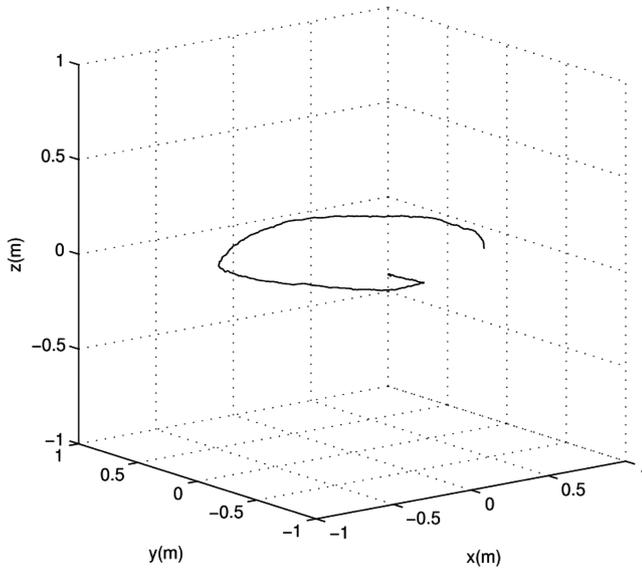


Fig. 8 Simulation results: 3-D visualization of the inspector spacecraft circumnavigation trajectory.

communicating between the Goggles and the SPHERES satellites. The last algorithmic element is the computation of the forces and torques based on the sensor measurements (vision and gyroscope), and then computing the thruster firing times based on these desired forces and torques.

From this table, it is clear that the relative navigation and control approach presented in this paper has a low computational requirement due to its inherent simplicity. Also, it is clear that the majority of the computation time is spent in computing the disparity map from the two stereo images. If an alternative sensor that provides depth information, such as a flash LIDAR or time-of-flight camera, was used, this step would not be necessary, and the computation time could be reduced by more than a factor of 5.

VI. Simulation Results

The simulation of the SPHERES and VERTIGO system was developed and used to evaluate the flight software before testing aboard the ISS. The simulation evaluation described here was built on the existing SPHERES MATLAB simulation environment, adding VERTIGO specific features. The specific satellite properties, including changes in mass, c.m., c.m. to ultrasonic receiver locations, and inertia tensor, were modified. Additional callback functions,

which would imitate the software processes on the VERTIGO hardware, were also included into the simulation environment. The following questions may be asked to establish the capability and robustness of rotational inspection around an unknown object, based on the techniques proposed previously:

- 1) Can the visual navigation system, together with the inertial sensors, be effectively used to navigate relative to the unknown object?
- 2) Is SPHERES equipped with VERTIGO Goggles able to follow a stable and controlled path during inspection maneuvers?

To gather initial results and information relevant to answering these questions, numerical simulations were conducted, and the results are presented in Figs. 8–10. The selected inspection scenario consists in the following: the test starts with the inspector satellite at a short distance from the object, and with the object in the field of view. The inspector satellite approaches and maintains a predefined relative distance from where the visual–inertial circumnavigation maneuver is initiated. Rotation around the object is achieved by changing the desired angular velocity in the inspector yaw axis to a nonzero value, as discussed in Sec. II. The target then starts to drift to the side of the camera view. The visual feedback loop registers this offset and commands a control translational force to maintain the target in the field of view. The translational and rotational dynamics of the inspector are controlled independently, for which the reconfigured thruster-mapping strategy accounts asymmetric mass distribution.

Figure 8 depicts the 3-D trajectory around the unknown object, and Fig. 9 illustrates the position and velocity for the inspector satellite with respect to an inertial reference frame, where the solid lines, the dashed lines, and the dotted lines correspond to the x , y , and z axes, respectively. As shown in Fig. 9, the x and y positions both describe a sinusoid as the inspector spacecraft moves around the object satellite. The next important criterion of the controller is the ability to maintain the commanded range from the target object. Figure 10 shows that the inspector maintains a range of 0.68 m, indicated by the solid line, which corresponds to the desired range, indicated by the dashed line. These results predict a good control authority on range.

VII. ISS Flight Results

VERTIGO was operated on 26 February 2013 and 12 March 2013, during ISS Expedition 34, by Flight Engineer Thomas Marshburn and Commander Kevin Ford, respectively. On-orbit operations consisted of installing the VERTIGO hardware to the SPHERES hardware, adding texture stickers to the target SPHERES, performing an on-orbit calibration of the cameras, uploading the navigation and control code into the flight hardware, executing the code, and downloading data to the station laptop for postprocessing analysis of the data.

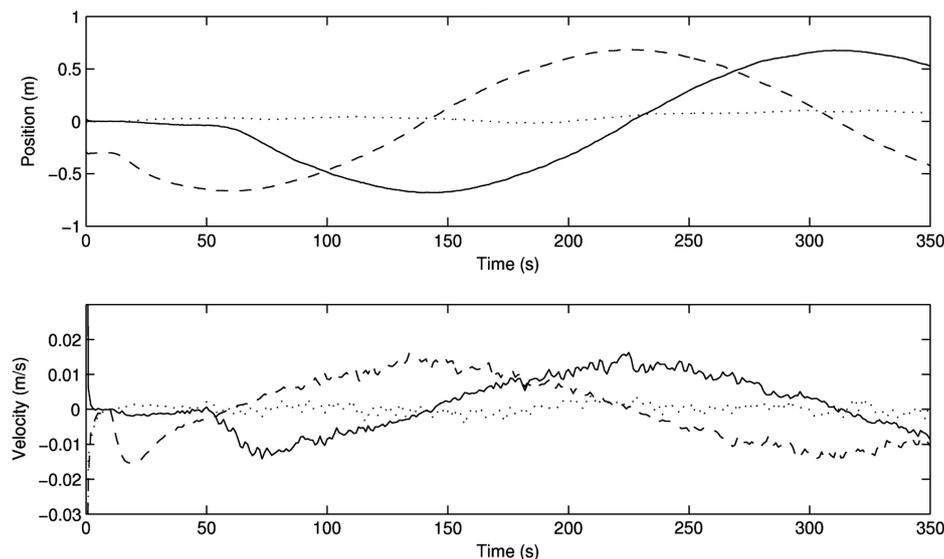


Fig. 9 Inspector spacecraft position and velocity along the x , y , and z axes.

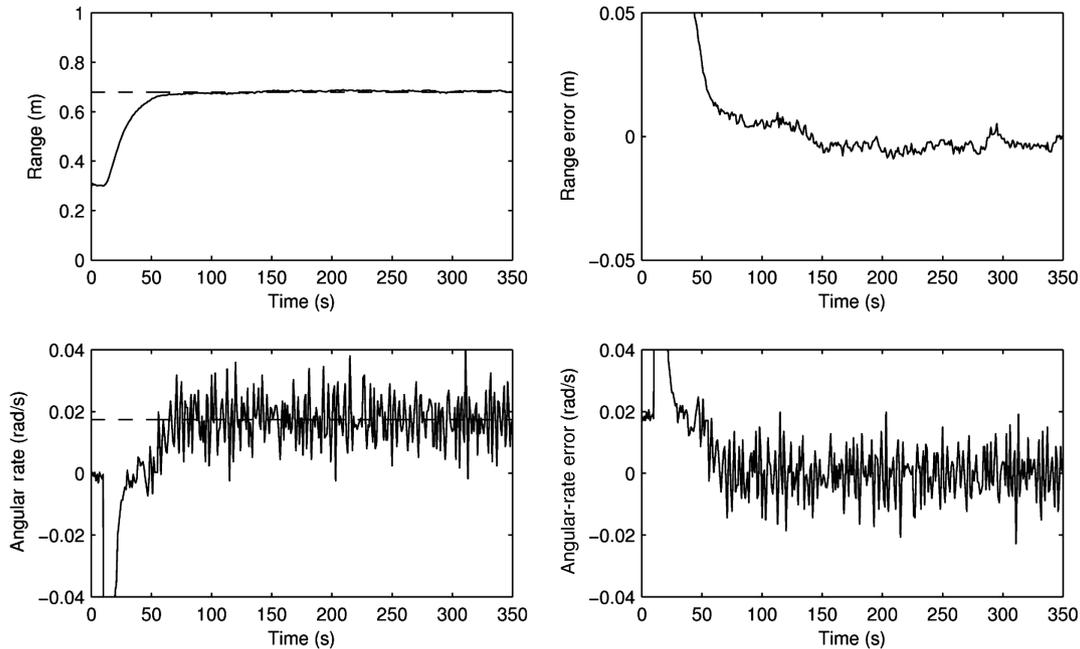


Fig. 10 Range to target spacecraft and angular velocity of the inspector spacecraft.

The objective of the first day of operation consisted mainly in performing an initial checkout of the integrated VERTIGO hardware. The circumnavigation maneuver was not demonstrated successfully during the first test session. Data from this test showed higher-than-expected noise levels in the relative velocity measurement of the object through the stereo-camera system. This high noise is most likely due to the first-order differentiation of the noisy relative-position measurements. (A possible solution to reduce the high noise levels would be to use a dynamic filter, such as a Kalman filter, for example.) The higher noise levels coupled through controller gains, ultimately resulting in firing times exceeding the maximum allotted time. The long firing times were, per design, scaled down to fit the allowable thrusting budget of the satellite, thereby reducing the control authority of the force and torque control loops. After a careful analysis of the flight data, it was found that this problem could easily be solved by retuning appropriately the controller gains. The updated controller gains were sent to the ISS in preparation for day operation, and the flight results successfully demonstrated appropriate firing times (i.e., not exceeding the maximum allotted time).

The second day of operation was mostly dedicated to performing the visual-inertial navigation and control experiments. Similar to the numerical simulation test, the on-orbit experimental tests were divided into three maneuvers, each ending after a predetermined lap of time, and leading to the next one. From an operational point of view, the first maneuver, lasting 10 s, consisted in letting both SPHERES in a free-floating mode upon being released in the test volume by the astronaut. Technically, the objective of the first maneuver was to let the state estimator converge from the acoustic-metrology measurements. The estimated state vector was then used with feedback control — during the second maneuver — to set the inspector and object satellites in the respective initial conditions inside the test volume. Forty seconds was allotted for this initial positioning maneuver. The desired three-axis initial positions for each spacecraft, defined with respect to a coordinate reference frame attached to the ISS, are the same as those used in the numerical simulations (i.e., $r_0 = [0.00 \ -0.68 \ 0.00]$ m and $r_0 = [0.00 \ 0.00 \ 0.00]$ m for the inspector and target spacecraft, respectively). Note that the origin of the coordinate reference frame system is located in the center of the test volume, which also corresponds to the position of the geometric center of the target spacecraft for this experiment. The third maneuver started at 50 s into the test. Maneuver 3 would only last a few seconds, and is used to initiate the spinning motion on the object satellite, if required. Maneuver 4 would last 300 s, and performed the actual visual-inertial inspection trajectory.

Figure 11 shows four frames taken from the actual footage during the test session experiment, on 12 March 2013. The first frame depicts both satellites at their respective initial positions, with the inspector spacecraft being the one shown at the bottom of the first frame, that is, the spacecraft located closer to the camera, which was used to record the footage (i.e., the ISS JEM camera). The remaining three frames show the inspector spacecraft at different instants of its autonomous inspection trajectory. As qualitatively illustrated in Fig. 11, the inspector successfully demonstrated the ability to complete a fully autonomous inspection trajectory around an unknown object in space, using only a pair of stereo cameras and a gyroscope.

The ultrasound-beacon-based navigation system described in Sec III provided the three-axis position and velocity of the inspector, which are plotted in Figs. 12 and 13. While the trajectory of the inspector satellite around the object was intended to remain in the xy plane, the actual trajectory has some noticeable motion in the z direction, as observed in Fig. 13. There has been some discussion as to the primary cause of this inclination, of which one or a combination of two ideas dominate. One probable cause is that the inspector satellite is not perfectly static or aligned with the xy plane at the beginning of the vision-based navigation and control maneuver (i.e., fourth maneuver). This is clearly shown in the position history in Fig. 13, in which, at $t = 50$ s, the desired initial position for the inspector is not perfectly reached. In other words, at the end of the initial positioning maneuver (second maneuver), the inspector spacecraft was still moving toward its desired initial position, as illustrated in the velocity plot in Fig. 13, introducing additional out-of-plane linear momentum. Furthermore, the inspector satellite would not be perfectly aligned in the inspector's roll frame. This initial misalignment would contribute to motion in the Japanese experiment module (JEM) perceived vertical direction. While it may seem that this problem can be avoided through increasing the time allotted for the initial positioning maneuver, it must be remembered that the position of the inspector satellite is subject to noise coupling from the acoustic ranging system and minimum firing time limitations on the satellite thrusters. All these factors contribute to constant non-zero velocity of both the inspector and object satellites.

Another source of error causing the off-plane motion might be due to a small mismatch in the expected and actual c.m. locations, particularly in the z direction, which could have resulted in a nonideal thrusting configuration scheme. Thrusting computations will not accommodate all thruster and mass variations in the system, as to always perfectly align the thrust vector with the c.m. of the satellite. Aspects such as CO_2 gas expenditure are, for example, not

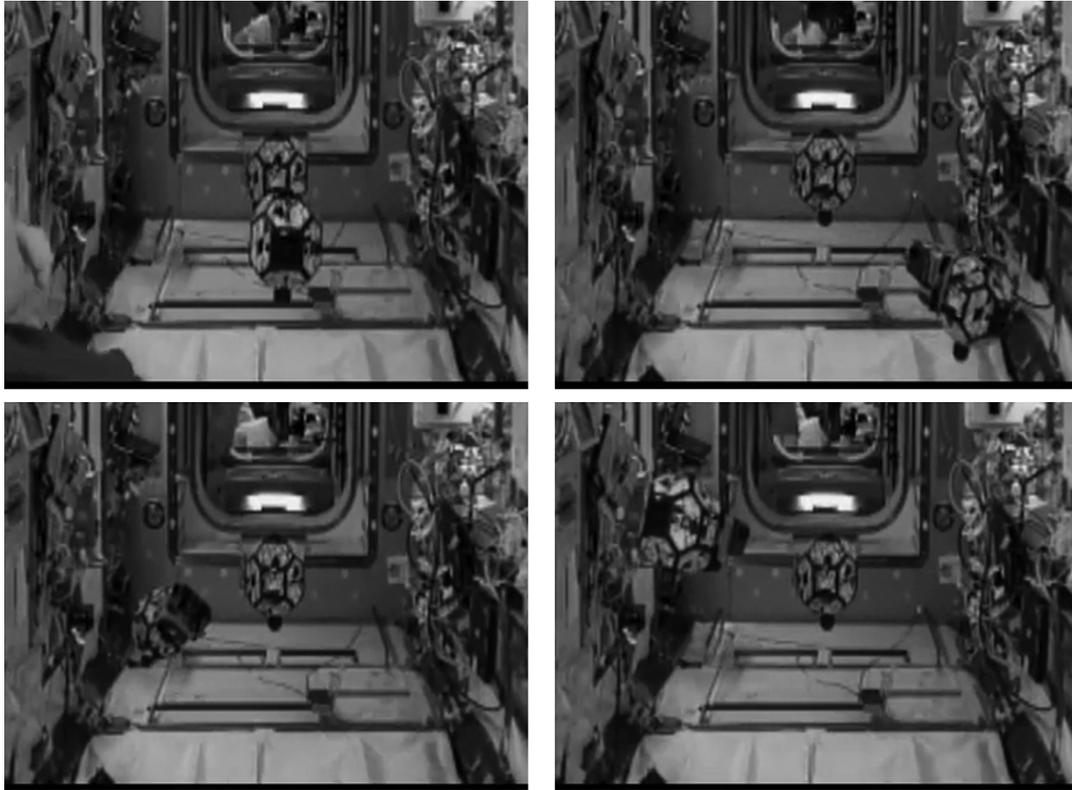


Fig. 11 On-orbit experimental results: autonomous spacecraft robotic inspection aboard the ISS.

considered. Therefore, during the initial acceleration phase, as the inspector satellite starts the circumnavigation maneuver, there will be large translational thrusting to transfer the linear momentum into the system. The expected xy navigation plane will, therefore, also be perturbed through the aforementioned errors.

The position of the inspector satellite over the course of the experiment is better visualized with the 3-D plot, provided in Fig. 12. This figure shows that, despite the off-nominal motion along the z axis, the inspector successfully followed a circular trajectory with a relatively constant radius corresponding to the desired inspection distance of 0.68 m.

The top plot in Fig. 14 shows the range between the object and inspector satellites as the solid line. The range set point during the visual-inertial circumnavigation maneuver is shown, indicated by

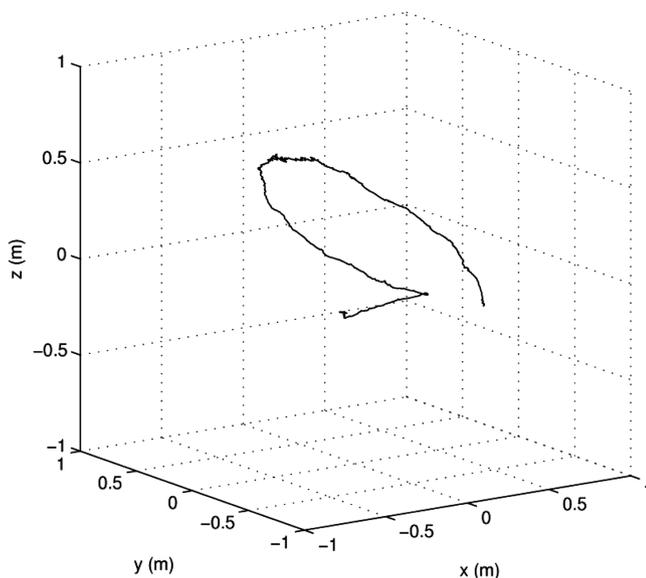


Fig. 12 On-orbit experimental results: 3-D visualization of the inspector spacecraft circumnavigation trajectory.

the dashed line. There is a clear nonzero bias in relative range between the inspector and the object. This constant range error, although relatively small, is attributed to insufficient feedforward translational thrust force. A few main factors may contribute to the reduced force. First is the lower-than-expected thruster performance from the CO_2 thrusters. Note that the SPHERES satellite thrusters with a hand gauge set the gas regulator, and the thrust scaling is furthermore taken as the mean performance over all thrusters of the three satellites, which are on orbit.

Second, part of this error may also be attributed to variation in vehicle mass. The circumnavigation test was run soon after a CO_2 gas tank swap, whereby the mass of the satellite is above the mean mass of the vehicle, which is assumed to be at a half-filled tank. While this error is not desirable, the error is comparatively small, and the feedback system remains stable. Two other potential contributors to the nonzero bias have been considered. While the feedforward thrust term is computed by using the desired range and rotation rate, as described by Eq. (7), any variation in the line-of-sight vector to the center of the circumnavigation path would result in a projection of the required feedforward force opposing the centripetal acceleration acting on the inspector.

Lastly, the vision system tends to see more of the direct nearest surface of the object, rather than the slanted sides. Depending on the lighting conditions, this would result in a biased range measurement to the object.

The rotation rates of the inspector satellite during the visual-inertial circumnavigation maneuver are an important indicator to the capability of the control law. The lower plot in Fig. 14 shows the yaw rotation rate of the inspector satellite. The roll and pitch rotation rates are omitted, as these values were stable around zero throughout the circumnavigation maneuver. Figure 14 illustrates some rotational motion during the first 50 s while the satellites are brought to their initial positions. At 50 s, the visual-inertial circumnavigation maneuver is initiated, and the yaw rotation rate is ramped from zero to the desired yaw rate set point of 1 deg/s. The ramp-up period was set to 15 s. The satellite followed closely the rotation-rate ramp-input command, but a rate oscillation in the order of 25% set-point rate was observed for the remaining portion of the maneuver.

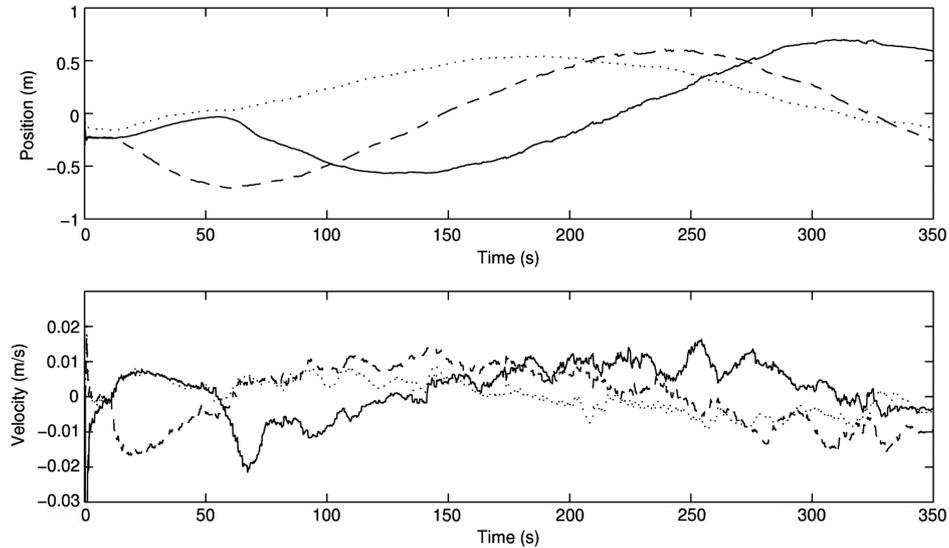


Fig. 13 On-orbit experimental results: inspector spacecraft position and velocity.

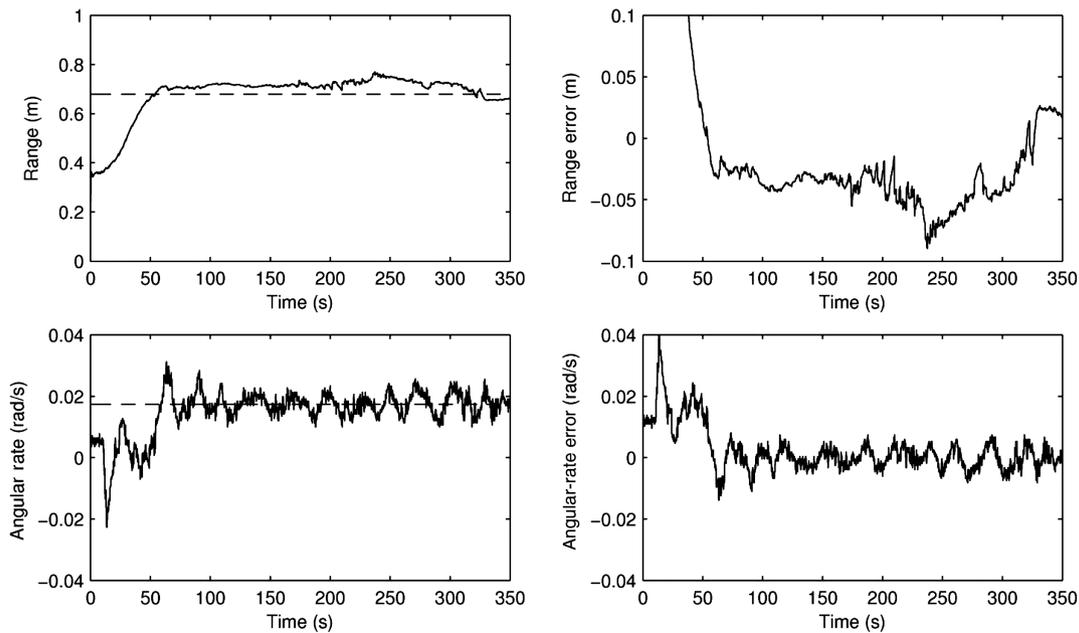


Fig. 14 On-orbit experimental results: range to target and angular velocity of inspector spacecraft.

Notice a similar oscillation pattern on the inspector velocities, as shown in Fig. 13. This indicates that there is coupling between the translational and rotational modes of the system via the visual-inertial measurements. The source of this coupling is again attributed to the variation in thruster performance and the nonperfect alignment of the forces and expected c.m. of the satellite. Furthermore, the suggested control architecture, as shown in Fig. 1, does not include any damping for the rotational rates of the inspector satellite. It is therefore suggested that an angular acceleration damping term be added to the controller to ensure that the aforementioned oscillations are actively suppressed.

VIII. Conclusions

This paper presented the system design and flight results from the VERTIGO experiment, the first on-orbit demonstration of relative navigation and control for the autonomous robotic inspection of a noncooperative spacecraft based on computer vision. Specifically developed hardware, techniques, and software modules, such as a pair of monochrome stereo cameras, two illuminating light-emitting

diodes, and a dedicated 1.2 GHz computer and power source; feature detection and matching; relative-position and -velocity determination; and control maneuvers, among others, have been presented and experimentally validated on ISS Expedition 34. Flight results successfully demonstrated a fully autonomous, reliable, safe, and accurate circumnavigation inspection trajectory around an unknown object by relying solely on camera hardware and inertial sensors. One main advantage of the proposed approach is its high-technology readiness level through its simplicity, which minimizes the computational requirements and increases the robustness. In contrast to previous flight demonstrations, VERTIGO is a fully noncooperative experiment, in which neither knowledge nor control of the target vehicle is available during execution.

As future work, the technologies presented in this paper to support the VERTIGO experiment will be employed to collect data, which can be used to build a visual and volumetric model of the unknown and noncooperative object. While such algorithms to reconstruct a model of the object have already been demonstrated on the ground, the next step will consist of incorporating these algorithms on the VERTIGO hardware for in situ operation on the ISS. Ultimately, it is

expected that these flight results will improve the confidence in these advanced technologies, so that they can be applied in present and future spacecraft robotics missions.

Acknowledgments

This research was funded by the Defense Advanced Research Projects Agency's Integrated Research Experiments program under contract NNH11CC25C. Steve Ulrich also acknowledges the Natural Sciences and Engineering Research Council of Canada for its financial support under the postdoctoral fellowship award PDF-438572-2013. The authors would like to acknowledge John Leonard and David W. Miller from the Massachusetts Institute of Technology for their support on the project. Furthermore, the authors would like to thank NASA Astronauts Thomas Marshburn and Kevin Ford for having operated the VERTIGO experiment on the International Space Station during Expeditions 34 and 35. The authors are also very grateful to the U.S. Air Force Space Test Program, the human space flight office at the NASA Johnson Space Center, and the NASA Marshall Space Flight Center payload operations team for their support of the operations. Special thanks go to Aurora Flight Sciences for their efforts in the development of the VERTIGO Goggles hardware.

References

- [1] Geller, D. K., "Orbital Rendezvous: When Is Autonomy Required?" *Journal of Guidance, Control, and Dynamics*, Vol. 30, No. 4, 2007, pp. 974–981. doi:10.2514/1.27052
- [2] Lampariello, R., "Towards Semi-Autonomous Grasping of a Non-Cooperative Target," *IEEE International Conference on Robotics and Automation*, Inst. of Electrical and Electronics Engineers, Piscataway, NJ, May 2012, pp. 1–28.
- [3] Hablani, H. B., "Autonomous Inertial Relative Navigation with Sight-Line-Stabilized Integrated Sensors for Spacecraft Rendezvous," *Journal of Guidance, Control, and Dynamics*, Vol. 32, No. 1, 2009, pp. 172–183. doi:10.2514/1.36559
- [4] Tweddle, B. E., "Computer Vision Based Navigation for Spacecraft Proximity Operations," M.S. Thesis, Dept. of Aeronautics and Astronautics, Massachusetts Inst. of Technology, Cambridge, MA, 2010.
- [5] Linares, R., Crassidis, J. L., and Cheng, Y., "Constrained Relative Attitude Determination for Two-Vehicle Formations," *Journal of Guidance, Control, and Dynamics*, Vol. 34, No. 2, 2011, pp. 543–553. doi:10.2514/1.50053
- [6] Romano, M., Friedman, D. A., and Shay, T. J., "Laboratory Experimentation of Autonomous Spacecraft Approach and Docking to a Collaborative Target," *Journal of Spacecraft and Rockets*, Vol. 44, No. 1, 2007, pp. 164–173. doi:10.2514/1.22092
- [7] Tweddle, B. E., "Relative Computer Vision Based Navigation for Small Inspection Spacecraft," *AIAA Guidance, Navigation, and Control Conference*, AIAA, Reston, VA, 2011, pp. 1–15; also AIAA Paper 2011-6296, 2011.
- [8] MacLean, S. G., and Pinkney, H. F. L., "Machine Vision in Space," *Canadian Aeronautics and Space Journal*, Vol. 29, No. 2, 1993, pp. 63–77.
- [9] Howard, R., Heaton, A., Pinson, R., and Carrington, C., "Orbital Express Advanced Video Guidance Sensor," *IEEE Aerospace Conference*, Inst. of Electrical and Electronics Engineers, Piscataway, NJ, May 2008, pp. 1–10.
- [10] Henshaw, C. G., Healy, L., and Roderick, S., "LIIVe: A Small, Low-Cost Autonomous Inspection Vehicle," *AIAA SPACE 2009 Conference and Exposition*, AIAA, Reston, VA, 2009, pp. 1–9; also AIAA Paper 2009-6544, 2009.
- [11] Tweddle, B. E., Saenz-Otero, A., and Miller, D. W., "Design and Development of a Visual Navigation Testbed for Spacecraft Proximity Operations," *AIAA SPACE 2009 Conference and Exposition*, AIAA, Reston, VA, 2009, pp. 1–14; also AIAA Paper 2009-6547, 2009.
- [12] Tweddle, B. E., "Computer Vision-Based Localization and Mapping of an Unknown, Uncooperative and Spinning Target for Spacecraft Proximity Operations," Ph.D. Thesis, Dept. of Aeronautics and Astronautics, Massachusetts Inst. of Technology, Cambridge, MA, 2013.
- [13] LaValle, S., *Planning Algorithms*, Cambridge Univ. Press, Cambridge, England, U.K., 2006, pp. 633–691.
- [14] Latombe, J. C., *Robot Motion Planning*, Kluwer Academic, Boston, MA, 1991, pp. 1–57.
- [15] Thrun, S., Burgard, W., and Fox, D., *Probabilistic Robotics*, MIT Press, Cambridge, MA, 2006, pp. 309–336.
- [16] Katz, J. G., Saenz-Otero, A., and Miller, D. W., "Development and Demonstration of Autonomous Collision Avoidance Algorithm on the ISS," *2011 IEEE Aerospace Conference*, Inst. of Electrical and Electronics Engineers, Piscataway, NJ, 2011, pp. 1–6.
- [17] Chamitoff, G. E., Saenz-Otero, A., Katz, J. G., and Ulrich, S., "Admissible Subspace Trajectory Optimizer (ASTRO) for Autonomous Robot Operations on the Space Station," *AIAA Guidance, Navigation, and Control Conference*, AIAA, Reston, VA, 2014, pp. 1–17; also AIAA Paper 2014-1290, 2014.
- [18] Mohan, S., Saenz-Otero, A., Nolet, S., Miller, D. W., and Sell, S., "SPHERES Flight Operations Testing and Execution," *Acta Astronautica*, Vol. 65, No. 7, 2009, pp. 1121–1132.
- [19] Nolet, S., "The SPHERES Navigation System: From Early Development to On-Orbit Testing," *AIAA Guidance, Navigation, and Control Conference and Exhibit*, AIAA, Reston, VA, 2007, pp. 1–17; also AIAA Paper 2007-6354, 2007.
- [20] Sidi, M. J., *Spacecraft Dynamics and Control: A Practical Engineering Approach*, Cambridge Univ. Press, Cambridge, England, U.K., 1997, pp. 265–289.
- [21] Tohru, I., Yoshimasa, O., and Kimio, K., "New Design Method for Pulse-Width Modulation Control Systems via Digital Redesign," *Journal of Guidance, Control, and Dynamics*, Vol. 22, No. 1, 1999, pp. 123–128. doi:10.2514/2.4358
- [22] Bernelli-Zazzera, F., Mantegazza, P., and Nurzia, V., "Multi-Pulse-Width Modulated Control of Linear Systems," *Journal of Guidance, Control, and Dynamics*, Vol. 21, No. 1, 1998, pp. 64–70. doi:10.2514/2.4198
- [23] Dantzig, G. B., *Linear Programming and Extension*, Princeton Univ. Press, Princeton, NJ, 1998, pp. 94–146.
- [24] Curti, F., Romano, M., and Bevilacqua, R., "Lyapunov-Based Thrusters' Selection for Spacecraft Control: Analysis and Experimentation," *Journal of Guidance, Control, and Dynamics*, Vol. 33, No. 4, 2010, pp. 1143–1160. doi:10.2514/1.47296
- [25] Mohan, S., "Quantitative Selection and Design of Model Generation Architectures for On-Orbit Autonomous Assembly," Ph.D. Thesis, Dept. of Aeronautics and Astronautics, Massachusetts Inst. of Technology, Cambridge, MA, 2010.
- [26] Trefethen, L. N., and Bau, D., *Numerical Linear Algebra*, Soc. for Industrial and Applied Mathematics, Philadelphia, 1997, pp. 81–82.
- [27] Cheng, Y., Maimone, M., and Matthies, L., "Visual Odometry on the Mars Exploration Rovers—A Tool to Ensure Accurate Driving and Science Imaging," *IEEE Robotics & Automation Magazine*, Vol. 13, No. 2, 2006, pp. 54–62. doi:10.1109/MRA.2006.1638016
- [28] Nistér, D., Naroditsky, O., and Bergen, J., "Visual Odometry," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Inst. of Electrical and Electronics Engineers, Piscataway, NJ, 2004, pp. 652–659.
- [29] Scaramuzza, D., and Fraundorfer, F., "Visual Odometry," *IEEE Robotics & Automation Magazine*, Vol. 18, No. 4, 2011, pp. 80–92. doi:10.1109/MRA.2011.943233
- [30] Huang, A. S., Bachrach, A., Henry, P., Krainin, M., Maturana, D., Fox, D., and Roy, N., "Visual Odometry and Mapping for Autonomous Flight Using an RGB-D Camera," *International Symposium on Robotics Research*, International Foundation of Robotics Research, Stanford, CA, 2011, pp. 1–16.
- [31] Bradski, G., and Kaehler, A., *Learning OpenCV: Computer Vision with the OpenCV Library*, O'Reilly, Cambridge, MA, 2008.
- [32] Geiger, A., Roser, M., and Urtasun, R., "Efficient Large-Scale Stereo Matching," *Computer Vision – ACCV 2010, Part 1, Lecture Notes in Computer Science*, Springer-Verlag, Berlin, Heidelberg, Vol. 6492, 2011, pp. 25–38. doi:10.1007/978-3-642-19315-6_3

D. Spencer
Associate Editor