Proceeding of the 6th International Symposium on Artificial Intelligence and Robotics & Automation in Space: i-SAIRAS 2001, Canadian Space Agency, St-Hubert, Quebec, Canada, June 18-22, 2001.

Robust 3D Vision for Autonomous Space Robotic Operations

Mark Abraham, Piotr Jasiobedzki, Manickam Umasuthan

MD Robotics, 9445 Airport Rd., Brampton, Ontario, Canada L6S 4J3

pjasiobe@mdrobotics.ca

Abstract

This paper presents a model based vision system intended for satellite proximity operations. The system uses natural features on the satellite surfaces and three dimensional models of the satellites and docking interfaces, and it does not require any artificial markers or targets. The system processes images from stereo cameras to compute 3D data. The pose is determined using a fast version of an Iterative Closest Point algorithm. The system has been characterized under representative space-like conditions (illumination, material properties and trajectories) and proven to be robust to this environment.

Keywords

Space vision system, 3D tracking, Model based pose estimation, Iterative Closest Point

1 Introduction

Recent space missions, such as ETS–VII – an autonomous rendezvous and docking of unmanned satellites [3, 11] and the International Space Station assembly [9], have demonstrated successful operations of vision systems on-orbit. These vision systems provided information about the position and orientation of observed spacecrafts and structures during unmanned and manned operations. However, these vision systems relied on the presence of visual targets or retro-reflectors. This restricted vision based operations to predefined tasks only (distances and viewing angles). It also required an ability to control the orientation of the observed spacecraft so, as the visual targets were always visible.

Future space operations, such as, satellite inspection and servicing, will require much more advanced vision systems that will be able to detect and recognize observed objects from arbitrary viewpoints [4]. Such a capability will enable servicing spacecrafts to locate, approach, and dock or capture target satellites autonomously and to perform various servicing tasks [6]. Satellite servicing operations performed on-orbit, for example, inspection, hardware upgrades and refueling will extend the life of the satellites and reduce operational costs by eliminating the launches of hardware, which is already on-orbit.

The technical challenge is to develop advanced and flexible vision systems capable of operating reliably in the space environment. Any approach taken to provide three-dimensional data to the robotic guidance system must be robust and reliable throughout a wide range of approach vectors and lighting conditions. The on-orbit environment consists of intense sunlight combined with dark shadows, which presents challenging lighting situations. The contrast in an image is often beyond the dynamic range of a camera resulting in the loss of some of the data within the scene.

In this paper we describe a vision system developed by MD Robotics that is capable of computing the pose of known objects using only their geometrical models and natural surface features. The system operates reliably under simulated on-orbit conditions (illumination, distance, viewing angle, and partial data loss due to shadows and reflections). Performance of the system has been tested in a laboratory environment representative for satellite proximity operations.

2 Vision System tasks and requirements

Space vision systems provide three-dimensional information about the position and orientation of observed spacecraft or structure. An image of Canadarm preparing to capture a free floating satellite, Spartan, is shown in Figure 1.

Figure 1 Canadarm preparing to grasp the Spartan satellite.

Operating range of such a vision system may extend from approximately 100 m to contact. It is expected that Global Positioning Systems will be used to bring the servicer within this range of the target spacecraft. Different sensors and algorithms will be used for different phases of the satellite operation. The operations are typically divided into phases according to the distance long, medium and short range. At the long range (100 – 20m) the vision system must determine bearing and distance to the satellite of interest and its approximate orientation and motion. At the medium range (20 –2m) the distance, orientation and motion must be determined very accurately and with high confidence. Only when this information is known the servicer spacecraft might approach the target satellite and dock with it using a short range module (below 2m).

The short range subsystem will be used for well defined operations of docking or grasping and the expected range of distance and viewing angles will be restricted by the allowed approach trajectories to the target spacecraft. Lights mounted together with cameras may be used to illuminate the scene. Computer vision techniques that rely on presence of well defined features on or around the interface can be used then. For example, techniques similar to the ones tried by two vision system tested in space [3] and SVS [9] can be used, or a variation on techniques applied in industrial automation can be adopted.

The medium range poses a much more difficult challenge as the viewing angles are unrestricted, the distance varies significantly, and the vision system mostly relies on ambient illumination from the Sun or Earth albedo. For the purpose of our work we assume that the vision system is activated within the medium range and provided with an approximate attitude towards a satellite of interest, its distance and orientation. Techniques for determining such information are currently investigated, and concepts and initial results are presented in a companion paper [7].

At the medium range it is expected that the vision system will be capable of estimating pose of the observed object at a rate of 0.3 – 3 Hz. This requirement is derived from a maximum relative motion of a satellite that still allows capturing or docking with it.

Expected accuracy of a vision system is related to a capture envelope of a mechanical interface. Depending on the interface this envelope can be from 2” (for small mechanisms) to 10“ (Canadarm), the angular misalignment is in the range of 2 – 10 degrees. However, these requirements are defined at contact and are mostly relevant for the short range system. For the medium range system it expected that accuracy in order of 1% of the distance and several degrees of the object orientation will be sufficient.

On-orbit illumination and the imaging properties of materials used to cover satellites pose significant challenges for any vision system. The illumination changes from day to night as the spacecraft is orbiting the Earth, for example for the International Space Station this period is 90 minutes. The objects can be illuminated by a combination of direct sunlight (an intense point-like source creating hard shadows), Earth albedo (extended diffuse and almost shadow-less source) and on-board lights. The insulation materials used in spacecraft manufacture are either highly reflective metallic foils or featureless white blankets that loosely cover the satellite body. This causes specular reflections and lack of distinct and visually stable features such as lines or corners.

3 Vision system architecture

The vision system currently under development at MD Robotics is a solution to the medium range vision problem (20 – 2 meters) for the satellite servicing operation. The system uses stereo cameras and a x86 family 6 computer that runs the developed software. The cameras are rigidly coupled together and may be mounted close to a docking interface of a servicer satellite, inside a cargo bay of a shuttle or on an end-effector of a robotic manipulator. We chose to use cameras instead of scanning rangefinders [2, 8] as they are lighter, cheaper, and contain no moving parts, which allows them to survive better the space environment. However, the cameras require ambient illumination or camera lights for their operation.

Architecture of the developed vision system is shown in Figure 2. The processing is divided into off-line and on-line processing. Off-line processing may be performed on the ground or in space, and includes calibration of the cameras and pre-computing data that will be used during the on-line phase. During the on-line phase the vision system acquires images from the cameras, computes the 3D data, determines and tracks 3D pose of the observed object. The vision system algorithms are embedded in a server that is responsible for communication with the external world, i.e., user interfaces, robot controllers and a spacecraft Guidance, Navigation and Control system.

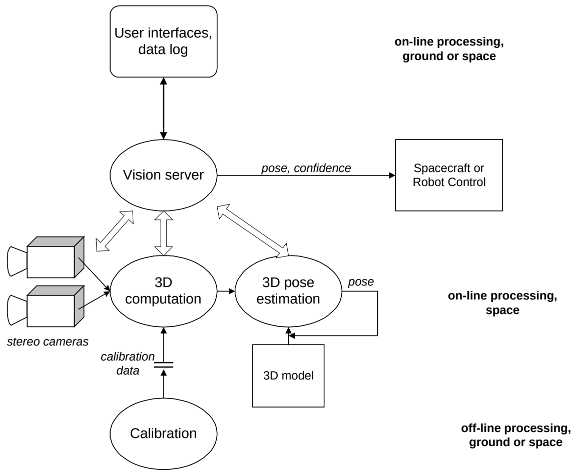

Figure 2 Vision system architecture

3.1 Camera calibration

The system calibration and data initialization is performed off-line and maybe be performed on the ground before the launch or in space. The camera calibration module estimates intrinsic and extrinsic parameters of the stereo cameras. The calibrated camera model includes focal length, geometrical distortions of the camera / lens assembly: radial and tangential, and the location of the principal point in the image plane. The camera calibration algorithm is an extension of recent work on camera auto-calibration [14].

Calibration is performed using several arbitrary (unknown) views of a planar calibration target. The target used in our experiments is shown in Figure 3, but any planar object with detectable surface features is sufficient.

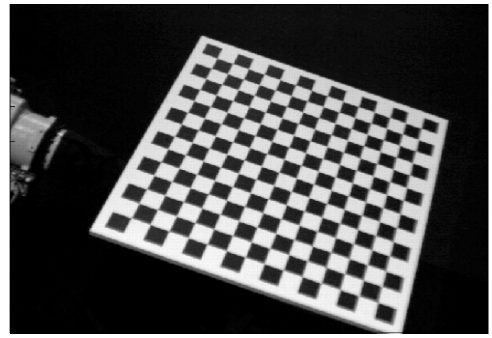

Figure 3 A calibration target used by the vision system

Classical approaches to camera calibration are fairly labour intensive. They require complex three dimensional targets, images are acquired at multiple camera/target positions, and often it is necessary to measure the relative position of the camera and target very accurately. Some algorithms can estimate the relative camera / target pose [13]. However, then the mathematical camera model may model only simple lens distortions, and large 3D targets are still required for calibration. Such calibration must be performed in ground based laboratories and this process cannot be repeated once the hardware is in orbit. This is a significant limitation as even small mis-calibration due to, e.g., thermal and aging effects reduce accuracy of any vision system. Our approach can be used to calibrate cameras on-orbit, as it only requires a planar object to be presented to the cameras in several arbitrary poses.

After the calibration the updated camera parameters (internal and external) are incorporated into the stereo system's model. In order to accelerate the on-line phase certain image and 3D transformations are computed off-line and stored as look up tables.

3.2 3D computation

The 3D computation module processes stereo images obtained from two calibrated cameras and computes 3D data. The process follows a typical processing sequence and it starts from warping the images, which corrects for geometrical lens distortions and rectifies the stereo geometry. Edges detected in the rectified images are matched and used to create a sparse 3D representation of the scene [6]. One of the stereo images of a mockup representing a satellite, which was used in experiments, is shown in Figure 4. Different views of the computed 3D representation are shown in Figure 5.

Figure 4 One of the stereo images

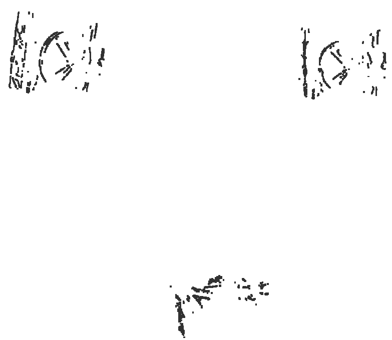

Figure 5 Computed 3d representation shown from different viewpoints

The 3D computation adapts to variable illumination and scene content, and uses only a minimum number of control parameters. In a typical scene our vision system computes approximately 1000 3D points.

3.3 Pose Estimation

To determine the position and orientation of the object, the 3D data points are matched to a known model using a version of the Iterative Closest Point algorithm (ICP) [1]. Our implementation of this algorithm consists of the following steps [6]:

1. Selection of the closest points between the model and the data set

2. Rejection of outliers

3. Computation of geometrical registration between matched data points and the model

4. Application of geometrical registration to the data

5. Termination of the iteration after stopping criterion is reached

The algorithm iterates the steps 1-4 until the stopping criterion (convergence and/or reaching the maximum allowed number of iterations) is met.

Detection of the closest points between the model and the data sets involves computing distances for each combination of data points and model features. This is a computationally intensive process and different model representations and acceleration schemes have been used [1, 15]. For models represented as points or meshes *k-d* trees are often used [12]. Man-made objects, which can be represented as collections of simple shapes, allow using efficient closed form solutions to compute the distances between points and model parts [6]. In the current version of our system we have implemented a very fast algorithm, which eliminates the need to perform any on-line distance calculations.

3.3.1 Pre-computed Distance Maps

As the computation of distances between data points and models is time consuming, we have eliminated it from the on-line phase and pre-computed all distances in advance off-line. This comes at a price of having to store the data and/or reduce the resolution of the distances.

The space surrounding the model is partitioned into voxels and the closest point between the centre of each voxel and the model is computed off-line. During the on-line operation, instead of computing the closest point to the model we simply need to lookup the distance based on the partition in which the data point lies. The data structure chosen to facilitate the pre-computed distance map is the octTree. An octTree is a tree structure where each node is either a leaf or has eight children. From any one point in 3D space one can move in the positive or negative direction for each dimension giving $2^3 = 8$ partitions. Since we have eight such partitions we refer to each one as an octant. Each octant can then be subdivided, with each sub-octant representing a region of space one eighth the size of its parent, see Figure 6.

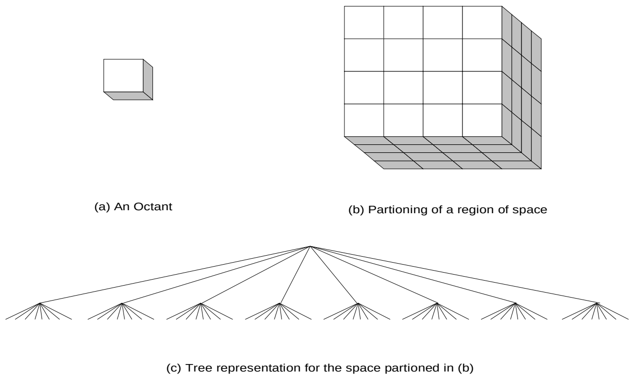

Figure 6 Octtree represnetation of a 3D space

The closest point to the model from each corner of the octant is stored at each node. When we wish to determine the closest point on the model from some arbitrary point **p** in space we traverse the tree and find the smallest octant containing **p**. Next we find the closest corner of the octant, **c,** to point **p** and use the closest point on the model from point **c** as an approximation of the closest point to **p**. This is a simple and very fast way of getting an approximation of the closest point to **p**. No calculation is required - we simply use the octTree as a lookup table.

Figure 7 shows a model of the object used in the experiments and Figure 8 its octTree representation at the level 2.

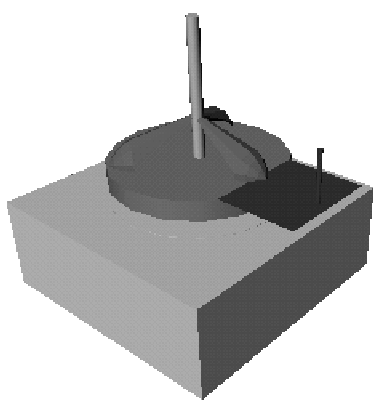

Figure 7 - Object Model

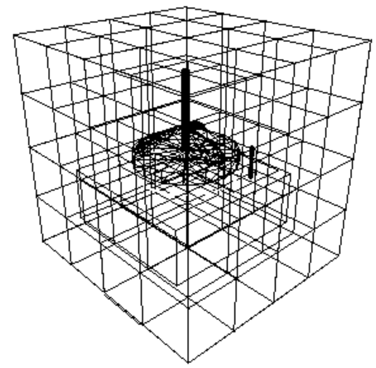

Figure 8 Level 2 OctTree Representation of the Model

In our experiments we have used level 7 as a good compromise between the size, accuracy and processing speed.

Experimental results show that looking up the closest point to the model using the octTree is over 1000 times faster than performing the calculation. The disadvantage to using octTrees is that all the pre-computed data needs to be stored. Some measures have been implemented to reduce the storage space required for octTrees. A trade-off between the size and resolution of octTrees is illustrated in Table 1 for an object within a space of 0.5 x 0.5 x 0.5 m.

Table 1 Accuracy and Size of OctTrees

Tree Level	**Size of Octant [**mm^3**]**	**Maximum Distance to Nearest Corner [mm]r**	**Model Size [MB]**
4	31	27.0	0.25
5	15	13.5	2
6	7.8	6.75	16
7	3.9	3.38	128
8	1.95	1.69	1024
9	0.976	0.85 m	8192

4 Experimental verification

Testing the performance of vision systems is becoming a separate sub-discipline in computer vision. Any system that is or will be installed as part an autonomous system must be fully characterized in representative environments and under expected operational scenarios before its deployment. The experiments conducted with our vision system attempted to determine the operational ranges where such a vision system can be used successfully. Such information allows one to use the system in a safe and predictable manner and to focus future development. The scope of the performed tests included measurements of:

- performance, accuracy and repeatability of the main software components of the vision system
- performance, accuracy and repeatability of the complete vision system for different scenarios
- effects of illumination on operation of the complete system

The testing has been performed using the Space Vision Testbed with two jointly calibrated robots [10]. The robot positions reported by their controllers were treated as accurate measurements of the relative pose of a mockup with respect to a stereo camera platform. Representative scaled mockups have been manufactured using real spacecraft insulation materials: a Spartan-like model of a satellite, see Figure 1 and Figure 5. The mockups were moved in space mimicking trajectories that might be executed during an actual space mission: direct and loop approach, fly-around.

The illumination was scaled down to 1% of the on-orbit level and provided by a computer controlled mobile spotlight with a subtense angle similar to the Sun and large screens simulating the Earth albedo. Several combinations of available light sources were used: direct Sun light (using one of two different spotlights), “Earth albedo” - directional diffuse light and diffuse ambient light. The spotlights were arranged in such a way to create hard shadows and local image saturation in some of the images.

5 Results

Testing system components involved testing the camera calibration accuracy, which was found to be a small fraction of a pixel. Testing of the 3D generation module allowed us to determine that it operates in stable fashion under variable light intensity (between 40%– 100%), direction and type, and observed objects and scene complexity. The system operated always successfully with 25% data loss and often with even 50%. Example images from experiment with a loop approach are shown in Figure 12. The right window displays one of the stereo images of a mockup. The left window displays a synthetic image obtained by rendering the mockup in the same pose and from the same direction as the camera. The numbers show the computed pose.

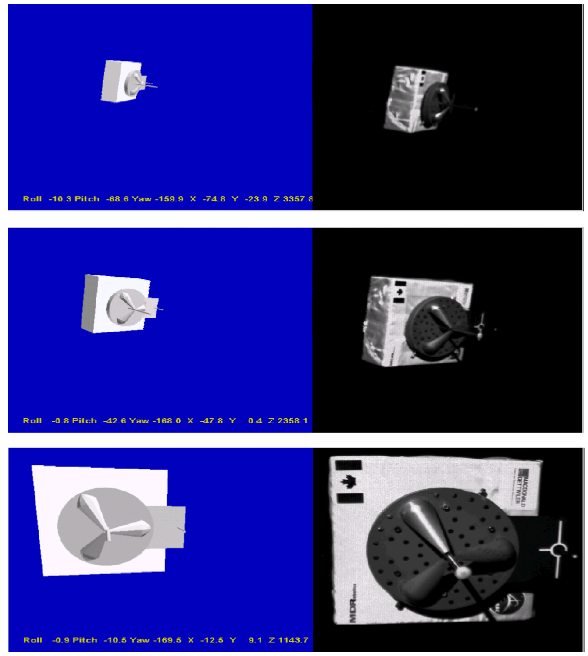

Figure 9 Several images from a test sequence

During this test the stereo camera platform representing the chaser satellite was approaching the mockup, which performed a rotation about Y (yaw) axis. The estimated distance to the mockup has been plotted as a function of the actual distance in Figure 10, and the estimated angle in Figure 11.

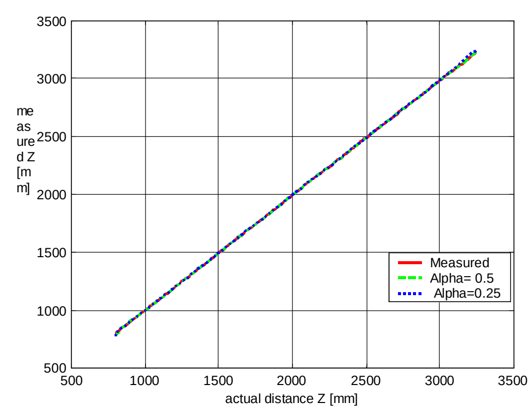

Figure 10 Estimated distance against the actual distance

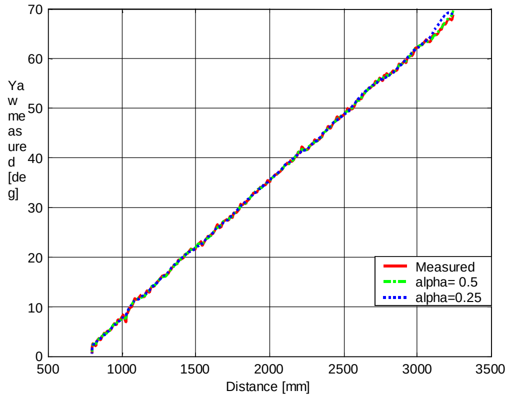

Figure 11 Estimated yaw angle as the function of the actual distance

Figure 12 shows images from another test; note the shadows cast on the observed object. The shadow edges are located on the object surface and they are actually beneficial as they provide additional 3D data (the data in the shadow is totally lost).

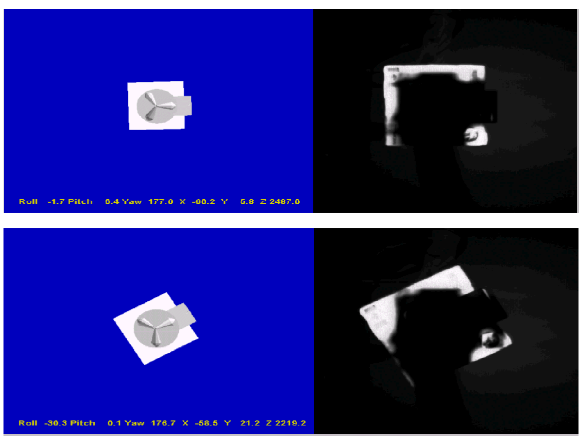

Figure 12 Actual and computed images from an experiment with shadows

The effective range of the tested stereo system (for the lens and cameras used) was found to be in the range of 5 – 20 stereo baselines. The maximum range was limited by the size of the object in the images and a small disparity, which was affecting the measurement accuracy and ability to compute the pose. The minimum range was limited by the fact that object exceeded the image size – we used only one model of the object.

Iterative algorithms such as ICP operate successfully when they are initialized within the convergence range of the algorithm [1]. Measuring this range requires a very large number of tests in order to test convergence from many initial conditions [5]. We have conducted exhaustive tests, in which we determined that the algorithm always converges in the range is of 15 – 30 degrees depending on the shape of the object. The high likelihood convergence range is larger than the guaranteed and it also depends on the object.

The processing time of our software is, on a x86 family 6 model 7 computer, in order of 1 second with most of the time spent computing 3D data.

6 Conclusions

This paper presents a design and the results of an experimental characterization of a vision system for satellite proximity operations. The system can track the 3D pose of a known object using only geometrical models and natural features detected on the object surfaces. As the system computes highly redundant data, partial data loss due to occlusion, shadows or local specular reflection does not affect the system operation.

The system was characterized using representative images obtained using a calibrated testbed. The images generated in experiments have been stored in a database and are available to collaborating research institutions for evaluation of their computer vision algorithms.

Pre-computing certain data off-line allows our system to rely on advanced and reliable algorithms and operate at 1 Hz on a desktop computer. The 3D computation, which uses most of the processing time, can be implemented in dedicated hardware off-loading the CPU and/or increasing the update rate.

Current work focuses on extending the vision system capabilities by incorporating additional modules that compute automatically the initial pose [7], implementing the short range module, and integrating the vision system in a closed loop control system.

The funding for this work has been provided in part by a contract from Canadian Space Agency to develop an Object Recognition and Pose Estimation Toolkit.

7 References

[1] Besl, P. & N. McKay, "A Method for Registration of 3-D Shapes", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 14, no. 2, February 1992, pp. 239-256.

[2] Blais F, Beraldin J, Cournoyer L.: "Integration of a Trackinbg Laser Range Camera with the Photogrammetry based Space Vision System". SPIE Aerosense, vol. 4025, April 2000.

[3] ETS-VII http://oss1.tksc.nasda.go.jp/ets-7/ets7_e/rvd/rvd_index_e.html

[4] Hollander S.,"Autonomous Space Robotics: Enabling Technologies for Advanced Space Platforms", Space 2000 Conference, AIAA 2000-5079.

[5] Hugli H., Schutz C: Geometric matching of 3D Objects: Assessing the Range of Successful Initial Conditions. Int. Conf. On Recent Advances in 3-D Digital Imaging and Modelling. 3-DIM.

[6] Jasiobedzki P, Talbot J, and Abraham M: Fast 3D Pose Estimation for On-Orbit Robotics. ISR 2000 Proceedings pp. 434-440

[7] Jasiobedzki P, Greenspan M, Roth G.: Pose Determination and Tracking for Autonomous Satellite Capture, I-SAIRAS 2001

[8] Laurin, Denis G.; Beraldin, J.-Angelo; Blais, Francois; Rioux, Marc; Cournoyer, Luc. "A Three-Dimensional Tracking and Imaging Laser Scanner for Space Operations", SPIE Vol. 3707, pp. 278-289.

[9] McCarthy J.: Space Vision System. ISR-1998, Birmingham, UK.

[10] Newhook P.: A Robotic Simulator for satellite Operations. I-SAIRAS 2001

[11] Oda M, Kine K, Yamagata F., "ETS-VII – a Rendezvous Docking and Space Robot Technology Experiment Satellite",46 Int. Astronautical Congress, Oct 1995, Norway, pp. 1-9.

[12] Simon D, Hebert M, Kanade T: Real-time 3D Pose Estimation Using a High Speed Range Sensor. IEEE Conference on Robotics and Automation, pp. 2235-2241

[13] Tsai RY A Versatile Camera Calibration Technique for High-Accuracy 3D Machine Vision Metrology Using Off-the-Shelf TV Cameras and Lenses. IEEE Journal of Robotics and Automation, v. RA-3, no.4 August, pp. 323-344.

[14] Zhang Z:. Flexible Camera Calibration By Viewing a Plane From Unknown Orientations. International Conference on Computer Vision. (ICCV'99), Corfu, Greece, pages 666-673, September 1999.

[15] Zhang Z: Iterative Point Matching for Registration of Free-Form Curves and Surfaces. IJCV, 13:2 119-152