

# Robust Uncalibrated Visual Servoing for Autonomous On-Orbit-Servicing

Azad Shademan, Amir-massoud Farahmand, Martin Jägersand

Robotics Research Group, Dept. of Computing Science, University of Alberta  
Edmonton, AB, CANADA

e-mail: a.shademan@ieee.org, amirf@ualberta.ca, jag@cs.ualberta.ca

## Abstract

Camera-type vision sensors are typically used in highly calibrated settings in On-Orbit-Servicing (OOS) missions. This is a limitation to autonomously service targets without specific markers. In addition, the harsh conditions of space demands a versatile algorithm that is robust to different types of visual errors. In this paper, we propose robust uncalibrated visual servoing scheme for OOS. Uncalibrated visual servoing is essentially free of both modeling and calibration, and works well in unstructured environment. The error is computed directly from image and is defined entirely in the visual space. Our approach is based on using a robust M-estimator to estimate the Jacobian matrix, which appears in the control law. We propose this algorithm to be used in the mating (capture) phase of a rendezvous mission. The proposed algorithm is evaluated in computer simulations as well as laboratory experiments. Our experimental setup includes a 7-DOF WAM arm equipped with a high speed camera. The laboratory experiment simulates capture of a grapple fixture using feedback from camera image.

## 1 Introduction

Autonomous On-Orbit Servicing (OOS) using robotic arms is a topic of great interest to the space community [1–4]. The majority of the OOS missions still rely on manual control. For example, the Space Shuttle’s robotic manipulator [5] is operated manually to dock free-flying objects [3]. Without autonomous OOS, manned missions with more complicated mission planning processes become necessary. Recent missions incorporate some concepts of autonomy in OOS. For example, one of the goals of DARPA’s Orbital Express Demonstration System (OEDS) was the autonomous capture of the free-flying target satellite by a robotic arm on a chaser satellite [3].

Feedback from vision sensors can be used for autonomous arm motions in OOS. One of the challenges in vision-based autonomous OOS missions comes from uncertainties in vision sensor output. Common vision sensors are video cameras [6] and laser cameras such as Neptec’s Laser Camera System (LCS) [7]. The uncertain-

ties in sensor output are generally due to either unexpected situations, *e.g.*, occlusions [4], or failure of visual tracking module due to mis-tracking, field-of-view constraint, etc. [8].

We have recently developed a robust algorithm for vision-based motion control of arms [8], which is typically studied as visual servoing in the robotics community [9, 10]. Our uncalibrated visual servoing algorithm is robust to uncertainties in sensor input and has been already evaluated with a Barret WAM arm with an eye-in-hand configuration successfully [8]. The results are promising to further the applications of supervised machine learning in unstructured space robotics. In this paper, we propose a vision-based autonomous capture technology for OOS built on uncalibrated visual servoing. Our algorithm is statistically robust to various types of outliers, such as lost features, mis-tracked features, occluded features, etc.

The advantages of the proposed technology are three-fold:

1. It does not require any calibration, neither calibration of robot kinematics, nor calibration of the vision sensor;
2. It does not depend on a specific sensor and can be used with either video cameras or laser cameras; in addition, it is general to any target and any robotic arm;
3. It automatically rejects sensor outliers and is robust to many unavoidable sources of error, including visual occlusion, visual mis-tracking, etc.

The remainder of this paper is organized as follows. In Section 2, we briefly mention the typical OOS procedure to put our method into context. The overview and problem formulation of the robust uncalibrated visual servoing is presented next in Section 3. The OOS simulation experiments and laboratory experiments with a Barrett WAM arm is presented in Section 4. Finally, we discuss the technology and conclude with some future directions in Section 5.

## 2 On-Orbit Servicing (OOS) Background

In this section, we provide a very brief background on a typical rendezvous mission. We hope that this section brings the proposed study into context. Details of rendezvous missions is beyond the scope of this paper. For a detailed discussion of the concepts in this section, please see [11].

A rendezvous mission is a very complex process, in which a chaser vehicle connects to a target spacecraft. The target can be either passive or active. There are many different phases in a typical rendezvous mission: launch and orbit injection, transfer to near-target orbit, far-range rendezvous, close-range rendezvous, and mating [11]. Most of these involve trajectory generation for the chaser vehicle, but not the robotic arm. *The only phase that involves the arm is the mating phase.*

The mating phase might refer to either *docking* or *berthing*. Docking is the process, where the chaser vehicle is actively controlled to align its capture interface with that of the target vehicle. In berthing, the robotic arm has a nominal role. Berthing is the process, where the chaser vehicle is controlled to a constant relative position (with respect to the target vehicle), followed by the robotic arm capture. The arm could be stationed on either the chaser vehicle or the target vehicle. The grapple fixture could be placed on the other vehicle. The robotic arm capture is an important phase in a servicing (berthing) mission. “Capture” is not exactly the final step; it is followed by some structural and utility connection tasks [11]. These tasks do not involve motion control of the arm. For the purpose of the current study, we assume that the chaser and target vehicles are already in their reception ranges and mating by capture is possible. Hence, we only consider the motion control of an arm to a capture position<sup>1</sup>.

Sensors are used to provide feedback during motion control of the arm to achieve higher performance and robustness to uncertainties. As already explained in the Introduction, vision sensors are good candidates for autonomous vision-based OOS. Namely, laser camera systems and CCD cameras are both viable options with their own advantages and disadvantages. An essential advantage of CCD cameras is their rich information content (largely ignored in the space robotics community). While we make no explicit assumption about the sensor type, and our methods could be extended to any consistent sensor, our presentation is primarily with a CCD camera and the pinhole camera model.

A typical target pattern for a camera rendezvous sensor includes five reflectors (markers) [11]: four coplanar reflectors (markers) evenly spaced on a circle, and a

fifth reflector directly above the circle center. The focal length and field-of-view (FOV) angle of the camera is assumed known (calibrated camera). In these settings, the relative orientation and position of the target from the camera can be found from the image of the reflectors. The size and position of the target pattern is calculated to satisfy the FOV and occlusions constraints. The problem with this kind of patterns is that multiple concentric patterns with different sizes are needed for visibility during far-range to close-range transitions. Another problem with reflector markers is that the light reflected from the Sun or other light sources could saturate the image and cause visual tracking errors. In addition, such features are not available on many satellites that are already in orbit and need servicing. Xu *et al.* [12] have proposed a control and planning algorithm for autonomous rendezvous of such *non-cooperative* targets. They consider model-based 3D reconstruction and pose calculation from stereo cameras with the typical calibration assumptions (known camera/robot and camera/vehicle transformation, known 3D geometric model of the target, etc.).

Our proposed algorithm entertains a very relaxed set of assumptions. Namely, we do not require camera calibration or calibration of camera pose with respect to the vehicle body. In case of visual tracking errors, our statistically robust algorithm rejects the outliers. Finally, we do not make structural assumptions on the configuration of reflectors or any other visible marker on the vehicle. The only assumption is that there must be at least four coplanar markers, no three of which are collinear. In the next section, we present our robust uncalibrated visual servoing approach.

## 3 Robust Uncalibrated Visual Servoing

### 3.1 Overview

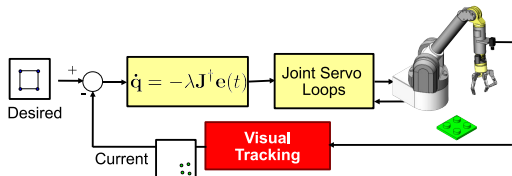
Visual servoing concerns closed-loop feedback of visual information to control a robot to a desired configuration, while minimizing an error norm in the image space [9]. There are two main classes for visual servoing control laws: image-based [13] and position-based [14]. For a complete survey of basic and advanced methods, please see [9, 10].

In Image-Based Visual Servoing (IBVS), the control law and the feedback are purely in the visual space. A typical control law for IBVS uses the Jacobian matrix  $\mathbf{J}$ . The Jacobian matrix relates the rate of change in visual features to either the velocity screw of the camera [13] or directly to the joint velocities [15]. In the case of direct mapping from joint velocities to visual feature velocities, the Jacobian matrix is called the *visual-motor Jacobian*. The control law is

$$\dot{q} = -\lambda \mathbf{J}^\dagger (s - s^*), \quad (1)$$

<sup>1</sup>The methods proposed in Section 3 are not specific to robotic manipulators and could be, theoretically, extended to the nonlinear feedback control of the chaser vehicle in docking.

where  $\dot{q}$  is the vector of joint (motor) velocities,  $s$  and  $s^*$  are the vectors of the current and desired visual features, respectively, and  $\mathbf{J}$  is the so-called visual-motor Jacobian, and  $\mathbf{J}^\dagger$  is its Penrose-Moore pseudo-inverse:  $\mathbf{J}^\dagger = (\mathbf{J}^T \mathbf{J})^{-1} \mathbf{J}^T$ . The block diagram for a typical IBVS system is illustrated in Figure 1.



**Figure 1.** IBVS block diagram.

Existing IBVS methods are either *calibrated* in the sense that they require some knowledge of the camera/robot model (not suitable for unstructured environments) [9, 10], or are *uncalibrated*, but assume that no portion of the visual-motor data is corrupted (susceptible to outliers) [15, 16]. Unfortunately, these assumptions do not generally hold in real unstructured environments of space missions, where the analytic form of the Jacobian in (1) is not available and must be replaced by its numerical estimate,  $\widehat{\mathbf{J}}$ . For example, the uncalibrated Jacobian can be updated at each control iteration by Broyden's method [15]:

$$\widehat{\mathbf{J}}_{k+1} = \widehat{\mathbf{J}}_k + \alpha \frac{(\Delta s - \widehat{\mathbf{J}}_k \Delta q) \Delta q^T}{\Delta q^T \Delta q}, \quad (2)$$

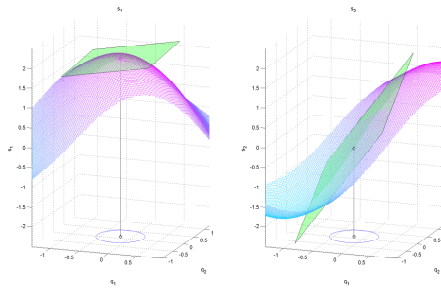
where  $\alpha$  is the update rate.

The uncalibrated Jacobian estimation becomes more challenging when there exists outliers (which is often the case in harsh unstructured settings). An erroneous Jacobian estimate can make the system unstable or drive the robot in the wrong direction. Therefore, an adaptive system working in an unstructured scene, should use a Jacobian estimator that is robust to outliers. This is in addition to the requirement of being independent from geometric structure, model, and calibration parameters.

### 3.2 Problem Formulation

We apply robust M-estimation [17] to estimate a robust Jacobian while rejecting the outliers. M-estimators generalize the idea of maximum likelihood to robust measures of scale and location. Outliers are caused by various visual tracking errors, such as occlusion or mis-tracking [8]. We directly use the visual-motor data to learn the visual-motor Jacobian and deal with the outliers at the same time.

The robust uncalibrated visual servoing problem is formulated as estimating the uncalibrated visual servoing



**Figure 2.** The Jacobian of a two-link bar with 2 features is a  $2 \times 2$  matrix which represents a hyperplane tangent to the visual-motor data. The hyperplane is represented by two planes for a query point.

Jacobian. Formally, for a given memory of visual-motor pairs and a new visual-motor query  $(s_c, q_c)$ , the Jacobian estimation problem can be formulated as the following minimization problem [8]:

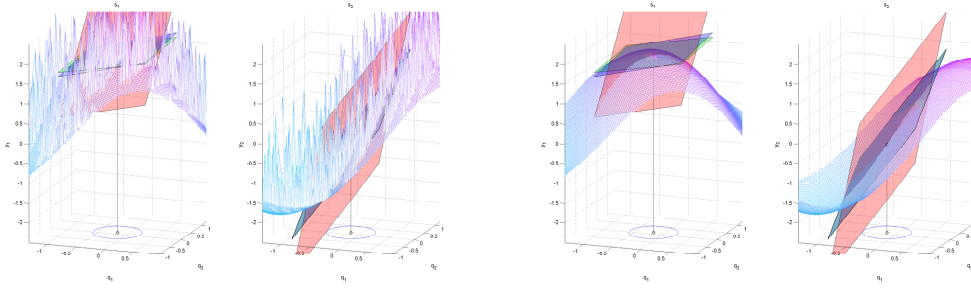
$$\widehat{\mathbf{J}}(q)|_{q=q_c} = \underset{\mathbf{J}}{\operatorname{argmin}} \sum_{k: q_k \in N(q_c)} \rho(\Delta s_k - \mathbf{J} \Delta q_k), \quad (3)$$

where  $N(q_c)$  is a neighborhood of  $q_c$  that contains joint-space neighbors,  $\Delta q_k$  are the neighboring joint differences:  $\Delta q_k = q_c - q_k$ , and  $\Delta s_k$  are the neighboring image feature differences:  $\Delta s_k = s_c - s_k$ . This method robustly fits the best hyperplane to the visual-motor data around  $q_c$ .

We use a robust M-estimator, such as  $\rho(e; \sigma) = \frac{e^2}{e^2 + \sigma^2}$ , to reject the outliers. Error  $e(\cdot)$  is measured in image-space and the measure of scale  $\sigma$  quantifies how the probability distribution is spread. For example, variance is a measure of scale for the normal distribution, but it is not robust to outliers. We need a robust measures of scale in addition to a robust M-estimator. The Median Absolute Deviation (MAD) is a computationally efficient robust measure of scale and appropriate for real-time applications. In essence, MAD estimates the variance of the inlier samples. The minimization problem in (3) is solved efficiently using the well-known Iteratively Reweighted Least Squares (IRLS) algorithm. Details can be found in [8] and is not repeated here.

### 3.3 Validation and Illustration of Robust Jacobian Estimation

Figure 2 illustrates the visual-motor manifold and the tangent Jacobian hyperplane at an arbitrary point for a planar robot with 2 degrees-of-freedom ( $q_1$  and  $q_2$ ) and 2 visual features ( $s_1$  and  $s_2$ ) without any outliers. Figure 3 (left) shows the same manifold in Figure 2, which is now



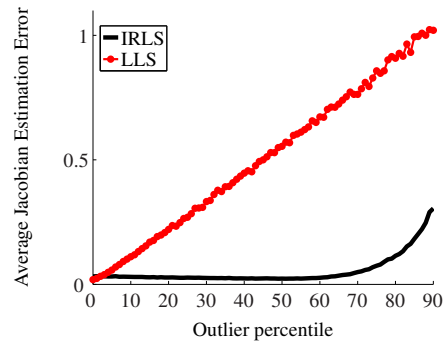
**Figure 3.** (Left) Visual-motor outliers due to numerous errors necessitate robust Jacobian estimation. (Right) The learned Jacobian hyperplane from least-squares (red) and robust (blue) are overlaid on the ground truth planes (green). For clarity, the outliers are not shown in the right image. The robust Jacobian is very close to the ground truth. *Figure best seen in color.*

corrupted by outliers. Least squares-based Jacobian estimation methods [16] work well when there are no outliers. However, they are significantly influenced by outliers and result in a wrong Jacobian estimate. In Figure 3 (right), the Jacobian hyperplane by least-squares regression is shown by red (light) planes and the proposed robust method in (3) is represented by blue (dark) planes. The robust Jacobian is very close to the ground truth (green planes). To verify the performance of the robust algorithm, we analyzed the Jacobian estimation error with both the least-squares based method (LLS) from [16] and the robust IRLS algorithm [8]. In controlled simulation experiments, we started with no outliers and grew the outlier-to-inlier percentile to 50% and beyond to a maximum of 90% (*i.e.*, only 10% of the visual-motor data were not corrupted). The normalized average error is depicted in Figure 4. When there are no outliers, both methods are similar and produce a small estimation error. As the outlier-to-inlier percentile grows, the LLS method is affected linearly. However, the IRLS method is robust to outliers and can handle outliers even when half (or more) of the data are outliers. In simulation experiments, up to 60% outliers could be tolerated. In practice, around 40% outliers could be tolerated. See [8] for more details.

## 4 Experimental Results

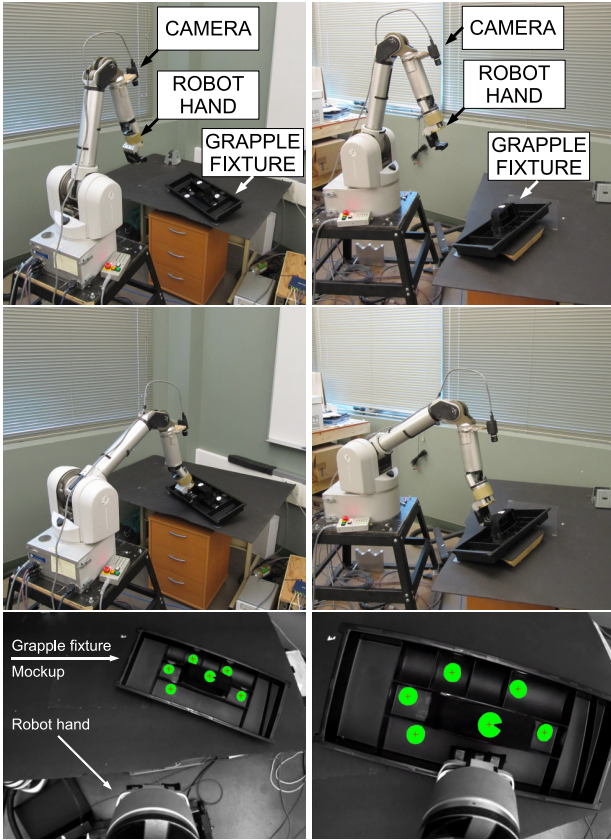
An uncalibrated eye-in-hand IBVS system with 4-DOF WAM is considered for visual servoing. Simulations are implemented in MATLAB using the Robotics Toolbox [18] and the Epipolar Geometry Toolbox [19]. In the MATLAB simulations, we use the coordinates of four markers as 8 features, *i.e.*,  $\hat{\mathbf{J}} \in \mathbb{R}^{8 \times 4}$ .

For robot experiments, we use a 7-DOF WAM arm that runs on RTAI-Linux and controlled with openman library [20]. The first 4 DOFs are controlled to position the



**Figure 4.** Simulation results of Jacobian estimation error. The error is found with respect to a reference Jacobian. Normalized average estimation error is reported for the least-squares based estimation (LLS) from [16] and the robust IRLS estimation from [8].

wrist in the desired capture configuration. The remaining 3 DOFs control the wrist and a Barrett Hand mounted on the end-effector of the 7-DOF Barrett WAM arm. The vision system consists of a Point Grey Grasshopper camera that captures  $640 \times 480$  MONO8 images at 60 Hz. The camera is mounted on link 5 of the WAM arm (after the elbow) overlooking the Barrett Hand and the workspace. For real-time visual tracking, we use the Visual Servoing Platform (ViSP) [21]. In the robot experiments, we use a mockup of a grapple fixture with the coordinates of five arbitrary markers (total of 10 image features). If visual tracker loses one of the markers, 4-DOF visual servoing continues with the remaining 4 markers. Therefore,  $\hat{\mathbf{J}} \in \mathbb{R}^{10 \times 4}$  if all markers are being tracked, or  $\hat{\mathbf{J}} \in \mathbb{R}^{8 \times 4}$



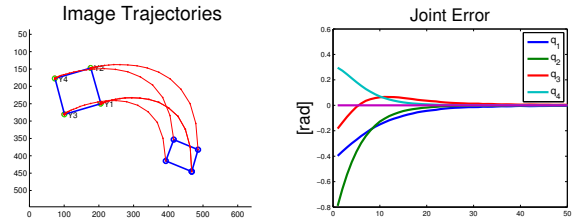
**Figure 5.** Experimental Setup. Top row shows the initial state. Middle row shows the capture-ready (desired) state. The bottom row shows the initial and desired images from left to right, respectively.

if one of the markers is lost. The grasping capabilities of the Barrett Hand is not used in this work, because the main focus of this paper is on visual servoing to the desired grasping point. We use a fixed configuration of the Barrett Hand. Figure 5 shows the experimental setup.

#### 4.1 OOS Simulation Experiments

The simulation experiments considers the kinematic model of the arm with the model of an object during the capture (mating) phase of an OOS mission. Our capture procedure is divided into a preparatory phase and a capture phase. In simulations, we have used visual servoing for the preparatory phase.

First, the arm is driven to a reliable tracking distance from the target. This is performed by the first-stage visual servo. The end-effector of the robotic arm approaches the target by minimizing the error in the visual space. The result of this experiment is depicted in Figure 6. Figure 6



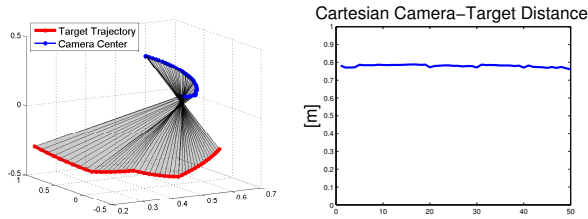
**Figure 6.** First-stage visual servo. The image trajectories and joint errors during IBVS. (Left) The trajectory is overlaid on top of the initial and desired images of the target. The target consists of 4 coplanar points. (Right) The joint error values in [rad] are shown during visual servoing.

(left) shows the desired image of the object (four points), the initial image of the objects and the image trajectory from the IBVS control (see Section 3). A Gaussian noise with standard deviation of 1 pixel is considered. Figure 6 (right) shows the joint error for the 4 DOF WAM arm. All joints converge to the desired value with a very small error after 50 iterations. A robust Jacobian is used in this experiment. As Figure 6 suggests, this approach works well to drive the robot arm into a reliable-tracking region.

Active targets need more preparations prior to the final capture. This includes target tracking and arm following. When the target is moving, the Jacobian is also constantly changing. Instead of estimating it at every iteration (this requires new visual motor memory at the new state), we use the Broyden update rule in (2) to update the estimate efficiently. The faster the object moves, the higher the servo loop frequency should be. In this stage, the visual servo works to regulate the relative distance of the arm end-effector to the target. This update compensates for variations in the Jacobian matrix elements due to target motion. Figure 7 (left) shows the target trajectory in bottom and the visually-servoed camera trajectory on top. The relative distance between the two remains at a constant value. This is shown by a 2D surface in Figure 7 (left). The numerical values are shown on a graph in Figure 7 (right).

#### 4.2 Laboratory Experiments

The aim of the laboratory experiments is to show the method in practice. The top row in Figure 5 shows the robot and the camera at their initial state. The middle row shows the setup at the desired capture-ready state. The bottom row shows the initial and final camera images. The visual servoing is entirely based on errors from the

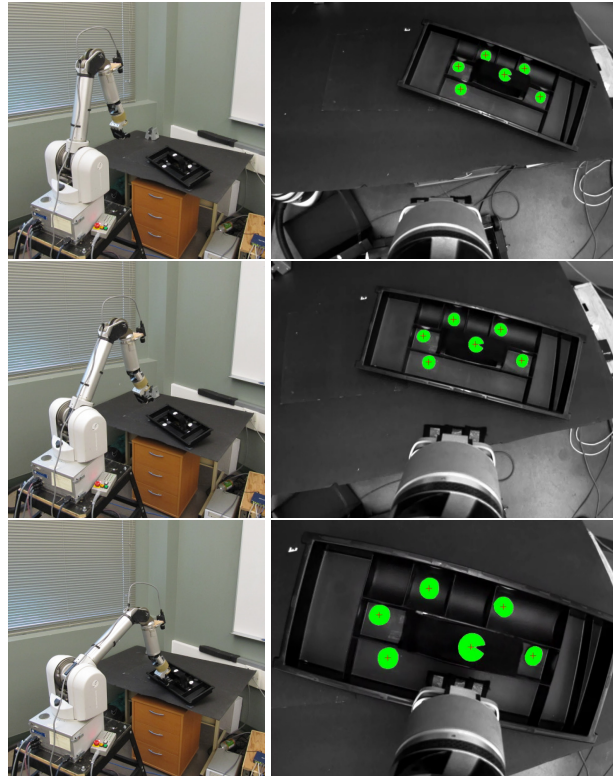


**Figure 7.** Target trajectory and camera trajectory during the tracking phase (second-stage visual servoing). The relative distance between the target and the camera remains constant at around 80 cm. The visual servo loop uses the uncalibrated Jacobian in (1), which is updated at each iteration by the Broyden formula (2).

visual space. The markers have been chosen arbitrarily. This is in contrast to the current practiced model-based approaches. We follow the same steps in the simulations: a preparatory phase followed by a capture. The visual servoing snapshots from the initial state to the capture-ready state is shown in Figure 8. Once the robot is in the capture-ready mode, the capture command can be issued. We have mainly focused on the visual servoing experiment. The grasping capabilities of the Barrett Hand have not been utilized in this work. The capture command is a simple motion of the wrist 3 DOFs, which is recorded at the capture-read state (teach-by-showing). Figure 9 depicts the robot pose and the camera image before and after capture.

## 5 Concluding Remarks

We propose a robust vision-based technology for autonomous On-Orbit-Servicing missions. Our proposed technology is not specific to a platform, does not require calibration, and is robust to outliers in the visual sensor. Our experimental setup is related to the setup at the Canadian Space Agency (CSA) [4], where one arm tracks and captures a satellite mockup attached to a second arm. They use adaptive Kalman Filter for motion estimation and pose tracking of a satellite. When occlusions occur, their system estimates the pose from the previous motions and the free-falling dynamics properties. Our proposed technology does not explicitly calculate the pose, but tracks object motion by regulating the visual error at zero. Additionally, our work compares to OEDS. The robotic arm of OEDS is mounted on the chaser satellite and model-based visual servoing is used to track and capture the target satellite [3]. However, our approach is

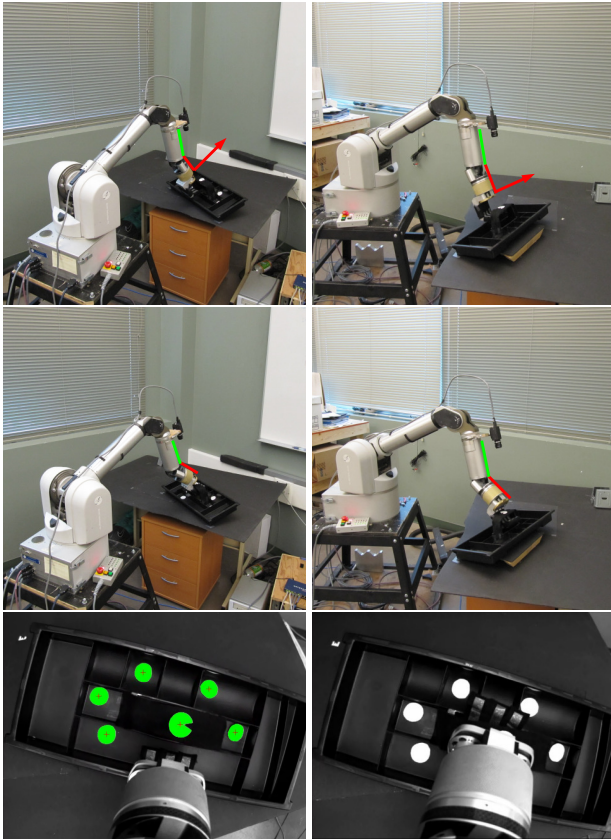


**Figure 8.** Visual Servoing from the initial point to the capture-ready (desired point).

purely image-based and completely model-free. In addition, the visual servoing in [3] is based on pose estimates from the vision sensor, which is found from the complete 3D CAD model of the target satellite and perfect calibration of the vision sensor and the robot. Another related project is the VIMANCO ESA project [6], where image-based visual servoing is used to improve autonomy and robustness for vision-based control of Eurobot. Although, their method does not require to place markers on the targets, it does require the 3D CAD model of the observed object. Our future work includes using other types of features and experiments with vision-based grasping of the Barrett Hand.

## Acknowledgment

This work is supported in part by NSERC, CFI, iCORE, and the Alberta Advanced Education & Technology. A. Farahmand acknowledges the support of AICML.



**Figure 9.** Capture command can be issued once visual servoing controls the arm to the capture point. The top two rows show the two views of the robot, before and after capture command. Arrows are overlaid for visualization purpose. The view from the camera is shown in the bottom row.

## References

- [1] K. Yoshida, "Engineering test satellite vii flight experiments for space robot dynamics and control," *Int. J. Robot. Res.*, vol. 22, no. 5, pp. 321–335, 2003.
- [2] S. Moosavian and E. Papadopoulos, "Free-flying robots in space: An overview of dynamics, modeling, planning and control," *Robotica*, vol. 25, pp. 537–547, 2007.
- [3] A. Ogilvie, J. Allport, M. Hannah, and J. Lymer, "Autonomous satellite servicing using the orbital express demonstration manipulator system," in *Proc. Int. Symp. Artificial Intell. Robot. Automat. in Space (i-SAIRAS)*, Hollywood, USA, 2008.
- [4] F. Aghili, K. Parsa, and E. Martin, "Robotic docking of a free-falling space object with occluded visual condition," in *Proc. Int. Symp. Artificial Intell. Robot. Automat. in Space (i-SAIRAS)*, Hollywood, USA, 2008.
- [5] D. King and C. Ower, "Orbital robotics evolution for new exploration enterprise," in *Proc. Int. Symp. Artificial Intell. Robot. Automat. in Space (i-SAIRAS)*, Munich, Germany, 2005.
- [6] K. Kapellos, F. Chaumette, M. Vergauwen, A. Rusconi, and L. Joudrier, "Vision-based control for space applications," in *Proc. Int. Symp. Artificial Intell. Robot. Automat. in Space (i-SAIRAS)*, Hollywood, USA, 2008.
- [7] C. Samson, C. English, A. Deslauriers, I. Christie, F. Blais, and F. Ferrie, "Neptec 3D laser camera system: From space mision STS-105 to terrestrial applications," *Canadian Aeronautics and Space Journal*, 2004.
- [8] A. Shademan, A.-M. Farahmand, and M. Jägersand, "Robust jacobian estimation for uncalibrated visual servoing," in *Proc. IEEE Int. Conf. Robot. Automat.*, Anchorage, AK, May 2010, pp. 5564–5569.
- [9] F. Chaumette and S. Hutchinson, "Visual servo control. part I: Basic approaches," *IEEE Robot. Automat. Mag.*, vol. 13, no. 4, pp. 82–90, Dec. 2006.
- [10] —, "Visual servo control. part II: Advanced approaches [tutorial]," *IEEE Robot. Automat. Mag.*, vol. 14, no. 1, pp. 109–118, Mar. 2007.
- [11] W. Fehse, *Automated rendezvous and docking of spacecraft*. Cambridge University Press, 2003.
- [12] W. Xu, B. Liang, C. Li, and Y. Xu, "Autonomous rendezvous and robotic capturing of non-cooperative target in space," *Robotica*, pp. 1–14, 2009.
- [13] B. Espiau, F. Chaumette, and P. Rives, "A new approach to visual servoing in robotics," *IEEE Trans. Robot. Automat.*, vol. 8, no. 3, pp. 313–326, Jun. 1992.
- [14] W. J. Wilson, C. C. W. Hulls, and G. S. Bell, "Relative end-effector control using cartesian position based visual servoing," *IEEE Trans. Robot. Automat.*, vol. 12, no. 5, pp. 684–696, October 1996.
- [15] M. Jägersand, O. Fuentes, and R. Nelson, "Experimental evaluation of uncalibrated visual servoing for precision manipulation," in *Proc. IEEE Int. Conf. Robot. Automat.*, vol. 4, April 1997, pp. 2874–2880.

- [16] A. M. Farahmand, A. Shademan, and M. Jägersand, “Global visual-motor estimation for uncalibrated visual servoing,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2007, pp. 1969–1974.
- [17] P. J. Huber, *Robust Statistics*, ser. Wiley Series in Probability and Mathematical Statistics. New York, NY, USA: Wiley-IEEE, 1981.
- [18] P. Corke, “A robotics toolbox for MATLAB,” *IEEE Robot. Automat. Mag.*, vol. 3, no. 1, pp. 24–32, Mar. 1996.
- [19] G. Mariottini and D. Prattichizzo, “EGT: a toolbox for multiple view geometry and visual servoing,” *IEEE Robot. Automat. Mag.*, vol. 3, no. 12, December 2005.
- [20] S. Leonard. openman: An open source C++ toolbox for control and simulations of manipulators. [Online]. Available: <http://sourceforge.net/projects/openman/>
- [21] E. Marchand, F. Spindler, and F. Chaumette, “ViSP for visual servoing: a generic software platform with a wide class of robot control skills,” *IEEE Robot. Automat. Mag.*, vol. 12, no. 4, pp. 40–52, Dec. 2005.