

Vision-Guided Self-Alignment and Manipulation in a Walking Robot

Kevin Nickels

Department of Engineering Science / Jet Propulsion Laboratory
Trinity University / California Institute of Technology
San Antonio, TX, USA / Pasadena, CA, USA
Email: knickels@enr.trinity.edu

**Brett Kennedy, Hrand Aghazarian,
Curtis Collins, Mike Garrett,
Avi Okon, Julie Townsend**
Jet Propulsion Laboratory
California Institute of Technology
Pasadena, CA, USA

Abstract — *One of the robots under development at the NASAs Jet Propulsion Laboratory (JPL) is the Limbed Excursion Mechanical Utility Robot, or LEMUR. Several of the tasks slated for this robot require computer vision, as a system, to interface with the other systems in the robot, such as walking, body pose adjustment, and manipulation. This paper describes the vision algorithms used in several tasks, as well as the vision-guided manipulation algorithms developed to mitigate mismatches between the vision system and the limbs used for manipulation. Two system-level tasks will be described, one involving a two meter walk culminating in a bolt-fastening task and one involving a vision-guided alignment ending with the robot mating with a docking station.*

1 Introduction

The Mobility and Robotic Systems Section at NASAs Jet Propulsion Laboratory (JPL) is tasked with investigating and developing system concepts and prototypes for potential future missions. One of these system prototypes is the LEMUR class of walking robots. LEMUR (Limbed Excursion Mechanical Utility Robot) was developed to investigate the capabilities and trade-offs of a small (10-15 kg) limbed robot. These robots are under consideration for on-orbit assembly operations, planetary science, and spacecraft inspection and maintenance. There are three LEMUR robots, the original LEMUR [1], LEMUR-IIa [2], and LEMUR-IIb [3], [4]. This paper will focus on LEMUR-IIa, shown in Figure 1.

1.1 Hardware Architecture

A very complete system description of LEMUR-IIa is given in [2]. Briefly, LEMUR consists of a PC-104 based computer system with six limbs, each capable of mobility and/or manipulation. Several tools have been developed for LEMUR, including a flashlight, palm-camera, and nut-driver. Feet are attached to the limbs via an industrial quick-connect, enabling task-specific feet to be developed and interchanged. A pair of cameras on a circular track enables 360 degree stereo vision around the robot. The kinematics have been designed to allow walking in any direction without time-consuming rotations of the robot body.

1.2 R4SA Software Architecture

A description of the Robust Real-Time Reconfigurable Robotics Software Architecture (R4SA) is given in [5]. The systems relevant to this paper are briefly described here.

A walking module accepts bearing, step-size, and total distance. It computes and performs an alternating tripod gait to move the robot in a given direction as requested. LEMUR is a holonomic walking robot - it can move in any direction without rotating its body. This module uses pedometry (step-counting) to estimate robot position.

A computer vision module provides the locations of any circular fiducials found in the image(s), as well as the position and orientation of fiducial triads. Optionally, some forward kinematics are used to restrict the area in which to search for fiducials.

A vision-guided manipulation module provides a corrective mapping between camera and limb coordinate systems. This module allows manipulation targets to be specified with respect to fiducials. The manipulation is robust to the target and/or the robot changing position.

A limb-control module provides low-level control of limbs for manipulation or walking purposes. This module provides forward and inverse kinematics for each limb.

A body pose module provides adjustments of the body

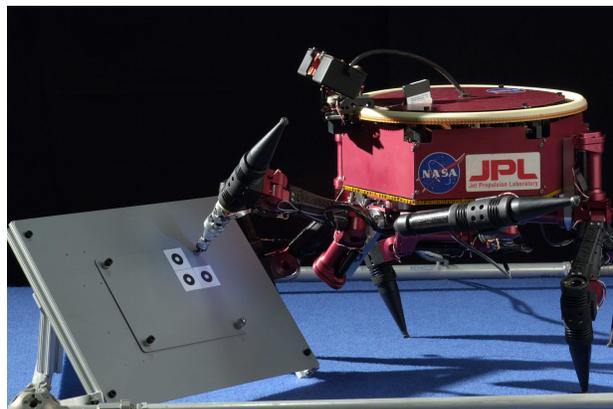


Figure 1. LEMUR-IIa Fastening



Figure 2. LEMUR Docking

position and orientation for a given foot placement. This is distinct from the walking module, which moves the legs to affect body position.

Each of these individual systems represents a well-developed and complex system that enables the creation of new functionality for the robot. To accomplish a task, most or all of these modules must communicate, creating a system of systems.

1.3 Task Descriptions

The first task described in this paper involves mating with a docking station. Zin Technologies, under contract to NASA, created the prototype docking station shown in Figure 2 for the use of LEMUR for an inspection and light maintenance task.

In order to mate with this docking station, three tooling balls have been placed underneath LEMUR as shown in Figure 3. In order to mate the robot with the docking station, all three balls need to be placed in the capture envelope of their respective parts of the docking station, requiring an accuracy of approximately 1 cm in displacement and 5 degrees in rotation. In this task, a fiducial triad is used to align the robot with the docking station prior to mating.

The second task described in this paper is a remote fastening task. LEMUR begins approximately two meters from a task board, and must approach the board by walking through the truss structure shown in Figure 4 and tighten a 5/16 inch acorn-head bolt. The accuracy required to place a socket on this bolt is approximately 4 mm.

Both of these tasks are representative of the current capabilities of this platform. Both tasks require the interaction of the vision, walking, and limb control systems.



Figure 3. Tooling Ball and Docking Fixture

2 Algorithm Descriptions

This section motivates and describes several algorithms that enable the accomplishment of the tasks described above.

2.1 Fiducial Detection and Processing

In autonomous operations, robust and repeatable detection of features is crucial. This constraint has led to a significant history at JPL of the use of fiducials, or artificial landmarks¹.

A circular fiducial finder that looks for a light-dark-light pattern in the horizontal, vertical and diagonal directions, then refines the ellipse center to sub-pixel accuracy, is used in these tasks. More description of this algorithm can be found in [7]. Any number of fiducial types (and associated fiducial finders) could be, and have, been used for these tasks, so long as they return the position of the fiducial in the image plane with sub-pixel accuracy.

It is often useful to be able to find the orientation as well as the position of a target. One method of doing this is to form a triad of fiducials, from which a local coordinate system can be derived. The following section describes the algorithm developed for doing this.

¹Indeed, there are several fiducials on Mars! [6]

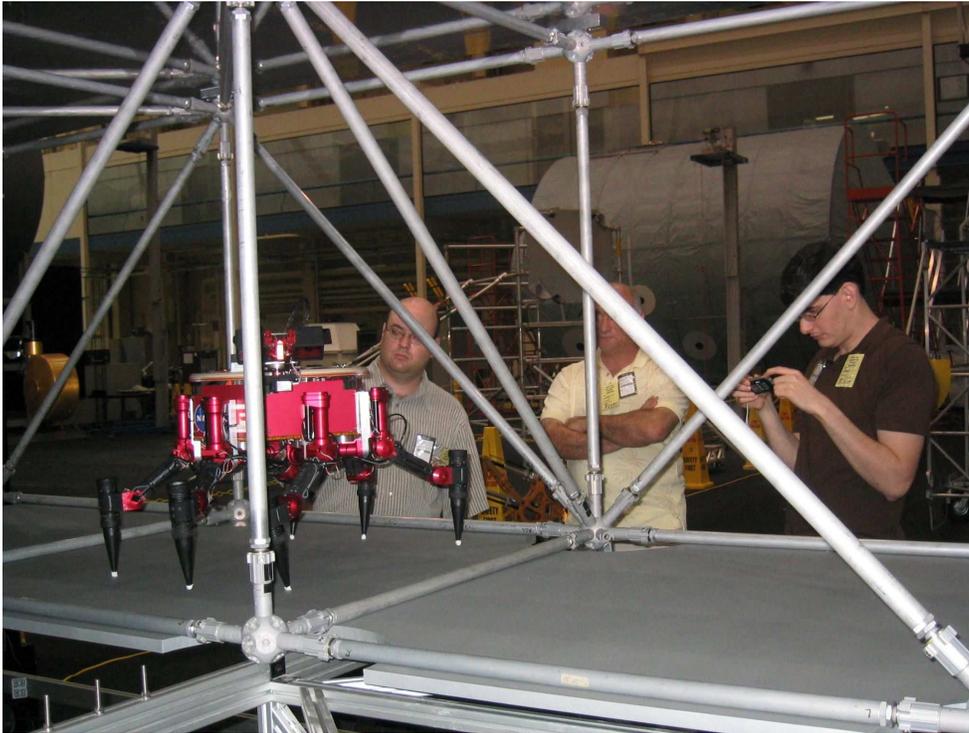


Figure 4. LEMUR Navigating Truss Structure

2.2 Robust Triad Detection

It is required that this subsystem recognize and detect triads of fiducials at arbitrary rotation in the image plane and significant rotation out of plane. Figure 7 shows an example image from the VGM task that illustrates these rotations.

In the first stage, heuristic filtering is used to ensure that only the three relevant fiducials are considered. In the docking task, prior knowledge of the approximate position of the robot and the triad location with respect to the docking station is used to reject false positives. If more than three fiducials are detected, clustering is performed to reject fiducials that are spatially distant from each other. Finally, when three fiducials are located that are an appropriate distance from their center of mass (COM), these are accepted as members of the triad. This type of task-specific interaction between the Triad module and the Fiducial Finder module is essential for a robust operational system.

The angles of the fiducials with respect to the COM (in the image plane) are computed, and the fiducials are sorted according to this. Fiducial one is defined to be the fiducial clockwise of the largest intra-fiducial angle. This ensures that the same physical fiducial is labeled one regardless of in-plane rotation. Fiducials two and three are labeled clockwise from one. This is done independently in the left and right images, then the three dimensional locations of the fiducials are computed by triangulation. Models for the individual cameras are calibrated according to the JPL

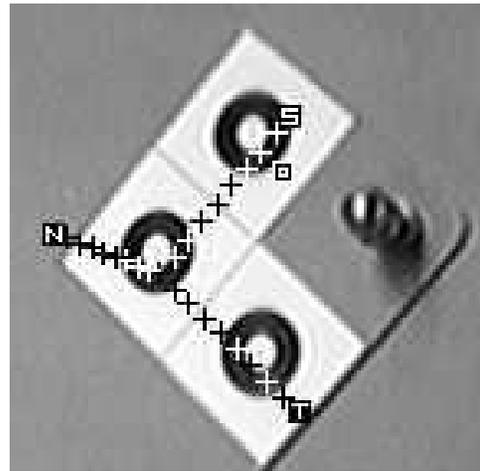


Figure 5. Fiducial Triad with TCS Marked

standard CAHVOR model [8].

Next, a local target coordinate system (TCS) is defined by the fiducials as follows. Fiducial two is taken to be the origin of the TCS. A unit vector from fiducial two to fiducial three is taken to be the s axis, and a unit vector from fiducial two to fiducial one is taken to be the t axis. The n axis, to form a right-handed coordinate system, is formed by $n = t \times s$. Figure 5 shows a sample triad with the TCS marked. All vectors (n , t , s) are expressed (initially) in the camera coordinate system, which is by convention placed at the optical center of the left camera.

2.3 Vision-Guided Manipulation

If the kinematics of the arm² were known perfectly, the camera system were perfectly calibrated, and the transform between the two systems known exactly, VGM could be performed fairly straightforwardly as follows. A target is specified in the TCS formed by a fiducial triad. The camera system locates the triad and computes the location and orientation of the target in the camera coordinate system (CCS). This is transformed into the arm coordinate system (ACS), and inverse kinematics are used to compute the desired target configuration of the arm. The arm is commanded to go to that configuration, and the tool exactly reaches the target.

Unfortunately, the ACS and CCS are not perfectly aligned, and indeed exhibit a nonlinear warping between them. Depending on the particulars of the target robotic system, this mismatch can be caused by errors in the fixed kinematic parameters of the arm, the calibration of the arm, the transform between the arm base and camera location, the calibration of the cameras, or any combination of these problems. Typical values of misalignment on the Lemur platform translate to 1-2cm of positioning error in the socket, if no correction algorithm is used.

The Hybrid Image-Plane Stereo (HIPS) algorithm was developed at JPL as a method for mitigating these types of errors. In HIPS, a fiducial is attached to the end of the arm, as close as possible to the tool. The arm is then moved to several poses (typically about 20) where the fiducial is visible in the cameras, and its location in the CCS and the ACS is recorded. A representative image is shown in Figure 6. This procedure is referred to as a “preplan step.” These data enable the computation of a mapping between the CCS (where the fiducial locations were measured) and the ACS (where the forward kinematics of the arm predict the fiducial location) to be generated. HIPS generates this mapping by taking the metric camera models, generated by calibration of the camera system [8], and warping them to best fit the data gathered. A complete description of HIPS can be found in [9]. Other representations for this ACS-CCS mapping have also been used[10].

Once the HIPS camera models are computed, VGM proceeds as above, with the exception that the HIPS models are used for the mapping from the image plane locations of the fiducials (and hence the TCS) to the CCS and finally to the ACS. This interaction between the camera calibration module, the arm control module, and the vision module illustrates the increasing complexity of operational robotic systems and the need for well-defined interfaces and robust unit testing.

²By “arm” we are referring to the limb where a tool is attached. Some tools can even be walked on, so an orientation with respect to the quick-connect is not implied. It is simply much shorter than referring to “the limb used for manipulation, in the posture where the tool happens to be attached”.



Figure 6. Preplan Image. Detected fiducial location is marked by +.

2.4 Vision-Guided Alignment

When walking, particularly on slick surfaces such as burnished aluminum, LEMUR experiences significant foot slip. Room Temperature Vulcanizing (RTV) silicone is used to mitigate this slip, but it is still sufficient to preclude open-loop docking or bolt-fastening. Typical values for the Bolt Fastening task are approximately 8-10cm of positioning error and 10 degrees of orientation error. Therefore a fiducial triad is placed in a known position with respect to the target, and visually guided alignment (VGA) is used to position the robot with respect to the triad. In the R4SA architecture, all application-layer algorithms are finite state machines (FSMs).

In each alignment module, the same general procedure is followed. The triad is located in the CCS, which is mapped to the Robot Coordinate System (RCS). The alignment of the RCS with the task object is computed, and checked against a desired alignment. This check is done in two phases (cross-path and along-path, in the bolt fastening task, and orientation and translation in the docking task). If the first alignment dimension is within tolerance, the second is checked. If the second is also within tolerance, the FSM terminates. Any out of tolerance alignment results in a correction followed by an FSM restart, ensuring that when the FSM terminates the robot is aligned in each dimension.

3 Task Sequencing

In this section, we describe the set of algorithms used to perform the tasks outlined in Section 3. These sequences are encoded as a set of FSMs in the R4SA architecture.

3.1 Vision-Guided Docking

In this task, the robot is initially positioned near the docking station, at an approximately known position and

orientation. The approach to the docking station could also make use of a walking-adjusted self alignment as described below, but for this demonstration this was not required.

A fixed walk command is issued, to position the robot approximately over the docking station. This walk is terminated by pedometry. Next, vision-guided alignment is performed, as follows:

- The triad is located in the CCS.
- If the body rotation with respect to the triad is out of tolerance, a yaw adjustment is issued, and the FSM restarts.
- If the body position with respect to the triad is out of tolerance, a body position adjustment is issued, and the FSM restarts.
- The FSM terminates when both body position and rotation are within tolerance.

Note that each stage of the alignment is re-checked on every pass through the FSM, ensuring that when the FSM terminates the robot is aligned in each dimension. This process typically begins approximately 1-3 cm out of tolerance in position and 10-20 degrees out of tolerance in orientation, and iterates at most twice before convergence.

When the VGA module terminates successfully, the robot is properly positioned above the docking station, and the final docking maneuver is issued (a body position adjustment straight down).

3.2 Bolt Fastening

Before this sequence is run, a “preplan step” as described in Section 3 is performed. Large amounts of walking and (worse) power cycles, as are common during development, can lead to miscalibration of the joint angles of the arm, and poor correspondence between the ACS and CCS. To mitigate this issue, a preplan step is performed as close as feasible in time to the actual manipulation task.

The Bolt Fastening task described here is a portion of a joint “In Space Assembly” demonstration undertaken by the NASA’s Jet Propulsion Laboratory (JPL), NASA’s Johnson Space Center (JSC), and Carnegie Mellon University (CMU). In the full demo, undertaken to investigate robotic assembly with heterogeneous robotic teams, a large spider-like walking robot (developed by JSC) carries a panel and mates it with the outside of a truss structure. Then, JPL’s Lemur walks up and fastens a bolt to attach the panel to the truss. Finally, JSC’s tendril-like inspection robot inspects the fastened bolt to ensure success. CMU acts as the overall planner and sequencer of the multi-robot demo. The task described in this paper is Lemur’s part of this sequence.

The task begins with a sequence of walk commands that take the robot through the truss structure, over a cross-bar, and under a diagonal support bar, on the right hand side of the truss. In this demonstration, the initial position is specified *a-priori* and the final position is specified with

respect to the task board, which is bolted to the truss. This can be seen in Figure 4. This procedure is sequenced by pedometry. When the robot has walked into the final truss cube, a series of small cross-path walk commands are issued, moving the robot to the left. This set terminates when the fiducial triad (just to the right of the image in Figure 4) is first spotted. At this point, the VGA module is initiated. This module works very similarly to the one developed for docking:

- The triad is located in the CCS.
- If the cross-path³ robot position with respect to the triad is out of tolerance, a cross-path walk (a walk at an angle of heading $\pm 90^\circ$) command is issued. The requested step size is set to the total adjustment divided by three (so that the walk can be completed in three steps) if the total adjustment size is small enough, or a predetermined maximum otherwise. After the walk sequence completes, the FSM restarts.
- If the along-path robot position with respect to the triad is out of tolerance, a forward walk (at the heading angle) command is issued. The requested step size is set as above. After the walk sequence completes, the FSM restarts.
- If the body position is within tolerance in all dimensions, control is passed to the VGM module.

Note that each stage of the alignment is re-checked on every pass through the FSM, ensuring that when the FSM terminates the robot is aligned in each dimension. As mentioned above, walking on slick surfaces implies foot-slip, so this fault-tolerant iteration has been shown to be necessary.

The VGM module is responsible for placing the tool on the bolt head:

- The limb adjacent to the manipulation limb is moved forward to stabilize the robot.
- The manipulation limb is raised and moved into a safe “manipulation standoff position,” Henceforth we will refer to this limb as the arm. In this standoff position, the arm is clear of the region where we expect the target to appear. Figure 7 shows a representative image from this position.
- The target triad is imaged and the HIPS models are used to compute the target position in the CCS.
- The arm’s inverse kinematics are used to compute joint angles that correspond to a position a few cm along the outward normal of the bolt. The arm is moved to this position, then downward along the normal to mate with the bolt head.
- Control is then passed to the torque-terminated fastening module.

The torque-terminated fastening module (TTFM) takes as input the a normal vector acquired by the VGM module,

³Cross-path is defined as right/left with respect to the truss structure, and along-path is forward/backward. Since LEMUR is holonomic, the orientation with respect to the robot coordinate system is arbitrary.

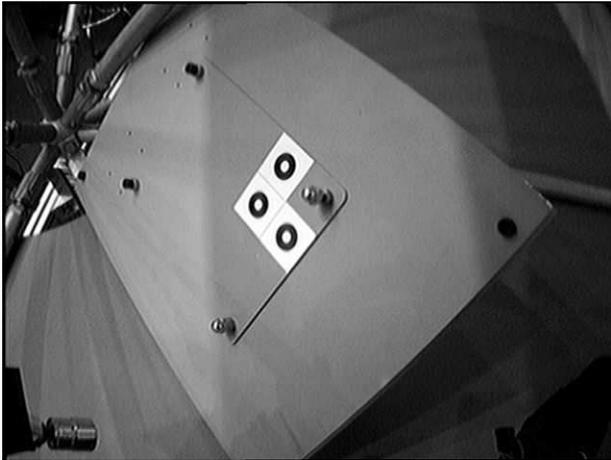


Figure 7. Manipulation Standoff Position. The tool is just visible in the lower left part of the image.

and operates as follows:

- A small voltage, sufficient to overcome sticktion but not quite initiate a movement of the socket in free-space, is applied to the driver tool.
- TTFM moves the tool along the normal to engage the bolt head, possibly turning the socket a small amount to align with the bolt.
- A voltage sufficient to slowly turn the drive tool and bolt is applied.
- The TTFM moves the tool along the bolt normal in synchronization with the turning of the socket, so that the relative position of the socket and bolt is (nominally) constant. The fastener pitch is specified a-priori.
- When a sharp increase in the drive current is observed, indicating that the bolt is tight, the TTFM terminates.

Finally, the arm moves to the manipulation standoff position to unseat the socket from the bolt, then back to standing position, and all limbs are re-posed to walking postures.

4 Conclusion

Several algorithms for visual guidance for robot positioning and manipulation have been developed at NASA's Jet Propulsion Laboratory. A simple method for robust processing of fiducial triads was described, as well as JPL's Hybrid Image-Plane Stereo method of correcting misalignments between camera and arm coordinate systems, support these algorithms.

The interactions between these modules have been described, and the data flow between them has been presented. While the executive control of these algorithms must change from task to task, a well-designed and robust interface allowed module testing and re-use between tasks.

This allows the capture of algorithms developed for specific tasks as well as the development of more complex robot behaviors.

In particular, these algorithms have been used to guide JPL's LEMUR-IIa robot in several tasks, including a bolt-tightening task following a 2 meter walk and an automated docking task following a walk.

Future work on the Lemur robot includes Cartesian-space control of the feet, integrating visual pedometry to dynamically detect and correct for foot slip during walks, and force control of the limbs. The HIPS algorithm will also need to be extended to handle the moving cameras, enabling manipulation from any angle.

Acknowledgment

The research described in this (publication or paper) was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

References

- [1] B. Kennedy, H. Agazarian, Y. Cheng, M. Garrett, G. Hickey, T. Huntsberger, L. Magnone, C. Mahoney, A. Meyer, and J. Knight, "Lemur: Limbed excursion mechanical utility rover," *Autonomous Robots*, vol. 11, pp. 201–205, 2001.
- [2] B. Kennedy, A. Okon, H. Aghazarian, M. Garrett, T. Huntsberger, L. Magnone, M. Robinson, , and J. Townsend, "The lemur ii-class robots for inspection and maintenance of orbital structures: A system description," in *Proc. 8th International Conference on Climbing and Walking Robots (CLAWAR'05)*, London, England, Sept. 2005.
- [3] B. Kennedy, A. Okon, H. Aghazarian, M. B. and X. Bao, Y. Bar-Cohen, Z. Chang, B. Dabiri, M. Garrett, L. Magnone, and S. Sher-rit, "Lemur iib: a robotic system for steep terrain access," in *Proc. 8th International Conference on Climbing and Walking Robots (CLAWAR'05)*, London, England, Sept. 2005.
- [4] T. Bretl, S. Lall, J. C. Latombe, and S. Rock, "Multi-step motion planning of free-climbing robots," in *Workshop on the Algorithmic Foundations of robotics (WAFR)*, Utrecht/Zeist, The Netherlands, July 2004.
- [5] B. Kennedy, H. Agazarian, Y. Cheng, M. Garrett, T. Huntsberger, L. Magnone, A. Okon, and M. Robinson, "Limbed excursion mechanical utility rover: Lemur ii," in *Proc. 53rd Annual International Astronautical Congress*, Houston TX, 2002.
- [6] J. N. M. et al, "Mars exploration rover engineering cameras," *Journal of Geophysical Research*, vol. 108, no. E12, p. 8071, Nov. 2003.
- [7] M. Seelinger, J. D. Yoder, E. Baumgartner, and S. Skaar, "High-precision visual control of mobile manipulators," *IEEE Trans. Robot. Automat.*, vol. 18, no. 6, pp. 957–964, 2002.
- [8] D. Gennery, "Least-squares camera calibration including lens distortion and automatic edition of calibration points," in *Calibration and Orientation of Cameras in Computer Vision*, A. Grun and T. Huang, Eds. Springer-Verlag, 2001, pp. 123–136.
- [9] E. T. Baumgartner, P. C. Leger, P. S. Schenker, and T. L. Huntsberger, "Sensor-fused navigation and manipulation from a planetary rover," in *Proc. SPIE Sym. on Sensor Fusion and Decentralized Cntrl. in Rob. Sys.*, Boston, Nov. 1998.
- [10] M. Bajracharya, M. DiCicco, and P. Backes, "Vision-based end-effector position error compensation," in *Proc. IEEE Aerospace Conference*, Big Sky, MT, Mar. 2006.