Autonomous Robotic Inspection for Lunar Surface Operations

Maria Bualat[1], Laurence Edwards[1], Terrence Fong[1], Michael Broxton[2], Lorenzo Flueckiger[2], Susan Y. Lee[3], Eric Park[2], Vinh To[3], Hans Utz[4], Vandi Verma[3], Clayton Kunz[2] and Matt MacMahon[5]

[1] NASA Ames Research Center, MS269-3, Moffett Field, CA 94035
Maria.G.Bualat@nasa.gov, Laurence.J.Edwards@nasa.gov, Terry.Fong@nasa.gov
[2] Carnegie Mellon University West Michael.J.Broxton@nasa.gov, Lorenzo.J.Flueckiger@nasa.gov, ckunz@mail.arc.nasa.gov, epark@mail.arc.nasa.gov
[3] Research Institute for Advanced Computer Science hutz@mail.arc.nasa.gov
[4] Perot Systems Corporation sylee@mail.arc.nasa.gov, vinh2@email.arc.nasa.gov, vandi@email.arc.nasa.gov
[5] University of Texas, Austin, 1 University Station A8000, Austin, TX 78712
Matt@MacMahon.org

Summary. In this paper, we describe NASA Ames Research Center's K10 rover as used in the 2006 Coordinated Field Demonstration at Meteor Crater, Arizona. We briefly discuss the control software architecture and describe a high dynamic range imaging system and panoramic display system used for the remote inspection of an EVA crew vehicle.

1 Introduction

In preparation for returning humans to the Moon by 2020, NASA is dedicating significant effort to building exploration systems that are affordable, reliable, and effective. A key part of this effort is to develop tools and techniques that will allow humans to operate efficiently and safely on the lunar surface over the long-term. Given that cost pressures will keep astronaut teams small (e.g., four-person crews), robots will be needed to extend the effectiveness of these teams well beyond their individual human capacity.

Robots, whether teleoperated or supervisory-controlled, can be used for "dull, dirty, or dangerous" tasks and other tasks that are not sensible or necessary for humans to perform unaided during exploration missions. Robots can also augment humans by differing from the human scale of performance in one or more parameters: force, size, speed, duration, precision, temperature, environmental exposure, cleanliness, etc. In particular, robots can be used

for a wide range of surface operation tasks including (but not limited to) site survey, construction, routine maintenance, visual inspection, instrument placement, and transport.

Involving robots in these tasks will provide three key benefits: (1) a significant reduction in the number and duration of extra-vehicular activity (EVA) sorties (and thus reduced cost and risk to crew), (2) more effective use of surface crew time (e.g., tasks can be off-loaded to ground control), and (3) enhanced mission capability (ability to reposition large payloads, greater sortie range, emergency resupply, etc.).

Coordinated Field Demonstration at Meteor Crater

During 2006, multiple NASA centers collaborated to develop human-robot systems for a variety of lunar surface activities. To provide a usage example of this collaboration a "Coordinated Field Demonstration" (CFD) was conducted involving robots from the NASA Ames Research Center (ARC), the Jet Propulsion Laboratory (JPL), and the NASA Johnson Space Center (JSC). The CFD took place during a two-week period in September 2006 near Meteor Crater, Arizona.

The scenario for the CFD focused on operations that occur after the crew returns from a "typical" lunar EVA sortie on a manned vehicle (the JSC SCOUT rover) to a Lunar base site. At the base site, three robots supported post-sortie operations: the JPL ATHLETE robot equipped with a modular Pressurized Rover Compartment provided crew shelter, the JSC Robonaut off-loaded and transfered a payload from SCOUT, and the ARC K10 rover performed a "walk-around" visual inspection of SCOUT. The visual inspection ensured that SCOUT was safe for the next EVA.

Visual Inspection Architecture

The ARC K10 rover performed a remote walk-around visual inspection of the SCOUT rover. This was in lieu of having an astronaut perform the same task. Our goal was to create a system that would provide a remote human operator with imagery of sufficient resolution and dynamic range to determine EVA vehicle integrity after a sortie. The system must autonomously collect the inspection data and present it to the remote user in an easily navigated form.

The inspection task architecture is as follows:

- The K10 rover, described in Section 2, automatically collects high-resolution panoramic imagery at multiple inspection points. We describe the onboard software and executive in Sections 3 and 4, respectively.
- Inspection panoramas use high-dynamic range imaging, described in Section 5, in order to capture detail in deep shadow and strong highlight regions.
- K10 images from the inspection of SCOUT are transmitted to the control station, where they are stitched into seamless panoramas and displayed in the Panorama Viewer, described in Section 6.

- Ground control operators perform inspection by "exploring" the high-resolution panoramas.

In Section 7, we describe the CFD results and conclude in Section 8.

2 Inspection Rover

Fig. 1. K10 "Blue" Rover

The K10 rovers (Figure 1) are a series of robots designed to be a cost-efficient, easily maintainable robotic test-bed using as many commercial off-the-shelf components as possible. Each robot features a 4-wheel steer, 4-wheel drive rocker chassis and a top speed of 1 m/sec, comparable to human walking speed. Hard points on all sides allow attachment of additional components including antennas, masts, arms, and other equipment.

K10's avionics are off-the-shelf with the exception of a few custom power electronics components. [1] The rover runs off of twenty 14.4V, 6.6 AH Li-Ion smart battery packs. The brain of K10 is an IBM Thinkpad X31 laptop that enables eased debugging and servicing while maintaining the reliability and computational power of a PC104 stack. K10's sensor suite includes a Novatel differential GPS system, a Honeywell digital compass, three sets of firewire stereo cameras, a Control Chief wireless e-stop, encoders and potentiometers for ground truth and wheel calibration, as well as temperature and power monitoring sensors.

K10's inspection camera rig consists of a stereo pair of Point Grey Scorpion cameras mounted atop a Directed Perception pan-tilt unit. The cameras have a resolution of 1600x1200 and are configured with synchronized shutters. The vergence angle of the stereo pair is 5 degrees with a baseline of 36.2 cm making the ideal target-to-rover distance about 1 meter. A third camera is mounted in the center of the camera rig for taking HDR panoramas. In order to resolve millimeter scale features at distances up to 5m, we selected a 35mm Schneider Optics Xenoplan lens for the HDR panorama camera.

3 Rover Software

The K10 rover is one of a number of hardware platforms supported at ARC. We use our rovers for diverse experiments calling for various sensor and actuator configurations and we conduct multiple field tests requiring integration of various scientific instruments. In addition, the robot controller uses complex

software components in applied computer vision and human-robot interaction. Finally the robot controller needs to smoothly integrate with interactive 3D visualization and monitoring systems for ground control.

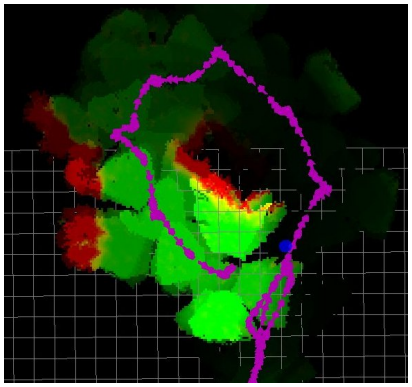

Fig. 2. Goodness map during an inspection task. The image shows safe areas (green) and dangerous obstacles (red). The confidence level (depicted by pixel brightness) decreases with time. SCOUTs rectangular profile is clearly seen in the center of the map. The other large red area is the ATHLETE robot sitting close to SCOUT.

To manage this complexity and create a scalable robotic system, we developed a Service-Oriented Architecture (SOA). Each component in this architecture provides specific functionality and exposes a clear interface to the others. The interfaces are defined using the Interface Definition Language (IDL) [2] and communication between components relies on CORBA [3]. The main components used for this particular field demonstration are: locomotion, localization, navigation and panorama acquisition. Most of these components are built using facilities provided by the CLARAty (Coupled Layer Architecture for Robot Autonomy) framework [4].

Locomotion

The locomotion service encapsulates the locomotion framework provided by CLARAty [5]. This framework provides a common locomotion interface for any wheeled vehicle. A generic algorithm handles steering and drive actuation accounting for the vehicles kinematics and its capabilities. Communication with the motors is delegated to adaptations of the generic CLARAty motors for the K10 specific hardware.

Localization

The localization service computes rover pose estimates required by the navigation system to reach absolute GPS waypoints. K10 uses an Extended Kalman Filter with inputs from rover odometry, compass, inclinometer and differential GPS.

Navigation

The navigation service computes a safe path to a given goal. The navigator evaluates the environment and generates trajectories for the locomotor to execute using the sense-think-act navigation scheme realized by the CLARAty

navigation framework [6]. Terrain reconstruction is performed with stereo vision using front and rear hazard detection camera pairs. The ARC stereo correlator [7] generates high fidelity 3D point clouds in 0.4 seconds on a 1.2GHz Pentium M. Rover path generation relies on the Morphin/D* algorithms implemented in CLARAty with specific input parameters for the K10 rover. An example of a goodness map and path generation is given in Figure 2.

Panorama and HDR Enabled Service

To obtain full coverage of the SCOUT at a nominal inspection distance of 2m, panoramas were acquired with approximately 90 degrees coverage in the horizontal direction and 45 degrees coverage vertically. The panoramic camera lens has a 12 degree horizontal and 9 degree vertical field of view. With 15% image overlap, achieving the desired SCOUT coverage requires acquisition of a 9x6 mosaic of images.

The panorama service requires a specific start up sequence since it requires the camera and the pan-tilt services to be running. Panorama requests are handled by first sending the pan-tilt unit to the minimum pan angle and minimum tilt angle. Images are then taken using the inspection camera. The pan-tilt unit moves in a raster pattern with small pan-tilt increment angles to insure image overlap. If the service is started with HDR enabled, several images with multiple exposure settings are taken at each step increment.

Benefits of the Service-Oriented Architecture

Each of the above-described subsystems is a service component which only relies on abstract (IDL-defined) interfaces to other components, they are resolved at service startup time. The SOA allows us to group the services into dynamic libraries that can be loaded and configured at run time. This reduces the time needed to recompile and relink software. The memory footprint of the controller is also reduced since unused sensors and algorithms are not loaded into memory. The highly decoupled nature of the system helps avoid over-bloated code because deprecated services can be removed with minimal effort and risk.

4 PLEXIL Exec

The K10 rover operates in an uncertain environment. For example, the execution time of a traverse may vary depending on whether or not the rover encounters obstacles, components fail, or subtasks take longer to complete than anticipated. A robust control strategy must therefore implement desired behavior flexibly while ensuring safety in all conditions. An executive is a general software system that implements a control plan that specifies what to do in given situations.

K10 uses the Universal Executive [8], a lightweight, efficient executive capable of executing PLEXIL plans. PLEXIL [8] is a language for representing control plans that is expressive and deterministic, and has formal semantics [9] that enable formal verification and validation. In addition, it allows for easy encoding of safety monitoring in the controller that runs parallel with task execution, thus ensuring safety constraints independent of individual tasks.

K10's task was to perform a visual inspection of the SCOUT vehicle. A number of primary and secondary inspection points were defined. At each of these locations, K10 was to take a HDR panoramic image of SCOUT and send it to ground control for analysis. HDR panoramas from primary inspection points were mandatory, while inspections from secondary locations were to be performed only if the rover was operating well within the given time limit.

During tests, the rover correctly skipped inspections at secondary tests to get back on schedule when obstacles slowed traversal between inspection points and when the rover was delayed by pose estimation inaccuracies. If no such delays were encountered, the robot acquired the whole set. During the final field demonstration runs, the time limit was very short, so the rover trimmed all secondary inspection points to accomplish the tight schedule.

5 HDR Imaging

The range of brightness values in most outdoor scenes far exceeds the dynamic range of commodity CCD and CMOS sensors. The luminance of a sunlit scene may vary by five or more orders of magnitude, whereas most solid-state sensors are 12-bit devices with a limited dynamic range of three orders of magnitude. Hence, even when the optimal exposure is measured and set for the scene, most images captured in a sunlit environment contain some regions that are either over- or under-exposed.

Due to the absence of atmospheric dispersion and the anisotropic nature of sunlight on the moon, severe shadows result in very dark areas that will be underexposed in digital imagery, and correspondingly, when exposure is adjusted for shadowed areas, the sunlit areas will be harshly overexposed.

We have addressed this issue by capturing a bracket of several images with varying exposures. A bracket consists of images with evenly spaced exposures that range from underexposed (to capture bright regions) to overexposed (to capture dark regions). The number of images in the bracket can be increased until the full dynamic range of the scene has been captured. In our experiments, a bracket of five images with an exposure ratio of 1 photographic stop (i.e. a factor of two in shutter speed) between images proved sufficient.

Once captured, the bracketed images are combined into a single high dynamic range (HDR) image using the weighted averaging technique of [10][11]. Combined in this manner, a HDR image contains well-exposed pixel information for every pixel in the image.

Fig. 3. Original low dynamic range mosaic (top) and high dynamic range mosaic (bottom).

The resulting HDR image contains all of the information needed by the ground inspection team, however it is not ready for display; print media and most display technologies have inherently low dynamic range, thus the dynamic range of a HDR image must be reduced before it can be presented to the user. Simply scaling the pixel values linearly to fit in the dynamic range of the display yields poor results because the human visual system's response to luminance is approximately logarithmic rather than linear. Instead, we use the logarithmic brightness mapping function described in [12], which was designed to closely mimic the physiological response to brightness in the human visual system. This operation, generically called tone mapping, produces natural looking images that are well exposed across a very wide range of real-world brightnesses.

Figure 3 shows a comparison of the inspection images before and after high dynamic range image processing. The low dynamic range mosaic is composed from a single exposure. The HDR images are composed from a bracket of five exposures, and then tone mapped for display using the algorithm in [12].

6 Panorama Stitching and Display

In order to provide a coherent context for visual inspection, individual images were projected onto a common spherical surface utilizing pointing information from rover telemetry, and merged into a single mosaiced panorama. As described above, the multiple exposures are combined into single images prior to panorama mosaic merging.

To reduce discontinuities in the merged panorama due to differences in brightness between individual images, a weighted average of pixel values was used where images overlapped. Pixel weighting is based on the distance of a given pixel from its containing images edge. Given the high resolution of the images and the number of images acquired, a straightforward distance calculation is computationally prohibitive. A fast distance transform technique [13] was implemented which enabled generation of blended merged 12000x6000 pixel mosaics in less than five minutes. The merged panoramas were then converted into a multi-resolution image structure for interactive viewing in the Panorama Viewer described below.

The Panorama Viewer was developed to allow inspection of the blended and merged panoramas. The multi-resolution structure of the final panorama image product enabled smooth, interactive navigation (zooming and panning) of the large images providing continuous contextual cues during the inspection process.

The Panorama Viewer user interface also provided means to quickly switch between panoramas acquired at adjacent inspection stations, again allowing the user to maintain a sense of continuity while navigating the data. The Panorama Viewer was developed as part of the Ames VizExplorer component of the Ensemble software framework. Ensemble is being developed in collaboration with JPL as a framework for mission operations software development. The Panorama Viewer utilizes the OpenSceneGraph C++ library [14] wrapped in a Java Native Interface (JNI) for high performance rendering.

7 Field Demonstration Results

Overall the Coordinated Field Demonstration was successful, with the robots from each NASA center reliably completing their tasks. At the end of each EVA sortie, the suited astronauts parked the SCOUT rover within the designated inspection area and K10 was commanded to begin its walk-around.

During the morning and early afternoon K10 had little problem properly navigating around SCOUT and taking inspection panoramas; however, after about 4pm the lighting conditions were poor enough to cause false obstacle detections leading to extremely long navigation times. Due to time constraints we did not acquire full HDR panoramas during every run. When running without acquiring panoramas at inspection points, K10's SCOUT inspection

took an average of fifteen minutes. A total of 18 full and partial inspection runs were performed; however, only two HDR panoramas were acquired.

Once the circumnavigation of SCOUT was complete, the inspection images were downlinked over the wireless network to the command trailer where the ground station automatically generated the stitched HDR panoramas. The HDR inspection panoramas contained sufficient resolution to enable a very detailed inspection of the SCOUT rover. Small details that would normally be obscured by deep shadow, such as a pebble stuck in the tread of the tire under the wheel well, were clearly visible.

The rover control software's Service-Oriented Architecture provided us with the flexibility to adapt inspection runs to the conditions and time constraints encountered in the field. For example, a single flag in the configuration file allowed switching between HDR and non-HDR panoramas. The availability of the rover component interfaces across the network (using the CORBA middleware) provides seamless multi-mode control. Autonomous execution could be suspended to allow direct teleoperation using a dedicated GUI, and then resumed.

8 Conclusion

Using robots to perform remote equipment inspections is one way of reducing risk to humans during exploration missions by reducing the number and duration of EVA. In order to enable remote visual inspection by a habitat- or an earth-based user, the inspection system must be robust and autonomous enough to keep mission operations simple and efficient. The imagery returned by the inspection robot must be of sufficient resolution and dynamic range that significant damage can easily be spotted. We have created an inspection task architecture that addresses these requirements and have demonstrated the system in the field. Further use studies would help to better characterize and improve the efficiency of the system.

9 Acknowledgments

The authors of this paper would like to acknowledge the support and sponsorship of the Exploration Systems Technology Development Program. We thank the interns and other members of the Intelligent Robotics Group at NASA Ames Research Center for their continuing support and inspiration. We would also like to thank our colleagues from NASA Johnson Space Center who provided us with invaluable support and made the Coordinated Field Demonstration possible, especially Ron Diftler and Darby Magruder.

References

1. M. Bualat, L. Kobayashi, S.Y. Lee, E. Park, "Flexible Rover Architecture for Science Instrument Integration and Testing," Proceedings of AIAA Space 2006, San Jose, California, September 2006.
2. Object Management Group. (2006) OMG IDL. [Online]. Available: http://www.omg.org/gettingstarted/omg idl.htm
3. Object Management Group. (2006) CORBA FAQ. [Online]. Available: http://www.omg.org/gettingstarted/corbafaq.htm
4. I. Nesnas, A. Wright, M. Bajracharya, R. Simmons, and T. Estlin, "CLARAty and challenges of developing interoperable robotic software," Proceedings of the 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems, Las Vegas, Nevada, October 2003.
5. M.G. Bualat, C.G. Kunz, A.R. Wright, I.A.D. Nesnas, "Developing an autonomy infusion infrastructure for robotic exploration," Proceedings of the 2004 IEEE Aerospace Conference, March 2004, Vol. 2, pp. 849-860.
6. C. Urmson, R. Simmons, I. Nesnas, "A generic framework for robotic navigation," Proceedings of the 2003 Aerospace Conference, March 2003, Vol. 5, pp. 2463-2470.
7. L. Edwards, M. Sims, C. Kunz, D. Lees, J. Bowman, "Photo-realistic terrain modeling and visualization for Mars Exploration Rover science operations," Proceedings of the 2005 IEEE International Conference on Systems, Man and Cybernetics, Vol. 2, pp. 1389-1395.
8. V. Verma, A. Jnsson, C. Pasareanu, M. Iatauro, "Universal Executive and PLEXIL: Engine and Language for Robust Spacecraft Control and Operations," Proceedings of AIAA Space 2006, San Jose, California, September 2006.
9. C Munoz, C Pasareanu, G Dowek, A Small-Step Semantics of PLEXIL, NASA Langley Research Center Technical Report.
10. E. Reinhard, G. Ward, S. Pattanaik, P. Debevec. High Dynamic Range Imaging: Acquisition, Display, and Image-Based Lighting. Morgan Kaufmann, San Francisco, 2006.
11. T. Mitsunaga, and S. K. Nayar, "Radiometric Self Calibration," 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'99), Volume 1, p. 1374.
12. F. Drago, K. Myszkowski, T. Annen and N. Chiba, "Adaptive Logarithmic Mapping For Displaying High Contrast Scenes," Computer Graphics Forum, Volume 22, Issue 3, 2003.
13. P. F. Felzenszwalb, D. P. Huttenlocher, "Distance Transforms of Sampled Functions," Technical Report TR2004-1963, Cornell Computing and Information Science, 2004.
14. The OpenSceneGraph high performance 3D graphics toolkit (2006) [Online]. Available: http://www.openscenegraph.org