Targeted Driving Using Visual Tracking on Mars: From Research to Flight

Won S. Kim, Issa A. Nesnas, and Max Bajracharya
Jet Propulsion Laboratory
California Institute of Technology
4800 Oak Grove Drive
Pasadena, California 91109
e-mail: Won.S.Kim@jpl.nasa.gov, Issa.A.Nesnas@jpl.nasa.gov, Max.Bajracharya@jpl.nasa.gov

Richard Madison
Charles Stark Draper Laboratory
555 Technology Square
Cambridge, Massachusetts 02139
e-mail: rmadison@draper.com

Adnan I. Ansar, Robert D. Steele, Jeffrey J. Biesiadecki, and Khaled S. Ali
Jet Propulsion Laboratory
California Institute of Technology
4800 Oak Grove Drive
Pasadena, California 91109
e-mail: Adnan.I.Ansar@jpl.nasa.gov, Robert.D.Steele@jpl.nasa.gov, Jeffrey.J.Biesiadecki@jpl.nasa.gov, Khaled.S.Ali@jpl.nasa.gov

Received 2 July 2008; accepted 27 December 2008

This paper presents the development, validation, and deployment of the visual target tracking capability onto the Mars Exploration Rover (MER) mission. Visual target tracking enables targeted driving, in which the rover approaches a designated target in a closed visual feedback loop, increasing the target position accuracy by an order of magnitude and resulting in fewer ground-in-the-loop cycles. As a result of an extensive validation, we developed a reliable normalized cross-correlation visual tracker. To enable tracking with the limited computational resources of a planetary rover, the tracker uses the vehicle motion

Journal of Field Robotics 26(3), 243–263 (2009)
Published online in Wiley InterScience (www.interscience.wiley.com). • DOI: 10.1002/rob.20283

estimation to scale and roll the template image, compensating for large image changes between rover steps. The validation showed that a designated target can be reliably tracked within several pixels or a few centimeters of accuracy over a 10-m traverse using a rover step size of 10% of the target distance in any direction. It also showed that the target is not required to have conspicuous features and can be selected anywhere on natural rock surfaces excluding rock boundary and shadowed regions. The tracker was successfully executed on the Opportunity rover near Victoria Crater on four distinct runs, including a single-sol instrument placement. We present the flight experiment data of the tracking performance and execution time.

1. INTRODUCTION

The Mars Exploration Rovers (MERs), Spirit and Opportunity, have been exploring the Martian surface for more than 4 years since landing in January 2004, well exceeding the 90-day primary mission lifetime. To date, they have logged a combined 17 km on different terrain types with about 25% of this traversal distance using autonomous vision-based navigation. The software release that was onboard at the time of landing provided robotic capabilities for autonomous navigation with local obstacle avoidance using stereo-based image processing (Maimone, Leger, & Biesiadecki, 2007). It also provided a capability for rover pose (position and orientation) estimation using visual odometry (Cheng, Maimone, & Matthies, 2005).

In parallel with the robotic software development for the MERs, NASA centers and universities were developing a number of robotic autonomy technologies for potential use in future missions under the competed NASA's Mars Technology Program. The technology development cycle can take from 1 to 3 years, after which the technology gets tested onto a research rover at the Jet Propulsion Laboratory (JPL) or at NASA Ames Research Center. The Mars Technology Program provides a unified software infrastructure called CLARAty (Coupled Layer Architecture for Robotic Autonomy) (Nesnas et al., 2003) for the integration, maturation, and deployment of these technologies onto the research rovers.

One of the technologies developed under this program was visual target tracking (VTT). This technology is a key component of targeted driving and single-sol instrument placement, which enable scientists to direct the rover to autonomously go to a remote target 10–20 m away on Mars and collect in situ scientific measurements. The benefit of this autonomous capability is a reduction in operational time from, at least, three sols to a single sol, thus increasing the overall science return for the mission. Following its successful validation, VTT was selected as one of the four technologies (Hayati et al., 2007) to be demonstrated on Mars. Since then, we integrated this capability into the MER flight software, and four distinct runs were successfully executed on the Martian surface as part of the operational checkouts process. A sequential combined execution of the targeted driving capability with the vision-based arm collision detection and auto placement capability (Hayati et al., 2007; Maimone et al., 2007) on the Opportunity rover demonstrated the first single-sol instrument placement on a planetary surface from 6 m away.

This paper focuses on the VTT for targeted driving. After the Background section, we present the overall concept of targeted driving operations in Section 3 and provide details of the VTT component in Section 4. In Section 5, we present the validation results. In Section 6, we describe the infusion of this capability into the MER flight software. Thereafter, we present the successful results of running this capability on the Opportunity rover on Mars in Section 7 and close with a conclusion in Section 8.

2. BACKGROUND

Visually localizing and tracking a target using a template designated in a previous image is a well-studied problem in computer vision, and several survey papers on image registration (Brown, 1992; Zitova & Flusser, 2003) and object tracking (Yilmaz, Javed, & Shah, 2006) are available. One recent approach is the SIFT (scale-invariant feature transform) algorithm that detects and uses scale-invariant features to match one frame to the next (Lowe, 2004). Such an approach can be computationally expensive for applications with limited computing power. In contrast to feature-based image matching, a number of area-based matching algorithms have been well studied. There are two search techniques for area matching:

brute-force and iterative search. The iterative method uses successive approximations based on image gradients to track a distinct target (Lucas & Kanade, 1981; Shi & Tomasi, 1994) but allows only very small target image displacement between frames. The use of multiresolution image pyramids (Anandan, 1989; Bergen, Anandan, Hanna, & Hingorani, 1992; Bouguet, 2000) allows relatively larger image displacements. When the translational displacement of the target image is larger than tens of pixels, the brute-force search by image correlation tends to be more reliable as long as the target image appearance does not change much in terms of scale and rotation. Image similarity measures used for area matching include sum of absolute differences (SAD) (Barnea & Silverman, 1972), sum of squared differences (SSD), and normalized cross-correlation (NCC) (Pratt, 1974; Rosenfeld & Kak, 1982). Normalization in NCC makes matching less sensitive to image brightness and contrast changes. Computation becomes expensive with a large search area. Coarse–fine template matching (Rosenfeld & Vanderbrug, 1977) reduces computation time significantly, by starting with a low-resolution template matching, followed by a full-resolution template matching.

Coupling the visual feedback provided by target tracking with the control of a robot also results in the well-studied problem of visual servo control (Ellekilde, Favrholdt, Paulin, & Petersen, 2007; Hutchinson, Hager, & Corke, 1996). Most terrestrial applications can run at high frame rates and can consequently assume that the target has only a small change in appearance from one frame to another. Unfortunately, for space applications, in which radiation-hardened processors lack the computational bandwidth of their terrestrial counterparts, this assumption does not hold and a high-frame-rate tracker is not feasible. For example, on the MERs' 20-MHz CPU, image capture and transfer takes tens of seconds for a pair of 12-bit full-resolution (1,024 × 1,024 pixels) stereo images (or several seconds for 8-bit quarter-resolution of 256 × 256 pixels). Because of these limitations, we have to drive the rover in small steps and perform VTT only at these discrete stops. This implies that the visual tracker must reliably handle large changes in image appearance between frames. Unfortunately, NCC itself can handle only slight changes in target image scale and rotation. Although there are generalized NCC with geometric transformation (Zitova & Flusser, 2003), the computational load increases very fast with the number of parameters while the false matching probability increases as well. Instead of determining the geometric transformation purely from images, which is difficult and time-consuming, we utilize the available rover pose estimation to scale and roll the template image prior to NCC, which enabled reliable tracking for space applications with a limited computational resource.

Some of the very first versions of VTT on planetary research rovers were demonstrated on the Marsokhod rover at NASA Ames Research Center (Wettergreen, Thomas, & Bualat, 1997) and on the Rocky 7 rover at the JPL (Nesnas, Maimone, & Das, 1999). The Marsokhod tracker used the sign of the difference of Gaussian (SDOG) to match the target templates to new images. The Rocky 7 tracker used three-dimensional (3D) information from stereo images combined with intensity information to track the target, which was used for both sample acquisition and instrument placement. A follow-on joint effort by these two teams led to the first visual target tracker (Nesnas et al., 2004) that was developed and deployed on the Rocky 8 rover using the CLARAty framework (Nesnas et al., 2006; Volpe et al., 2000). Another joint effort under the ASTEP (Astrobiology Science and Technology for Exploring Planets) program also led to the development of a single- and multiple-target instrument placement on the K9 rover at Ames Research Center (Pederson, Bualat, Smith, & Washington, 2003).

The resultant tracker was a two-stage visual tracker consisting of a first-stage normalized cross-correlator (NCC) and a second-stage multiresolution affine tracker for fine matching (Bajracharya, Diaz-Calderon, Robinson, & Powell, 2005; Nesnas et al., 2004). Subsequent validation experiments, however, indicated that pure NCC for the first stage was not sufficient to cope with the large changes in image appearance. Therefore, we added a preprocessing step to the NCC to compensate for image scale and roll using the rover's pose estimator and removed the second stage to improve the overall reliability of the tracker (Kim, Ansar, & Madison, 2006). The final tracker used *template image scale and roll* in the preprocessing phase feeding into the normalized cross-correlator stage. This visual target tracker was able to reliably track any target on natural rock surfaces with little contrast.

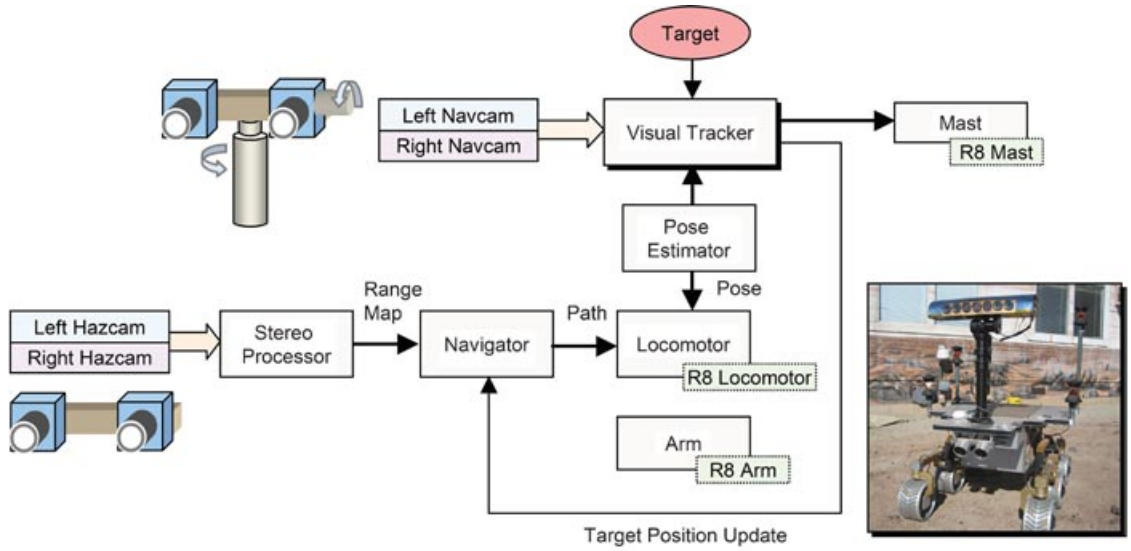


Figure 1. Functional diagram of the overall targeted driving capability. Navcams are mounted on the mast, and hazcams are mounted on the rover body.

3. TARGETED DRIVING SYSTEM

Targeted driving enables operators to autonomously drive planetary rovers to visually designated targets that are up to 10–20 m away. In a typical operational scenario, scientists and rover planners first select a target from the downlinked navigation camera (navcam) images of the previous day and then uplink a command sequence that specifies the designated target position, the rover's goal distance (and optional position offset) from the target, and subsequent operations for targeted driving. Then the rover autonomously drives to the goal position, avoiding obstacles while continually tracking the target using the rover's articulated stereo camera pair, and reaches the goal position within a few centimeters of accuracy. The initial target position can be specified in one of the three ways: (1) two-dimensional (2D) image coordinates in the left navcam image together with the mast's azimuth/elevation angles, (2) 3D coordinates in the local rover frame, or (3) 3D coordinates in the global site frame.

As shown in a functional diagram of the targeted driving capability (Figure 1), the visual tracker closes the feedback loop through navigator path planning, locomotor driving, rover pose estimator, and mast-mounted camera pointing and stereo imaging. The navigator uses a goodness map generated from the range data of the body-mounted stereo hazard avoidance cameras (hazcams). Both the Morphin navigator (Urmson, Simmons, & Nesnas, 2003) and the GESTALT navigator (Maimone et al., 2007) were used during the validation of this capability, with the latter used when operating on Mars. For pose estimation, we typically use wheel odometry integrated with inertial sensing (Ali et al., 2005) for flat hard-packed terrains. For slopes or high-slip terrains, we additionally use visual odometry (Cheng et al., 2005) combined with wheel odometry and inertial sensing for improved position estimation accuracy. The pose estimate is then fed into the algorithm to point the mast camera to the target (Kim, Ansar, & Steele, 2005).

4. VISUAL TARGET TRACKER

4.1. Functional Description

Figure 2 shows a functional diagram of the visual tracker. After the target is selected in the image, we initialize a fixed-size template centered at the target point to represent the target. Then the tracker performs the following actions at each tracking update:

1. After the rover stops, estimate the target position relative to the new rover pose using the target position updated at the previous rover

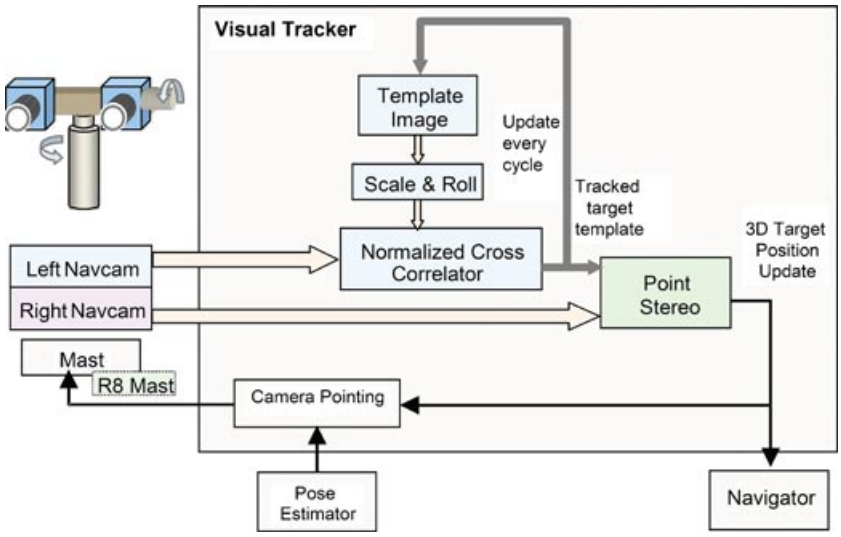


Figure 2. Functional diagram of the visual target tracker.

stop and the change in rover pose reported by the pose estimator(s).

2. Compute the azimuth and elevation angles of the camera mast so that the left navcam points to the estimated target at the center of the image. Point the mast and capture navcam stereo images.
3. Use the estimated target position and the left navcam camera model to compute the scale (magnification/shrinking) factor and roll angle of the target template image to compensate for the image appearance change due to the rover roll motion. Scale and roll the target template image accordingly.
4. Apply NCC to the left navcam image within a search window to find the new target image point that matches best to the scaled and rolled template image.
5. Update (replace) the target template image with the window around the newly found target point in the left navcam image.
6. Perform point stereo to find the corresponding target image point in the right navcam image and compute the newly updated target 3D position by triangulation.

Figure 3 shows an example of VTT results, demonstrating good tracking despite drastic change in target appearance.

4.2. Theoretical Calculation of Target Approach Accuracy

To meet the instrument placement accuracy requirement of less than 3 cm, we need to visually track the designated target during the traverse. To illustrate this, we compute the target positioning accuracy with and without a visual target tracker for an 8-m rover traverse from an initial position 10 m away from the target to the final goal position 2 m away. When a mission planner selects a target on a downlinked image, there is an inherent target designation error because the target position measured by the stereo camera views is limited by the stereo down-range error ΔR, which is proportional to the square of the target distance from the rover:

$$\Delta R = \frac{R^2}{f_p B}\Delta d, \tag{1}$$

where B is the stereo baseline, R is the range, Δd is stereo disparity error, and f_p is the camera's equivalent focal length in pixels (Kim, Ansar, Steele, & Steinke, 2005). The stereo disparity error Δd is assumed to be 1 pixel for the 3-σ stereo range error. The camera focal length f can be converted to pixels by

$$f_p = f/\text{pixel\_size}. \tag{2}$$

Here, we will consider two different mast-mounted cameras equipped with a 1/3-in. (8.5 cm)

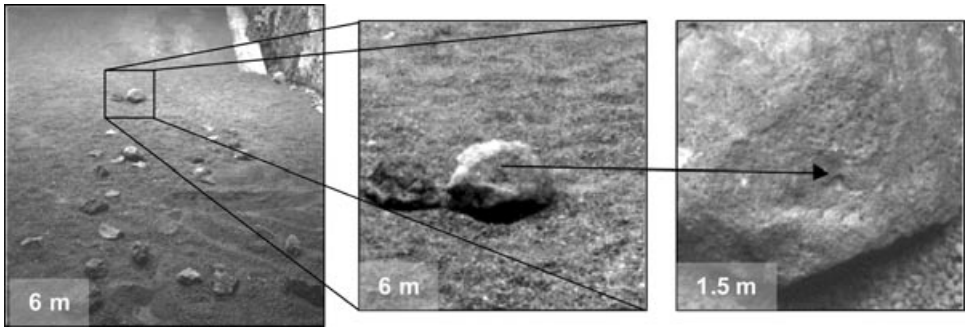


Figure 3. Dramatic change in target appearance. A full frame, full resolution (1,024 × 1,024 pixels) with a target selected at the center of the large rock in the MER indoor test facility (left). The 161 × 161 pixel NCC search window at the start of the traverse (middle). The search window with the target point in the center at 1.5 m away (right). The two target images bear little resemblance after 4× enlargement.

charge-coupled device (CCD) image sensor with a $1{,}024 \times 768$ pixel resolution: (1) pancam with $B = 300$ mm and $f = 16$ mm with a 17×13 deg field of view and (2) navcam with $B = 200$ mm and $f = 6$ mm with a 49×37 deg field of view. Because the effective image size of the 1/3-in. CCD image sensor is approximately 4.8×3.6 mm, each CCD pixel is a square with the pixel size of $4.8/1{,}024$ mm $= 3.6/768$ mm $= 4.69$ μm (manufacturer's specification of the actual pixel size was 4.65 μm). From Eqs. (1) and (2), the target designation errors at a 10-m distance, which is the stereo range error at 10 m, are computed, and they listed in the second column of Table I.

Without visual target tracker, the commanded goal position defined relative to the designated target position is unchanged throughout the rover traverse. Thus, at the end of the 8-m traverse, the target approach error without visual tracker is roughly the root sum square (RSS) of the target designation error and the rover navigation estimation error over an 8-m traverse:

$$\Delta R_{\text{no\_tracking,10m}} = \sqrt{\Delta R^2_{\text{stereo,10m}} + \Delta R^2_{\text{nav,8m}}}. \quad (3)$$

If we assume that the rover navigation error based on the wheel and visual odometers is roughly 2% of the rover travel distance, then $\Delta R_{\text{nav,8m}} = 16$ cm. The computational results of Eq. (3) are listed in the third column of Table I.

When a visual target tracker is employed, the target position and its related commanded goal position are updated more accurately as the rover approaches the target. If the target is perfectly tracked, the target position error will be reduced to the stereo range error at a 2-m distance with $\Delta d = 1$ pixel. However, because a 1-pixel (±0.5-pixel) image expands to a 5×5-pixel image as the rover approaches the target from 10 m away to 2 m, the visual tracker cannot track more accurately than ±2.5 pixels. This can be conveniently considered as a target drift of

Table I. Target designation error and target approach errors without and with visual tracking. Visual tracker compensates for the initial target designation error by continually updating the target position during the course of target approach.

Camera specifications (1/3-in. CCD sensor)	Target designation error (3σ) due to stereo range error at 10-m distance (cm)	Target approach error (3σ) at 2 m using visual odometry without tracker (cm)	Target approach error (3σ) at 2 m with visual tracker with N-pixel drift at 2 m (cm)
Pancam, $f = 16$ mm, $B = 30$ cm	9.7	18.7	$\sqrt{(0.4)^2 + (0.06N)^2}$
Navcam, $f = 6$ mm, $B = 20$ cm	38.8	42.0	$\sqrt{(1.6)^2 + (0.16N)^2}$

$\Delta x_d = \pm 2.5$ pixels, where the cross-range error (Kim, Ansar, Steele, & Steinke, 2005) is given by

$$\Delta R_{\text{drift}} = \frac{R}{f_p} \Delta x_d. \tag{4}$$

From Eq. (4), a 1-pixel drift at a 2-m target distance results in 0.6-mm lateral error for pancam and 1.6-mm lateral error for navcam. At the end of the 8-m traverse, the total approach error with visual tracking is approximately the RSS of the stereo down-range error at 2 m and the cross-range error due to drift:

$$\Delta R_{\text{tracking,10m}} = \sqrt{\Delta R^2_{\text{stereo,2m}} + \Delta R^2_{\text{drift}}}. \tag{5}$$

The computational results of Eq. (5), with $R = 2$ m, $\Delta d = 1$ pixel, and a drift $\Delta x_d = N$ pixels, are shown in the last column of Table I. Even with an extreme 10-pixel drift, the target approach error with visual tracker is 0.7 cm for pancam and 2.3 cm with navcam. It clearly demonstrates that VTT improves the target approach accuracy by an order of magnitude.

4.3. Image Registration and Tracking Using Rover Motion Estimation

As shown in Figure 2, we chose NCC with template scale and roll correction as the final template matching configuration of the visual target tracker. We use the available rover pose estimation to determine the scale and roll.

4.3.1. Template Image Scale and Roll Correction

Based on the perspective projection imaging geometry, the target image scale is inversely proportional to the target distance from the camera. Therefore, the template scale factor can be computed by the ratio of the old target distance before the rover's move D_{old} to the new target distance after the move D_{new}:

$$\text{scale\_factor} = \frac{D_{\text{old}}}{D_{\text{new}}} = \frac{|\mathbf{p}_{\text{old}} - \mathbf{C}_{\text{old}}|}{|\mathbf{p}_{\text{new}} - \mathbf{C}_{\text{new}}|}. \tag{6}$$

The target distance is the vector norm from the left navcam lens center $\mathbf{C}$ to the target 3D position $\mathbf{p}$, where $\mathbf{C}$ and $\mathbf{p}$ are updated from the rover pose estimation. When the rover moves closer to the target, the template image gets enlarged with scale_factor greater than 1. When the rover moves away from the target, scale_factor becomes less than 1.

Figure 4 illustrates the importance of template image roll compensation. A large roll angle can be observed not only during traverses over rocky terrain but also when the rover passes by the target on perfectly flat terrain. The rover may pass the target to approach it from a different angle. Figure 4 shows a 30-deg roll between two images on a flat terrain traverse. Without the proper roll compensation, this run would not have succeeded.

The template image roll compensation rotates the image only about the camera optical axis perpendicular to the image plane, changing neither the perspective viewpoint nor the target image shape. Because the mechanical pan-tilt mast of the rovers provides only two degrees of freedom for azimuth and elevation control, the third degree of freedom of roll control can be achieved only by software. Given the two camera models obtained from the rover pose estimation before and after the rover's move, we employ a practical approach to determine the roll angle change between the two images.

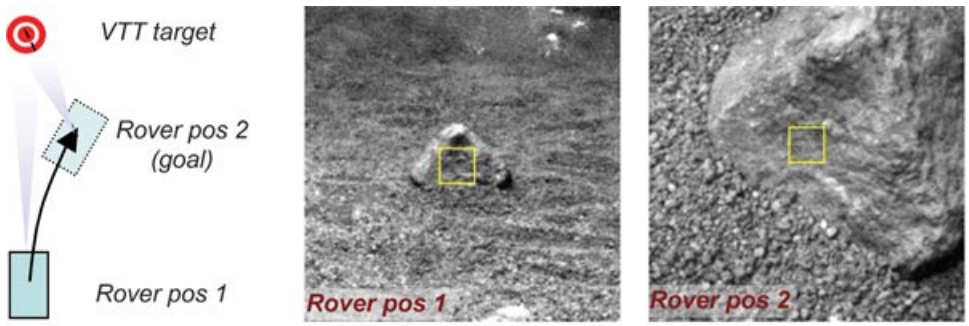


Figure 4. A 30-deg roll in image at 6.8 m away (pos1) that at 1.8 m away (pos2).

We define a hypothetical 3D point $\mathbf{p}_{0\text{-adj}}$ adjacent to the target point $\mathbf{p}_0$ such that the vector from $\mathbf{p}_0$ to $\mathbf{p}_{0,\text{adj}}$ is parallel to the horizontal axis of the camera model before the rover's move. The projection of the vector on the template image makes zero roll angle, aligned with the horizontal axis of the template image. In the CAHVOR camera model (Gennery, 2006) used on the MER, the horizontal unit axis vector $\mathbf{H}_x$ along the x axis can be derived from $\mathbf{H}$, $\mathbf{V}$, and $\mathbf{A}$ vectors, which are not mutually orthogonal:

$$h_s = |\mathbf{A} \times \mathbf{H}|, \tag{7}$$

$$h_c = \mathbf{A} \bullet \mathbf{H}, \tag{8}$$

$$\mathbf{H}_x = \frac{\mathbf{H} - h_c \mathbf{A}}{h_s}. \tag{9}$$

Then the hypothetical adjacent point is given by

$$\mathbf{p}_{0,\text{adj}} = \mathbf{p}_0 + k\,\mathbf{H}_x, \tag{10}$$

where k is set to an arbitrary, small length of 5 cm. Now using the new camera model after the rover's move, we can project both points, $\mathbf{p}_0$ and $\mathbf{p}_{0,\text{adj}}$, on the new camera image plane. The projected vector from these two points determines the template roll angle needed. Because the image y axis is defined downward, a clockwise rotation yields a positive roll angle.

4.3.2. NCC

After the preprocessing step that scales and rolls the template, we match the template to the target using NCC given by

$$N = \frac{\sum (I - \bar{I})(J - \bar{J})}{\sqrt{\sum (I - \bar{I})^2 \sum (J - \bar{J})^2}}, \tag{11}$$

where $\bar{I}$ and $\bar{J}$ are the pixel intensity averages of the template image and the matching image window of the same size. The range of N is $-1 \le N \le 1$. The NCC value is computed for every possible matching window within the image search area, and the image window location that yields the maximum correlation is selected as the best match. To speed up the computation, we employed multiresolution coarse-to-fine template matching. This technique increased the computation speed by a factor of 10. On the MER processor, this algorithm takes about 4 s to process for a 21×21 template over a 161×161 search image.

4.3.3. Template Update

We typically update the template at every tracking update cycle to accommodate noticeable changes in target appearance (up to 10% scale change and 5.7-deg rotation). Figure 3 illustrates a drastic change between initial and final target images, strongly suggesting that template updates are necessary. Because the total number of rover steps and corresponding template updates is small (<20), continuous updates did not cause noticeable cumulative drifts. Therefore, other template update schemes (Matthews, Ishikawa, & Baker, 2004) were not considered.

4.4. Point Stereo and Target Position Update

After we match the target point in the left navcam, we find the corresponding point in the right navcam image. Because the full-frame stereo vision is time-consuming, we use a point stereo that searches for the corresponding right image point only along the epipolar line with a few pixels (default 2 pixels) of vertical offset tolerance (Bajracharya et al., 2005). Once the 3D target position is computed by point stereo, the target position is updated. We limited the search disparity range to one half of the search window width, which increased the computational speed by a factor of 10. On the MER processor, this algorithm takes about 1 s to process.

4.5. Subframe Exposure Control

To achieve robustness against changes in lighting conditions, we use automatic exposure control over a subframe region rather than over the entire image area. Controlling the exposure on small subframe regions around the target reduces the impact of a large global effect due to the change in proportion of the very bright sky region in the image relative to darker terrain surface. By doing local autoexposure adjustment over a small subframe region, we maintain a good contrast within the region surrounding the target. This feature, which was available as part of the MER flight software (Litwin & Maki, 2005), was used in VTT.

4.6. Target Loss Detection and Fault Protection

Flight software qualification requires a fault protection mechanism to handle target losses, which can result from anomalous conditions such as occlusion, shadow, and indistinct features. We used four main

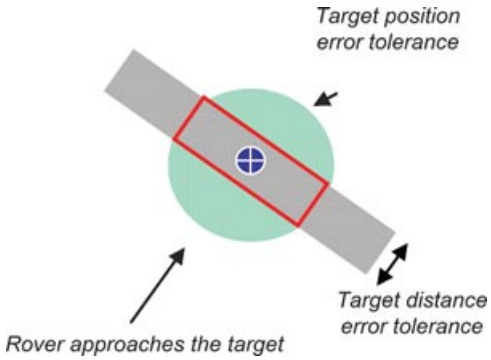


Figure 5. Both target distance and position error tolerances are recomputed at each tracking update.

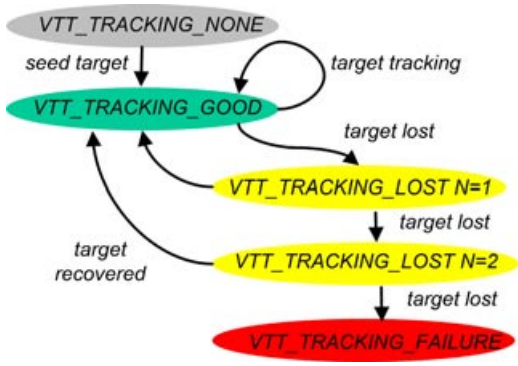


Figure 6. Visual tracking state transition diagram for fault protection.

criteria to detect tracking outlier and declare target loss:

1. NCC match score < 0.7
2. point stereo match score < 0.7
3. target distance correction between frames $>$ target distance error tolerance (Figure 5)
4. cumulative target position correction $>$ target position error tolerance (Figure 5)

The first and second criteria avoid potential false matches by eliminating the cases when the correlation score is too low. The threshold is set to 0.7 empirically based on experiments. The third criterion checks the target distance change after each stereo update primarily along the bore-sight direction of the camera. The fourth criterion checks the cumulative target position change relative to the initial target position.

When the VTT target is lost, the tracker enters a "lost" state but continues to update up to N times (Figure 6), at which the default value is 2. If the target is recovered during these N frames, the system returns to normal state. Otherwise, VTT declares FAILURE and remains in that state until the target is cleared.

5. TECHNOLOGY VALIDATION

The thorough validation process not only helped home in on the most robust tracking algorithm for the given mission constraints, but it also produced sufficient experimental results to secure approval for flight infusion. The technology development and research environment used for the visual tracking delivery was the CLARAty test bed that allowed testing this capability on planetary research rovers at JPL's outdoor Mars Yard.

Extensive experiments were performed to compare eight different tracker configurations for space applications (Kim, Steele, Ansar, Ali, & Nesnas, 2005; Kim, Steele, Ansar, Ator, & Catron, 2006), and the results showed that the NCC with the template image scaling using the rover motion estimation performed best. For brevity, here we present only the experimental results comparing this best configuration with the pure NCC. Template image scaling refers to magnifying or shrinking the image without changing the template window size. A 21×21 pixel template window was used because it provided the best result for natural rock images tested in the Mars Yard.

5.1. Validation Results from Tracking with Straight Traverses toward Target

In this experiment, we compared the performance of the two tracker configurations for straight traverses toward the target. First, we drove the Rocky 8 rover in the JPL Mars Yard in a straight line toward a target 10 m away and collected navcam stereo images at steps of 5% of the remaining distance. Because the target image size changes inversely with the target distance, the step size magnifies the target image by a constant 5% for every step. Therefore, the step size was gradually reduced from 0.5 m at 10 m to 0.12 m at 2.26 m for a total of 7.74-m traverse. We ran different tracker configurations for the same data set on each of the same 20 initial target points. The tracker tracks only one target per run.

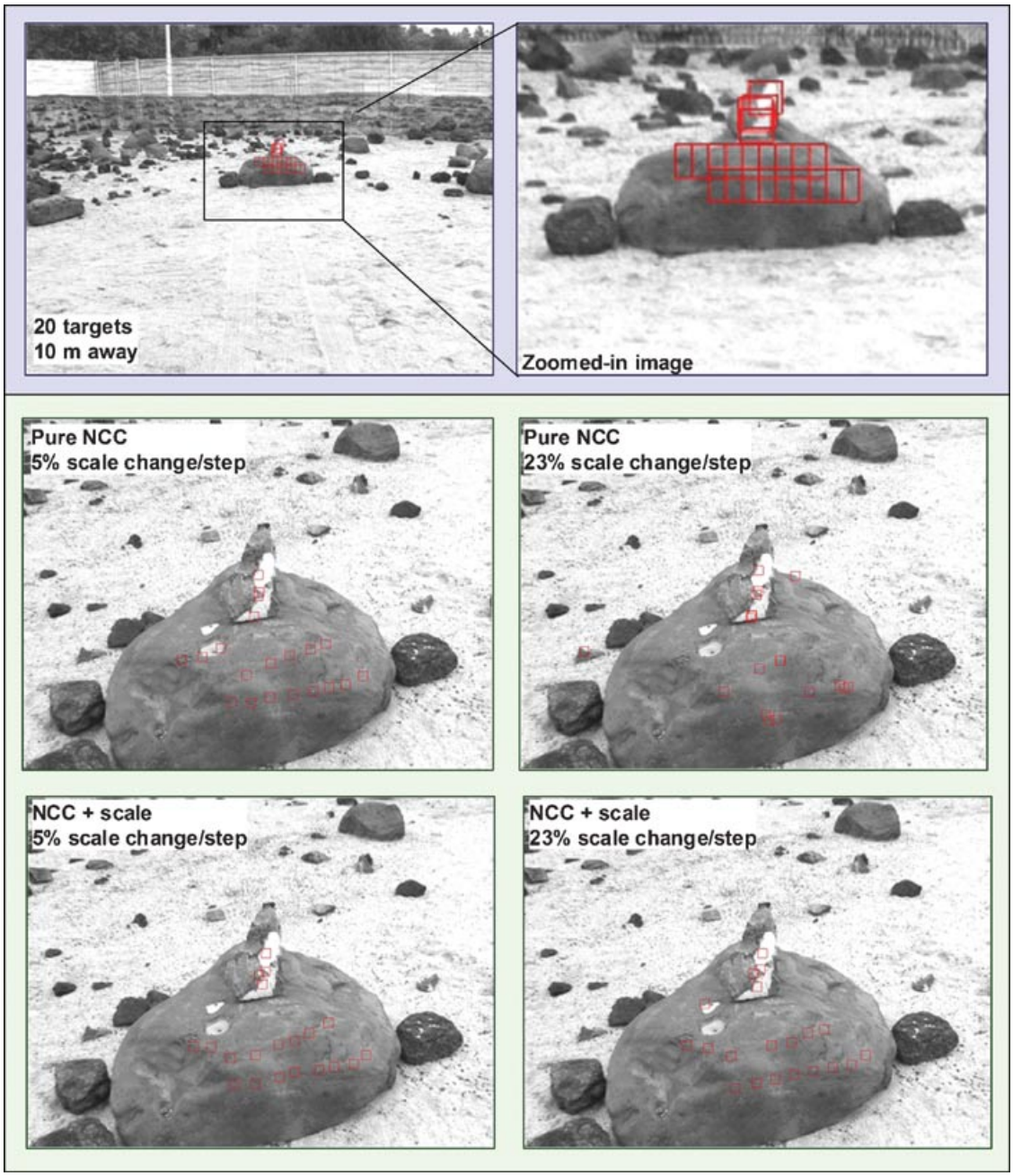


Figure 7. Tracking results for straight traverse toward target. Initial target selection at 10 m (top two images); final tracked targets at 2.3 m after 30 steps (bottom four images).

The top left image in Figure 7 shows the initial image used for target selection with an overlay of the template windows of the 20 targets selected. The top right image of Figure 7 shows a zoomed-in view. Note that targets on the large rock were intentionally selected on two lines with equal spacing. This aligned pattern of multiple targets helps examine tracking drifts easily by observing the evenly arranged smooth curve pattern of final target positions overlaid. We examined the tracker's performance by

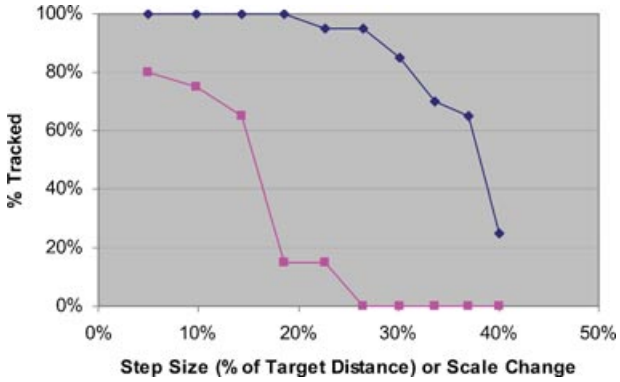


Figure 8. Comparing tracking performance with straightforward rover motions. NCC with template image scale (diamonds) performed much better than pure NCC (squares).

skipping images in the set to simulate a larger step size. For an image skip of N, the resulting step size is $1 - 0.95^{(N+1)}$.

The bottom four images of Figure 7 show the final tracked target positions. At 2.3 m, target images were magnified by a factor of 4 relative to the initial targets 10 m away. Further, the viewing angle changed significantly: mostly looking forward in the initial image while mostly looking down in the final one. In the second row of Figure 7, pure NCC tracked 16 targets of 20 (80% tracking) with a step size = 5% of target distance and tracked 3 targets (15% tracking) with a step size = 23% target distance. By contrast, in the bottom row of Figure 7, NCC with template image magnification (scale) tracked all 20 targets (100% tracking) with a 5% step size and lost only 1 target (95% tracking) with a 23% step size. The evenly arranged smooth curve pattern of final target positions for the bottom left image of 100% tracking indicates that tracking drifts are not noticeable despite that the targets are picked arbitrarily on two lines with equal spacing without considering conspicuous features. Based on the above results, only NCC with image scale was used in the subsequent tracking experiments with sideways and turn-in-place rover motions. A tracking percentage plot of the above experiments is shown in Figure 8, clearly demonstrating that template image scale is essential to achieve reliable tracking performance.

5.2. Tracking with Sideways Rover Motions

The rover motions chosen in this experiment were sideways crab motions encircling the target point with 5-deg heading change per step, while keeping the rover facing the target. These turn-around-the-target motions have an effect of changing the azimuth angle of the target relative to the rover. The beginning and end images in these experiments are shown in Figure 9, where the rover encircled the target by 70 deg in total at a 7-m radius. Experimental results in Figure 10 indicate that good tracking over 70 deg

Figure 9. Tracking with sideways rover motions of 70 deg in total encircling the target. Beginning (left) and end (right) images show that only one face of the big rock is seen on both images. Targets were selected on this surface. Note that image appearance changes significantly. Excluding a badly selected target (top location) on the occluding boundary, the images illustrate that 2 targets of 11 were lost at 15-deg step size.

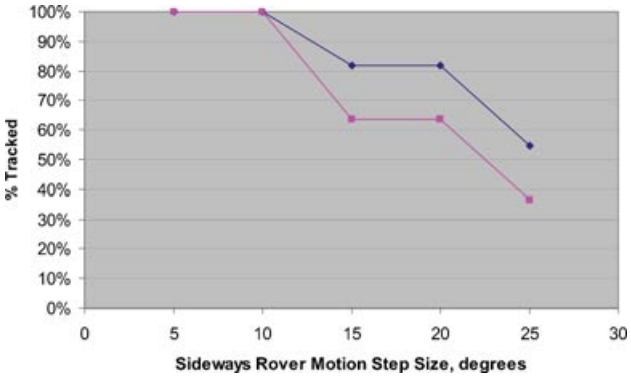


Figure 10. Tracking results with sideways rover motions as a function of step size and search window size. The smaller search window size (200 pixels; diamonds) had less chance to lose tracking than the larger search window (600 pixels; squares).

was possible with a step size of up to 5–10 deg. As the search window size increases, the tracking percentage decreases slightly because the chance of target loss by picking up false target increases. Among 12 targets, 1 target happened to be selected on the occluding boundary where its target window (top one in Figure 9) overlapped with another rock in the background. Because of large discontinuities at occluding boundaries, the tracker failed for this target. As an operating guideline, the target should not have been selected on occluding boundaries. Excluding the target on rock boundaries, the tracker successfully tracked all targets for 5- and 10-deg step sizes.

5.3. Tracking with Turn-in-Place Rover Motions

It is often desirable to change the heading by turn-in-place motion during visual tracking, for instance, to face the rover toward the target at the starting position or after the rover reaches the goal position. To determine the appropriate step size for the turn-in-place motion, visual tracking was performed with every 5 deg of turn-in-place motion. Tracking experiments with different step sizes indicated that tracking was good over the entire range of 5–45 deg for turn-in-place motions. Unlike the previous turning-around-target motion, image appearance did not change even with a large turn-in-place motion because of exact compensation by mast camera pointing using pan control. For this reason, there is no limit in turn-in-place step size as long as the rover pose estimator is sufficiently accurate.

Figure 11. System-level testing on Rocky 8 in the JPL Mars Yard.

5.4. Determining Rover Drive Step Size

On the basis of the above visual tracking validation experiments with straightforward, sideways, and turn-in-place rover motions, we determined that an appropriate rover step size is 10% of the target distance in all directions. The 10% limit is sufficient to cover any direction of rover motion because it limits the maximum target azimuth angle due to the rover sideways motion to 0.1 radian or 5.7 deg. Thus, a separate limit on target azimuth angle change is not necessary. The rover step size is 1 m when the rover is 10 m away from the target and 20 cm at 2 m away.

5.5. Visual Tracking with Visual Odometry on Various Terrains

As a system-level test, visual tracking combined with visual odometry was performed on four types of terrains (Figure 11): (1) flat ground, (2) shallow (6–10 deg) inclines, (3) steep (15–20 deg) inclines, and (4) oblique path up/down hill. Among the primary considerations in testing was the feasibility of VTT in the presence of large vehicle slip. A total of 26 runs were performed in a variety of situations, some of which pushed the tracking tests beyond normal operating conditions. The results are summarized in Table II. There were two failures. One failure happened during an extreme rover motion case of excessive target azimuth angle change of more than 10 deg combined with target occlusion. The other failure happened when there was a dramatic sun lighting change due to a patch of opaque clouds passing across the sun, which caused a large drift in tracking target position. This dramatic change is unlikely on Mars because clouds on Mars are thin and fluffy.

Table II. Visual tracking results with visual odometry on various terrains.

Terrain type	No. of trials	No. of failures	Reason for failures
Flat ground	11	1	1 Failure due to excessive azimuth change
Shallow incline	4	0	None
Steep incline	6	0	None (despite 100% slippage sometimes)
Oblique path	5	1	1 Failure due to dramatic lighting change

From the above observations, VTT was 100% successful for all normal operating conditions, demonstrating its robustness. In many cases, significant slip occurred during the traverse on slopes. For steep hills, the slip was sometimes as large as 100%, and yet tracking was 100% successful.

6. INFUSION INTO MER

The MER flight software (Reeves, 2005) consists of many modules. Many of these modules start their own independent task(s) and monitor message queues to communicate with other tasks through message passing. Some modules, however, simply provide library functions that are called and managed by other modules. The Autonomous Navigation (NAV) module (Maimone et al., 2007) is such an example. Likewise, the VTT module also contains a collection of visual tracking functions but does not have its own task context. The Mobility Manager (MOBM) task (Biesiadecki & Maimone, 2006), while communicating with other tasks via message passing, handles queued messages including commands to perform visual tracking and calls appropriate VTT and NAV module functions when needed. Similar to the NAV module, the VTT module ported from CLARAty is mostly written in C++. Global variables and public functions are defined with a C interface, so that MER C programs can call VTT public functions.

VTT is fully integrated into the existing MER software, so that it can run in any combinations of rover driving: blind driving (using the wheel odometry and initial sensing estimator only), Visodom (visual odometry), and Autonav. Figure 12 illustrates the details of a single rover step. The upper portion shows the MOBM/NAV functional flow of the original MER flight software, and the lower portion shows the insertion of VTT functions. If Visodom is enabled, this is done first. Then, if Autonav is enabled, the hazard avoidance map is updated and path selection for autonomous navigation is performed. Finally, the rover is driven a single step. When VTT is enabled, the VTT routine executes right after Visodom and right before Autonav. The VTT routine specifies the camera pointing and performs the VTT update. The Autonav goal position may optionally be assigned to the

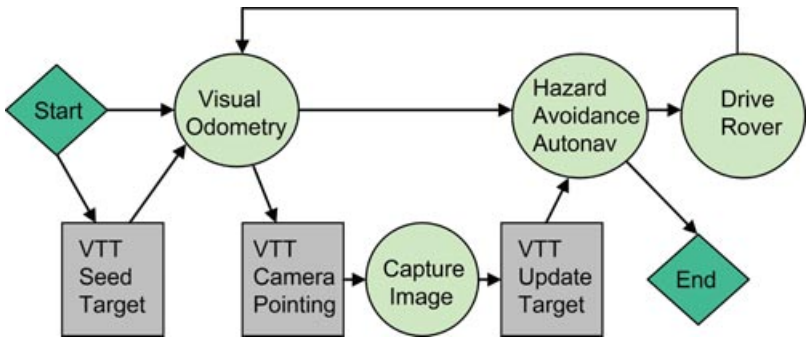


Figure 12. Functional flow of VTT integrated into the MER flight software. The upper portion shows the existing functional flow for one cycle of rover motion with hazard avoidance and visual odometry when enabled. The lower portion shows the insertion of VTT functions right after Visodom and right before Autonav. VTT module functions (boxes) are additions to the existing MER software (circles).

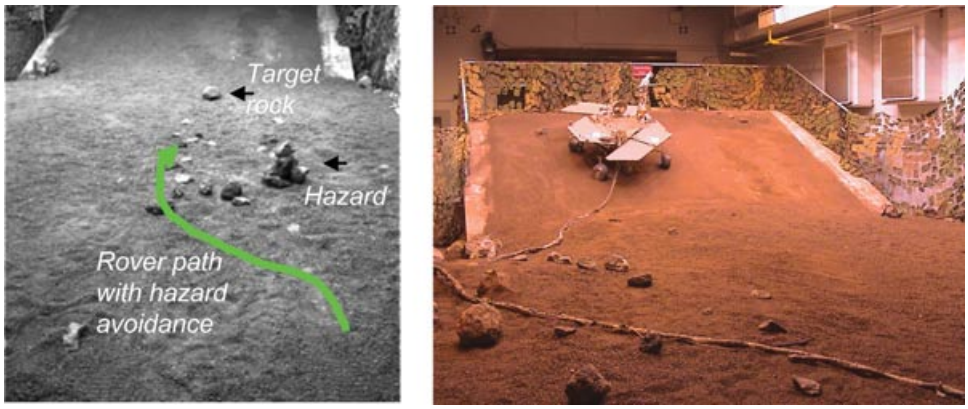


Figure 13. Visual tracking with Autonav rover path with hazard avoidance (left) and MER engineering model driving on a 25-deg ramp (right).

new VTT target position. The actual image capture is done by the IMG (image) module. MER provides the rover pose estimator (Ali et al., 2005) based on wheel odometry and inertial sensing using LN200 inertial measurement unit (IMU). MER also provides visual odometry (Cheng et al., 2005).

A total of 14 regression tests were successfully performed on the MER engineering model in the surface system test bed shown in Figure 13. This included testing visual tracking with blind driving, with visual odometry, and with autonomous navigation using both body-mounted and mast-mounted cameras. In tests in which the target rock was small (less than 10 cm in height), the integrated software had no difficulty driving toward the target. However, in tests in which the target was selected on a large rock, the obstacle avoidance algorithm tended to avoid the target. Therefore, we had to adjust the navigation parameters to steer the rover toward the designated target. Tests were also conducted with the mast-mounted navcams that combined two image wedges. A single-wedge, 45-deg map was not wide enough to find a collision avoidance path when there was an obstacle on the way.

7. DEPLOYMENT ON MARS

Prior to general use by rover planners and scientists, three operational checkouts of VTT were performed on the Opportunity rover roaming along the rim of Victoria Crater on Mars.

7.1. First Operational Checkout

In the first checkout performed on sol 992 (November 8, 2006), VTT was performed only to track the target without controlling the rover's movement. A small, 7-cm-wide rock located about 4 m away from the rover was chosen as the target [Figure 14(a)]. The rover path [Figure 14(b)] on that day was manually commanded to go around the target in seven steps over a total azimuth angle change of 53 deg, while keeping the target distance at about 4 m. The target azimuth angle changed from −3.4 deg (position 1) to 50.1 deg (position 8) relative to the rover. The rover moved straight 42.5 cm each for the first two steps (rover positions 2 and 3) and then 55 cm each for the next three (positions 4, 5, and 6). Thereafter, the rover made a 20-deg turn in place (position 7), followed by a final 1-m straight move (position 8). Figure 15 shows samples of navcam subframe images (161 × 161 pixels) used for VTT. The target window (21 × 21 pixels) overlays clearly indicate that the tracking was successful over all eight images despite dramatic changes in target rock appearance.

Table III shows details on tracking performance. The template image scales were ranged from 0.97 to 1.04, all of which were well within the nominal limit of the ±10% change because the rover path was more or less at the same distance from the target over all seven steps. Changes in the target image roll angle were also small, ranging from 0.06 to 6.25 deg. On the other hand, changes in the target image azimuth angle relative to the rover were much larger, up to 15.44 deg (Table 3). Because VTT does not do

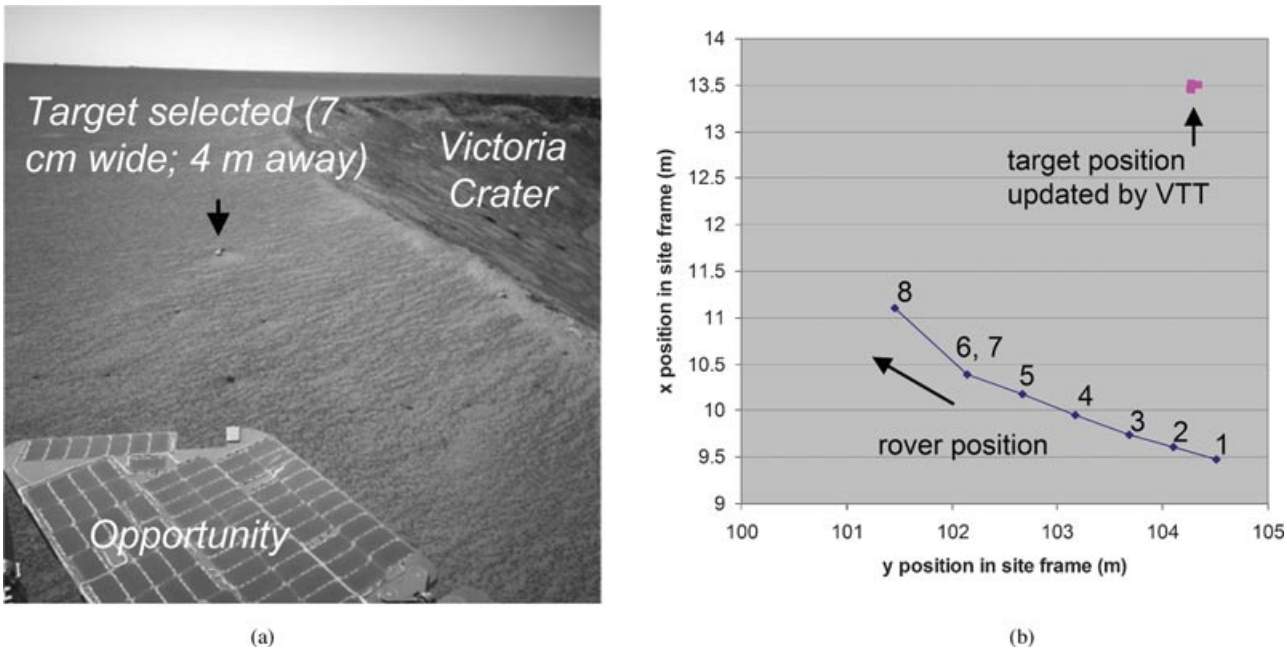


Figure 14. (a) A target was selected at the center of the rock from this full-frame navcam image (45 × 45 deg field of view). (b) Rover path.

azimuth compensation (difficult due to image appearance change), the maximum azimuth change is nominally limited to 0.1 radian (5.7 deg). Despite the excessively large azimuth angle change (Figure 15), visual tracking was successful for all eight images. The NCC match score ranged from 0.79 to 0.94, where the nominal threshold is 0.7. Not surprisingly, the lowest NCC score (0.79) occurred at rover position 8 with the largest azimuth angle change. An unanticipated benefit of the visual tracking capability, which was pointed out by a scientist after this successful run, was the use of the multiple views of a single target for photometric analysis.

7.2. Second Operational Checkout

In the second checkout of sol 1100 (February 26, 2007), a 12-cm-wide rock located about 10 m away from the

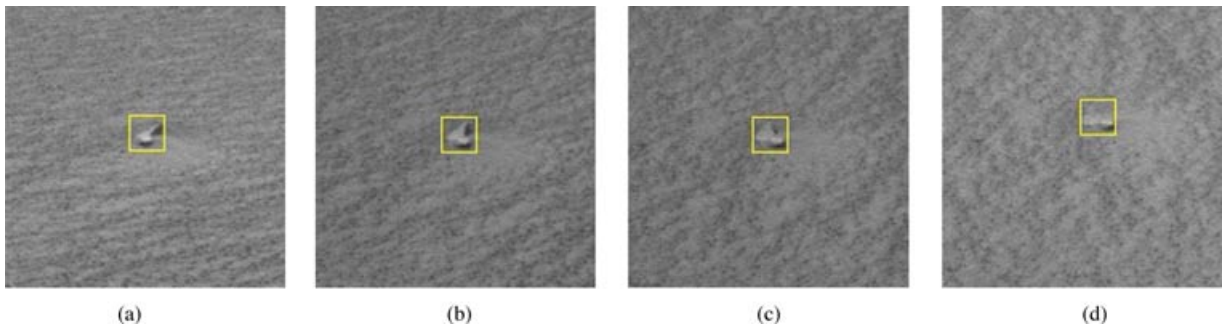


Figure 15. Subframe images of 161 × 161 pixels (7 × 7 deg field of view), taken during the VTT run, with target window overlay: (a) dist = 4.00 m, azimuth = −3.4 deg; (b) dist = 3.67 m, azimuth = 17.5 deg; (c) dist = 3.73 m, azimuth = 34.7 deg; and (d) dist = 3.68 m, azimuth = 50.1 deg.

Table III. VTT details for operational checkout 1.

	Rover position							
Parameter	1	2	3	4	5	6	7	8
Magnification	–	1.04	1.03	1.03	1.01	1.00	0.97	1.04
Roll change (deg)	–	2.46	2.46	2.92	3.27	3.27	0.06	6.25
Azimuth change (deg)	–	6.01	6.38	8.53	8.58	8.62	0.00	15.44
NCC score	–	0.88	0.88	0.85	0.87	0.87	0.94	0.79

rover was chosen as the target [Figure 16(a)]. The Opportunity drove autonomously within 2 m of the target rock in 15 steps [Figure 16(b)] while performing visual tracking at each stop and updating the drive goal. At each stop, the next rover step size was set to 10% of the target distance. The template image magnifications were between 1.09 and 1.13, which were close to 1.1 as expected for the 10% step size. Figure 16(b) also shows the updated target positions in the site (world) frame at all stops. As the rover approaches the target, the stereo range error becomes smaller and the visually tracked target position becomes more accurate. The total change between the initial and final target positions was 36 cm, which was the amount of correction made by visual tracking. Pancam images of the target were also taken at a couple of locations during the drive. Excluding the pancam image collection and wait durations, this 8-m run using VTT took 19 min in total. Figure 17 clearly illustrates that the target rock was tracked very well over all 18 images despite the target rock image being magnified by a factor of 5 in total over the course. The NCC match scores and stereo match scores were all greater than 0.85, well above the 0.7 threshold, indicating good tracking performance.

In this checkout, VTT used the IMU and wheel odometer–based MER pose estimator without visual odometry. If the rover pose estimator and the mast camera pointing were perfect, the target image point would appear right at the center of the image at every stop. Because there is generally some error, the target image will appear a little off from the center. The visual tracking will correct this error by finding the best match of the previous template target image. In this checkout, the maximum correction made by visual tracking was only 11 pixels, whereas the NCC search range was set to ±61 pixels for both rows and columns. This much smaller amount of correction is quite expected because this checkout was done on flat surface with negligible slippage and thus the IMU and wheel odometer–based estimator was fairly accurate.

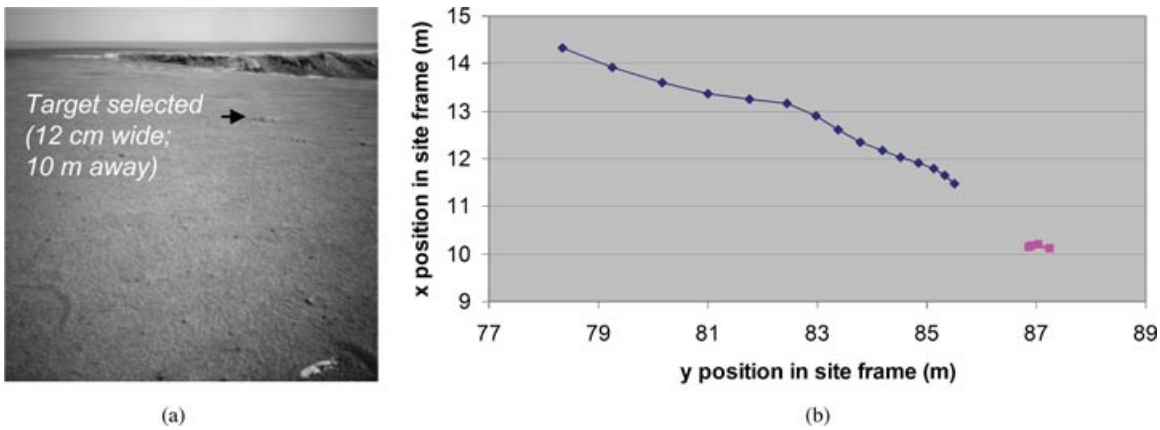


Figure 16. (a) A target was selected from this full-frame navcam image at the center of the rock located at the rim of Victoria Crater. (b) Actual rover path.

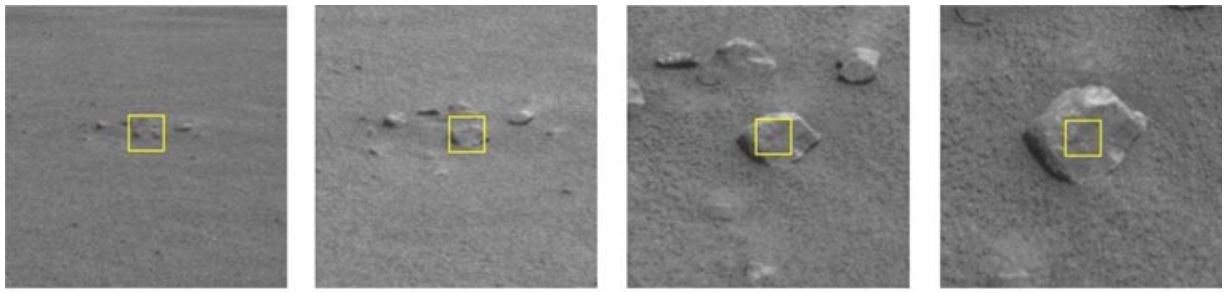


Figure 17. Subframe images of 161 × 161 pixels, taken during the VTT run, with target window overlay: (a) dist = 9.65 m, (b) dist = 6.11 m, (c) dist = 3.05 m, and (d) dist = 1.92 m.

7.3. Third Operational Checkout

The third VTT checkout ran successfully on Opportunity on sol 1194 (June 2, 2007). This checkout involved using VTT with Visodom, followed by using VTT with Autonav. To get enough distance between the rover and the target rock called Paloma [Figure 18(a)], the rover first backed up from Paloma using VTT along the way. After backing up at a 7-m distance [Figure 18(b)], the rover drove toward the target for 2.1 m in four steps using VTT with Visodom. Thereafter, the rover drove 1.4 m further in four more steps using VTT with Autonav. Figure 19(a) shows an example of a navcam image used for Visodom. For a flat terrain with scarce salient features, it is a common MER operational practice to use wheel tracks of the rover as Visodom features. Figures 19(b) and 19(c) show an example of two wedges of navcam images used for Autonav map building. Because hazcam images often had not made good stereo range maps for featureless flat terrain, navcam images were used. However, because a single navcam's field-of-view angle (45 deg) was not wide enough, two wedges were used to build a sufficiently wide terrain map: one for the left-side and the other for the right-side stereo range map. When the first terrain map was built, the

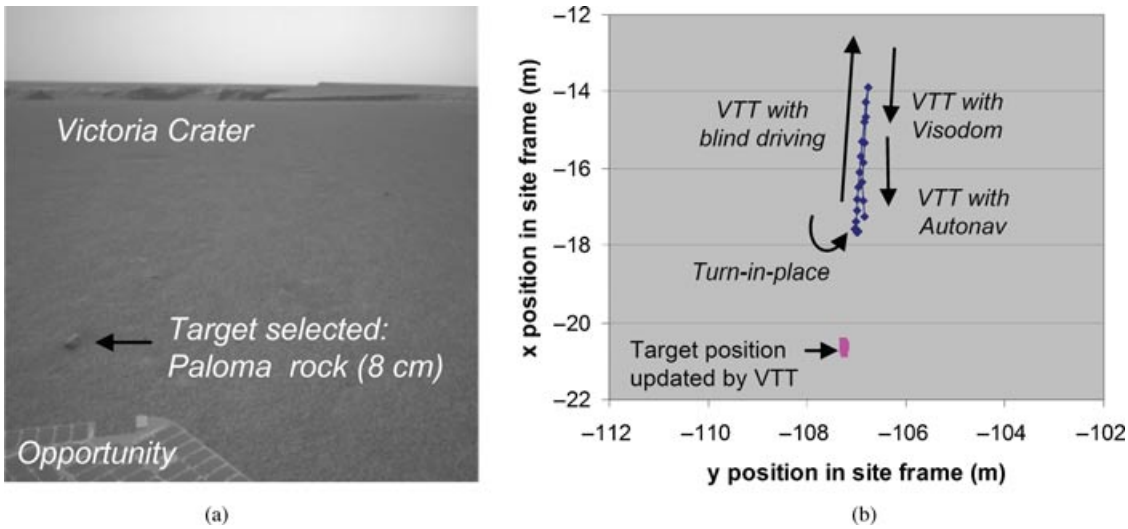


Figure 18. (a) A target was selected from this full-frame navcam image at the center of the rock located at the rim of Victoria Crater. (b) Actual rover path.

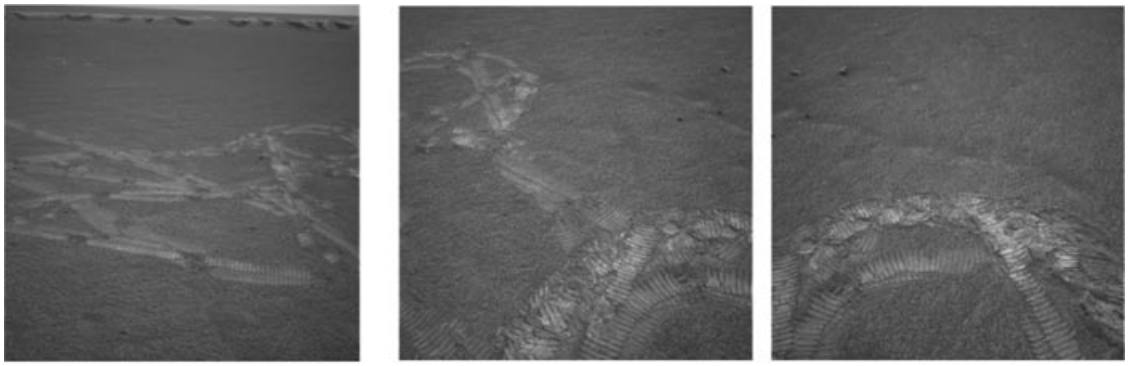


Figure 19. (a) An example of a navcam image used for Visodom. (b) and (c) An example of left-wedge and right-wedge pair of navcam images used for Autonav.

rover needed to move forward at least 2 m more before Autonav could be used, because Autonav would not be able to find a path without a terrain map right underneath the rover.

The operational sequence and execution times are summarized in Table IV. VTT was successful for all motions: turn in place, backing up from the target, VTT with Visodom, and VTT with Autonav (in this checkout, there were no obstacles in the way). The initial 70-deg turn-in-place operation was done in several incremental motions of a 10-deg turn in place combined with a 10-cm alternate back-and-forth step. This is a usual procedure adopted for the Opportunity rover to avoid rover trenching due to its broken front-right steering wheel motor.

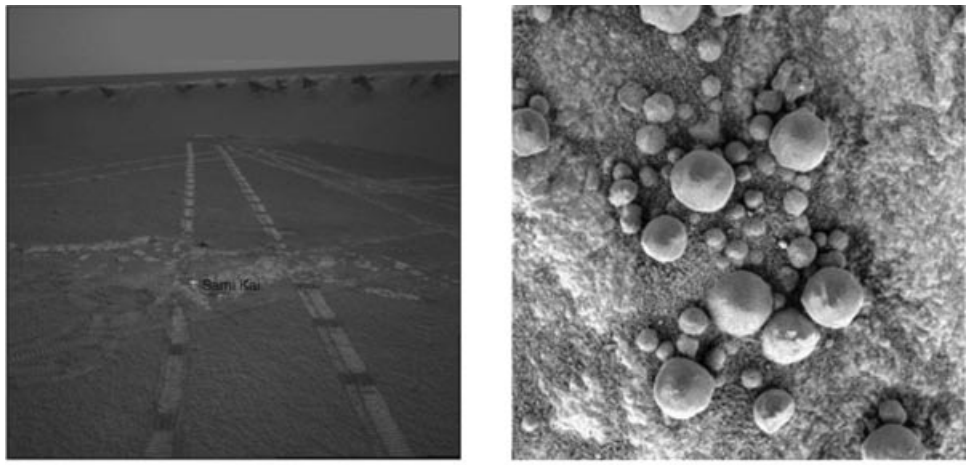
During the backward course, the VTT template image was shrunken prior to NCC match. Each VTT update took 0.5–1 min, among which only 5 s is for computation, 30 s for image capture and transfer of a stereo pair of 12-bit full-frame images, and the rest for other unrelated background tasks. Each Visodom update took 1–2 min, and thus VTT was overall two to three times faster than Visodom. Computationally, VTT was 10–20 times faster than Visodom.

7.4. Single-Sol Instrument Placement

The final VTT checkout was combined with the AutoPlace checkout on Opportunity. The successful completion of this run on sol 1216 (June 23, 2007)

Table IV. Operational checkout 3 summary.

Operational mode	No. of VTT updates (0.5–1 min each)	No. of Visodom updates (1.5–2 min each)	No. of Autonav updates (5 min each)	Total execution duration (min)
VTT seed at 3 m away from target	2			2
Turn-in-place CCW by 10-deg increments	7			10
Move back from 4 to 7 m away	10			11
VTT with Autonav to build initial map			1	5
VTT with Visodom over 2.1 m	4	6		15
VTT with Autonav over 1.4 m	4		4	23
Total	27	6	5	66



(a) (b)

Figure 20. (a) Navcam image used for target selection. (b) MI image of hematite-rich "blueberries" and soil acquired after an autonomous placement on sol 1216 near the rim of Victoria Crater. The patch of terrain viewed in the image is roughly 3 cm.

marks the first single-sol instrument placement on a planetary surface. In this checkout, a true "go-and-touch" was performed by a sequential execution of VTT followed by AutoPlace. A target named Sami Kai was chosen on a slab of bedrock near Victoria Crater from the previous sol's navcam panorama. At the start of the checkout, Sami Kai was 4.7 m away from the rover. The rover performed a VTT-guided drive to within 2 m of the target and then performed AutoPlace to touch a Moessbauer and acquire a single microscopic imager (MI) stack of five images. The AutoPlace software detected potential terrain collisions with some of the tools for the original target and picked a target about 8 mm away. The VTT tracking, handoff of the target from VTT to AutoPlace, and AutoPlace were successful. Figure 20(a) is the navcam image in which the Sami Kai target was picked, and Figure 20(b) shows one of the MI images of Sami Kai.

8. CONCLUSIONS

We successfully infused the targeted driving capability with VTT into the MER flight mission and have used it on Mars. This capability enabled scientists to autonomously drive the rovers to a target with the target positioning accuracy of within a few centimeters. In particular, within the limited computation resources of the planetary rovers, we developed a particular class of the visual tracker that works reliably using NCC with template scale and roll preprocessing based on the available rover pose estimation. Through extensive validation in the JPL Mars Yard and the MER indoor test facility, the visual tracker was successfully deployed on Mars. All four operational checkouts including single-sol instrument placement were successfully executed on the Opportunity rover near Victoria Crater. An unanticipated benefit of the visual tracking capability, which was pointed out by a scientist, was the use of the multiple views of a single target for photometric analysis. The Opportunity is now out of Victoria Crater, and scientists are interested in using the targeted driving capability when an interesting cobble is spotted in the future. This capability will also be available on the future Mars Science Laboratory rover.

ACKNOWLEDGMENT

This work was performed at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration. This work is supported by the NASA Mars Technology Program, Mars Science Laboratory Focused Technology Program, and Mars Exploration Program. The authors would like to acknowledge contributions by Clayton Kunz, Esfandiar Bandari, Maria Bualat, Matt Deans, and Liam Pedersen at NASA Ames Research Center. The authors also would like to thank Mark Maimone for valuable suggestions during the VTT integration into MER,

Richard Volpe for suggesting the template image roll, and Samad Hayati and Gabriel Udomkesmalee for their support. Reviewers' comments greatly enhanced the quality of this paper.

REFERENCES

Ali, K., Vanelli, C., Biesiadecki, J., Maimone, M., Cheng, Y., San Martin, A., & Alexander, J. (2005, October). Attitude and position estimation on the Mars Exploration Rovers. In IEEE Conference on Systems, Man and Cybernetics, The Big Island, HI.

Anandan, P. (1989). A computational framework and an algorithm for the measurement of visual motion. International Journal of Computer Vision, 2(3), 283–310.

Bajracharya, M., Diaz-Calderon, A., Robinson M., & Powell, M. (2005, March). Target tracking, approach, and camera handoff for automated instrument placement. In IEEE Aerospace Conference, Big Sky, MT.

Barnea, D. I., & Silverman, H. F. (1972). A class of algorithms for fast digital image registration. IEEE Transactions on Computing, 21, 179–186.

Bergen, J. R., Anandan, P., Hanna, K. J., & Hingorani, R. (1992, May). Hierarchical model-based motion estimation. In European Conference on Computer Vision, Santa Margherita Ligure, Italy (pp. 237–252).

Biesiadecki, J. J., & Maimone, M. W. (2006, March). The Mars Exploration Rover Surface Mobility Flight Software: Driving ambition. In IEEE Aerospace Conference, Big Sky, MT.

Bouguet, J. (2000). Pyramidal implementation of the Lucas Kanade Feature Tracker: Description of the algorithm. Santa Clara, CA: Microprocessor Research Laboratories, Intel Corporation.

Brown, L. S. (1992). A survey of image registration techniques. ACM Computing Surveys, 24(4), 325–376.

Cheng, Y., Maimone, M., & Matthies, L. (2005, October). Visual odometry on the Mars Exploration Rovers. In IEEE Conference on Systems, Man and Cybernetics, The Big Island, HI.

Ellekilde, L., Favrholdt, P., Paulin, M., & Petersen, H. G. (2007). Robust control for high-speed visual servoing applications. International Journal of Advanced Robotic Systems, 4(3), 279–292.

Gennery, D. (2006). Generalized camera calibration including fish-eye lenses. International Journal of Computer Vision, 68(3), 239–266.

Hayati, S., Rankin, A., Kim, W., Leger, P., Chien, S., & Ali, K. (2007, September). Advanced robotics technology infusion to the NASA Mars Exploration Rover (MER) Project. In IFAC Symposium on Intelligent Autonomous Vehicles, Toulouse, France.

Hutchinson, S., Hager, G., & Corke, P. (1996). A tutorial on visual servo control. IEEE Transactions on Robotics and Automation, 12(5), 651–670.

Kim, W. S., Ansar, A., & Madison, R. (2006, September). Improvement in visual target tracking for a mobile robot. NASA Tech Briefs.

Kim, W. S., Ansar, A., & Steele, R. (2005, July). Rover mast calibration, exact camera pointing, and camera handoff for visual target tracking. In International Conference on Automation and Robotics (ICAR), Seattle, WA.

Kim, W. S., Ansar, A., Steele, R., & Steinke, R. (2005, October). Performance analysis and validation of a stereo vision system. In IEEE Conference on Systems, Man and Cybernetics, The Big Island, HI.

Kim, W. S., Steele, R., Ansar, A., Ali, K., & Nesnas, I. (2005, August). Rover-based visual target tracking validation and mission infusion. In AIAA Conference, Space 2005, Long Beach, CA.

Kim, W. S., Steele, R., Ansar, A., Ator, D. E., & Catron, S. S. (2006). Visual target tracking technology validation report (JPL internal document, JPL D-33416). Pasadena, CA: Jet Propulsion Laboratory.

Litwin, T., & Maki, J. (2005, October). Imaging services flight software on the Mars Exploration Rovers. In IEEE Conference on Systems, Man and Cybernetics, The Big Island, HI.

Lowe, D. (2004). Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision, 60(2), 91–110.

Lucas, B. D., & Kanade, T. (1981, August). An iterative image registration technique with an application to stereo vision. In 7th International Joint Conference on Artificial Intelligence (IJCAI), Vancouver, Canada (pp. 674–679).

Maimone, M., Nesnas, I., & Das, H. (1999, June). Autonomous rock tracking and acquisition from a Mars Rover. In International Symposium on Artificial Intelligence and Robotics in Space (i-SAIRAS), Noordwijk, The Netherlands.

Maimone, M. W., Leger, P. C., & Biesiadecki, J. J. (2007, April). Overview of the Mars Exploration Rovers' autonomous mobility and vision capabilities. In IEEE International Conference on Robotics and Automation (ICRA) Space Robotics Workshop, Rome, Italy.

Matthews, I., Ishikawa, T., & Baker, S. (2004). The template update problem. IEEE Transactions on Pattern Analysis and Machine Intelligence, 26(6), 810–815.

Nesnas, I. A., Bajracharya, M., Madison, R., Bandari, E., Kunz, C., Deans, M., & Bualat, M. (2004, March). Visual target tracking for rover-based planetary exploration. In IEEE Aerospace Conference, Big Sky, MT.

Nesnas, I. A., Maimone, M., & Das, H. (1999, November). Autonomous vision-based manipulation from a rover platform. In IEEE Symposium on Computational Intelligence in Robotics & Automation, Monterey, CA.

Nesnas, I. A., Simmons, R., Gaines, D., Kunz, C., Diaz-Calderon, A., Estlin, T., Madison, R., Guineau, J., McHenry, M., Shu, I., & Apfelbaum, D. (2006). CLARAty: Challenges and steps toward reusable robotic software. International Journal of Advanced Robotic Systems, 3(1), 23–30.

Nesnas, I. A., Wright, A., Bajracharya, M., Simmons, R., Estlin, T., & Kim, W. (2003, April). CLARAty: An architecture for reusable robotic software. In SPIE Aerosense Conference, Orlando, FL.

Pederson, L., Bualat, M., Smith, D. E., & Washington, R. (2003, May). Integrated demonstration of instrument placement, robust execution and contingent planning. In International Symposium on Artificial Intelligence and Robotics in Space (i-SAIRAS), Nara, Japan.

Pratt, W. K. (1974). Correlation techniques of image registration, IEEE Transactions on Aerospace and Electronic Systems, 10, 353–358.

Reeves, G. (2005, October). Mars Exploration Rovers flight software. In IEEE Conference on Systems, Man and Cybernetics, The Big Island, HI.

Rosenfeld, A., & Kak, A. C. (1982). Digital picture processing. New York: Academic Press.

Rosenfeld, A., & Vanderbrug, G. J. (1977). Coarse-fine template matching, IEEE Transactions on Systems, Man, and Cybernetics, 7, 104–107.

Shi, J., & Tomasi, C. (1994, June). Good features to track. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR'94), Seattle, WA (pp. 593–600).

Urmson, C., Simmons, R., & Nesnas, I. (2003, March). A generic framework for robotic navigation. In IEEE Aerospace Conference, Big Sky, MT.

Volpe, R., Nesnas, I. A., Estlin, T., Mutz, D., Petras, R., & Das, H. (2000). CLARAty: Coupled layer architecture for robotic autonomy (Tech. Rep. JPL D-19975). Pasadena, CA: Jet Propulsion Laboratory.

Wettergreen, D., Thomas, H., & Bualat, M. (1997, September). Initial results from vision-based control of the Marsokhod Rover. In IEEE International Conference on Intelligent Robots and Systems (IROS), Grenoble, France (pp. 1377–1382).

Yilmaz, A., Javed, O., & Shah, M. (2006). Object tracking: A survey. ACM Computing Surveys, 38(4).

Zitova, B., & Flusser, J. (2003). Image registration methods: A survey. Image and Vision Computing, 21(11), 977–1000.