Machine Vision and Applications (2010) 21:713–725
DOI 10.1007/s00138-009-0188-9

ORIGINAL PAPER

Wide-baseline stereo vision for terrain mapping

Clark F. Olson · Habib Abi-Rached

Received: 11 January 2008 / Revised: 30 July 2008 / Accepted: 7 January 2009 / Published online: 6 February 2009

Abstract Terrain mapping is important for mobile robots to perform localization and navigation. Stereo vision has been used extensively for this purpose in outdoor mapping tasks. However, conventional stereo does not scale well to distant terrain. This paper examines the use of wide-baseline stereo vision in the context of a mobile robot for terrain mapping, and we are particularly interested in the application of this technique to terrain mapping for Mars exploration. In wide-baseline stereo, the images are not captured simultaneously by two cameras, but by a single camera at different positions. The larger baseline allows more accurate depth estimation of distant terrain, but the robot motion between camera positions introduces two new problems. One issue is that the robot estimates the relative positions of the camera at the two locations imprecisely, unlike the precise calibration that is performed in conventional stereo. Furthermore, the wide-baseline results in a larger change in viewpoint than in conventional stereo. Thus, the images are less similar and this makes the stereo matching process more difficult. Our methodology addresses these issues using robust motion estimation and feature matching. We give results using real images of terrain on Earth and Mars and discuss the successes and failures of the technique.

Keywords Stereo vision · Wide-baseline matching · Terrain mapping · Motion estimation

1 Introduction

Terrain mapping is critical to mobile robot navigation in outdoor environments. If a robot cannot sense the pathways and obstacles to the goal location, then considerable trial-and-error may be performed before finding a navigable route. Maps are also useful for performing localization of the robot and finding the goal, particularly when Global Positioning System (GPS) data is not available.

We are interested in performing terrain mapping for the robotic exploration of Mars. An important goal for Mars exploration is to maximize the amount of scientific data returned during a mission. The amount of such data is increased if a rover is able to traverse to distant science targets in a single command cycle. However, this requires the rover to be able to navigate to science targets seen in orbital images, but not previously seen by the rover. Inaccurate navigation to these goals leads to a reduction in the time for gathering scientific data, since communications from Earth are usually received only once per day. When the rover fails to correctly reach the target, it must again be commanded to traverse to the correct location.

Previous work [16,31,35,49] has addressed several issues in mapping and localization for rovers. However, this work has not solved the problem of on-board mapping of distant terrain by a rover, which would allow a rover to improve its long-range planning and localization capabilities. Current rovers use stereo vision for mapping nearby terrain, but

Portions of this work have previously appeared in the Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems [34] and in the Proceedings of the 12th International Conference on Advanced Robotics [33]. A summary of an early version of this work appears in [36].

C. F. Olson (✉)
Computing and Software Systems, University of Washington,
Box 358534, 18115 Campus Way NE, Bothell, WA 98011, USA
e-mail: cfolson@u.washington.edu

H. Abi-Rached
Department of Electrical Engineering, University of Washington,
Box 352500, 253 EE/CSE Building, Seattle, WA 98195, USA

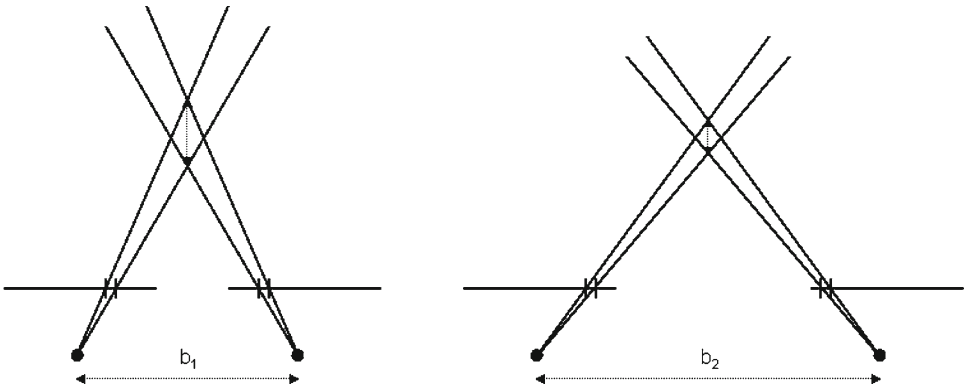

Fig. 1 As the baseline distance increases, the stereo range error is decreased, assuming that the error in estimating the disparity does not increase linearly with the baseline distance

the accuracy of such techniques degrades with the square of the distance to the terrain. This can be improved through the use of a larger baseline distance (the distance between the stereo cameras). On the other hand, a rover cannot have two cameras with a large, fixed baseline distance because of the limited size of the rover.

Rather than using two images taken by different cameras at the same time, wide-baseline stereo vision uses two images taken by the same camera at different times. The wide baseline can improve the quality of the stereo range data for distant terrain. This can be observed by noting that the stereo disparity d is inversely proportional to the depth of the point z and directly proportional to the baseline distance b. So, $z = cb/d$, where c is constant. In practice, the disparity must be estimated with some error and this yields error in the estimate of the distance to the point. However, as the baseline is increased, the disparity increases proportionally and this causes a reduction in the effect of the error in the disparity estimate. This effect is illustrated in Fig. 1.

However, two new problems are introduced. First, we can no longer carefully calibrate the difference in location and orientation between the camera positions, since they are not fixed as in conventional stereo vision. These positions are only known up to the accuracy of the rover localization as it moves in the terrain. In addition, finding the stereo correspondences between the images is more difficult, since the terrain is seen with a larger difference in perspective than in conventional stereo.

Our methodology addresses these problems. We perform the following steps:

1. *Motion refinement*: We refine an estimate of the camera position using matches between image features detected in the images (an initial estimate is not crucial, but it is typically available from rover odometry). We do not assume that the initial estimate is precise. A refinement step is performed in order to improve the relative camera position estimate, consisting of the following substeps:
   (a) *Feature selection*: We detect features in the first image that can be precisely localized using a method based on the approaches of Förstner and Gülch [3] and Shi and Tomasi [41].
   (b) *Feature matching*: For each feature, we look for a corresponding feature in the second image using a hierarchical search over the entire image with multiple candidates at each level. A high-pass filter is applied to both images prior to matching to remove most illumination effects.
   (c) *Outlier rejection*: We reject correspondences where the estimated precision is low, multiple candidates are similar, or the disparity is not in agreement with other matches.
   (d) *Motion estimation*: The Levenberg-Marquardt algorithm [38] is used to optimize the motion parameters with respect to the epipolar constraints for the detected correspondences. A robust objective function is used to minimize the distances between the matched feature locations in the second image and the locations reprojected from the first image into the second image using the estimated motion parameters.
2. *Image rectification*: We rectify the images using the algorithm of Fusiello et al. [5] so that the correspondences between the images lie along the same image row. This allows the disparity for each pixel to be computed efficiently.
3. *Stereo matching*: We use our maximum-likelihood image matching measure [32] to compute disparity estimates for each pixel in the first image. The search efficiency is improved by eliminating redundant computations between neighboring pixels. We estimate the subpixel disparity value and the precision of the disparity by fitting a parameterized surface the estimated likelihoods. Outliers are rejected using the precision estimates and by eliminating small regions of the disparity image that are inconsistent with surrounding estimates.

4. *Triangulation*: The image disparities are used to triangulate the three-dimensional position of each location in the terrain.

While we discuss novel techniques as a part of the components of our system, the primary purpose of this work is to describe a complete system for wide-baseline stereo vision. Given maps created at several overlapping locations, several methods can be used to merge them into a larger encompassing map [9,36,37,44]. The following sections describe the above steps in further detail and present experiments testing the efficacy of the methodology.

2 Previous work

Indoor mapping has received considerable attention in the robotics community. Thrun [44] gives a good review of such work. Outdoor mapping has received much less attention. Most work on outdoor mapping has concentrated on stereo vision or laser rangefinders, owing to the complex terrain, but other sensors have been used.

Early work at Carnegie Mellon University used scanning laser rangefinders to perform outdoor mapping [7,12–14]. Gennery [6] and Matthies [23,24] used stereo vision data for similar purposes at the Jet Propulsion Laboratory (JPL).

More recent techniques deal with highly complex data. Maimone et al. [19] used trinocular stereo vision to map difficult terrain for use in the Chornobyl nuclear facility. Huber and Hebert [9] built maps for large data sets of unstructured terrain with widely varying resolution. Mandelbaum et al. [21] constructed maps with single or stereo cameras over long distances using structure and motion estimation. Li et al. [17] generated maps of Mars using data acquired by the Spirit and Opportunity rovers.

Terrain mapping has also been performed using aerial images. Hung et al. [10] generated terrain models from overlapping aerial images. At the National Center for Scientific Research in France (CNRS), an autonomous blimp was used to acquire stereo imagery for mapping [11,15]. Xiong et al. [49] developed methods to generate maps from the images captured as a spacecraft descends to a planetary body. Montgomery et al. [27] generated maps using an autonomous helicopter for safe landing. Williams et al. [47] used sonar for mapping and navigation in underwater robotics.

Wide-baseline stereo (also called motion stereo [8,29]) can map terrain that is more distant than conventional stereo, because the larger baseline allows improved triangulation. Much recent work in this area has considered the problem of finding correspondences between the wide-baseline images [1,18,22,25,39,40,46]. In contrast, our work uses a relatively simple method for finding correspondences and concentrates on building a system capable of generating dense and accurate depth maps from wide-baseline images. The use of an alternative method for detecting correspondences has the potential for improving our system.

Other interesting work includes the method of Strecha et al. [42] for dense mapping of wide-baseline images. This technique generates excellent results, but is not suitable for a mobile robot because of the required processing time. Finally, Okutomi and Kanade [30] describe a system for generating stereo data using a set of images with different baselines. However, the baselines in this work are not wide.

3 Feature selection and matching

The first step in accurately recovering the terrain map is to refine the estimated motion between the camera positions. An initial estimate is typically given by rover odometry or other localization methods. This estimate can be refined using motion estimation, if we know correspondences between the images. This section is concerned with locating such correspondences. Since the images are not captured at the same time, the illumination of the terrain may be different in the two images. In order to remove most of the effects of such changes, we convolve both images with a high-pass filter, replacing each pixel with the deviation from the average local brightness.

It is impractical to determine correspondences for every location in the observed terrain for several reasons. Some locations that can be seen in one image will not be observed in the other. Correspondences for other locations cannot be found precisely, since the terrain has a uniform appearance. Furthermore, the processing time needed to detect correspondences for each location would be prohibitive. We select 256 locations with distinctive appearance from the first image to match with the second image. The distinctiveness of the feature is determined using a method based on the similar interest operators developed by Förstner [3] and Shi and Tomasi [41]. The operator scores each pixel based on the strength of the gradients in a neighborhood around the pixel. For a pixel to score highly, the neighborhood must have strong gradients with multiple orientations. Requiring the gradients to have multiple orientations allows linear edges to be discarded, which is desirable, since it is hard to localize features along the edge. This operator examines the covariance matrix of the local gradients. The larger eigenvalue of the covariance matrix is an estimate of the strength of the gradients in the neighborhood and the ratio of the eigenvalues measures the degree to which they are in multiple orientations. Following Shi and Tomasi [41], we score each pixel according to value of the smaller eigenvalue.

Once the image locations have been scored, we divide the image into an even grid of 16 sub-images and select local maxima within each sub-image. This ensures that we select

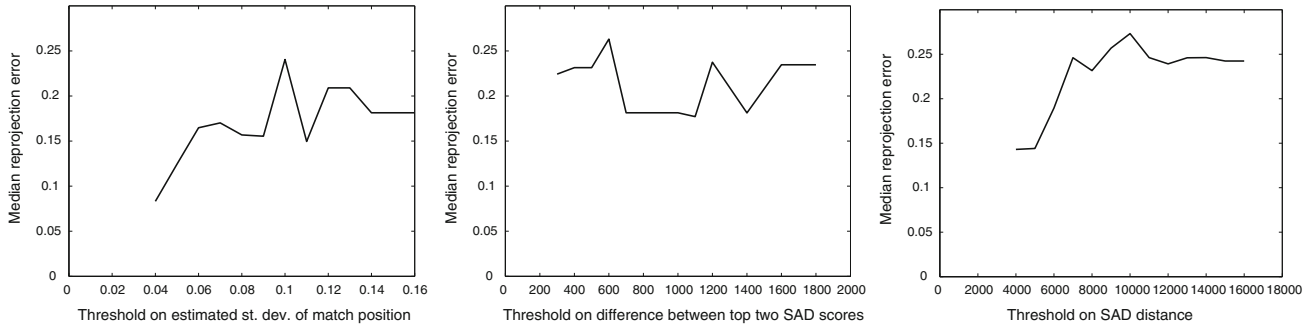

Fig. 2 Experiments indicate that the methodology is insensitive to the thresholds chosen for accepting candidate feature matches. **a** Plot of the median reprojection error versus the threshold on the estimated standard deviation of match position. **b** Plot of the median reprojection error versus the threshold on the SAD distance between top two matches. **c** Plot of the median reprojection error versus the threshold on the SAD score of top match

features from all parts of the image. However, the maxima in each sub-image are subject to a threshold, since we do not want featureless sub-images to contribute to the set of selected features.

For each selected feature, a match is sought in the second image using a coarse-to-fine strategy. The method first detects candidate matches in an image that has been downsampled by four in both rows and columns (the Mars rover images that we test on are $1{,}024 \times 1{,}024$ pixels, but other image sizes are also used). We perform matching using the sum-of-absolute-differences (SAD) distance over a large window (21×21) so that considerable image context is incorporated. However, the large context can lead to a large SAD distance between image patches, even for correct matches, if there is a change in appearance between images. For this reason, multiple candidate matches are selected at the low resolution if the SAD distance is not more than 50% larger than the distance for the best candidate (up to a maximum of ten candidates).

Each candidate match is refined at the highest resolution using a small (9×9) search space and candidates are removed if they do not remain within 50% of the best SAD distance. Finally, candidates undergo an (optional) affine refinement step that uses iterative optimization to find the best fit for each candidate match over linear transformations. We store the best candidate if it meets four criteria (the others are discarded as unreliable):

1. The estimated standard deviation in the position of the localized match is <0.15 pixels.
2. The difference in the SAD distance between the best match and the second best is $\geq$800.
3. The SAD distance is <12,000 (for a 21×21 window).
4. The vertical disparity of the match is consistent (up to a threshold) with the median disparity of the other candidates meeting the previous criteria.

Experiments varying these thresholds (see Fig. 2) indicate that the technique is not sensitive to the precise values of these thresholds, owing to the consistency check and the robust objective function used in the motion estimation. The low tails in two of the experiments at certain thresholds is because many of the matches were pruned and overfitting of the remaining matches was possible. Note that these criteria will often throw away correct matches in addition to incorrect matches. It is important that most incorrect matches are discarded while keeping sufficient correct matches to perform motion estimation.

Figure 3 shows an example of selected and matched features using these techniques. In this case, 256 features were selected in the first image. Of these, candidate matches were found for 74 features in the second image. After outlier rejection (which discards mostly correct matches in this case), 65 correct matches remained to be used as input to the motion refinement step.

4 Motion estimation

Given correspondences between the stereo images, we can update the estimated motion between the camera positions by enforcing geometric constraints on the correspondences [43,49]. This optimization is over the rotation R and translation T between the camera positions C_1 and C_2, which include the position and orientation.

$$C_2 = RC_1 + T. \tag{1}$$

In estimating the motion, only five of the six motion parameters (three for translation and three for rotation) can be estimated. The distance between the camera positions (i.e., the baseline distance) cannot be recovered, since there is a scaling of the entire scene that would yield the same image pair for any such baseline distance. We use the rover onboard

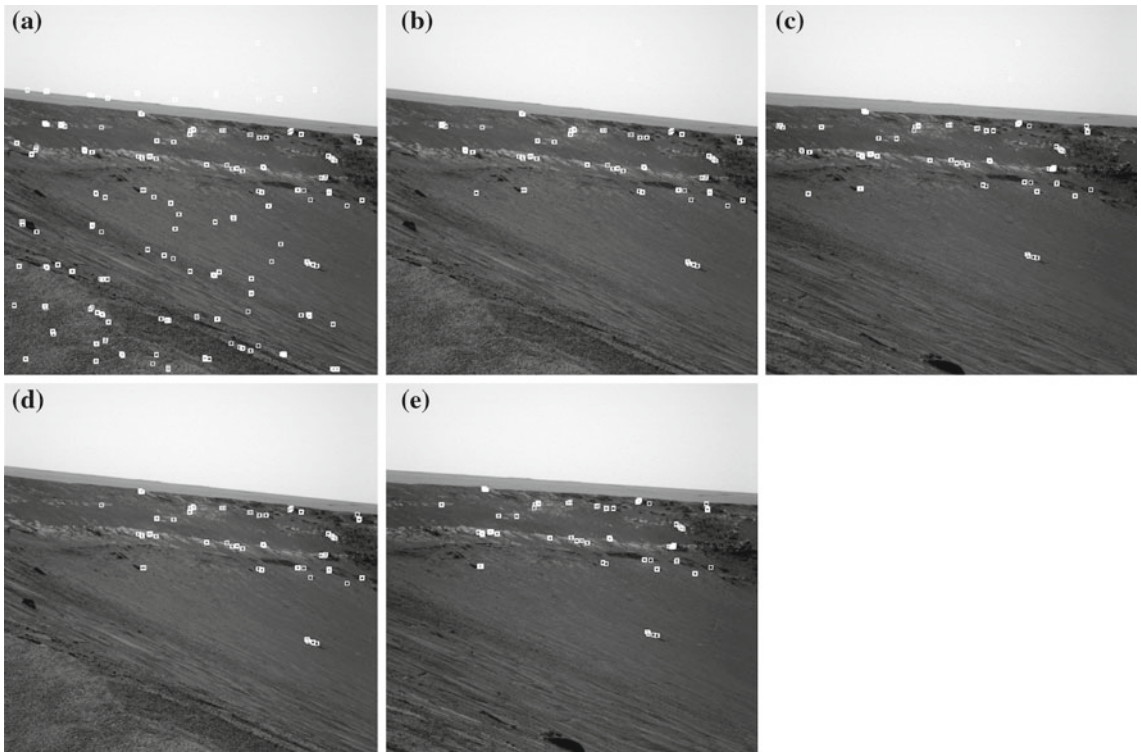

Fig. 3 Feature matching example. **a** Features selected in the first image. **b** Initial features matched in first image. **c** Initial features matched in second image. **d** Matched features in the left image after outlier rejection. **e** Matched features in the right image after outlier rejection

estimate for the positions where the images were captured to compute the baseline distance. This relies on the rover localization capabilities [16,17,20,31] in order to estimate the distance that the rover has traveled between capturing the images.

In the process of optimizing the motion estimate, we use a state vector that includes the five recoverable motion parameters and an estimate of the depth to each of the recovered features (relative to the first camera position). We initialize the depths to values lower than the expected distance to the terrain. This has led to good results in our experiments. We have noticed that the optimization has converged to a local minimum occasionally when the depths are initialized to larger values.

The objective function that we use is an M-estimator that robustly combines the squared error for each feature correspondence, where the feature error is the distance between the feature position detected in the second image (c_i, r_i) and the estimated position in the second image $(\hat{c}_i, \hat{r}_i)$ calculated by reprojecting the feature from the first image into the second image according to the current motion estimate and estimated feature depth.

$$D_i = \sqrt{(c_i - \hat{c}_i)^2 + (r_i - \hat{r}_i)^2}. \tag{2}$$

We use a variation of the M-estimator discussed in [4] to combine the distances as follows:

$$\sum_{i=1}^{N} \frac{\sigma^2 D_i^2}{\sigma^2 + D_i^2}, \tag{3}$$

where $\sigma = \text{median}(D_1, \ldots, D_N)$ and N is the number of correspondences. The objective function is optimized using the Levenberg-Marquardt method [38].

When we applied this technique to the images in Fig. 3, the average reprojection error per feature was 0.096 pixels. In a series of more difficult tests, the median reprojection error was 0.15 pixels. The techniques were also tested on a data set with a known homography between image pairs [26]. In this experiment, we compared the locations that the points were predicted to be projected according to the motion estimate with the location specified by the homography. Figure 4 shows the results. The histograms indicate that the errors are usually small.

5 Rectification

The location of any position in the terrain that is seen in the first image is constrained to lie along a line in the second image. This is called the epipolar constraint and the line is called the epipolar line. If the camera parameters and the motion between the images are known, then we can make use of this constraint. The location of the position in the second image along this line depends on the

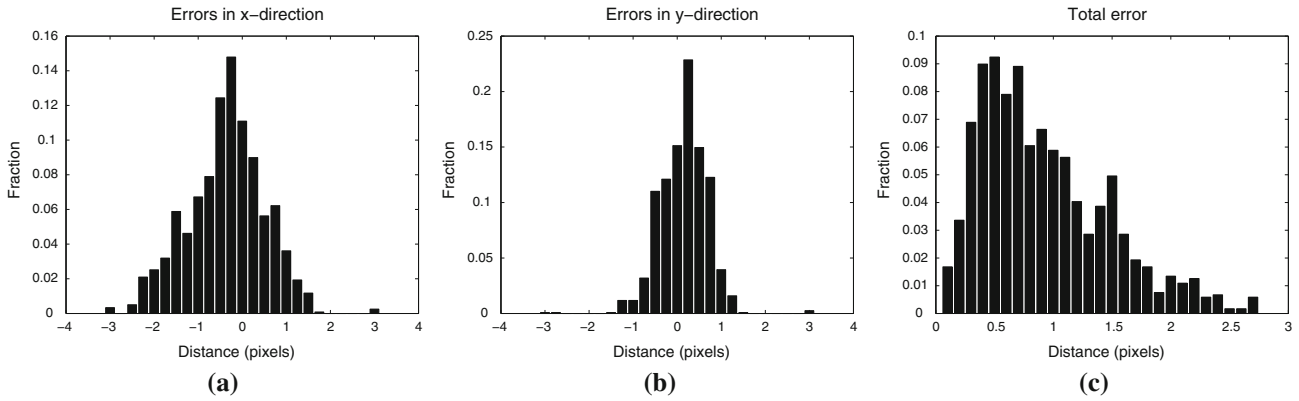

Fig. 4 Motion errors computed by comparing the projected feature locations according to the estimated motion to the location predicted by the known homography. **a** Histogram of errors in the x-direction. **b** Histogram of errors in the y-direction. **c** Histogram of overall error distances

distance of the terrain from the camera. A common and useful trick in stereo vision is to rectify the images to appear as if they were captured by cameras with horizontal axes that are parallel to the baseline and vertical axes that are perpendicular to both the baseline and the optical axes. This corresponds to a virtual rotation of each camera about its center of projection and can be achieved by a linear transformation of the image (this assumes a pinhole camera model. In practice, most lens distortion can be removed prior to this step, if a model of the distortion is computed in advance). When this rectification has been performed, all of the epipolar lines become horizontal and they lie along the same image row as the corresponding image feature. This allows efficient algorithms to be used to find the stereo correspondences.

We use the method of Fusiello et al. [5] to perform rectification. If R_1 and C_1 are the camera rotation matrix and center of projection for the first image in the world coordinate frame, then a point at $[x\ y\ z]^T$ is transformed according to the perspective projection into an image point $[u_1\ v_1]^T$ by taking the intersection of the image plane with the line between the point and C_1.

To accomplish the rectification, we must determine the rotation that brings the cameras into the correct configuration. The translation does not change. After the rectifying transformation, both cameras will have the same rotation matrix R, since they will have the same (virtual) orientation. The epipolar lines will be horizontal if the baseline is parallel to the x-axis in the camera reference frame.

We achieve the rectifying transformation by setting the rotation row vectors as follows ($R = [r_1\ r_2\ r_3]^T$):

1. r_1 is a unit vector in the direction of $C_1 - C_2$.
2. r_2 is a unit vector orthogonal to r_1 and to the z-axis in the local frame of reference of camera 1.
3. r_3 is a unit vector perpendicular to both r_1 and r_2.

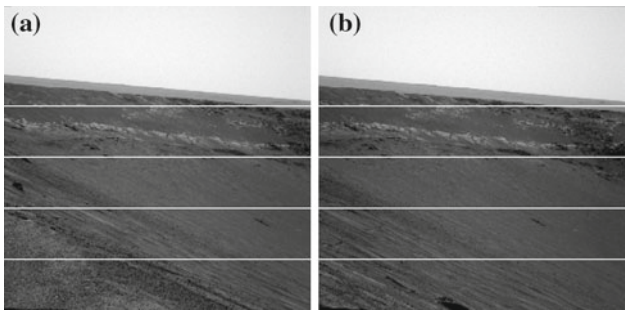

Fig. 5 Images after rectification. Lines have been added to show the relative position of features in the two images. **a** Left image. **b** Right image

With the new rotation matrix, the camera projection matrices become

$$P_i = A[R\ -RC_i] = [Q\ q_i], \tag{4}$$

where A represents the camera intrinsic parameters, $Q = AR$ and $q_i = -ARC_i$. The rectifying transformations are given by

$$T_i = Q(AR_i)^{-1}. \tag{5}$$

We apply these linear transformations to the stereo images. This sometimes transforms the image pixels such that they are moved outside of the image (for example, if the original forward axes converged). We recenter the images in order to retain as much data as possible within the image boundaries by moving the center of gravity of the transformed pixel locations to the center of the image. Figure 5 shows the images from Fig. 3 after rectification. The lines drawn on the image are on corresponding rows. It can be seen that corresponding features lie along the same rows in the images.

6 Stereo matching

In wide-baseline stereo matching, there is often a large (positive or negative) disparity between corresponding points in the two images. This necessitates a large search space in the horizontal dimension for the correct match. However, after the image rectification we do not need to search in the vertical direction. This allows us to combine efficient stereo matching techniques with a robust image matching measure.

To find the correspondences, we use an area-based strategy, where a small neighborhood around each pixel in the first image is used to compare against small neighborhoods in the second image. In our experiments, the use of a simple measure such as sum-of-squared-differences (SSD), SAD, or normalized correlation produced poor results. We use a maximum-likelihood image matching strategy that produces better results [32]. Typical measures compare only the pixels that are aligned between the neighborhoods in the two images. For example, the SSD squares the differences between these pixels and sums the results. The maximum-likelihood matching strategy allows matches between pixels that are not directly overlapping, if the intensities are sufficiently similar, using a pixel similarity measure that combines the difference in intensity with distance between them. To compute the measure, pixels in the images are considered to be vectors in the three-dimensional space spanning the pixel row, column, and intensity. However, a difference of one unit in intensity is not as important as a difference of one unit in the row or column of the pixel, so a weighting factor K is used to discount the distance in intensity. Let

$$v_i = \begin{bmatrix} p_i^{(\text{row})} \\ p_i^{(\text{column})} \\ K p_i^{(\text{intensity})} \end{bmatrix}. \tag{6}$$

For simplicity, the distance between p_i and p_j is computed with the L_1 norm:

$$d(p_i, p_j) = \left|\left| v_i - v_j \right|\right|_1. \tag{7}$$

We have found empirically that $K = 1/4$ works well. However, the method is not sensitive to this value.

Now, let $d^{\delta}_{r,c}$ be the distance (computed with respect to the above definition) from the pixel at (r, c) in the first image to the closest pixel in the second image when it is displaced by disparity δ. This can be computed efficiently using a three-dimensional distance transform [31]. The likelihood function over the neighborhood of a pixel at (r, c), given the disparity is

$$L^{\delta}(r, c) = \prod_{i=-w}^{w} \prod_{j=-w}^{w} f(d^{\delta}_{r+i,c+j}), \tag{8}$$

where $f(\cdot)$ is the probability density function (PDF) of the distances and w determines the size of the neighborhood (this assumes independence between the distances, which works well in practice). We use a PDF that is a weighted mixture of a Gaussian for inliers and a constant for outliers [32].

In order to perform dense matching between the rectified images using the measure described above, we use an efficient search strategy common in stereo vision [2,28]. This strategy makes use of the observation that a brute-force implementation performs many redundant computations for adjacent positions of the template at the same disparity. We eliminate the redundant computation by storing the information for reuse as necessary for fast matching.

Given previously computed values $L^{\delta}(r-1, c)$, $L^{\delta}(r, c-1)$, and $L^{\delta}(r-1, c-1)$, we can compute $L^{\delta}(r, c)$ efficiently with the following formulas:

$$r_+ = r + w \tag{9}$$

$$r_- = r - w - 1 \tag{10}$$

$$c_+ = c + w \tag{11}$$

$$c_- = c - w - 1 \tag{12}$$

$$L^{\delta}(r, c) = \frac{L^{\delta}(r-1, c) L^{\delta}(r, c-1) f(d^{\delta}_{r_-,c_-}) f(d^{\delta}_{r_+,c_+})}{L^{\delta}(r-1, c-1) f(d^{\delta}_{r_-,c_+}) f(d^{\delta}_{r_+,c_-})} \tag{13}$$

In practice, it is faster to compute $\log L^{\delta}(r, c)$, since $\log f(\cdot)$ can be precomputed for the necessary values and this results in only the use of addition and subtraction (rather than multiplication and division).

We select the disparity that maximizes the likelihood function as the candidate disparity for each pixel. A subpixel disparity estimate and an estimate of the standard deviation of the error for each pixel are computed by fitting a curve to the likelihood scores near the maxima [31]. The disparity is discarded if the standard deviation is too large or if the overall likelihood of the location is not large enough. A final outlier rejection step is used that discards any disparities that do not form a large enough coherent block in the output.

Figure 6 shows the disparity map that was computed for the example in the previous figures, which show the Endurance crater on Mars imaged by the Opportunity rover. Dense results are achieved for the far side of the crater where it appears in both of wide-baseline images. Results on the left side are correctly pruned, since this part of the crater is not visible in both images. Data at the bottom is pruned either for this reason or because the change in appearance is large. Figure 7 shows a second example, which is a conventional stereo pair of Mars. Dense results are obtained over most of the image in this case, since the cameras were pointing in nearly the same direction and the baseline was small. Data is correctly pruned from the left side of the image, since this region is not visible in both images. The results are very close to those obtained by conventional stereo. This is an encouraging result, since conventional stereo benefited from the

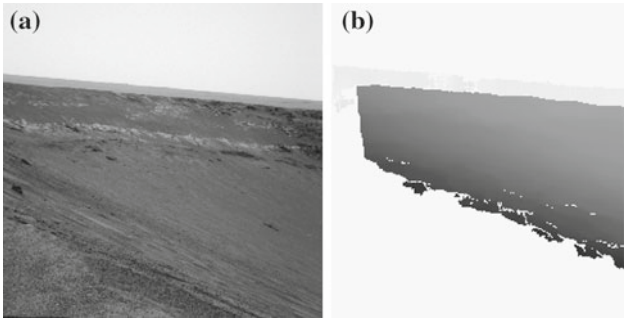

Fig. 6 Computed disparity map for Endurance crater wide-baseline stereo pair. **a** Left image. **b** Disparity image. Dark values represent larger disparities

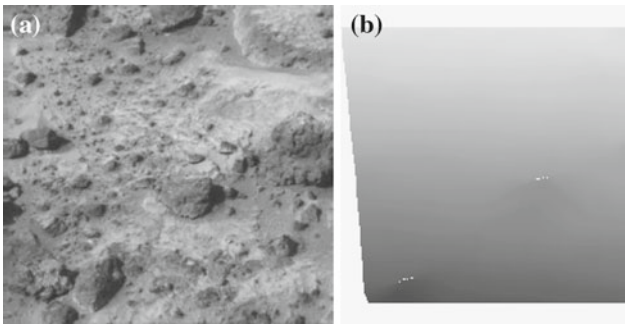

Fig. 7 Computed disparity map for Mars Pathfinder narrow-baseline stereo pair. **a** Left image. **b** Disparity image. Dark values represent larger disparities

carefully calibrated external parameters, whereas our methodology did not.

7 Triangulation

The triangulated position p_1 of a point with respect to the (rectified) first camera frame can be computed using the coordinates of the feature in the first image $(r,\ c_1)$, the coordinates of the corresponding feature in the second image $(r,\ c_2)$ and the camera parameters. Let A_1 and A_2 be the camera projection matrices (which will differ only in the projected location of the center of projection), R be the camera rotation in the global reference frame (it is the same for both cameras after rectification), and C_1 and C_2 be the camera centers of projection.

Define:

$$S_1 = (A_1 R)^{-1} \tag{14}$$

$$S_2 = (A_2 R)^{-1} \tag{15}$$

$$a = S_1^{(0,0)} c_1 + S_1^{(0,1)} r + S_1^{(0,2)} \tag{16}$$

$$b = S_1^{(2,0)} c_1 + S_1^{(2,1)} r + S_1^{(2,2)} \tag{17}$$

$$c = S_2^{(0,0)} c_2 + S_r^{(0,1)} r + S_2^{(0,2)} \tag{18}$$

$$d = S_2^{(2,0)} c_2 + S_r^{(2,1)} r + S_2^{(2,2)} \tag{19}$$

$$l_1 = (dC_1^{(0)} - C_2^{(0)} - cC_1^{(2)} + C_2^{(2)})/(bc - ad) \tag{20}$$

The triangulated position is given by

$$p_1 = C_1 + l_1 S_1 \left[c_1\ r\ 1\right]^T, \tag{21}$$

8 Experiments

We have performed experiments with these techniques on images of natural terrain of sandy and rocky terrain (similar to Mars). Some tests used actual images of Mars from the Spirit and Opportunity rovers or from the Mars Pathfinder mission. The first experiment tested a variety of baselines using data collected at JPL using the Rocky 8 rover prototype. A sequence of 11 images was captured with roughly 20 centimeter intervals between the camera locations in the JPL Mars Yard using the navigation cameras on the rover mast. This allows us to consider stereo pairs with baseline distances ranging from 20 centimeters to 2 meters. All of the images in the sequence were captured with a camera orientation that is largely perpendicular to the direction of travel, see Fig. 8.

Figure 9 shows the disparity maps that were generated from this sequence. The first image in the sequence was used in each wide-baseline pair, with every other image serving as the second image in one pair. The disparities are rendered relative to the position of the pixel in the first image. It can be seen in this data that the coverage of the stereo data is larger for image pairs with a smaller baseline. One reason for this is that there is a smaller overlap in the terrain visible in the image pair when the baseline is increased. This causes the triangular region on the lower-left that contains no disparities. A second reason for the lower coverage is that, as the camera positions become farther apart, the change in the appearance of the terrain becomes larger (the terrain is viewed from a different perspective). This makes the correspondence problem more difficult and the verification techniques eliminate more of the disparity data. It is worth noting that the goal of wide-baseline stereo vision is to map the more distant terrain, so that failure on nearby terrain (which can be mapped with a smaller baseline) is not a large drawback. The verification techniques are largely successful in eliminating outliers. However, they are not perfect. An outlier region that was not successfully pruned can be seen in the lower-left of the second image. In addition, correct data is sometimes pruned by the techniques.

We can evaluate the quality of the more distant range data in this experiment by examining the shape and accuracy of the range data to the vertical wall present at the top of each image. Conventional stereo techniques with a 20 cm baseline

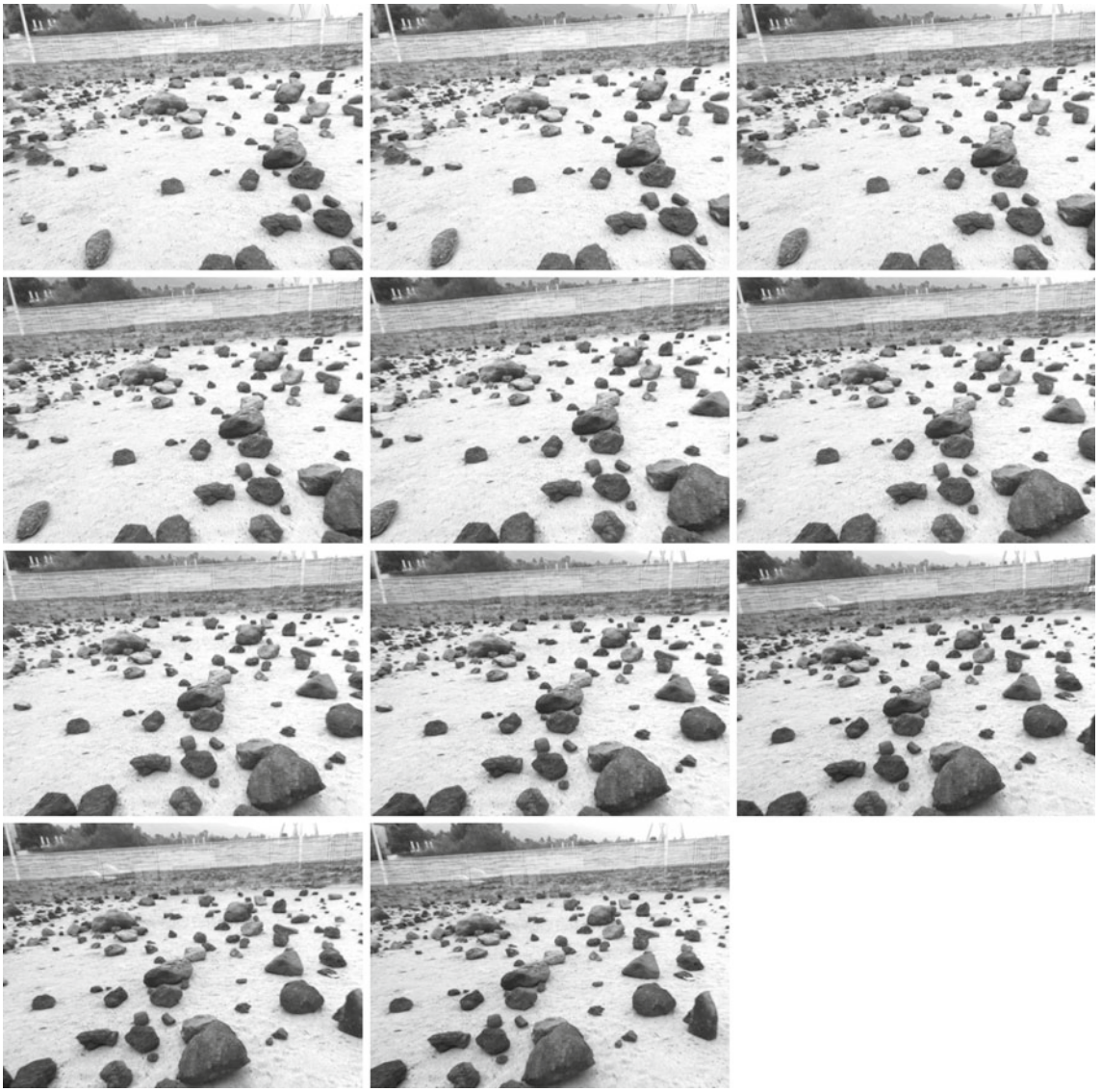

Fig. 8 Image sequence from the JPL Mars Yard. The images were taken at intervals of approximately 20 cm

did not achieve high accuracy in the shape of the wall, since it was roughly 20 m from the rover. Our techniques also did not achieve high accuracy with a 20 cm baseline, see Fig. 10. The estimated shape of the wall (which should be planar) was improved as the baseline was increased. Quantitative error was present in some cases because of a small number of feature matching errors that caused the motion estimation to converge to an incorrect local minimum. The incorrect matches were the result of artificial objects (for example, the poles rising beyond the wall). Needless to say, errors owing to the similarity of artificial objects would not occur in completely natural terrain, such as on Mars.

Figure 11 shows an experiment with more distant terrain. The wide-baseline stereo pair was captured by Rocky 8 during field testing. In these images, the foreground has been cropped, since the techniques were not successful in solving the correspondence problem for this nearby terrain. For this experiment, the baseline distance for the stereo pair was approximately 5 m, but the baseline was not perpendicular to the camera axis. This resulted in a small rotation of terrain in the rectified images. In this case, the images were captured with narrow field-of-view cameras, so the resolution of the images is high. The range data to the ridge seen in the images had high qualitative accuracy, even though the ridge was over 1 km distant from the camera.

The current computation time required for the code is less than a minute on a 2.5 GHz personal computer. It is likely that a careful implementation could improve upon this significantly. For a Mars rover, the wide-baseline stereo operation would be performed infrequently. Since commands are

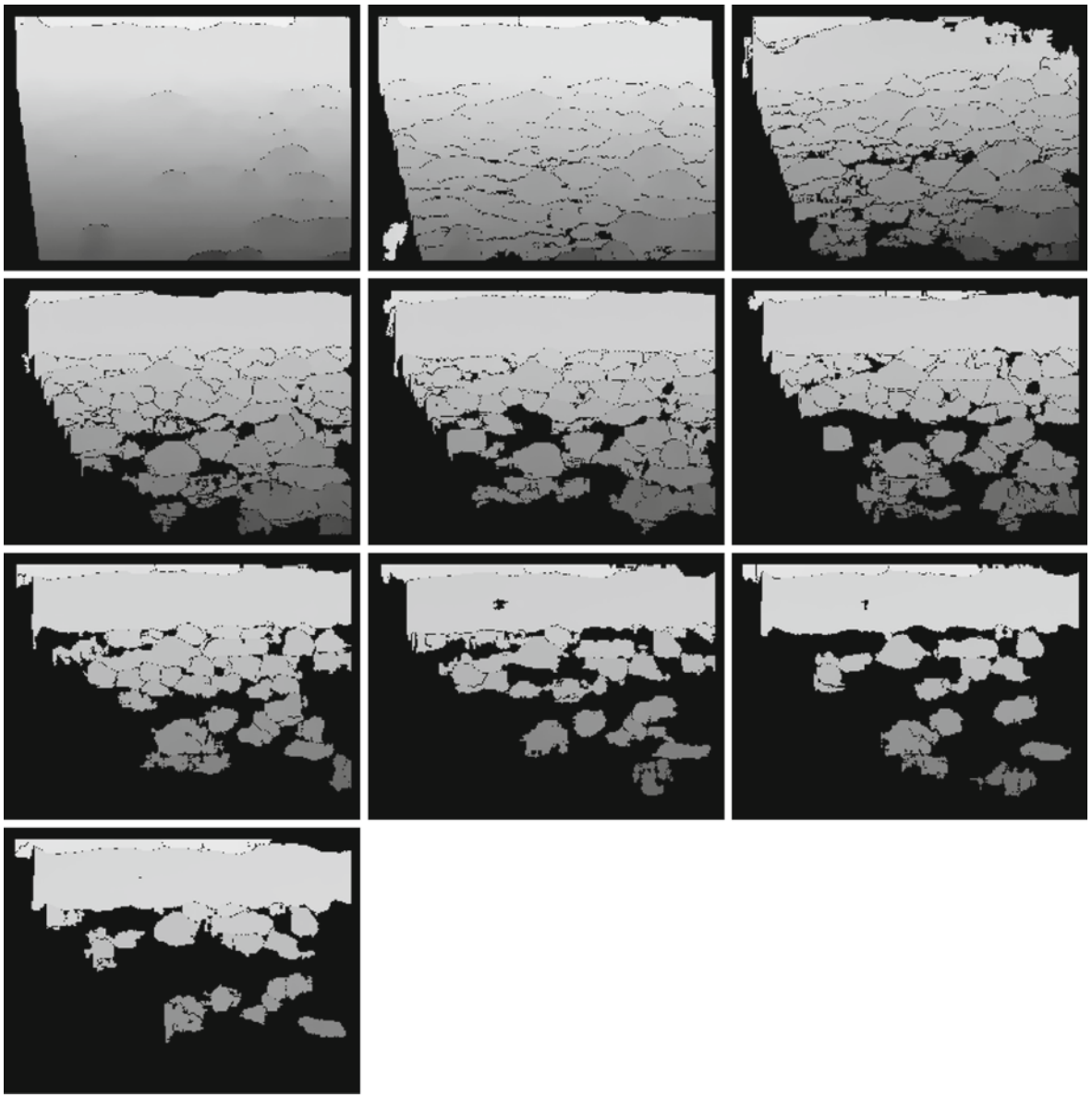

Fig. 9 Disparity maps generated from Mars Yard sequence. Each disparity map was created using the wide-baseline stereo algorithm with image 1 as the first image and image $n+1$ as the second image. The baseline distance ranges from 20 for the first disparity map to 2 m for the last disparity map. Black pixels indicate that the disparity was pruned at that location

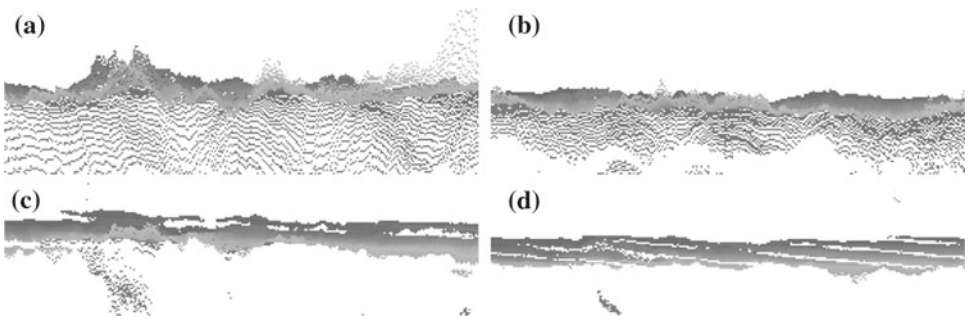

Fig. 10 Overhead view of the point cloud generated using wide-baseline stereo for the wall seen at the top of the images in Fig. 8. **a** 20 cm baseline. **b** 40 cm baseline. **c** 100 cm baseline. **d** 200 cm baseline

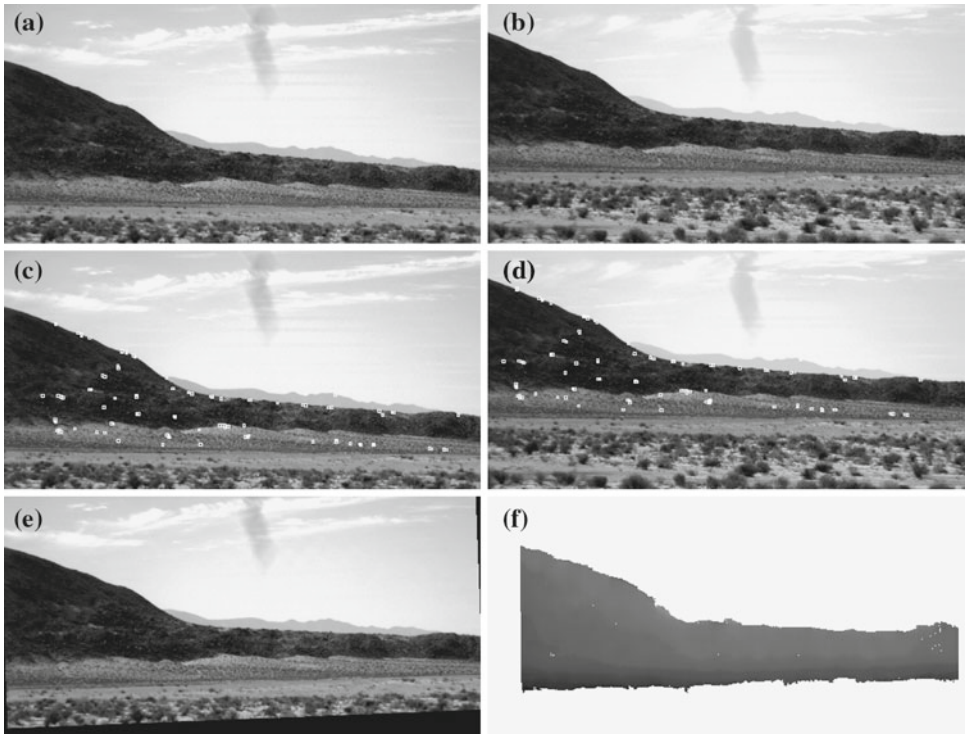

Fig. 11 Wide-baseline stereo pair of the California desert captured by Rocky 8 prototype. **a** Left image. **b** Right image. **c** Feature matches in left image. **d** Feature matches in right image. **e** Rectified left image. **f** Disparity map for left image

received infrequently from Earth, a short wait for results is likely to be acceptable.

9 Failure modes

It is important to consider the conditions under which the wide-baseline stereo techniques fail. One failure mode for these techniques is the inability to establish correspondences between the wide-baseline images. This can occur for multiple reasons. For example, the images may not capture the same terrain or the terrain may be uniform, such that no distinctive features exist. Nothing can be done about these situations, except to modify the rover operation in order to capture a better pair of images.

Problems can also occur in feature matching when the appearance between the images changes drastically. We can see this in the nearby terrain in several of the examples given here. Large baselines are better for mapping distant terrain than nearby terrain. Often conventional stereo can be used for mapping the nearby terrain. We have found that this can be improved upon by capturing images of the terrain at more than two rover positions, with baselines ranging from small to large. This sequence of baselines allows mapping of both near and far terrain. It can also improve the depth estimate on terrain captured in more than two images.

We have seen that incorrect feature matching sometimes adversely affects the motion estimate and this can cause poor results. In our experiments, such incorrect tracking has occurred primarily for artificial objects, rather than natural terrain, owing to objects with a very similar appearance. It is likely that the presence of incorrect matches in these cases can be addressed through the use of a random sampling technique, such as RANSAC, to eliminate the incorrect matches prior to the iterative estimation step [45].

In order for the rectification to be accurate over the entire image, it is important that the sparse correspondences used in the motion estimate cover the image well. Areas that are not well covered by such correspondences tend to be fit poorly and, thus, the rectification performs less well in these areas. We have concentrated less on this problem than others, since this problem occurs primarily for closer terrain. However, this could be improved through the use of feature matching strategies that are robust to affine transformations [1,46,48].

10 Summary

We have developed an algorithm to perform wide-baseline stereo on a mobile robot. Unlike conventional stereo vision, these techniques allow the robot to map terrain that is many meters (up to a few kilometers) distant. This has required the solution to two problems. We have addressed inexact knowledge of the relative positions between the cameras using nonlinear motion estimation. This step automatically selects features and determines correspondences between the images in order to refine the motion estimate to satisfy epipolar constraints. We use robust template matching techniques to deal with the increased change in appearance between

the images for dense matching. This method tolerates outliers and perspective distortion. High-quality stereo matching results are achieved even with a significant change in the image viewpoint.

Acknowledgements We gratefully acknowledge funding of this work by the NASA Mars Technology Program. We thank Max Bajracharya, Dan Helmick, and Rich Petras for collecting the Rocky 8 field test data, Dan Clouse for the data collection performed in the JPL Mars Yard, Mark Maimone for help with the MER data, Richard Madison for pointing out some issues during testing, Ming Ye for assisting with implementation of the dense matching, Jonathan P. Hendrich for assisting with the CLARAty integration, Rusty Gerard for porting the code to Windows, and the MER team for an inspiring mission with publicly available images.

References

1. Baumberg, A.: Reliable feature matching across widely separated views. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, vol. 1, pp. 774–781 (2000)
2. Faugeras, O., Hotz, B., Mathieu, H., Viéville, T., Zhang, Z., Fua, P., Théron, E., Moll, L., Berry, G., Vuillemin, J., Bertin, P., Proy, C.: Real time correlation-based stereo: algorithm, implementations and applications. Tech. Rep. RR-2013, INRIA (1993). http://www.inria.fr/rrrt/rr-2013.html
3. Förstner, W., Gülch, E.: A fast operator for detection and precise locations of distinct points, corners, and centres of circular features. In: Proceedings of the Intercommission Conference on Fast Processing of Photogrammetric Data, pp. 281–305 (1987)
4. Forsyth, D.A., Ponce, J.: Computer Vision: A Modern Approach. Prentice Hall, Englewood Cliffs (2003)
5. Fusiello, A., Trucco, E., Verri, A.: A compact algorithm for rectification of stereo pairs. Mach. Vis. Appl. **12**, 16–22 (2000)
6. Gennery, D.B.: Visual terrain matching for a Mars rover. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 483–491 (1989)
7. Hebert, M., Caillas, C., Krotkov, E., Kweon, I.S., Kanade, T.: Terrain mapping for a roving planetary explorer. In: Proceedings of the IEEE Conference on Robotics and Automation, vol. 2, pp. 997–1002 (1989)
8. Huber, J., Graefe, V.: Motion stereo for mobile robots. IEEE Trans. Ind. Electron. **41**(4), 378–383 (1994)
9. Huber, D.F., Hebert, M.: A new approach to 3-d terrain mapping. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 1121–1127 (1999)
10. Hung, Y.P., Chen, C.S., Hung, K.C., Chen, Y.S., Fuh, C.S.: Multipass hierarchical stereo matching for generation of digital terrain models from aerial images. Mach. Vis. Appl. **10**(5–6), 280–291 (1998)
11. Jung, I.K., Lacroix, S.: High resolution terrain mapping using low altitude stereo imagery. In: Proceedings of the International Conference on Computer Vision, pp. 946–951 (2003)
12. Kelly, A., Stentz, A., Hebert, M.: Terrain map building for fast navigation on rugged outdoor terrain. In: Mobile Robots VII, Proceedings of SPIE 1831, pp. 576–589 (1992)
13. Krotkov, E.: Terrain mapping for a walking planetary rover. IEEE Trans. Robot. Autom. **10**(6), 728–739 (1994)
14. Kweon, I.S., Kanade, T.: High-resolution terrain map from multiple sensor data. IEEE Trans. Pattern Anal. Mach. Intell. **14**(2), 278–292 (1992)
15. Lacroix, S., Jung, I.K., Mallet, A.: Digital elevation map building from low altitude stereo imagery. Robot. Auton. Syst. **41**(2–3), 119–127 (2002)
16. Li, R., Ma, F., Matthies, L.H., Olson, C.F., Arvidson, R.E.: Localization of Mars rovers using descent and surface-based image data. J. Geophys. Res. Planets **107**(E11) 4.1–4.8 (2002)
17. Li, R., Squyres, S.W., Arvidson, R.E., Archinal, B.A., Bell, J., Cheng, Y., Crumpler, L., Marais, D.J.D., Di, K., Ely, T.A., Golombeck, M., Graat, E., Grant, J., Guinn, J., Johnson, A., Greeley, R., Kirk, R.L., Maimone, M., Matthies, L.H., Malin, M., Parker, T., Sims, M., Soderblom, L.A., Thompson, S., Wang, J., Whelley, P., Xu, F.: Initial results of rover localization and topographic mapping for the 2003 Mars exploration rover mission. Photogramm. Eng. Remote Sens. **71**(10), 1129–1142 (2005)
18. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vis. **60**(2), 91–110 (2004)
19. Maimone, M., Matthies, L., Osborn, J., Rollins, E., Teza, J., Thayer, S.: A photo-realistic 3-d mapping system for extreme nuclear environments: Chornobyl. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, vol. 3, pp. 1521–1527 (1998)
20. Maimone, M., Cheng, Y., Matthies, L.: Two years of visual odometry on the Mars exploration rovers. J. Field Robot. **24**(3), 169–186 (2007)
21. Mandelbaum, R., Salgian, G., Sawhney, H., Hansen, M.: Terrain reconstruction for ground and underwater robots. In: Proceedings of the IEEE Conference on Robotics and Automation, pp. 879–884 (2000)
22. Matas, J., Chum, O., Urban, M., Pajdla, T.: Robust wide-baseline stereo from maximally stable extremal regions. Image Vis. Comput. **22**, 761–767 (2004)
23. Matthies, L.: Passive stereo range imaging for semi-autonomous land navigation. J. Robot. Syst. **9**(6), 787–816 (1992)
24. Matthies, L.: Stereo vision for planetary rovers: Stochastic modeling to near real-time implementation. Int. J. Comput. Vis. **8**(1), 71–91 (1992)
25. Mikolajczyk, K., Schmid, C.: Scale & affine invariant interest point detectors. Int. J. Comput. Vis. **60**(1), 63–86 (2004)
26. Mikolajczyk, K., Tuytelaars, T., Schmid, C., Zisserman, A., Matas, J., Schaffalitzky, F., Kadir, T., Gool, L.V.: A comparison of affine region detectors. Int. J. Comput. Vis. **65**(1/2), 43–72 (2005)
27. Montgomery, J.F., Johnson, A.E., Roumeliotis, S.I., Matthies, L.H.: The JPL autonomous helicopter testbed: a platform for planetary exploration technology research and development. J. Field Robot. **23**, 245–267 (2006)
28. Mühlmann, K., Maier, D., Hesser, J., Männer, R.: Calculating dense disparity maps from color stereo images, an efficient implementation. Int. J. Comput. Vis. **47**(1/2/3), 79–88 (2002)
29. Negahdaripour, S., Hayashi, B.Y., Aloimonos, Y.: Direct motion stereo for passive navigation. IEEE Trans. Robot. Autom. **11**(6), 829–843 (1995)
30. Okutomi, M., Kanade, T.: A multiple-baseline stereo. IEEE Trans. Pattern Anal. Mach. Intell. **15**(4), 353–363 (1993)
31. Olson, C.F.: Probabilistic self-localization for mobile robots. IEEE Trans. Robot. Autom. **16**(1), 55–66 (2000)
32. Olson, C.F.: Maximum-likelihood image matching. IEEE Trans. Pattern Anal. Mach. Intell. **24**(6), 853–857 (2002)
33. Olson, C.F., Abi-Rached, H.: Wide-baseline stereo experiments in natural terrain. In: Proceedings of the International Conference on Advanced Robotics, pp. 376–383 (2005)
34. Olson, C.F., Abi-Rached, H., Ye, M., Hendrich, J.P.: Wide-baseline stereo vision for Mars rovers. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 1302–1307 (2003)

35. Olson, C.F., Matthies, L.H., Schoppers, M., Maimone, M.W.: Rover navigation using stereo ego-motion. Robot. Auton. Syst. **43**(4), 215–229 (2003)
36. Olson, C.F., Matthies, L.H., Wright, J.R., Li, R., Di, K.: Visual terrain mapping for Mars exploration. Comput. Vis. Image Underst. **105**(1), 73–85 (2007)
37. Parker, L.E., Schneider, F.E., Schultz, A.C.: Merging partial maps without using odometry. In: Multi-Robot Systems. From Swarms to Intelligent Automata, vol. III, pp. 133–144. Springer (2005)
38. Press, W.H., Teukolsky, S.A., Vetterling, W.T., Plannery, B.P.: Numerical Recipes in C. Cambridge University Press, London (1988)
39. Pritchett, P., Zisserman, A.: Wide baseline stereo matching. In: Proceedings of the International Conference on Computer Vision, pp. 765–760 (1998)
40. Schaffalitzky, F., Zisserman, A.: Viewpoint invariant texture matching and wide baseline stereo. In: Proceedings of the International Conference on Computer Vision, vol. 2, pp. 636–643 (2001)
41. Shi, J., Tomasi, C.: Good features to track. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 593–600 (1994)
42. Strecha, C., Tuytelaars, T., Van Gool, L.: Dense matching of multiple wide-baseline views. In: Proceedings of the International Conference on Computer Vision, vol. 2, pp. 1194–1201 (2003)
43. Szeliski, R., Kang, S.B.: Recovering 3d shape and motion from image streams using non-linear least squares. J. Vis. Commun. Image Represent. **5**(1), 10–28 (1994)
44. Thrun, S.: Robotic mapping: a survey. In: Lakemeyer, G., Nebel, B. (eds.) Exploring Artificial Intelligence in the New Millennium, pp. 1–35. Morgan Kaufmann (2003)
45. Torr, P.H.S., Murray, D.W.: The development and comparison of robust methods for estimating the fundamental matrix. Int. J. Comput. Vis. **24**(3), 271–300 (1997)
46. Tuytelaars, T., Gool, L.V.: Matching widely separated views based on affine invariant regions. Int. J. Comput. Vis. **59**(1), 61–85 (2004)
47. Williams, S., Dissanayake, G., Durrant-White, H.: Towards terrain-aided navigation for underwater robotics. Adv. Robot. **15**, 533–549 (2001)
48. Xiao, J., Shah, M.: Two-frame wide baseline matching. In: Proceedings of the International Conference on Computer Vision, pp. 603–609 (2003)
49. Xiong, Y., Olson, C.F., Matthies, L.H.: Computing depth maps from descent images. Mach. Vis. Appl. **16**(3), 139–147 (2005)

Author biographies

Clark F. Olson received the B.S. degree in Computer Engineering in 1989 and the M.S. degree in Electrical Engineering in 1990, both from the University of Washington, Seattle. He received the Ph.D. degree in Computer Science in 1994 from the University of California, Berkeley. After spending 2 years doing research at Cornell University, he moved to the Jet Propulsion Laboratory (JPL), where he spent 5 years working on computer vision techniques for Mars rovers and other applications. Dr. Olson joined the Faculty at the University of Washington, Bothell in 2001. His research interests include computer vision and mobile robotics. He teaches classes on the mathematical principles of computing and database systems, and he continues to work with NASA/JPL.

Habib Abi-Rached was born in Zahle, Lebanon. The son of a lawyer and a French literature teacher, he traveled throughout the world. He earned a degree in Electrical Engineering from the University of Lebanon, followed by a Diplome d'Ingenieur, a Mastere Specialise and a Diplome D'Etudes Approfondies in Computer Science from ENSEEIHT in Toulouse. In 2006, he earned a Doctor of Philosophy at the University of Washington in Computer Science. Currently he works for SPC, a bio-informatics company in Seattle.