Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition
Kauai, Hawaii, December 2001 · Volume 1 · Pages 392-397

Computing Depth Maps From Descent Imagery

Yalin Xiong
KLA-Tencor
160 Rio Robles St.
San Jose, CA 95134
yalin.xiong@kla-tencor.com

Clark F. Olson
University of Washington, Bothell
Computing and Software Systems
18115 Campus Way NE, Box 358534
Bothell, WA 98011

Larry H. Matthies
Jet Propulsion Laboratory
California Institute of Technology
4800 Oak Grove Drive M/S 125-209
Pasadena, CA 91109

Abstract

In the exploration of the other planets of our solar system, images taken during a lander's descent to the surface of a planet provide a critical link between orbital images and surface images. The descent images not only allow us to locate the landing site in a global coordinate frame, but also provide progressively higher-resolution maps for mission planning. This paper addresses the generation of depth maps from the descent images. Our approach has two steps, motion refinement and depth recovery. During motion refinement, we use an initial motion estimate in order to avoid the intrinsic ambiguity in descending motions. The objective of the motion refinement step is to adjust the motion parameters such that the epipolar constraints are valid between adjacent frames. The depth recovery step correlates adjacent frames to match pixels for triangulation. Due to the descending motion, the conventional rectification process is replaced by a set of anti-aliasing image warpings corresponding to a set of virtual parallel planes. We demonstrate experimental results on synthetic and real descent images.

1 Introduction

Future space missions that land on Mars (and other planetary bodies) are likely to include a downward-looking camera mounted on the vehicle as it descends to the surface. The images taken by the camera during the descent provide a critical link between orbital images and lander/rover images on the surface of the planet. By matching the descent images against orbital images, the descent vehicle can localize itself in global coordinates and, therefore, achieve precision landing. Through analysis of the descent images, we can build a multi-resolution terrain map for safe landing, rover planning, navigation, and localization. This paper addresses the issue of generating multi-resolution terrain maps from a sequence of descent images. We use motion estimation and structure-from-motion techniques to recover depth maps from the images. A new technique for computing depths is described that is based on correlating the images after performing anti-aliasing image warpings corresponding to a set of virtual planar surfaces.

It is well known that, in a descending motion against a planar surface, the motion recovery problem is ill-posed, since translations parallel to the surface appear similar to rotations about axes parallel to the surface. For space missions, it is likely that the motion will be nearly perpendicular to the planetary surface. Motion recovery is, therefore, not generally reliable for this scenario. However, if we have an independent means to measure the orientation of the camera, we can obtain stable motion recovery. For planetary exploration missions, such measurements can be provided by the inertial navigation sensors on the landing spacecraft.

For the Mars Polar Lander mission, which was unable to return data due to loss of the lander, it was planned that the camera would take an image every time the distance to the ground halved. In other words, there would be roughly a scale factor of two between adjacent frames in the sequence. A similar scenario is likely in future missions. The large change of scale prohibits us from tracking features and correlating images across many frames. We limit our correlation and depth recovery to adjacent frames for this reason.

The descending motion also causes problems in correlating the images. Since the epipoles are located near the center of the images, it is not practical to rectify adjacent frames in the same manner that traditional stereo techniques do. Instead, we "rectify" the images by considering a set of parallel planar surfaces through the terrain. Each surface corresponds to a projective warping between the adjacent images. The surface that yields the best correlation at each pixel determines the depth estimate for that location. This rectification not only aligns images according to the epipolar lines, but also equalizes the image scales using anti-aliased warpings.

Of course, many other approaches have been proposed for recovering motion and depth from image sequences [2, 4, 6, 8, 9, 11]. This work differs from most previous work in two ways. First, almost all of the motion is forward along the camera pointing direction. Second, large movements in the camera position occur between frames, usually doubling the resolution of the images at each frame. This work is, thus, in part, an application of previous work to the problem of mapping spacecraft descent imagery and, in part, new techniques for dealing with the above problems. The technique produces dense maps of the terrain and operates under the full perspective projection.

In the next two sections, we describe the motion refinement and depth recovery steps in detail. We then discuss our experiments on synthetic and real descent images. The results demonstrate the various terrain features that can be recovered. Near the landing site, small obstacles such as rocks and gullies can be identified for planning local rover

©2001 IEEE. Published in the Proceedings of CVPR'01, December 2001, Kauai, Hawaii. Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works, must be obtained from the IEEE.

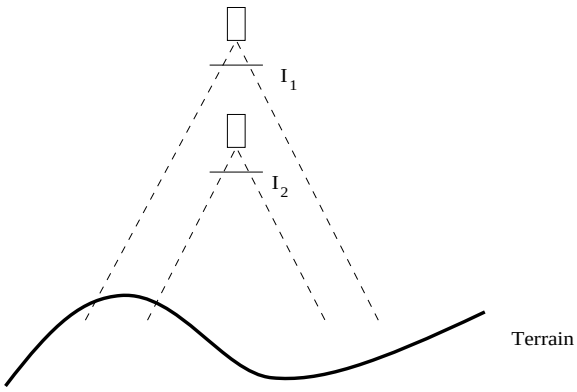


Figure 1: Descent motion.

navigation. Further from the landing site, larger features such as mountain slopes and cliffs are visible for use in long-range planning.

2 Motion refinement

Recovering camera motion from two or more frames is one of the classical problems in computer vision. Linear [7] and nonlinear [10] solutions have been proposed. For descent motions (as in Fig. 1), generic motion recovery from matched features is ill-posed owing to a numerical singularity. Since the camera can be rigidly attached to the lander, and the change in the lander orientation can be measured accurately by an inertial navigation system onboard, we can eliminate the singularity problem by adding a penalty term for deviating from the measured orientation. The following two subsections briefly explain our feature tracking and nonlinear optimization for motion refinement.

2.1 Feature tracking

For each pair of adjacent frames in the sequence, we track features that have been selected in the higher resolution frame into the lower resolution frame. Forstner's interest operator [3] is used to evaluate the trackability of the features in the higher resolution frame. We select the features with high scores, while disallowing features that are too close together.

Once the image resolutions have been equalized (through downsampling or anti-aliasing warping, if necessary), feature tracking can be performed in a straightforward manner. For every feature in the reference image, we search an area in the target image for a match. The location of the search area is derived from the initial estimate of the vehicle ego-motion and its altitude. The initial estimates do not need to be precise. The size of the search area is determined by how uncertain the initial estimates are. Once the search area is located, we detect the feature match through normalized correlation.

2.2 Nonlinear motion estimation

The objective of motion refinement is to establish the precise camera motion between two adjacent frames such that the epipolar constraints are satisfied to subpixel accuracy. It is unrealistic to expect the onboard inertial sensors to track the camera orientation with such precision. It is, therefore, crucial to be able to refine the motion parameters prior to recovering the depth map.

The tracked features provide a rich set of observations to constrain the camera motion, even though the relationship between the locations of the tracked features and the camera motion parameters is nonlinear. Let us assume that the projection matrix of the camera (including the calibrated internal parameters) is $\mathbf{M}$, the location of feature i at time t is $[X_i^t, Y_i^t, Z_i^t]^T$ in the camera frame of reference, its image location at time t represented in homogeneous coordinates is $[x_i^t, y_i^t, z_i^t]^T$, and the camera motion between time t and time $t+1$ is composed of a translation $\mathbf{T}$ and rotation $\mathbf{R}$ (3×3 matrix). The projection of the feature at time t is:

$$\begin{bmatrix} x_i^t \\ y_i^t \\ z_i^t \end{bmatrix} = \mathbf{M} \begin{bmatrix} X_i^t \\ Y_i^t \\ Z_i^t \end{bmatrix}, \tag{1}$$

and the projection at time $(t+1)$ is:

$$\begin{bmatrix} x_i^{t+1} \\ y_i^{t+1} \\ z_i^{t+1} \end{bmatrix} = \mathbf{M} \begin{bmatrix} X_i^{t+1} \\ Y_i^{t+1} \\ Z_i^{t+1} \end{bmatrix} = \mathbf{M} \left(\mathbf{R} \begin{bmatrix} X_i^t \\ Y_i^t \\ Z_i^t \end{bmatrix} + \mathbf{T} \right). \tag{2}$$

Therefore, the feature motion in the image is:

$$\begin{aligned} \begin{bmatrix} x_i^{t+1} \\ y_i^{t+1} \\ z_i^{t+1} \end{bmatrix} &= \mathbf{M} \left(\mathbf{R}\mathbf{M}^{-1} \begin{bmatrix} x_i^t \\ y_i^t \\ z_i^t \end{bmatrix} + \mathbf{T} \right) \\ &= \mathbf{U} \begin{bmatrix} x_i^t \\ y_i^t \\ z_i^t \end{bmatrix} + \mathbf{V}, \end{aligned} \tag{3}$$

where $\mathbf{U} = \mathbf{M}\mathbf{R}\mathbf{M}^{-1}$ is a 3×3 matrix and $\mathbf{V} = \mathbf{M}\mathbf{T}$ is a 3-vector. Let $[c_i^t, r_i^t] = \left[x_i^t/z_i^t, y_i^t/z_i^t\right]$ denote the actual column and row location of feature i in image coordinates at time t. We, then, have the predicted feature locations at time $t+1$ as:

$$\hat{c}_i^{t+1} = \frac{u_{00}x_i^t + u_{01}y_i^t + u_{02}z_i^t + v_0}{u_{20}x_i^t + u_{21}y_i^t + u_{22}z_i^t + v_2}, \tag{4}$$

$$\hat{r}_i^{t+1} = \frac{u_{10}x_i^t + u_{11}y_i^t + u_{12}z_i^t + v_1}{u_{20}x_i^t + u_{21}y_i^t + u_{22}z_i^t + v_2}, \tag{5}$$

where u_{ij} and v_i are elements of $\mathbf{U}$ and $\mathbf{V}$ respectively.

There are two ways to optimize the camera motions in the above equations. One is to reduce the two equations into one by eliminating z_i^t. We would then minimize the summed deviation from the equation specifying a nonlinear relation between $[c_i^t, r_i^t]$ and $[\hat{c}_i^{t+1}, \hat{r}_i^{t+1}]$. Though this method is concise and simple, it poses a problem in the context of least-squares minimization in that the objective function does not have a physical meaning.

The other approach to refine the motion estimate is to augment the parameters with depth estimates for each of the features. There are two advantages to this approach. First, the objective function becomes the distance between the predicted and observed feature locations, which is a meaningful measure for optimization. In addition, in the context of mapping descent images, we have a good initial estimate of the depth value from the spacecraft altimeter. Incorporating this information will, thus, improve the optimization in general.

Let us say that the depth value of feature i at time t is d_i^t and the camera is pointing along the z-axis, the homogeneous coordinates of the feature are $[x_i^t, y_i^t, z_i^t]^T = d_i^t[c_i^t, r_i^t, 1]^t$. Therefore, the overall objective function we are minimizing is:

$$\sum_{i=1}^{N} \left(\left(c_i^{t+1} - \hat{c}_i^{t+1}\right)^2 + \left(r_i^{t+1} - \hat{r}_i^{t+1}\right)^2 \right), \qquad (6)$$

where N is the number of features, and $\hat{c}_i^{t+1}$ and $\hat{r}_i^{t+1}$ are nonlinear functions of the camera motion and depth value d_i^t given by Eq. (4) and (5). We perform nonlinear minimization using the Levenberg-Marquardt algorithm with the estimated position as the starting point. Robustness is improved by removing points that yield large residuals.

Eq. (6) specifies the objective function for two adjacent images. A long sequence of descending images requires a common scale reference in order to build consistent multi-resolution depth maps. The key to achieving this is to track features over more than two images. From Eq. (3), the depth value of feature i at time $t+1$ can be represented as

$$d_i^{t+1} \begin{bmatrix} c_i^{t+1} \\ r_i^{t+1} \\ 1 \end{bmatrix} = \mathbf{U} d_i^t \begin{bmatrix} c_i^t \\ r_i^t \\ 1 \end{bmatrix} + \mathbf{V}. \qquad (7)$$

Thus, the overall objective is to minimize the sum of Eq. (6) for all adjacent pairs while maintaining the consistent scale reference by imposing the constraint in Eq. (7) for all features tracked over more than two frames.

3 Depth map recovery

The second step of our method generates depth maps by performing correlations between image pairs. In order to compute the image correlation efficiently, we need to rectify the images in a manner similar to binocular stereo. Unfortunately, it is impossible to rectify the images along scanlines because the epipolar lines intersect near the center of the images. If we resample the images along epipolar lines, we will oversample near the image center, and undersample near the image boundaries.

In order to avoid this problem, we adopt a slicing algorithm to perform the correlation efficiently. The main concept is to use a set of virtual planar surfaces slicing through the terrain as shown in Figure 2. A similar concept was used by Collins [2]. Collins applied the idea to perform matching between features extracted from the images. In contrast, we perform dense matching between intensity windows in the images.

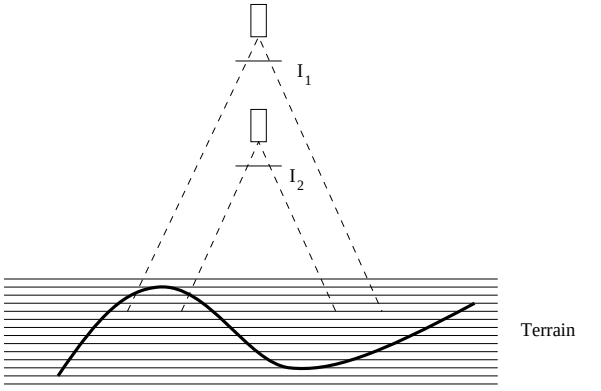


Figure 2: The terrain is sliced with virtual parallel planes.

The virtual planar surfaces are similar, in concept, to horopter surfaces [1] in stereo. For every planar surface k, if a terrain surface patch lies on the planar surface, then there exists a projective warping $\mathbf{P}_k$ between the two images for this patch. If we designate the first image $I_1(x, y)$ and the second image $I_2(x, y)$, then for every virtual planar surface, we can compute the sum-of-squared-differences (SSD) as:

$$C_k(x, y) = \sum_{m=x-W}^{x+W} \sum_{n=y-W}^{y+W} \left(I_1(m, n) - I_2^k(m, n)\right)^2, \qquad (8)$$

where $2W + 1$ is the size of the correlation window and $I_2^k(x, y)$ is a warped version of $I_2(x, y)$:

$$I_2^k(x, y) = I_2 \left(\frac{p_{00}x + p_{01}y + p_{02}}{p_{20}x + p_{21}y + p_{22}}, \frac{p_{10}x + p_{11}y + p_{12}}{p_{20}x + p_{21}y + p_{22}} \right), \qquad (9)$$

where p_{ij} are elements of the 3x3 matrix $\mathbf{P}_k$. Due to the large resolution difference, an anti-aliasing resampling [5] or a uniform downsampling of $I_2(x, y)$ is applied before the image warping. In practice, if the camera heading directions are close to be perpendicular to the ground, a uniform downsampling before warping is sufficient. Otherwise, a space-variant downsampling should be used to equalize the image resolutions.

The estimated depth value at each pixel is the depth of the plane z_k whose corresponding SSD image pixel $C_k(x, y)$ is the smallest:

$$z(x, y) = z_k, \qquad (10)$$

where

$$C_k(x, y) \leq C_j(x, y), j = 1, \ldots, M, \qquad (11)$$

and M is the number of planar surfaces. To further refine the depth values, the underlying SSD curve can be interpolated by a quadratic curve and the "subpixel" depth value can be computed [12] as:

$$z(x, y) = z_k + \frac{\delta z(C_{k+1}(x, y) - C_{k-1}(x, y))}{2(C_{k+1}(x, y) + C_{k-1}(x, y) - 2C_k(x, y))}, \qquad (12)$$

where δz is the depth increment between adjacent planar surfaces. In order to improve the localization of this operation, we compute the SSD between the windows with a

Gaussian modulation function, so that the pixels closer to the center of the window have more weight than the pixels at the edge of the window.

The projective warping matrix $\mathbf{P}_k$ is derived from the parameters of the camera motion and the planar surfaces. For an arbitrary point $\mathbf{X}$ in some reference frame, its projection is expressed as $\mathbf{x} = \mathbf{M}(\mathbf{X} - \mathbf{C})$, where $\mathbf{C}$ is the position of the camera nodal point and $\mathbf{M}$ is the projection matrix. Note that $\mathbf{C}$ and $\mathbf{M}$ encapsulate the camera motion between the images, since they are represented in a common reference frame. Let $\mathbf{C}_1$ and $\mathbf{M}_1$ represent the higher camera, $\mathbf{C}_2$ and $\mathbf{M}_2$ represent the lower camera in Fig. 2, and $\mathbf{N}^T\mathbf{X} + z_k = 0$ represent the set of planar surfaces. For any pixel in image 2 (i.e. the lower camera), its location must lie on a 3d ray:

$$\mathbf{X} = s\mathbf{M}_2^{-1} \begin{bmatrix} c_2 \\ r_2 \\ 1 \end{bmatrix} + \mathbf{C}_2, \tag{13}$$

where c_2 and r_2 are the column and row location of the pixel and s is a positive scale factor. If the pixel is from a point on the planar surface, then the following constraint must be satisfied:

$$s\mathbf{N}^T\mathbf{M}_2^{-1} \begin{bmatrix} c_2 \\ r_2 \\ 1 \end{bmatrix} + \mathbf{N}^T\mathbf{C}_2 + z_k = 0. \tag{14}$$

Therefore, the scale factor s must be

$$s = -\frac{\mathbf{N}^T\mathbf{C}_2 + z_k}{\mathbf{N}^T\mathbf{M}_2^{-1}[c_2, r_2, 1]^T}. \tag{15}$$

We can then re-project the point onto the first image using Eq. (13) and (15):

$$\begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix} = \mathbf{M}_1(\mathbf{X} - \mathbf{C}_1) = \mathbf{P}_k \begin{bmatrix} c_2 \\ r_2 \\ 1 \end{bmatrix}, \tag{16}$$

where $\mathbf{P}_k$ is a 3×3 matrix specifying the projective warping:

$$\mathbf{P}_k = \mathbf{M}_1(\mathbf{C}_2 - \mathbf{C}_1)\mathbf{N}^T\mathbf{M}_2^{-1} - (\mathbf{N}^T\mathbf{C}_2 + z_k)\mathbf{M}_1\mathbf{M}_2^{-1}. \tag{17}$$

Note that the depth recovery is numerically unstable in the vicinity of the epipoles, located near the center of the image. Pixels near the epipoles usually have a small amount of parallax, even with large camera motions. The SSD curves in these areas are very flat and, thus, accurate depth recovery is difficult. These regions can be easily filtered, if desired, by imposing a minimum curvature threshold at the minima of the SSD curves.

4 Experiments

Figures 3(a) and 3(b) show a synthetic set of nested descent images (400 × 400 pixels). For this set of images, the terrain model is composed of a slowly-varying terrain surface overlaid with rocks distributed according to a statistical model. The height of the camera decreases from approximately 25 meters above the ground to about 6 meters above the ground. The field of view of the camera is 70 degrees.

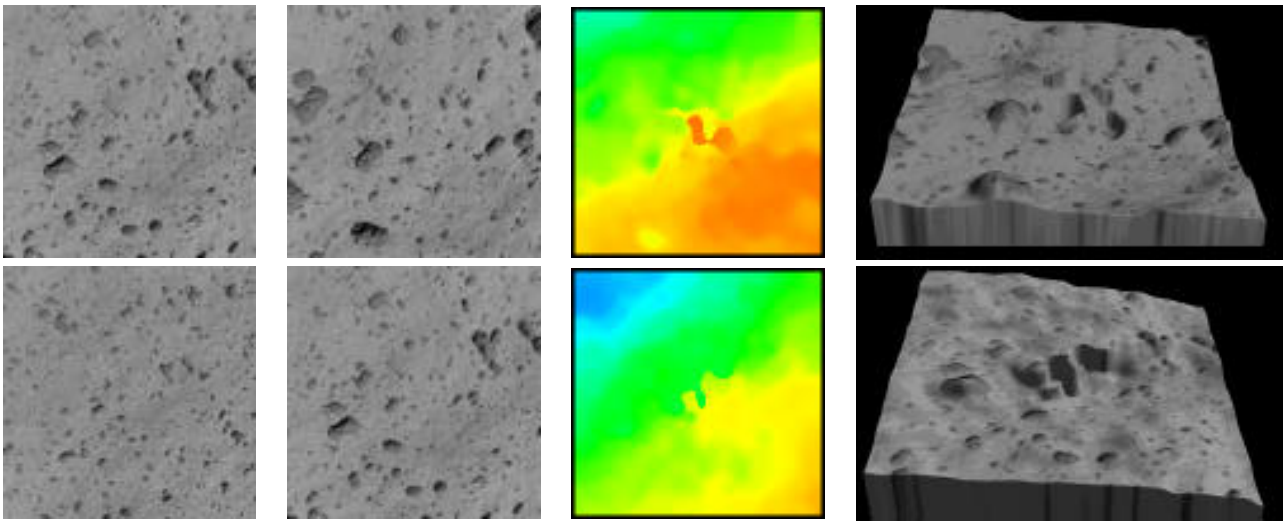
Figure 3(c) shows the recovered depth maps in false-color. The maps have root-mean-square errors of 4.6 cm and 9.7 cm, respectively. Note that the areas close to the focus-of-expansion (at the center of the image) have larger error than the rest of the image, owing to the geometrical instability at the focus-of-expansion. Figure 3(d) shows a visualization of the depth maps, with the image draped over the terrain. In both image pairs, the general downward slope of the terrain from back-to-front and left-to-right can be observed. In addition, individual rocks can be distinguished, particularly in the lower-elevation image pair.

For these experiments, we generated our initial estimates of the camera motion by perturbing the actual camera values by a random noise of magnitude two degrees. This level of accuracy in the orientation can be achieved by the onboard inertial navigation system during an actual landing. The overall quality of the recovered depth maps is satisfactory for both navigation in the vicinity of the landing and long term planning to goals far away from the landing.

Our techniques were also tested using a set of descent images that was collected in the desert area near Silver Lake, California using a helicopter. Figure 4 shows several frames from this sequence. The initial camera motions were estimated using control points on the ground. Several of the images contain significant lateral motions due to the difficulty in maintaining the x-y position of the helicopter during the data collection. Column (b) of Fig. 4 shows the false-color depth maps that were recovered from the sequence and column (c) shows the image draped over the visualized terrain.

Since these are real images captured using a moving helicopter, the focus-of-expansion for each image pair is not at the center of the image (although it is reasonably close in rows 3 and 6). In rows 1 and 2, the focus-of-expansion can be seen above the center of the image, while it is near the bottom-right corner in rows 4 and 5. In row 7, the focus-of-expansion is off of the image to the left. The instability can be seen in these locations where the rendered map becomes wavy or choppy. As the distance from the focus-of-expansion becomes large, the terrain elevations become more accurate. In row 5, the lower elevation image did not completely overlap the higher elevation image, resulting in the lack of height data in the lower-left corner of the result for that image pair.

For the images in this data set, the terrain slopes downward from left to right, which can be observed in the rendered maps. Some of the interesting terrain features include the bushes visible in row 1 and the channels in rows 3-7. Note that the areas in which the helicopter shadow is present yield good results, despite the movement of the shadow. This can be attributed to the robust methods that we use for both motion estimation and template matching.



(a) (b) (c) (d)

Figure 3: Synthetic descent images. (a) Image at higher elevation. (b) Image at lower elevation. (c) False-color elevation map. (d) Rendered terrain map with image overlaid. (The rows have different height scales.)

Overall, this data set indicates that we can robustly compute maps that are useful for navigation over both small and large scales using real images, albeit under somewhat different conditions than would be encountered by an actual Mars lander.

5 Summary

We have presented techniques for extracting depth maps from a sequence of descent images, such as those that would be acquired by a lander descending to a planetary surface. The method consists of two primary steps: motion estimation and depth recovery. Motion estimation is performed by tracking features and minimizing a least-squares objective function using nonlinear methods. The depth map is then recovered using a novel technique where the terrain is sliced by virtual planes, similar to horopter surfaces in stereo. Each plane can be thought of as a vertical disparity. The plane yielding the lowest SSD is selected as the depth for each pixel and subpixel estimation techniques are used to improve the estimate. We have performed experiments with this method on synthetic and real image sequences. The experiments have resulted in maps with sufficient accuracy for performing navigation and planning.

Acknowledgments

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

References

[1] P. J. Burt, L. Wixson, and G. Salgian. Electronically directed "focal" stereo. In *Proceedings of the International Conference on Computer Vision*, pages 94–101, 1995.

[2] R. T. Collins. A space-sweep approach to true multi-image matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 358–363, 1996.

[3] W. Förstner. A framework for low-level feature extraction. In *Proceedings of the European Conference on Computer Vision*, pages 383–394, 1994.

[4] K. J. Hanna. Direct multi-resolution estimation of ego-motion and structure for motion. In *Proceedings of the IEEE Workshop on Visual Motion*, pages 156–162, 1991.

[5] P. Heckbert. Survey of texture mapping. *IEEE Computer Graphics and Applications*, 6(11):56–67, November 1986.

[6] D. J. Heeger and A. D. Jepson. Subspace methods for recognition rigid motion. I. algorithm and implementation. *International Journal of Computer Vision*, 7(2):95–117, January 1992.

[7] H. C. Longuet-Higgins. A computer algorithm for reconstructing a scene from two projections. *Nature*, 293:133–135, September 1981.

[8] J. Oliensis and Y. Genc. Fast algorithms for projective multi-frme structure from motion. In *Proceedings of the International Conference on Computer Vision*, volume 1, pages 536–543, 1999.

[9] S. Soatta and P. Perona. Reducing "structure from motion": A general framework for dynamic vision part 1: Modeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(9):933–942, September 1998.

[10] R. Szeliski and S. B. Kang. Recovering 3d shape and motion from image streams using non-linear least squares. *Journal of Visual Communication and Image Representation*, 5(1):10–28, March 1994.

[11] C. Tomasi and T. Kanade. Shape and motion from image streams under orthography: A factorization method. *International Journal of Computer Vision*, 9(2):137–154, 1992.

[12] Y. Xiong and L. H. Matthies. Error analysis for a real-time stereo system. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1087–1093, 1997.

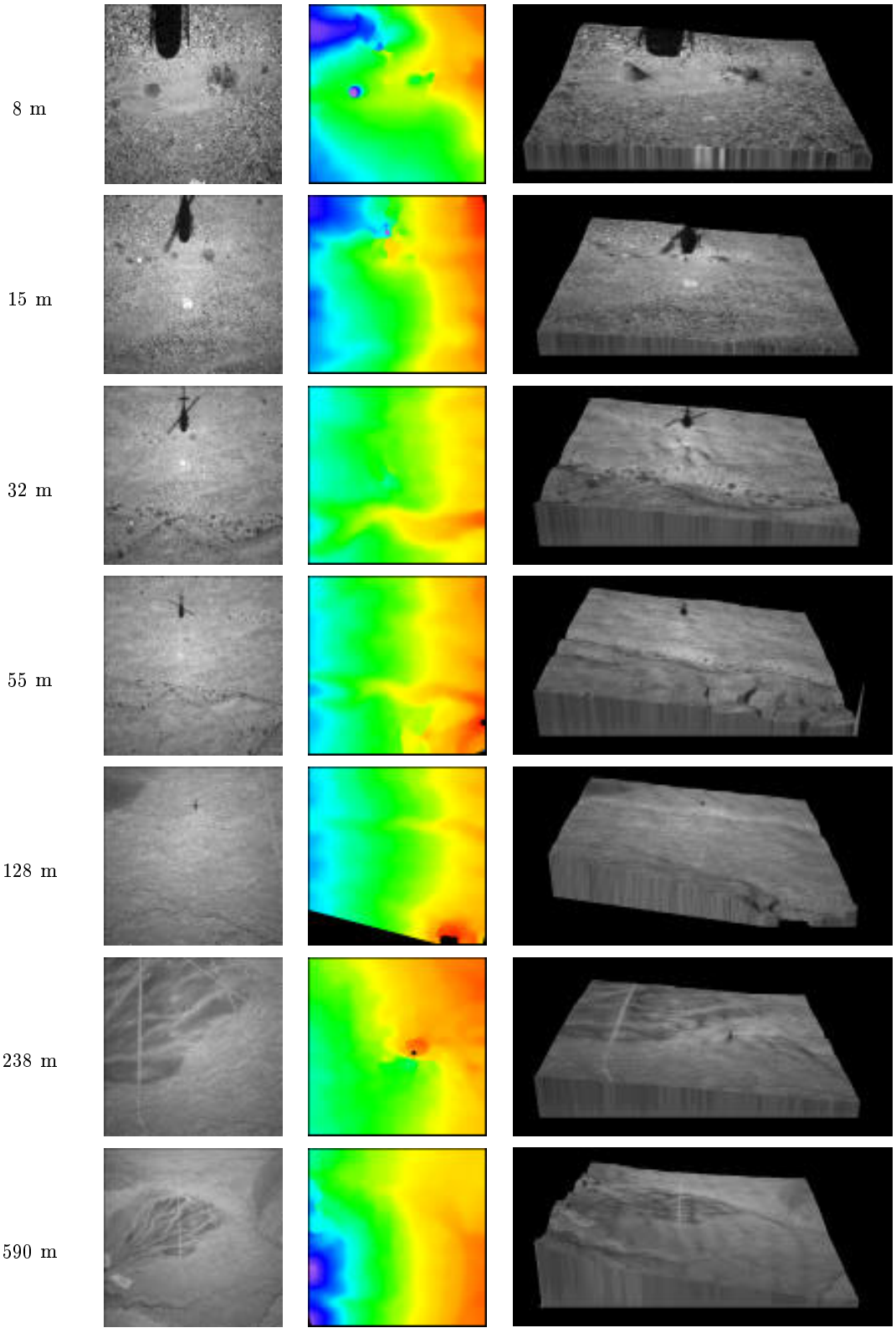


Figure 4: Descent sequence captured with a helicopter. (a) Image sequence (896 × 896 pixels). (b) Estimated terrain map (false-color). (c) Rendered terrain map with image overlaid. (The rows have different height scales.)