Proceedings of the 8th ESA Workshop on Advanced Space Technologies for Robotics and Automation (ASTRA 2004)
ESTEC-Noordwijk, The Netherlands

Far-field Terrain Navigation Using Geometric and Toposemantic Vision

Alec Avedisyan[(1)], David Wettergreen[(2)], Terrence Fong[(1)] , and Charles Baur[(1)]

[(1)]Institut de production et robotique
Ecole Polytechnique Fédérale de Lausanne
CH-1015 Lausanne Switzerland
Email:{alec.avedisyan | terrence.fong | charles.baur} @ epfl.ch

[(2)]The Robotics Institute
Carnegie Mellon University
Pittsburgh, PA 15213 USA
Email: dsw@ri.cmu.edu

ABSTRACT

In this paper, we describe how passive vision can be used to improve "far-field navigation" of planetary rovers, especially for detecting negative obstacles such as cliff edges, ditches, and escarpments. Far-field navigation fills the gap between close-up sensing for obstacle avoidance and satellite imagery. In particular, we can identify dangerous and/or interesting areas in distant terrain by processing color camera images.

INTRODUCTION

The NASA "Life in the Atacama" project is developing a planetary rover to conduct scientific transects of the Atacama Desert (Fig. 1), in order to locate microorganisms and chlorophyll-based life forms. To ensure that this robotic astrobiologist is relevant to the demands of planetary exploration, we do not rely upon artificial satellites, such as GPS. Instead, we are investigating other means for estimating position, orientation, and velocity.

Our approach is based on evaluating terrain at several resolutions and depth (downrange distance). We evaluate terrain based on slope, roughness, discontinuity in terms of the time, power and, in some cases, risk of traverse. We plan long routes using satellite imagery (30m resolution), identify paths that skirt steep slopes and drop-offs by examining the far-field (10-100m), and avoid obstacles in the near-field (0-10m).

In this paper, we describe our approach to far-field terrain navigation. First, using the horizon as a reference, we generate a coarse depth map of the scene. This allows us to depth segment the image and to identify the far-field (Fig. 2).

Next, we apply several *toposemantic* techniques to each image region. We use the term toposemantic to refer to the detection of topological relationships between objects (terrain patches, obstacles, etc.) based on semantic properties (e.g., color). Using statistical analysis, morphological operators, and 2D filters, we compute terrain metrics including obstacle density, roughness, and occlusion. By combining these metrics we produce a *terrain traversability map.* This map allows us to directly estimate the difficulty of traverse and location of danger zones.

Fig. 1. Hyperion rover in the Atacama Desert (Chile).

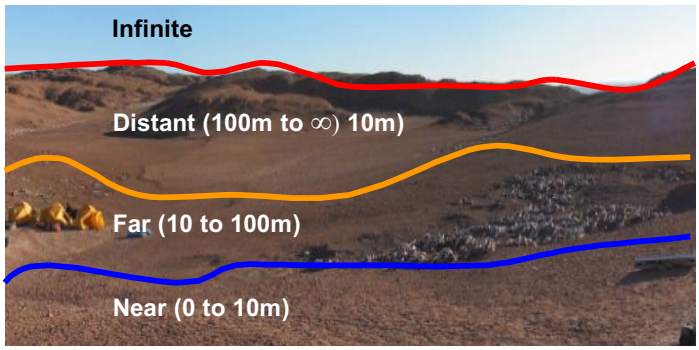

Fig. 2. Terrain zones, classified by down-range distance.

GEOMETRIC VISION

Depth Processing

To estimate distances in the scene, we have developed a stereovision system that meets the following two requirements: (1) weak calibration procedure that uses natural terrain images only; and (2) minimal restriction on camera separation (stereo baseline). These requirements follow from the difficulty of achieving and maintaining precise calibration in a planetary exploration system [1, 2].

The stereo method that we use is based on SIFT [3]. SIFT is a very effective algorithm for locating scene features in the presence of rotation, translation, image distortions, scaling, and partial illumination changes. In our work, we assume that the stereo cameras have similar characteristics.

Image Rectification

We perform image rectification using the most distant points contained in the image. For this task, we apply a sky detection algorithm (described in a following section), which allows us to identify the skyline.

Given the skyline, we locate matching SIFT keypoints and then compute distance between the points to obtain a translation vector. To correct for rotation, we compute the slope between two points on the same picture and calculate the difference. We then use the translation and rotation transform to rectify the image pair (Fig. 3).

Depth Calculation

We compute depth by matching SIFT keypoints between image pairs and using simple stereo camera geometry [4] and camera model [5]. Because we only match keypoints, the resulting depth map is sparse. However, based on tests with images acquired in the Atacama, we have concluded the detected features are sufficiently dense for our needs. In addition, we have found that the SIFT keypoints are localized to higher accuracy than would be obtained using conventional pixel-based correlation.

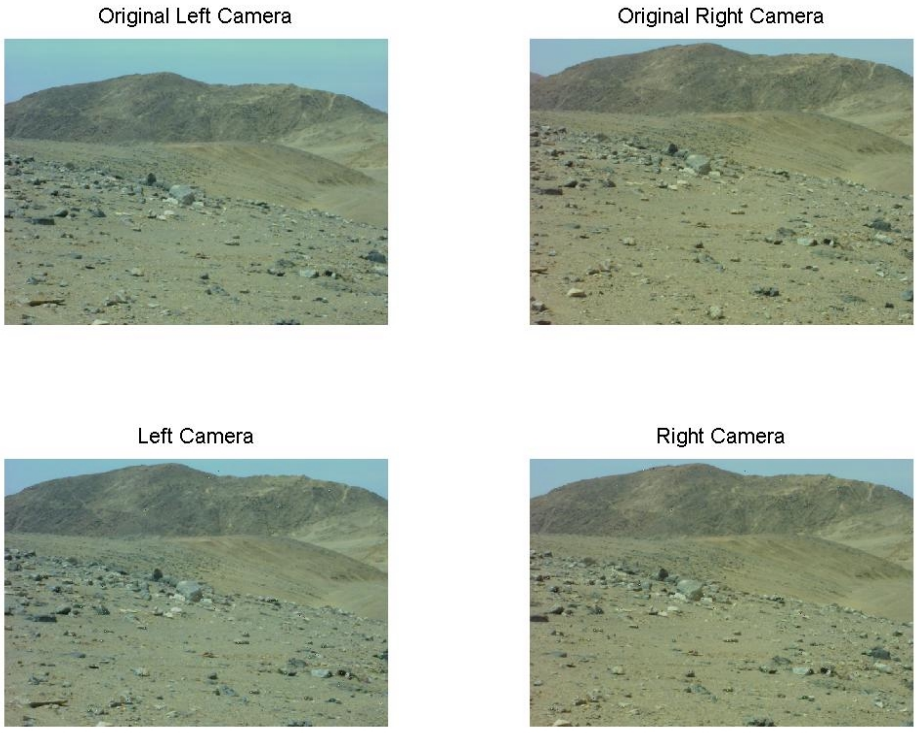

Fig. 3. Stereo images: before calibration (top), after calibration (bottom).

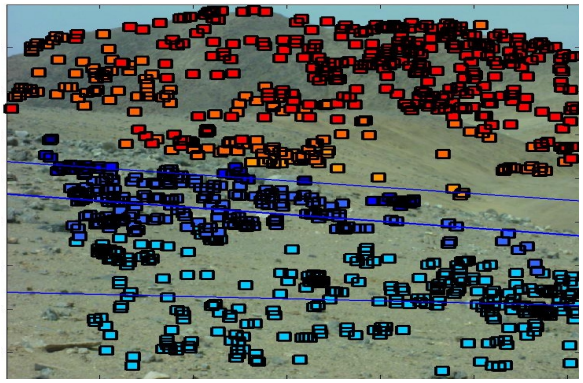

Fig. 4. Matched keypoints (colors show distances in Table 1).

Table 1. Downrange distance (depth) in [m].

From [m]	To [m]	Area's name
1	10	**Near field**
10	20	**Mid field**
20	30	**Far field 1**
30	50	**Far field 2**
50	100	**Far field 3**
100	300	**Distant field 1**
300	500	**Distant field 2**
500	1000	**Distant field 3**
1000	+∞	**Infinite**

Image Segmentation

We compute the real 2D coordinates of points from their image projection in the following manner. First, a unique reference point, which is common to both images is chosen, and the center (on X and Y) of the calibrated image is defined. All distances are then computed relative to this point.

Using the SIFT keypoints we compute a set of fifth-order interpolated curves. Fig. 5 shows a several equidistant image curves. Fig. 6 shows a *distance map* created by segmenting at the depths in Table 1.

Fig. 5. Equidistant image curves.

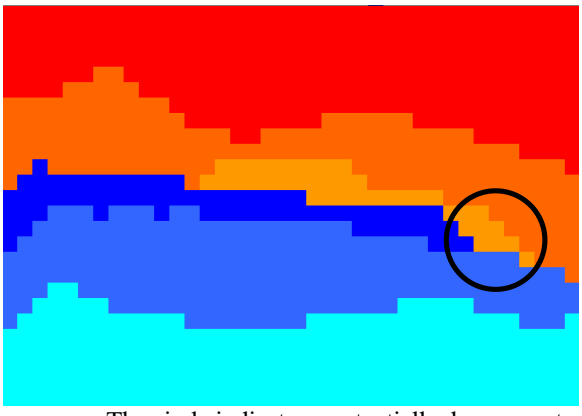

Fig. 6. Distance map. The circle indicates a potentially dangerous terrain region.

Distance Rules

Once we have obtained a distance map we apply heuristic evaluations, which we call "distance rules", to estimate qualitative terrain properties [6]. What do we mean by distance rules? Consider a simple example: if we scan the image and find that the furthest distance we can see is limited (e.g., tens of meters), we can conclude that an obstacle in front of the robot obstructs distant views. Similarly, if we can see continuously to the distant horizon, then we can conclude that the terrain is open.

Thus, using a lookup table with zone transition probabilities, we can approximate which transitions are dangerous or safe for the rover. For example if we look at the center right of the Fig. 6, we observe transition from blue to orange spans from a few tens to hundreds of meters and is likely to be dangerous (i.e., the occlusion may hide a sharp drop-off).

TOPOSEMANTIC VISION

Images contain a vast amount of non-geometric information (color, texture, etc.), which we can use to evaluate terrain characteristics. Consider, for example, the appearance of Fig. 7:

- The red circles show the areas that differ (in terms of texture, color or shape) from the rest of the image. A texture recognition algorithm, therefore, could enable identification of different ground coverage (sand, rocks, vegetation, etc.)
- The blue lines indicate areas where there is an abrupt change in image texture (frequency, direction, etc). Such changes in texture are most often caused by obstruction and changes in surface continuity. Applying frequency-based analysis would allow us to detect potential drop-offs and obstacles.
- The dotted red line marks the horizon line. Since we are only concerned with evaluating terrain, we can discard all image regions that contain the sky. This will speed processing as well as assist estimation of distant obstacles and slopes.

To extract and make use of this non-geometric information, we apply a variety of methods (including statistical analysis, morphological operators, and 2D filters).

Sky Detector

Knowing which portion of an image contains the sky is extremely useful. For example, we can employ the horizon line to estimate if the rover is going up or down, leaning left or right, etc. The sky, however, can have a varied appearance: clear or cloudy, different colors (notably at sunrise and sunset), and difficult to distinguish from the ground (e.g., when there is fog).

In the desert, the sky is generally homogeneous (i.e., clear or evenly distributed clouds). Our sky detection approach, therefore, is to find large (contiguous) image regions, which are uniformly colored. One way to do this is simply to reduce the number of image colors and then identify the largest blob.

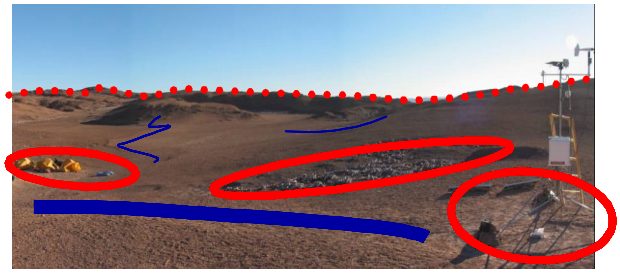

Fig. 7. Toposemantic terrain analysis.

Fig. 8. Sky detector results. Left, clear sky (day). Right, cloudy sky (dusk).

Currently, we are using a fast RGB-based method to do this [6]. In our tests, we have found that the algorithm is resistant to color variation (it is not dependent on pre-defined colors) and works well when the sky is clear, cloudy, or gradient colored. The algorithm also remains relatively stable in the presence of objects (e.g., boulders, occluding ridges, etc.) that cut the horizon line. Fig. 8 shows the detected horizon line for two common sky conditions in the Atacama.

Occlusion Detector

Image occlusions often indicate the presence of obstacles or geographic discontinuities. In order to detect occlusions, we need to identify image regions that are locally similar and then look for abrupt changes.

Our approach makes use of *perspective foreshortening*, the inverse square loss of resolution with distance. Specifically, because wide-area desert images generally contain uniformly textured surfaces (sand, smooth boulders, etc.), nearby objects appear with more image detail than distant objects. Thus, high-pass image filtering is an effective method for detecting occluding objects.

Fig. 9 shows a portion of Fig. 3, cropped to show a region from 0 to 100m downrange. Note the change in texture running diagonally from center-top to middle-right. This corresponds to an occluded drop-off. Fig. 10 shows the occluded regions detected by high-pass filtering.

Texture Classification

In our current system, we detect and classify texture using a Fast Fourier Transform (FFT). The basic approach is: (1) tessellate an image into uniform regions (a patch); (2) extract a frequency vector for each patch via FFT. At present, we divide an image into concentric circles and we create a vector for each circle (Fig. 11).

Fig. 9. Cropped image (taken from Fig. 3) showing 0 to 100m down-range.

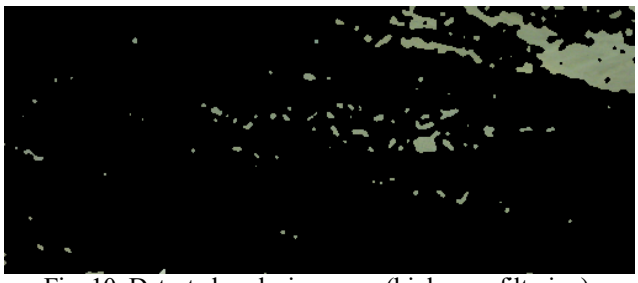

Fig. 10. Detected occlusion areas (high-pass filtering).

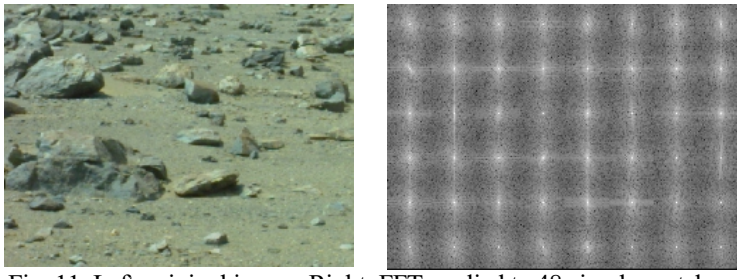

Fig. 11. Left, original image. Right, FFT applied to 48 circular patches.

Additional Detectors

In addition to the detectors described in this paper, we have implemented several other toposemantic detectors:

- Color detector
- Ground roughness map
- Rock detector
- Texture classification map
- Unique color detection

The design and implementation of these detectors is described in detail in [6].

TRAVERSABILITY MAP

By combining geometric and toposemantic information, we can produce a *traversability map.* This map allows us to directly estimate the difficulty of traversing the terrain and the location of danger zones.

As shown by Fig. 12, we build the traversability map by fusing information in two steps. First, we combine information from the ***rock detector map*** and the ***texture classification map*** in order to obtain an estimate of ground "roughness". We use the ***distance map*** to weigh the results based on down-range distance: closer regions are given more importance than distant areas.

The FFT of Fig. 3 (bottom left) is shown in Fig. 13. We use this FFT to produce the texture classification map (Fig. 14). This map shows both smooth regions (marked in green and light blue) and stony region (marked in purple).

By applying the distance rules to the distance map, we obtain the ***distances rules map***. Fig. 15 shows the result of processing Fig. 6. Red areas indicate100% danger and black areas indicate 75% danger.

Sub-sampling and thresholding the detected occlusion areas shown in Fig. 10, yields the occlusion detector map (Fig. 16).

We then sum the ***distances rules map*** and the ***occlusion detector map***. This produces a map with a coefficient of danger for each ground segment.

Finally, we produce the ***traversability map*** by combining the traversability measures from each map. Fig. 17 shows a traversability map overlaid on the source image: unsafe regions are shown with colored lines. The most dangerous areas are shown in dark blue.

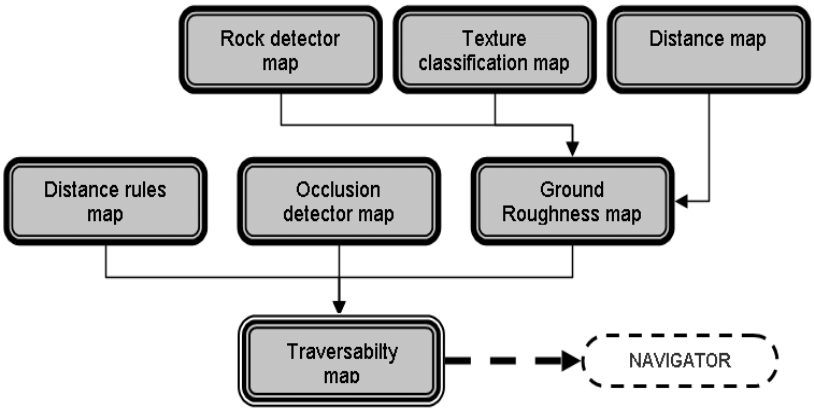

Fig. 12. Traversability map construction

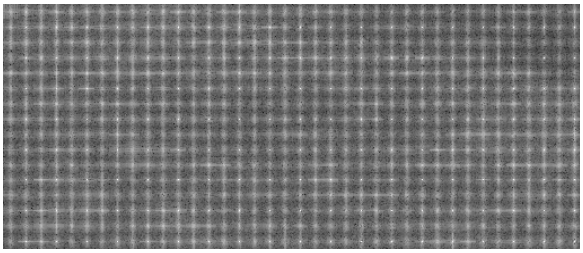

Fig. 13. FFT of Fig. 3 (bottom left)

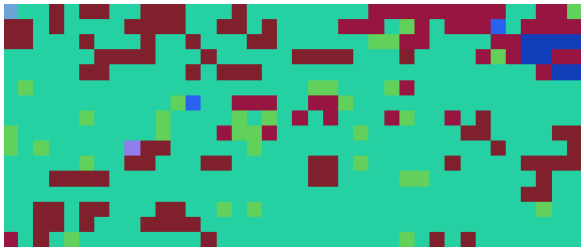

Fig. 14. Texture classification map

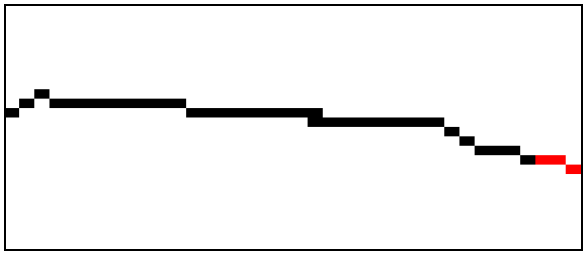

Fig. 15. Distance rules map

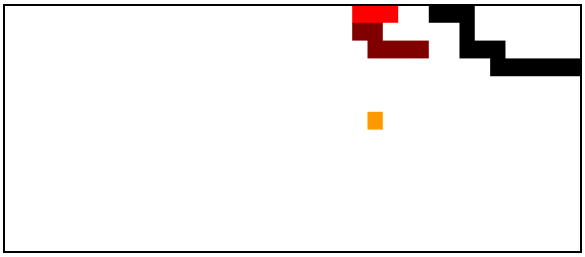

Fig. 16. Occlusion detector map

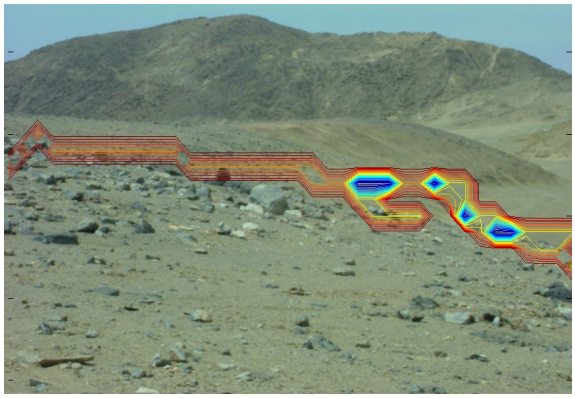

Fig. 17. Traversability map overlaid on source image.
The four dark blue regions indicate areas of significant danger.

FUTURE WORK

Although it is sufficient for our needs, our sky detector is simplistic. Replacing the current color reduction algorithm with principal component analysis would improve its robustness. Similarly, adding a learning method would improve performance, particularly under (rapidly) changing sky conditions.

The current traversability map only provides an indication of safe and dangerous regions. Although such information is important for navigation, the goal of planetary exploration is not simply autonomous driving, but rather autonomous exploration. Thus, it would be useful to generate a complementary "science target map", which would indicate regions of scientific interest. To generate this map, we would combine the traversability detectors with science detectors.

ACKNOWLEDGEMENTS

We would like to thank Sébastien Grange and Yannick Fournier for their assistance. This work was supported by NASA under grant NAG5-12890 (David Lavery, Program Executive).

REFERENCES

[1] M. Vergauwen, M. Pollefeys, L. Van Gool. *A stereo vision system for support of planetary surface exploration*, Machine Vision and Applications, Vol. 14(1), pp.5-14, 2003.

[2] C. Olson, L. Matthies, M. Schoppers, and M. Maimone, "Robust Stereo Ego-motion for Long Distance Navigation", In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2000.

[3] D. Lowe, Distinctive image features from scale invariant keypoints, *International Journal of Computer Vision,* 60(2), 2004.

[4] C. Graetzel, "Interface utilisateur basée sur les gestes visuels pour chirurgie", Technical Report, Virtual Reality and Active Interfaces Group, Swiss Federal Institute of Technology (EPFL), 2003.

[5] R. Tsai, "An Efficient and Accurate Camera Calibration Technique for 3D Machine Vision", In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1986.

[6] A. Avedisyan, Far-field terrain navigation, Microengineering Diplôme, Swiss Federal Institute of Technology, Lausanne, Switzerland, 2004.