[1]AIAA-01-4597

AUTONOMOUS ROCK DETECTION FOR MARS TERRAIN

Authors: V. Gor, R. Castano, R. Manduchi, R. C. Anderson, and E. Mjolsness.
Affiliations: Jet Propulsion Laboratory
California Institute of Technology
Pasadena, CA

ABSTRACT

In this paper we introduce a general framework for an image based autonomous rock detection process for Martian terrain. A rock detection algorithm, based on this framework, is described and demonstrated on examples of real Mars Rover data. An attempt is made to produce a system that is independent of parameters to ease on-board implementation for real time in-situ operation. The process utilizes unsupervised hierarchical approaches for object detection and is easily expandable to more complex data sets. Currently, it uses intensity information to detect small rocks and range information (derived from a pair of intensity images) to detect large rocks in the image. The range-based and intensity-based algorithms tend to be complimentary, with one working when the other fails, together they detect most of the rocks in Mars images. The Rock Detection System presented in this paper is one module in autonomous exploration system. This module closes the loop between data acquisition, data analysis and decision-making in situ. It can be used to prioritize what information will be sent back to Earth, where to take more scientific measurements using more time-consuming instrumentation, and which surface regions to explore further. In this manner the system contributes to reducing data downlink and maximizing science return per bit of data.

INTRODUCTION

Recent developments in technology have increased the gap between the ability of the instruments to collect the data, and the capacity of data downlink. Currently, we are capable of collecting much more data than we are capable of returning to Earth. Spacecraft and telemetry limitations place severe constraints on the scope of possible scientific return. However the continued growth of computing resources allows for the scope of planned investigations to be enlarged significantly beyond those considered in the past by maximizing scientifically important information content per bit of downlinked data. This is achieved through autonomous in-situ scientific data analysis. In this analysis scientific requests and reasoning behind scientific data analysis are coded into computer algorithms, which transform scientific intuition into an autonomous on-board system capable of directing the mission along the way of most scientific research in-situ, without heavy downlink or significant time delay. Good examples of missions where in-situ processing enables major increase in scientifically valuable data return are Mars surface exploration missions.

To a geologist, rocks provide clues for the paleo-history of the planet. Depending on the type and distribution of rocks (e.g. igneous, sedimentary, or metamorphic) found, a scientist can deduce what the area was like the time the rocks were being formed and deposited. By understanding the physical parameters such as the temperature and pressure, and the distribution of the materials on the surface, the environment the rock was created in (e.g. atmospheric conditions, interior conditions, surface conditions, etc) can be identified. All of these parameters can be recognize by looking at the physical properties and location of the rocks. For example, igneous rocks with large crystals can identify the depth of formation; the larger the crystal, the longer it takes for the rock to cool from the magma. In-situ volcanic rocks on the surface can indicate not only a source region, but can also be

[1] Copyright © 2001 by the American Institute of Aeronautics and Astronautics, Inc. The U.S. Government has a royalty-free license to exercise all rights under the copyright claimed herein for Governmental purposes. All other rights are reserved by the copyright owner.

used to distinguish the size of the volcanic body, volatile content and discharge rate.

Rocks can also explain the post-formational history (or secondary history) of the region. This history tells us what has happened to the rock since it was formed including such things as the effect of climate (i.e. weathering) and erosion and whether the rock has been transported. A sedimentary rock, a rock formed either by water, wind, or ice deposition, can identify the mode of deposition and how far the material has been carried just by the physical shape (or rounding) of particles. Angular fragments indicate that the source area is nearby; whereas a highly rounded particle indicates a large distance of transport. Rocks that form because of impacting can tell us the temperature of formation and the environment prior to impact. Therefore, a scientist can gain a tremendous amount of information just by looking at the distribution and the physical properties of rocks.

In future Mars surface exploration missions both mobile robotic platforms and landers will have the capability of collecting more sensor data than can be transmitted to earth. It will be critical that these platforms are capable of analyzing information onboard and selecting data that is most likely to yield valuable scientific discoveries. In this paper, we describe a system that can be used to focus efforts on regions in a scene that are of interest to the scientists. At the most basic levels such regions are the ones where high-priority targets are identified. We describe methods that will enable a rover to identify and statistically characterize possible targets by identifying rocks in an image.

Image data is of particular interest for scientists. Cameras are a common instrument for data collection (the MER rovers scheduled for launch in 2003 will have ten cameras onboard). Cameras are inexpensive instruments in terms of both time and energy to collect data, but are very demanding in terms of storage space or transmission bandwidth. It is evident that images are an example of data in which more can be collected than can be stored or transmitted to Earth. Further, image data is very rich in information content and can provide a great deal of insight about the planetary surface.

ON-BOARD SYSTEM CONFIGURATION

In the past, our research has focused on a variety of autonomous science processing systems, which were based on different scientific goals. Some of these systems required algorithm training and labeling of objects, which has to be done by scientists on Earth. Such systems upload compact feature recognizers produced by scientists, then using these recognizers they detect features of scientific interest in image data and downlink only the scientifically interesting data to Earth. An example of such system is described in the downlink report [(1)], which provides feasibility and cost efficiency estimates of an on-board science feature extraction module. This system incorporates supervised-learning techniques and will be explained in next section.

Other systems require more or less frequent parameter settings, which are also done manually on Earth. Two examples are the autonomous change detection software[2] integrated within the context of Mars orbiter simulation and the autonomous satellite (small-body) detection algorithm[3,11] implemented in mission simulator performing autonomous repointing for close observation of satellite. The latter system was incorporated within our Flight System Testbed, which supported communication of science processing software with an autonomous camera controller and planner. All these systems require real-time interactions with Earth for either training or parameter setting. In this paper we will introduce a framework, which stresses systems independence of the parameters and of real time interaction with Earth-based scientists.

Such systems are often imposed onto missions with limited DSN availability. The less DSN communication necessary, the easier the integration of the science-processing module within the spacecraft and the simpler and cheaper the mission design and implementation are. In the design of a spacecraft system, one should strive for full autonomy and limit the number of exports as much as possible. Such a goal will not only reduce the load on the communication network (downlink and uplink), but will also insure system operations during times when the network is not available (often a significant amount of time). Here we attempt to develop a Mars Rock Detectort hat is a parameter-independent (or fully autonomous) system for use on the surface of Mars.

The Rock Detection System presented in this paper is only one module in the autonomous exploration system. This module is in the loop between data acquisition, data analysis and decision-making in real time on board the spacecraft. (Figure 1). Some of the decisions may involve camera repointing or replanning of optimal rover path in order to further investigate objects of interest. They also can involve significant data reduction, where non-interesting or

repetitive data is ignored, and only the statistical summaries of most regions along with the details of scientifically interesting regions are downlinked to Earth.

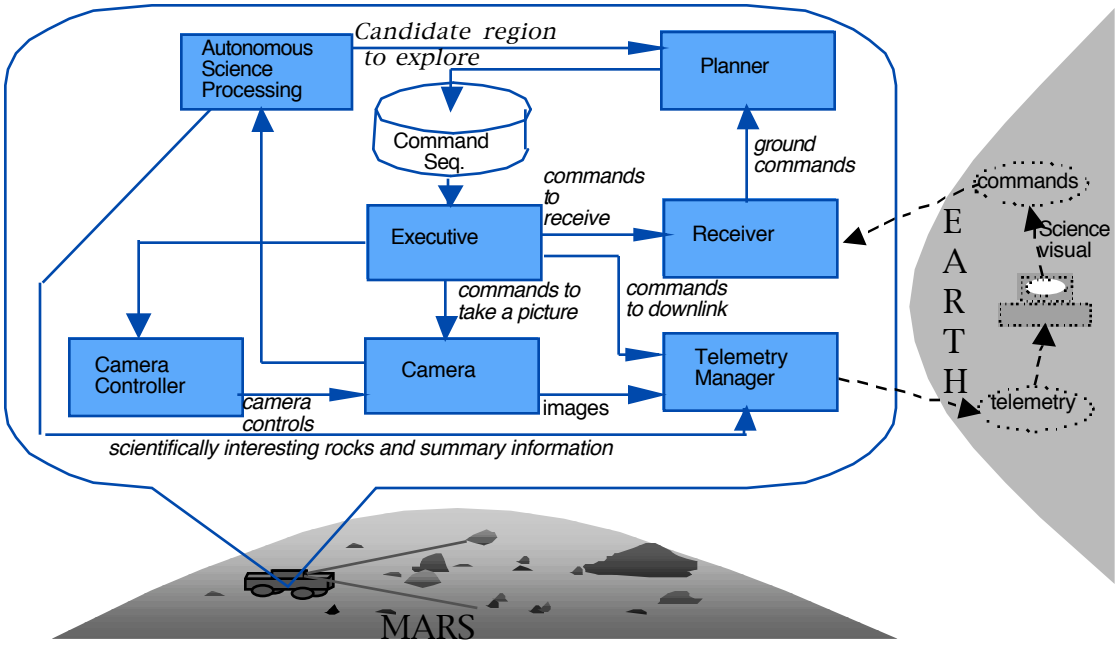


Figure 1. An example of autonomous exploration system.

THE PROCESS OF DETECTING ROCKS

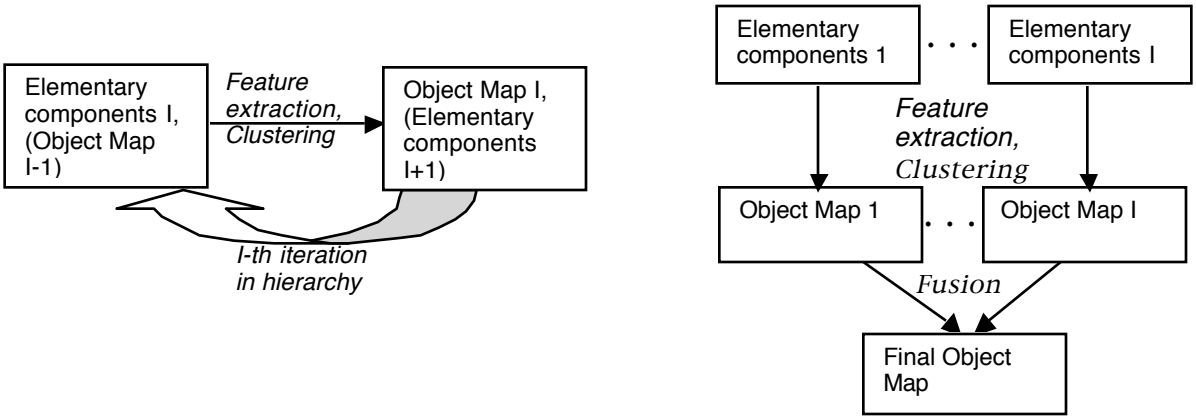
Background and goals

There are many methods for recognizing objects in a scene. Since humans are especially good at performing this task, many of these methods try to mimic human visual processing (neural networks, for example) in their attempt to recognize patterns[5]. These types of approaches fall into the category of supervised learning techniques, where the probabilistic models (or other generalized models) of object attributes (characteristics) are learned from numerous examples. The same object attributes are then computed for a new image region, and each region is characterized based on previously learned attribute distributions[1]. These techniques are good not only for detecting objects, but also for dividing them into different types specified in the initial step. These methods are excellent for creating interactive tools for scientists working on missions with available downlink/uplink and with a science team trained on the algorithm. Nevertheless, in many missions sufficient downlink is not available to allow science-rover interactions for science data analysis. For example, in future Mars rover missions, the path-planning unit (Figure 1) can be required to plan and replan rovers path and operations to maximize scientific return real time without any delays imposed by communications with Earth. In addition, there could be fleet of rovers (swarm intelligence) collecting and processing data, communicating and reaching decisions[6]. In these cases, having humans heavily involved in these highly synchronized real-time loops is practically impossible. This is why often in the initial design of the spacecraft system, stricter system requirements are required. Such are the requirements for independent space-vehicle operation and full (or minimal human-interaction) autonomy.

Full autonomy can only be approached through unsupervised techniques. These are the techniques in which no examples of object data are provided a priori, and few details and/or parameters are specified. The major components of an unsupervised image-processing system are feature extraction and clustering). In feature extraction, elementary components (regions from which features will be derived) are segmented from the image, and the features summarizing component characteristics which are representative of an object's unique signature are extracted for every component. This signature might be represented by a set of rules acting on the feature or its elementary components. In clustering, the features are divided or linked into compact groups based on data representative distance criterion. By alternating

between feature extraction and clustering, one can recognize objects in the scene without heavy dependence on parameters or prior data analysis.

A feature extraction/clustering loop can be hierarchical or symmetric. In hierarchical analysis the process builds more complicated features from the simpler ones, producing at each step an improved representation of the target objects until desired accuracy is achieved (Figure 2a). In each iteration, new elementary components are represented by objects produced in the previous iteration. In symmetric analysis (Figure 2b) initial representations of the scene are depicted with orthogonal elementary components (orthogonality implies that detection sets derived from such components are mostly non-overlapping). These representations are processed independently, each recognizing different types of the same objects. Both hierarchical and symmetric approaches reduce algorithm dependence on the parameters and increase the robustness of the algorithms, making every later component in the processing independent of changes in parameters in the previous module. In the system described in this paper we use both hierarchical and symmetric techniques for object detection.



a) Hierarchical object recognition

b) Symmetric object recognition

Figure 2. General framework for object recognition.

Even though, our current roadmap leads to a system with very few parameters, we realize that no system is truly parameter independent or assumption free. In this study we approach parameter independence by avoiding extensive learning techniques and allowing only those parameters, that either prove to be robust (can be set once, relying on tolerance of later algorithm components to changes in this parameter), can be set automatically (from image information), or are system mission dependent and can be set a priori (derived from camera and mission parameters).

High-level rock signature definition

The appearance of rocks in an image varies greatly from one image to another. Rock texture and color in normal camera image are affected greatly by camera view angles, lighting source and weather conditions. Some rocks can appear flat, grainy, bright or dark depending on time of the day the image was acquired. At a basic level, raw image data provides only partial characteristics of rocks. Nevertheless, in most cases the texture and intensity of a rock are distinct from background texture and intensity. Therefore, we define a primitive rock feature in the intensity image as a small region with texture and intensity different from its neighbors (Figure 5f, 6f, 7b). Such regions of distinct texture and intensity, if isolated in the scene, will represent a rock. This method is only effective for detecting small rocks, or big rocks at far ranges, that consist of a small percentage of pixels in an image. Both geometry and lighting affect small rocks the least, and in the desert-like environment, similar to Mars's surface, compact small regions of uniform intensity and color are good indicators of small rock presence. Large rocks, on the other hand, or rocks located close to the camera consist of many non-isolated regions of uniform texture and intensity. A simple method is used to combine non-isolated uniform texture and intensity regions (Figure 6f) into larger rock regions. Different rock features are used to segment larger rocks from the image.

Since rovers often have more then one camera on-board, range information can be derived from intensity information using disparity or pixel displacement from left image to the right[7].

Range–based obstacle avoidance is necessary for autonomous navigation, and therefore we can safely assume that range data is present on-board. The MER rovers in '03, for example, will rely on stereo range data for extended autonomous traverses.

Range data is usually not available for each pixel of original image. Some areas are be occluded by larger rocks; others do not have enough gradient information in them to calculate pixel displacement (pixel displacement is calculated through correlation techniques). Nevertheless, large rocks generally have abundant gradient information, assuring accurate range calculations. In fact, the abundance of gradient information in images in which a rock covers a large portion of the image tends to cause the intensity and texture based segmentation to fail to isolate the large (Figure 6f). In this study, we derive the height data (height for each pixel) from the range information and define a rock in the height image as a connected volume rising sharply above its local background. But for the recognition of smaller rocks height data usually lacks resolution. Therefore, it is better to use intensity-based algorithms for small rock detection and range-based algorithms to detect large rocks.

In this section, the intensity-based algorithm that detects small rocks in the image and the range-based algorithm that detects larger rocks are described. Examples were both algorithms are symmetrically executed on the same images combining (fusing) results, are presented.

Intensity based segmentation and rock detection

The main component of the intensity-based small rock detection is the edge–flow segmentation process[10]. This process utilizes a predictive coding model to identify the direction of change in color and texture at each image location at a given scale, and constructs an edge flow vector. The approach facilitates the integration of different image attributes such as color, texture and misleading discontinuities into a single framework for boundary detection. The algorithm first computes local edge energy and estimates the corresponding flow direction. The local edge energy is iteratevely propagated to its neighbor if the edge flow of the corresponding neighbor points in the similar direction. The edge energy stops propagating to its neighbor if the corresponding neighbor has an opposite direction of edge flow. In this case, these two image locations have both their edge flows pointing at each other indicating the presence of a boundary between two pixels. After the flow propagation reaches a stable state, all the local edge energies will be accumulated at the nearest boundaries.

The algorithm requires one significant control parameter: image scale. It controls both the edge energy computation and the local flow direction estimation, so that only edges larger than the specified scale are detected. For the purposes of segmenting the image into *small* components, it is safe to set image scale permanently to one small number. Moreover, since we are aiming for a non-parametric approach, this parameter must be set a priori. We selected a constant image scale of 5, and produce an image consisting of small regions (primitive components) that are separated by detected boundaries.

Although the edge-flow algorithm relies heavily on texture and color information, it uses the direction of change in these attributes to detect component boundary. The algorithm produces smooth and closed boundaries (Figure 5f, 6f, 7b), but it does not consider the actual average values of texture and intensity attributes within each primitive component for the purpose of linking these components together into larger regions. Therefore, the additional module that links these components (Figure 3) is necessary to separate large and mainly uniform background from small rocks.

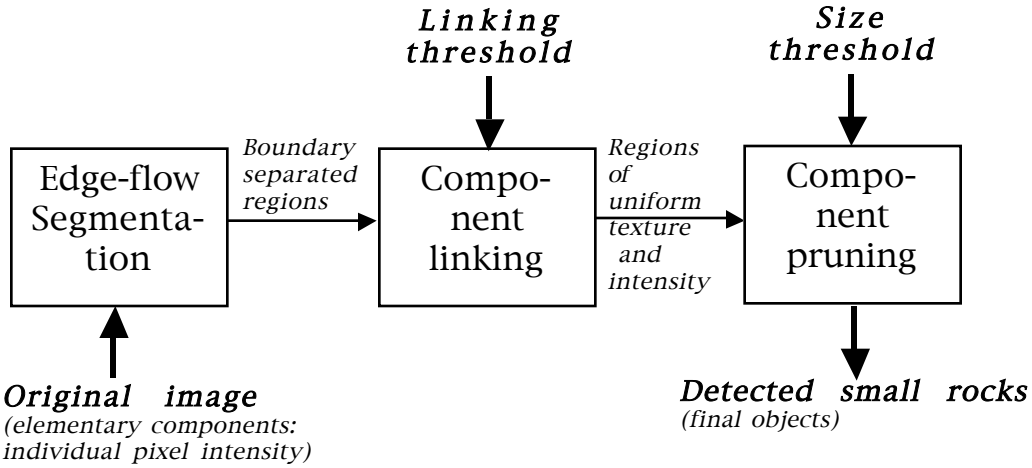


Figure 2. Intensity-based small region segmentation process.

The component linking module iteratively links adjacent components together, starting with the two components closest in intensity, if their average intensity difference is less than a specified threshold. Obviously, a linking threshold is required by this module. This threshold is surprisingly robust for small rock detection. In our experiments we were able to keep it constantly set to 10. Nevertheless, for truly autonomous system this threshold must be set automatically based on image information[4]. One possible method to set this threshold is to calculate the differences in intensity using the histogram for all adjacent components and to cluster them into two classes: one with smaller differences (corresponding to regions which must be linked), and another one with large differences. These classes will be separated and most of the adjacent regions in the image should correspond to background and do not differ in intensity. Automatic setting of linking threshold has not been developed yet and is an immediate goal for future development of the algorithm.

The general idea behind component pruning is to cluster all components into two groups based on some component attributes: rock surfaces and background. An attribute intuitively imposed by the algorithm and aimed at detecting small rocks in desert-like background is size. Since the previous module linked similar components together, the background component is being linked into big continuous regions, while differently textured rocks, remain as small isolated regions. Ideally size distributions should cluster into large and small, but currently the size threshold is set manually mainly because intensity-based algorithms are still used to detect large rocks in smeared images.

The Edge-flow segmentation process can be viewed as feature extraction and clustering acting on elementary components: pixels, and producing objects: boundary separated regions. Component linking and pruning basically perform the functions of clustering and feature extraction acting on a feature's average intensity, derived from new elementary components: boundary separated regions. This is the hierarchical process. Final detected rocks are used to produce an object map (Figure 5g, 6g, 7c), which can be used later in the symmetric process of combining different maps.

Range-based segmentation and rock detection

While intensity information is readily available from visual cameras, range infomation is derived via measuring displacement of image pixels in two images taken from different viewpoints. Such displacement is measured by correlating local visual information in two corresponding images[7]. The range data is originally given in Cartesian coordinate system (X-Y-Z coordinates for each pixel). Such range measurements are dominated by the horizontal distance to the observed point, which is not explicity relevant to rock detection. The information critical to rock detection is the height of the rock above the ground plane, since rocks are expected to be higher than sand and general background.

Rock height information (Figure 5b, 6b) is extracted from the range data (Figure 5a, 6a) by fitting a least-squares plane (ground level) to the range data and then calculating the distance between each range

point and the plane. Initially, a plane is fitted into all given points (at coarser resolution), then the error is calculated between plane points and original points for every pixel. The errors are assumed to be Gaussian-distributed (reasonable assumption for relatively flat terrain). Therefore, all outlier points located further than three standard deviations from mean are rejected (eliminated from point set), and the plane is refitted using the reduce number of points. The process is repeated until a small number of points (not less then 100, for example) remain in the set. To fit the plane coefficients into the large set of points, the Moore-Penrose Pseudoinverse[8] is used as shown below. This pseudoinverse minimizes the least squares error between the estimated plane and the original points, while minimizing the norm of the plane coefficient vector. The method is very efficient, since it requires the computation of inverse for only 3x3 matrix, independent on number of points involved in the fit. The method is as follows.

We assume a Cartesian representation of the plane: $C_1X_i + C_2Y_i + C_3Z_i + C_4 = 0$, and assume the usage of range coordinates for N pixels of the image. C_1, C_2, C_3 and C_4 are plain coefficients and X_i, Y_i, Z_i are coordinates of range for pixel $i (i < N)$. The plane equation can be rewritten as: $A_1X_i + A_2Y_i + A_3Z_i = -1$, where $A_1 = C_1/C_4, A_2 = C_2/C_4, A_3 = C_3/C_4$. The overall linear system is depicted in (Eq. 1)

$$\begin{bmatrix} X_1 & Y_1 & Z_1 \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ X_N & Y_N & Z_N \end{bmatrix}_{R} \bullet \underbrace{\begin{bmatrix} A_1 \\ A_2 \\ A_3 \end{bmatrix}}_{A} = \underbrace{\begin{bmatrix} -1 \\ \cdot \\ \cdot \\ -1 \end{bmatrix}}_{U} \qquad \text{(Eq. 1)}$$

The solution is given by

$A = R^+ \bullet U$,

where R^+ is the Moore-Penrose pseudoinverse R. In our problem, the number of points N is always larger than the number of unknown plane coefficients (3), therefore we use the full-column rank form of Moore-Penrose pseudoinverse:

$R^+ = (R^t \bullet R)^{-1} \bullet R^t$ (Eq. 2)

Combining (Eq.1 and Eq 2) we get the plane coefficients:

$A_{LS} = (R^t \bullet R)^{-1} \bullet R^t \bullet U$ (Eq 3)

And the minimized Least Squares Error is given by

$E_{A_{ls}} = \|U - R \bullet A\|^2$ (Eq 4)

The resulting height image contains more accurate rock-related information than range image (Figure 5ab, 6ab) and makes the overall rock-detection process significantly more robust for all ranges.

Once the height data is obtained, in an ideal world, one could theoretically detect rocks by selecting connected regions over certain heights. But because of the errors in ground plane calculation (at large scale the ground is not necessarily planer) global thresholds in height become unreliable. It is necessary to use only relative heights which preserve local ground to rock relationships. We bin the heights through k-means segmentation[9] into a moderately high number of K bins, and we analyze the relationship between connected components of the same height and their neighbors (Figure 5c, 6c). At this stage, connected components of same heights become elementary components for Rule-based rock synthesis.

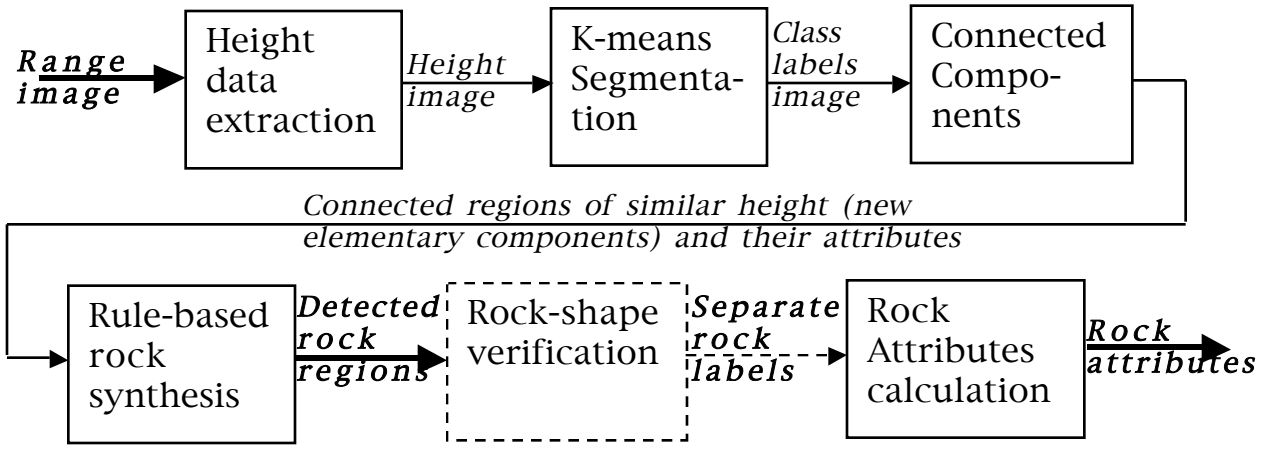


Figure 4. Range-based large rock segmentation process.

The rock synthesis process is designed in such a way, that the actual value of K is not important as long as it is large. We permanently set the value of K to 21. Note, if the K-means algorithm fails to converge, it means that there is not enough variance in height data to split it into K classes. Were this to occur, K can be automatically set to lower number, but this iteration may not be necessary, since non-convergence also indicates inadequate range data for the rock detection method using range data. In this case, we are most likely observing the scene at far range, and intensity-based algorithms should be used.

The K-means segmentation process has an interesting affect on the height data. This process assumes Gaussian distributed classes, but, in reality, only heights corresponding to background are Gaussian distributed and fit well into a few (ideally 1) k-means classes. We naturally get few well-spread connected components representing background (Figure 5c, 6c). The height distributions for rocks and dips depend heavily on the perspective angle and rock/dip shape. These distributions are rarely Gaussian; therefore many more classes are necessary to represent rocks (the K-means algorithm approximates non-Gaussian distributions by fitting large number of Gaussian clusters in it). As a result we get heavily populated classes for background, and lightly populated numerous classes for rocks (Figure 5c, 6c). Rocks appear as layers of monotonically decreasing elementary components, rising above wider spread background regions. From a shallow perspective angles (or side view of the rocks) elementary components representing rocks appear as narrow elongated regions. For steep perspective angles, as in a top down view, rock sides appear to be much wider. In this manner, K-means label map gives a view of regions topological information, from which rocks can be extracted.

By moving from absolute height data to connected components of same inter-region heights we gain three significant advantages. First we obtain the representation of the data where rocks exhibit a unique signature. Second, we provide non-parametric (assume K is always 21) data smoothing, which eliminates small height variations. And finally by replacing pixel data with larger connected regions of similar internal height, we achieve significant data reduction and algorithm speed-ups during rock synthesis step. Now, rocks have to be built from larger components rather than from individual pixels.

The rock synthesis module (Figure 4) builds rocks from elementary components by examining only relative positions and heights of topological regions. Shape-describing attributes of topological regions such as elongation and size are presently not used. Elongation $\left[\frac{(outside\_perimeter)^2}{area}\right]$, even though an extremely valuable attribute for shallow perspective angles, is not generally reliable since it can vary greatly with perspective changes. Size attribute is often a great discriminant of background components, since background classes are mostly the larger populated classes in the image, but a size discriminator fails if the background is heavily populated by rocks or if we observe a group of rocks at close view. In these cases the background can be split into smaller regions. In an attempt to achieve generality and parameter independence we start by

concentrating on only relative (binary) height and position attributes between components, such as higher or lower, adjacent or non-adjacent. We use a number of non-quantitative rules to describe rock height-based signature and separate rocks from background. Three rules have been implemented and others are under consideration.

In order to describe the rules, a few definitions must be made for clarity:
Neighbors – adjacent components in the image;
Lower neighbor – adjacent component of lower height;
Tops – elementary components whose neighbors are all of lesser height;
To point up – one component points up to another component, if all intermediate neighbors while moving from first component to the later are monotonically increasing in heights;
Sides – adjacent components with monotonically changing heights, that point up to the same set of tops.

With these definitions in mind the rules can be stated:
1) If the component points to only one top, then the top is the top of the rock and current component is part of the same rock as top. This rule uses the following topological characteristic of the scene: if more then one local tops are surrounded by same height region, this region tends to represent background, and rock tops tend to be tops of separate rocks.
2) If the component points up to more than one top, but there exists a lower neighbor of this component that points up to the same tops as a component itself (that is if side exists), then these tops and component represent the same rock. Moreover, all the intermediate neighbors between the component and the top are also part of the same rock. This means that if a down-slope signature of a surface is distinctly obvious, the surface must be side of the rock.
3) The remaining regions are backgrounds.
These rules basically emphasize the tops and the sides as unique rock signature characteristics.

The rock synthesis module outputs regions with detected rocks. This is the current output of the process and current final objects. In this output, smaller classes of rocks are sometimes combined into the same region, and often rock outlines are not very accurate. The hierarchical implementation allows further improvement on detected rocks. One of our goals is to implement further segmentation of rock regions into separate rocks, while improving on rock outlines. This will be done in a rock shape-verification module (currently not implemented). In this module we plan to fit elliptic parabaloids into rock regions, then calculate the error between the actual rock surface and parabaloid. We hope that the presence of continuous gradient in error will indicate the boundary between two rocks in rock region. We also assume that if the determined parabaloid continues to fit height data beyond the outlines of the rock region, such outlines should be extended.

The last module, the Rock Attribute Calculation module currently calculates the attributes of rock regions, such as average intensity, height, area, compactness, later they will include parameters of the fitted elliptic parabaloid, and possibly will provide various texture attributes if requested by the scientists. Such attributes will be used for rock classification, which might integrate supervised approaches and is beyond the scope of this article.

Fusion of intensity-based and range–based rock detection results

Experimental results have shown that combining intensity and range attributes for all-size rocks detection early in the hierarchy generates significantly more false alarms, and does not aid detection rates significantly. There are three reasons for this. The first one is that these two algorithms are truly orthogonal in a sense that intensity-based approaches usually succeed where range –based approaches fail and vice versa. This is happening because range data is usually good when there is enough gradient present in the data and the correlation results are accurate. Same gradient interfere with texture/intensity segmentation making it worst. If images look smooth and blurred (low-frequency image), rocks appear as blobs surrounded with slowly-varying background and intensity-based algorithms work well even for big rocks, while range-based algorithms fail (inaccurate ranges). On another hand, if good range information is present for even smaller rocks, we use range for rock detection, since range is generally more reliable attribute of rocks. Therefore, good range information takes precedence over intensity.

The other reason that the range and intensity attributes are not currently fused early in the hierarchy is the absence of well-corresponding data, meaning that range data is often offset away from real rock, especially near the boundaries of the image (Observe the offset of range-based detected rocks in the bottom left of image in Figure 6h). In the future, if the correspondence problem between range data and intensity data is solved, we plan to

combine intensity and range information to improve range-detected rock outlines. This can be performed by extending rock outlines to the boundary of overlapping intensity-based elementary component.

Finally range information is often absent in parts of an image, either because of the occlusions produced by big rocks or because of insufficient gradient to correlate pixels between the left and right images. In addition range calculations can not be performed near the boundaries of the image because of the area required by correlation window size and displacement search region. Due to these reasons, we prefer to fuse *final* detections of range-based and intensity based algorithms (symmetric approach). We use intensity-based algorithms for small rock detection and range- based algorithms for large rock detection (as described in background section). But for some data sets (example 3) intensity data is so smeared and far-viewed, that range does not have enough resolution to be useful. In other data sets (example 2) rocks are imaged at such a close range, that small rock detection does not add much to results. Nevertheless, on many images (Example 1) both small and large rocks are of interest. In these cases we detect whatever size rocks we can detect with range data, and then detect all the smaller rocks with intensity data. If intensity and range-based detections overlap, range-based detections always take precedence over intensity-based detections. The examples of this approach are demonstrated in next section.

The overall symmetric framework (Figure 2b) of fusing rock detection maps based on different data features allows for easy plug in of additional object maps based on different attributes and possibly recognizing other types of rocks in new data. For example, region compactness attribute $\left[\frac{perimeter^2}{area}\right]$ is an excellent indicator of a region's structural texture. (In structural texture "texture primitives" are detected and their arrangements are used to generalize texture structures. Currently, we are using statistical texture only: Gabor filters). The compactness attribute might allow us to see shallow smooth rocks on sandy granular background, even if the rocks are same color as the sand. In this case we would introduce a new object map and combine final detections.

ALGORITHM PROCESSING EXAMPLES

In this section we demonstrate the performance of rock detection algorithms on three examples. The first two examples are real images of Mars surface taken by Mars Pathfinder Rover. The rover had a pair of cameras on-board (IMP or Imager for Mars Pathfinder) for stereo imaging allowing range to be generated from "Left eye/Right eye " image pair. The third example is taken from FIDO test. Although the FIDO data contains both intensity and range information, the range data lucks resolution and does not contribute to the overall rock detection.

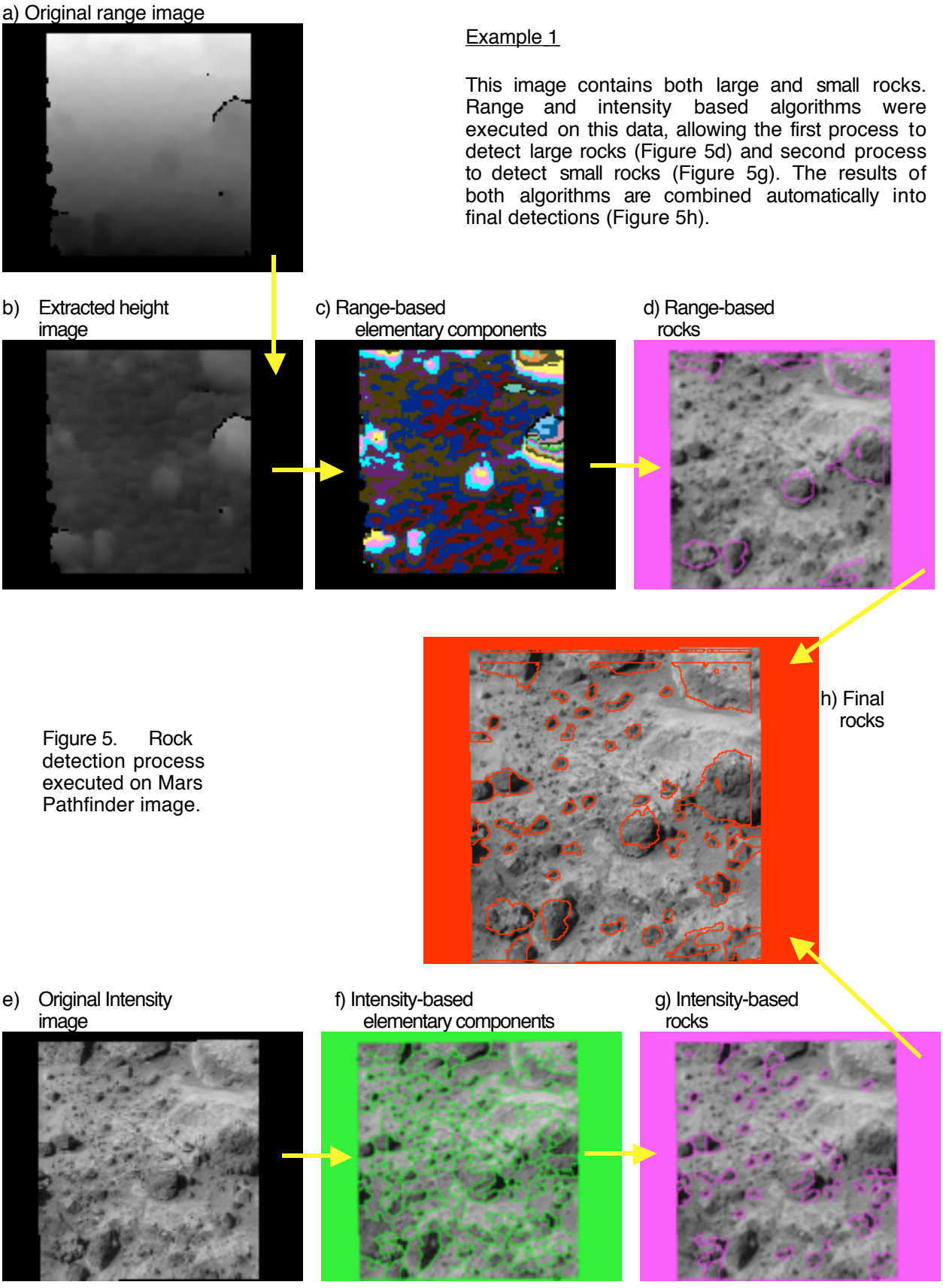


Figure 5. Rock detection process executed on Mars Pathfinder image.

Example 1

This image contains both large and small rocks. Range and intensity based algorithms were executed on this data, allowing the first process to detect large rocks (Figure 5d) and second process to detect small rocks (Figure 5g). The results of both algorithms are combined automatically into final detections (Figure 5h).

Example 2

This is a close view of rocks on MARS. At close views, range data is especially good (even though it is offset from the rocks near image-boundaries). Range based algorithm takes precedence over intensity-based algorithm and detects most of the rocks in this image. Intensity based algorithms are also ran on this data, even mostly tiny pebbles are left for it to detect. Such small rocks probably do not contribute much to science knowledge, and at very close views intensity-based rocks can be generally ignored.

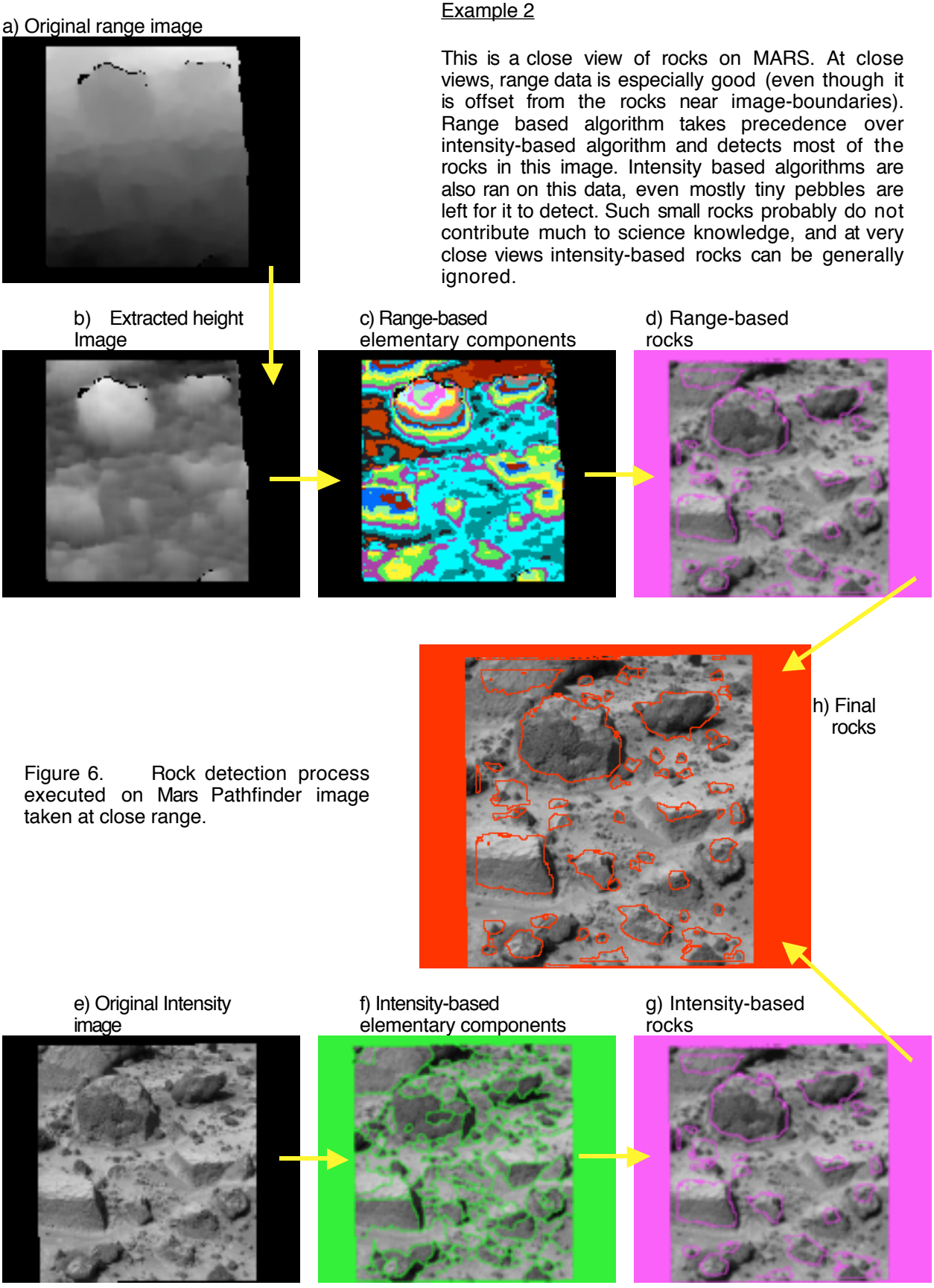


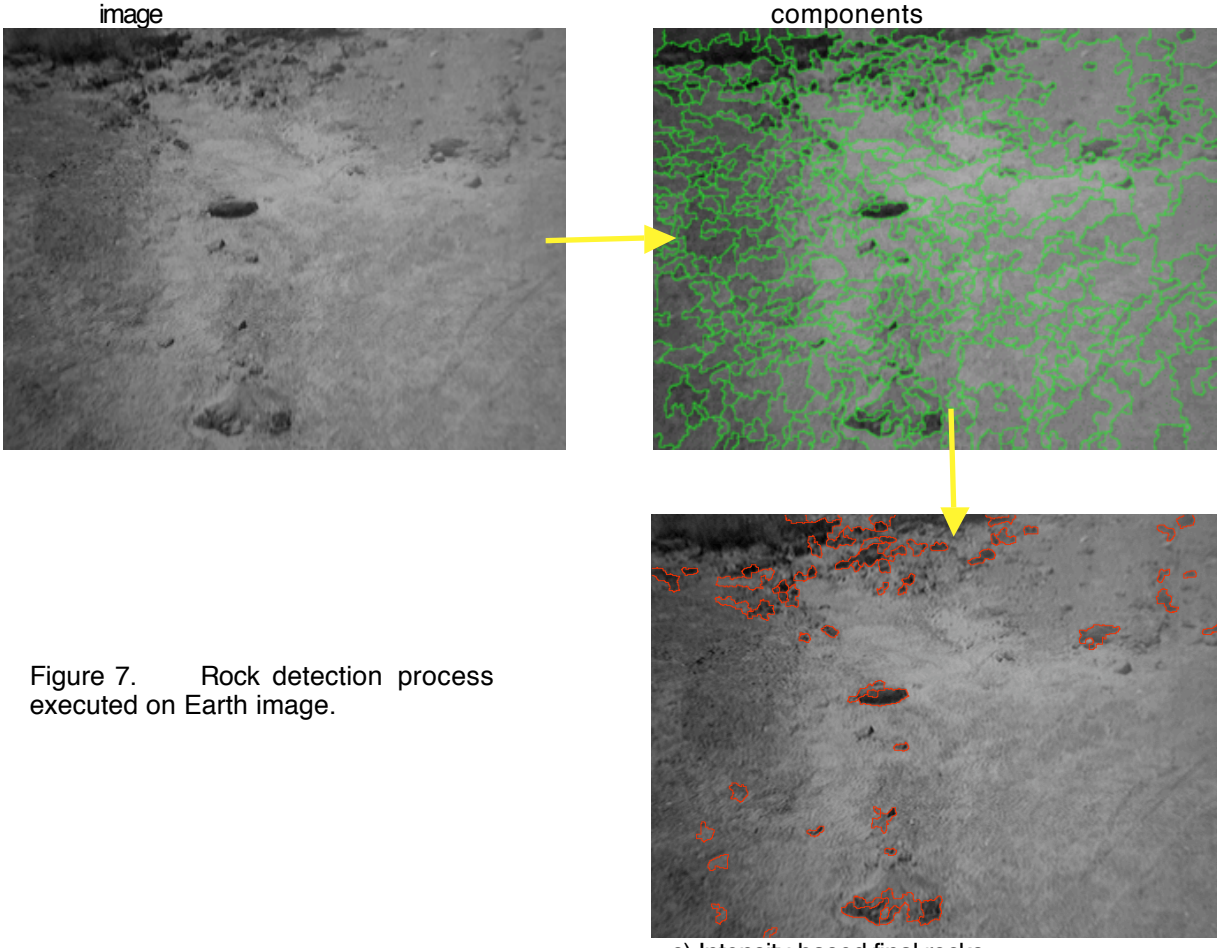
Figure 6. Rock detection process executed on Mars Pathfinder image taken at close range.

Example3

This image was taken on Earth at relatively far range. It contains some vegetation and is blurred. The range data lacks resolution and basically does not contribute to final results. Nevertheless the intensity-based rock detection works reasonable on this image (results are presented in 7c).

a) Original intensity image

b) Intensity-based elementary components



c) Intensity-based final rocks

Figure 7. Rock detection process executed on Earth image.

CONCLUSIONS AND FEASABILITY FOR ON-BOARD IMPLEMINTATION

In this paper we introduced a general framework for fully a automated rock detection process to be used on the surface of Mars, and demonstrated the initial version of the process. The process utilizes both symmetrical and hierarchical approaches for feature extraction and object recognition and is easily expandable to more diverse data sets.

Implemented on-board a rover, the described system could aid in deciding what information to send back to Earth. One of the major requirements imposed on our system is feasibility and ease of on-board implementation. At the level of algorithm development, three issues must be addressed to justify the ease of on-board implementation: fast execution time, parameter independence and radiation tolerance. Parameter independence was the topic addressed in this paper. We demonstrated the system that can detect rocks without extensive training and with minimal use of parameters. Specifically, only robust parameters and parameters that can be set automatically were used in this algorithm. In addition, the presented algorithm is radiation tolerant and cosmic noise resistant since the algorithm looks at image components of larger

sizes then normal cosmic noise hits, and features tolerant to small noise speckles are extracted from the image regions. Generally, if features larger then cosmic noise ray hits (which are usually few pixels large) are of interest, cosmic noise can be successfully removed by filtering out isolated small regions from the image.

The current implementation of the rock detection algorithm executes on a dual-processor Sparc Ultra 60 in less then 2 minutes. We believe that with optimal implementation of the algorithm and with off-the shelf image processing hardware and/or parallel hardware, real-time speeds will be achieved, ensuring the feasibility of rock detection system's implementation on-board the space-vehicle.

Acknowledgments

The research described in this paper was funded by the TMOD Technology Program and the NASA Remote Exploration and Experimentation task and was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

REFERENCES

1) Wayne Schober, Faiza Lansing, Keith Wilson, "High Data Rate Instrument Study", JPL Publications 99-4.

2) "Quake finder", http://www-aig.jpl.nasa.gov/public/mls/quakefinder.

3) V. Gor, P. Stolorz, T. Mann, W. Colwell, "DARE as an On-board Science Processing System for Spacecraft, ", Proc. AIAA Conf., 1999.

4) Nobuyuki Otsu, "A Threshold Selection Method from Gray-Level Histograms", IEEE, 1979.

5) B. D. Ripley, "Pattern Recognition and Neural Networks", Cambridge University Press, 1996.

6) T.Estlin, G.Rabideau, D.Mutz and S.Chien, "A Dynamic,Distributed Planning System for Coordinating Multiple Rovers", Third International Symposium on Robotics and Automation (WAC(ISORA)),Maui, Hawaii, 2000.

7) L. Matthies, "Stereo vision for planetary rovers: stochastic modeling to near real-time implementation, "International Journal Computer Vision, Vol. 8, No. 1, pp. 71-91, July, 1992.

8) Peters, G. and J.H. Wilkinson, "The Least Squares Problem and Pseudo-inverses".

9) Christopher Bishop, " Neural Networks for Pattern Recognition", Clarendon Press, Oxford, 1995.

10) W. Ma and B. Manjunath, "Edge Flow: A Framework of Boundary Detection and Image Segmentation", 1997 IEEE.

11) "Software Simulates Observations from a Spaceborne Camera", NASA Tech Briefs, V21, #11, November 1997, p.73.

12) "Autonomous Small Planet In situ Reaction to Events (ASPIRE)", http://www-aig.jpl.nasa.gov/public/mls/aspire/aspire.html.