# AUTOMATED ROCK SEGMENTATION FOR MARS EXPLORATION ROVER IMAGERY

Yonghak Song

School of Civil Engineering, Purdue University
West Lafayette, IN 47906, USA - song10@purdue.edu

**Commission IV WG 7**

**KEY WORDS:** Active Contour, Edge Flow, Feature Extraction, Level Set Method, Mars, Multi-resolution, Segmentation, Texture

**ABSTRACT:**

Rock segmentation is important for the success of the Mars Exploration Rover mission and its scientific studies. In this paper, a framework for automated rock segmentation using texture-based image segmentation and edge-flow driven active contour is developed and implemented. Three schemes: wavelet based local transform, multi-resolution histograms, and inter-scale decision fusion are combined and applied for texture-based image segmentation. The result is refined by active contour based on level set method, which is propagated in the edge flow vector field. Test images taken by the panorama and navigation cameras on the rover Spirit at the Gusev Crater landing site are used in this study. This paper presents the theory, implementation, and the test results along with discussions on the performance of the proposed method.

## 1. INTRODUCTION

The Mars Exploration Rover (MER), Spirit and Opportunity have collected a large amount of Mars surface imagery since their arrival on Mars in 2004. As the most important tasks of the MER mission, route planning and geologic analysis demand the identification of observed rocks. For route planning, rocks must be detected before producing rock maps at the landing sites. In terms of geologic and planetary science, rocks might hold the clues to past water activity and carry important information about environmental characteristics and processes.

Rock segmentation in an image is essential for rock mapping. Currently, rock segmentation in MER imagery is mostly accomplished by manual labelling which is extreme time consuming and tedious. Further more, the increasing amount of data being collected by the rovers or similar missions makes manual operation impractical and automated solution demanded. In addition, automated rock segmentation is also needed as part of the on-board processing. Improvement in the mobility and lifespan of MER allows for more images to be collected than the capability of the outer space communication bandwidth to transmit to the Earth. This fact highlights a crucial demand for effective data compression schemes that can prioritize regions in an image based on their scientific values. Automated rock segmentation will benefit such on-board data compress schemes (Roush et al., 1999).

To meet these needs on automated rock segmentation, this study presents an automated solution consisting of two stages: texture-based image segmentation as initials and active contours based boundary refinement. For the texture-based image segmentation, three texture analysis approaches are used: multi-channel approach, multi-resolution histogram, and inter-scale decision fusion. These three approaches are integrated and embedded into a framework for rock detection using discrete wavelet transforms. This texture-based image segmentation can roughly segment the rocks in the MER images, but can not yield satisfactory rock boundaries. To resolve this problem, the initial boundaries are refined by means of active contours based on the level set method. This boundary refinement allows us to achieve not only finer boundaries but also topologically correct rock segmentation results. Finally, the suggested framework for automated rock detection is applied to Mars surface images collected by MER PANCAM using various filters and NAVCAM, all at the Gusev Crater landing site.

The rest of this paper is organized as follows. Section 2 briefs the previous work for automatic rock extraction, while Section 3 explains the proposed methodology and describes the detail process. Presented in Section 4 are our implementation and its results on MER images. The paper concludes in Section 5 with the evaluation about the properties and performance of the proposed method with perspectives on future efforts. .

## 2. RELATED WORK

There have been a number of efforts towards automatic rock extraction from imagery. For mining studies, Crida and Jager (1994) propose a knowledge-based approach for rock recognition from imagery, which consists of two parts. The first part includes three stages: blob edge detection, boundary completion, and blob extent calculation. The second part involves testing the hypothesis, where the classification of interesting regions detected as blobs in the first part is performed according to twelve rule-based features. Initially, the feature vectors are classified by thresholding and then the remaining vectors initially rejected as non-rocks are reclassified using a supervised k-nearest neighbour classification. Although it is one of the good initial efforts for rock detection, it suffers from heavy computation and the difficulty of threshold determination. Gilmore et al. (2000) show that rock is texturally distinctive features and can be detected successfully in Mars-like desert pavement environment since rocks differ significantly from soils in terms of texture. They use Gabor-filter for texture feature extraction and maximum-likelihood method for classification. They focus on general strategy rather

than specific rock extraction technique. Gor et al. (2000) integrate intensity data and range data by using unsupervised classification. They detect the height-map discontinuities that indicate the top of rocks and then perform a region growing segmentation. However, this algorithm needs image scale as a significant control parameter and the range data produced from stereo imagery.

More recently, Castão et al. (2004) detect rocks using edges extracted from multi-resolution images. Small rocks are detected by finding small closed contours from the edge image generated by Sobel and Canny operators, while large rocks are detected in the same way using a resolution-reduced image. When rocks are detected at both high and low resolutions, the ones detected at the highest resolution are retained. On the other hand, if rocks are detected only at the low resolution, they refit the boundary using snakes (Kass et al., 1988). This rock detection algorithm is efficient when intensity differences between rocks and background (soil) are significant to show clearly linked boundaries. Thompson et al. (2005) propose rock detection from colour image based on machine learning approach. Their rock detection algorithm consists of two steps; segmentation and detection. Image segmentation is performed by split-and-merge method using three bands: hue, saturation and intensity. They then detect rocks using belief network, of which the input vector contains colour, texture, and shape. However, the difficulty remains that a rock may have non-homogeneous intensity and colour, which varies in terms of the illumination and geometry of the rock surface. Dunlop et al. (2007) propose an approach to rock detection and segmentation using super-pixel segmentation followed by a region-merging to search for the most probable groups of super-pixels. A model of rock appearances learned from the training data set identifies all rocks by scoring candidate super-pixel groups with incorporating features from multiple scales such as texture, shading, and two-dimensional shape. Although this rock segmentation algorithm based on supervised multi-scale segmentation provides promising results for rock detection, some problems such as training set determination and boundary localization still remain. A comparison on the performance of rock extraction algorithms is provided by Thompson and Castano (2007).

## 3. METHODOLOGY

The proposed framework for rock segmentation in this study consists of two stages: rock detection using texture-based image segmentation and boundary refinement using the edge-flow driven active contours. The first stage is to provide initial rock detection through the following steps. First, multi-channels containing different texture properties are generated by applying a wavelet transform to the input image. Specifically, four coefficient channels of Haar wavelet transform, including approximation, horizontal, vertical, and diagonal detail coefficients are used as the resultant channels. After the multi-resolution histograms are obtained, their changes across the resolutions are measured by the generalized Fisher information content to extract texture feature, which represents the spatial variation on the image. Finally, the inter-scale decision fusion designed by adopting the hierarchical and interactive k-means algorithm is performed to achieve the initial segmentation. As the second stage, the initial rock boundaries are refined using edge-driven active contours based on the level set method to compensate inaccurate localization of the initial segmentation. The refinement starts with the computation of the edge flow direction and the edge energy to generate the edge flows. These edge flows form a vector field as an external force to enforce the initial boundaries move towards the pixels with high probability being rock boundaries. After that, an edge penalty function is yielded by solving a Poisson equation to satisfy the condition that the Laplacian of the edge penalty function is equivalent to the divergence of the edge flow vector field. Finally, the initial rock boundaries propagate under the constraints of the prepared edge flow vector field and edge penalty function to yield the refined rock segmentation.

### 3.1 Rock detection using texture-based segmentation

**Texture feature extraction.** This study extracts the texture features by employing a multi-channel, multi-resolution approach. This is accomplished through image decomposition and diffusion by Haar wavelet transform. Haar wavelet decomposition works through averaging two adjacent values in a one-dimensional function at a given resolution to form a smoothed signal, namely approximation coefficients. The differences between the values and their averages become the detail coefficients. In discrete data set such as digital image, the construction of Haar wavelet coefficients can be interpreted as two dimensional filtering with four local transform filters: smoothing filter and horizontal, vertical, and diagonal edge detection filters. To achieve the Haar wavelet transformed image of size $m$ by $n$, the image is convolved with each filter and then down-sampled by 2. As an outcome of this procedure, an approximation coefficient and three detail coefficients of size $m/2$ by $n/2$ are produced. This filtering and down-sampling process can be iterated, leading the image from fine to coarse resolution. This decomposition ability of Haar wavelet transform allows the multi-channel approach to transform an image into a set of feature maps by using local transforms to achieve additional and condensed information for texture analysis.

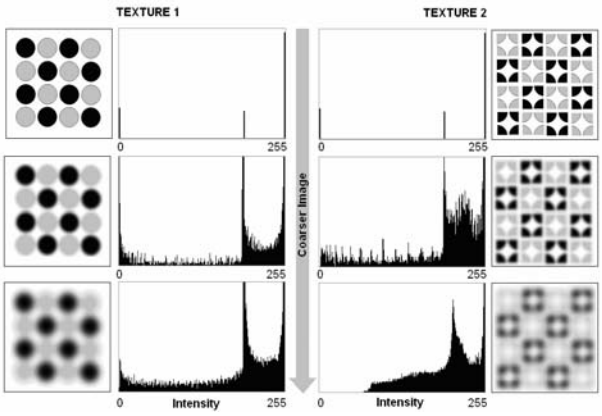



Figure 1. Histograms of multi-resolution images generated by Haar wavelet transform

Let the four channels formed by the wavelet transform coefficients be ( $L_{LL}$, $L_{LH}$, $L_{HL}$, $L_{HH}$ ). From each channel, the texture features are extracted by measuring the change of histograms across different resolutions, namely the multi-resolution histogram method (Hadjidemetrous et al., 2004). Figure 1 shows that although the histograms of two input images with different shades are identical at the high resolution, they differ considerably in coarser resolutions due to the different spatial structures in the two original images Such

histogram change reflects the variation of spatial information, i.e., texture, and can be measured by the generalized Fisher information content $K_J$

$$K_J = \sum_{i=0}^{m-1} -v_i \ln v_i \frac{h_i(L_J) - 4h_i(L_{J+1})}{\sum_{i=0}^{m-1} h_i(L_J)} \quad (1)$$

where $L_J$ and $L_{J+1}$ are two consecutive coefficients at decomposition level $J$ and $J+1$ generated from the original image $L_0$ , $h_i(\cdot)$ denotes the bin count, $v_i$ does intensity (gray value), and $m$ does the number of bins.

As aforementioned, the Haar wavelet coefficients consist of four components $L = \{L_{LL}, L_{LH}, L_{HL}, L_{HH}\}$ such that $K_J$ in equation (1) is extended as equation (2)

$$K_J = K(L_J, L_{J+1}) = \begin{bmatrix} K(L_{LL\,J}, L_{LL\,J+1}) \\ K(L_{LH\,J}, L_{LH\,J+1}) \\ K(L_{HL\,J}, L_{HL\,J+1}) \\ K(L_{HH\,J}, L_{HH\,J+1}) \end{bmatrix} \quad (2)$$

This measurement is concatenated with measurements in the next two levels until the measurement $K_{J+n}$ between $L_{J+n}$ and $L_{J+n+1}$ to form a texture feature vector. As a result, the texture feature vector $V$ is formed as equation (3), where the images with the decomposition levels of $J$ to $J+n+1$ used for computing $K_J$ to $K_{J+n}$.

$$V = [K_J, K_{J+1}, \cdots K_{J+n}]^T \quad (3)$$

**Texture feature classification.** For inter-scale decision fusion, the multi-scale texture features are extracted with windows of various sizes. If the feature vector $V$ is composed of $n$ Fisher information $K$ from level $J$ to $J+n$ computed by the window with size $M \times 2nw$ by $N \times 2nw$ where scale level $nw$ is integer, the feature vector $V^{nw}$ is rewritten as

$$V^{nw} = [K_J^{nw}, K_{J+1}^{nw}, \cdots K_{J+n}^{nw}]^T \quad (4)$$

With the same manner, the m-th feature vector for inter-scale decision fusion is determined by a window with size $M \times 2(nw+m)$ by $N \times 2(nw+m)$ as

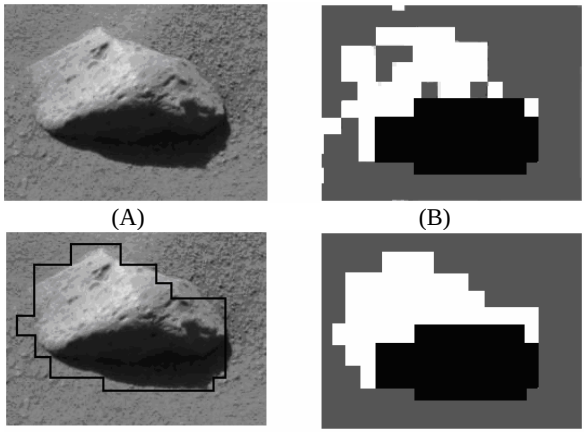
$$V^{nw+m} = [K_J^{nw+m}, K_{J+1}^{nw+m}, \cdots K_{J+n}^{nw+m}]^T. \quad (5)$$

Once the multi-scale texture feature vectors are ready, the k-means clustering for inter-scale decision fusion is performed as below. The k-means clustering starts with the lowest level (the coarsest resolution) feature vector $V^{nw+m}$. As a result, image pixels belong to one of the clusters such that each image pixel has a label. Let the normalized label value be denoted by $LB^{nw+m}$, the input feature vector $C^{nw+m-1}$ for clustering at next level $nw+m-1$ can be written as equation (6)

$$\begin{aligned} C^{nw+m-1} &= [LB^{nw+m}, V^{nw+m-1}]^T \\ &= [LB^{nw+m-1}, K_J^{nw+m-1}, K_{J+1}^{nw+m-1}, \cdots K_{J+n}^{nw+m-1}]^T \end{aligned}. \quad (6)$$

Clustering at level $nw+m$ is repeated using the cluster centres computed from the clustering result at level $nw+m-1$. These steps are iterated until the clustering result at level $nw+m-1$ is stable. After that, the same steps are repeated with $V^{nw+m-1}$ and $V^{nw+m-2}$ vectors. This procedure is concatenated until the clustering result is achieved at the highest level $nw$. As shown in Figure 2, this inter-scale decision fusion yields better clustering result than the classical classification at single scale due to the use of information extracted from multi-scale features.

On the one hand, the texture-based image segmentation yields compact rock detection results, however, they are still not fine enough to directly determine rock boundaries as shown in figure 3 (D). It leads the need for boundary refinement as to be discussed in the next stage.



(A) (B) (D) (C)

Figure 2. Rock detection using texture-based segmentation [Input rock image (A), with K-means clustering (B), with inter-scale decision fusion clustering (C), and detected rock (D)]

### 3.2 Rock boundary delineation using active contours

**Active contours by level set method.** This study exploits an active contour for boundary refinement. Level set method is suggested to describe the evolution of a (contour) curve by Osher and Sethian (1988). In contrast to the traditional snake method, the numerical schemes for the active contours based on level set method benefit automatic handling of the topological change during the curve propagation. In this method, a curve is represented as a level set of a given function, i.e., the intersection between this function and a horizontal plane. To be specific, the zero level set $\Psi(t) = \{(x,y) \mid \phi(x,y,t) = 0\}$ of a time-varying surface function $\phi(x,y,t)$ , gives the position of a contour at time $t$. The evolution equation for a contour curve propagation is defined as equation (7) (Sethian, 1990).

$$\phi_t + F \mid \nabla\phi \mid = 0 \quad \text{given } \phi(x, t=0) \quad (7)$$

For level set method, the evolution equation evolves the contour curve with three simultaneous motions determined by each speed function with

$$F = F_p + F_c + F_a. \quad (8)$$

In the above equation, $F_p$ denotes the expanding speed of the contour defined by a constant speed $F_0$ in its normal direction such as $F_p = F_0$. $F_c$ is the moving speed proportional to the

curvature $k$ such that it is defined as $F_c = -\varepsilon k$, where $\varepsilon$ is a coefficient. Finally, $F_a$ represents the speed moving passively by an underlying velocity field $\vec{U}(x,y,t)\cdot\vec{N}$, in which $\vec{N} = \nabla\phi / |\nabla\phi|$, and thus $F_a = \vec{U}(x,y,t)\cdot\vec{N}$. Plugging this speed function rewrites the evolution equation as equation (9).

$$\phi_t + F_0 |\nabla\phi| + \vec{U}(x,y,t)\cdot\nabla\phi = -\varepsilon k |\nabla\phi| \qquad (9)$$

The first term after the time derivative on the left is concerned with the propagation expansion speed and should be approximated through the entropy satisfying schemes. The second term is related with the advection speed and can be simply approximated by upwind scheme with the appropriate direction. The third term is curvature speed alike a non-linear heat equation, to which an appropriate solution approach is the central difference scheme since the information propagates in both directions. The following paragraphs will present details.

**Edge flow.** In this study the active contour is deformed by the edge flow towards the image pixels that have high probability to be the segment boundaries (Ma and Majunath, 2000). The method was originally designed for boundary detection or image segmentation considering regional image attributes.. Figure 3 illustrates edge flows generated from the image that move towards an expected boundary edge. Each flow vector indicates direction towards the closest edge and an edge can be found at locations where the flow vectors meet from opposite directions. This edge flow method requires edge linking step for a proper image segmentation result, which can be done through active contour since it propagates the closed polygons.

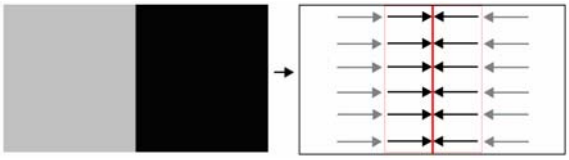


Figure 3. Boundary detection using edge-flow

The general form of an edge flow vector $\Gamma$ at an image location $s$ with an orientation $\theta$ is defined in equation (10) as a function of the edge energy $E(s,\theta)$, the probability $P(s,\theta)$ of finding the image boundary in the direction $\theta$, and the probability $P(s,\theta+\pi)$ in the opposite direction $\theta+\pi$

$$\Gamma(s,\theta) = \Gamma(E(s,\theta), P(s,\theta), P(s,\theta+\pi)) \qquad (10)$$

The first component measures the energy of local image information change and the rest two components determine the contour flow direction. The prediction error $Err(s,\theta)$ at pixel location $s=(x,y)$ is defined as equation (11) using the smoothed image $I_\sigma(x,y)$ obtained by applying the Gaussian kernel $G_\sigma(x,y)$ with a variance $\sigma^2$. The error function essentially estimates the probability of finding the nearest boundary in two possible flow directions

$$Err(s,\theta) = \left| I_\sigma(x + d\cos\theta, y + d\sin\theta) - I_\sigma(x,y) \right| \qquad (11)$$

From these prediction errors, an edge likelihood $P(s,\theta)$ using relative error is obtained

$$P(s,\theta) = \frac{Err(s,\theta)}{Err(s,\theta) + Err(s,\theta+\pi)} \qquad (12)$$

The probable edge direction is then estimated by

$$\theta' = \arg\max_{\theta} \int_{\theta-\pi/2}^{\theta+\pi/2} p(s,\theta')d\theta' \cdot \qquad (13)$$

On the other hand, the edge flow energy $E(s,\theta)$ at scale $\sigma$ is defined as the magnitude of the gradient of the smoothed image $I_\sigma(x,y)$ along the direction $\theta'$.

$$E(s,\theta) = \left| \frac{\partial}{\partial\vec{n}} I_\sigma(x,y) \right| = \left| I(x,y) * \frac{\partial}{\partial\vec{n}} G_\sigma(x,y) \right| = \left| I(x,y) * GD_{\sigma,\theta}(x,y) \right| \qquad (14)$$

where $\vec{n}$ represents the unit vector in the gradient direction, $GD_\sigma(x,y)$ is the first derivative of the Gaussian along the x-axis, and $GD_{\sigma,\theta}(x,y)$ is the first derivative of the Gaussian along orientation $\theta$

$$GD_{\sigma,\theta}(x,y) = GD_\sigma(x',y') \qquad (15)$$

where $$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

Once the flow direction and the edge energy are computed, the "edge flow" field is computed as the vector sum in equation (16)

$$\vec{\Gamma}(s) = \int_{\theta-\pi/2}^{\theta+\pi/2} [E(s,\theta')\cos\theta' \quad E(s,\theta')\sin\theta']^T \, d\theta' \qquad (16)$$

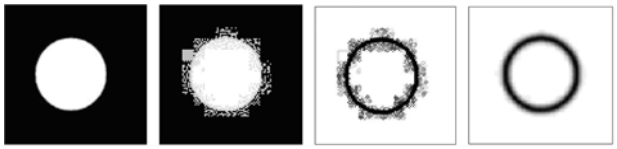
**Boundary refinement using active contours.** The edge flow vector field computed in the aforementioned steps is used as the external force to enforce the contour move towards edges. The contour curve evolution can be formulated as equation (17) where $\vec{\Gamma}$ is the edge flow vector field and $\vec{N} = \nabla\phi / |\nabla\phi|$

$$C_t = (\vec{\Gamma}\cdot\vec{N})\vec{N} + k\breve{g}\vec{N} - F_0\breve{g}\vec{N} \qquad (17)$$

The edge penalty function $\breve{g}$ attracts the contour towards the boundary and has a stabilizing effect when there is a large variation in the image attribute value. It is produced from the edge-flow vector field $\vec{\Gamma}$ by solving the Poisson equation as equation (18), where $\Delta$ is the Laplacian (Sumengen et al.,2002).

$$\nabla\cdot\vec{\Gamma} = -\Delta\hat{g} \qquad (18)$$

Comparing with the traditional gradient edge penalty function, edge penalty function derived from edge flow is more rigid to noise as shown in figure 4.



(A) (B) (C) (D)

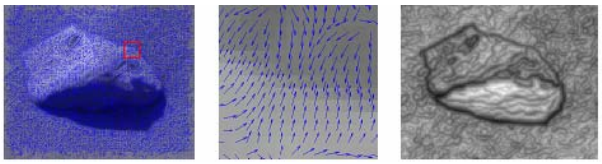
Figure 4. Comparison of edge penalty functions

[Original image (A), noise added image (B), edge penalty function using gradient (C), and using edge flow (D)]

To summarize, we achieve the level set formulation of the edge flow-driven active contour as equation (19)

$$\phi_t - \left( \nabla \cdot \left( \frac{\nabla \phi}{|\nabla \phi|} \right) + F_0 \right) \breve{g}(x,y) |\nabla \phi| - \vec{\Gamma} \cdot \nabla \phi = 0 \quad (19)$$

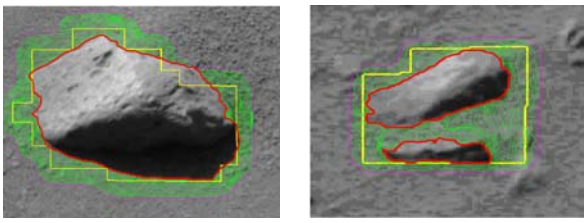
This active contour scheme propagates initial rock boundaries obtained from the first texture-based segmentation stage for refinement with edge flows. Figure 5 demonstrates the edge flow generation and edge penalty function computation, which are obtained through applying the aforementioned procedures to a rock image. The box marks the zoomed in area.



(A) (B) (C)

Figure 5. Edge flow and edge penalty function from rock image [Edge flow vector field (A), edge flows corresponding to the zoomed-in area (B), and computed edge penalty function (C)]

This refinement stage using active contours based on level set method can offer not only finer rock boundaries as shown in figure 6(A), but also correct topological errors caused by coarse resolution of the texture-based image segmentation as shown in figure 6(B).



(A) (B)

Figure 6. Rock boundary refinement by edge-flow driven active contours. [Yellow line denotes initial rock boundaries before refinement, red line is rock boundaries after refinement, and green lines is boundary propagation during refinement].

## 4. IMPLEMENTATION AND RESULTS

The suggested framework for automated rock segmentation is applied into Mars surface images for automatic rock detection to examine its performance. Mars surface images for this implementation are collected by rover Spirit using MER PANCAM with various filters. Additionally, implementation is extended to NAVCAM image (Eisenman, 2004). Each image consists of 1024 by 1024 pixels with 256 gray levels.

After pre-processing such as histogram equalization, the texture features are extracted by the proposed wavelet-based texture feature extraction method. In this experiment, three resolution levels are used for feature extraction and each resolution level contain four Fisher information contents of each channel such that the extracted texture feature vector $V$ is composed of 12 Fisher information contents $\{K_1, K_2, K_3\}$. Also, the texture feature vectors with three scale levels $\{V^1, V^2, V^3\}$ are used for inter-scale decision fusion. They are extracted using windows of three sizes. Equation (20) shows the resultant texture feature vectors in this implementation, where the window size for calculating the Fisher information $K_j^{nw}$ is determined by $16 \times 2nw / 2^j$.

$$V^1 = \begin{bmatrix} K_1^1 \\ K_2^1 \\ K_3^1 \end{bmatrix} \quad V^2 = \begin{bmatrix} K_1^2 \\ K_2^2 \\ K_3^2 \end{bmatrix} \quad V^3 = \begin{bmatrix} K_1^3 \\ K_2^3 \\ K_3^3 \end{bmatrix} \quad (20)$$

After multi-scale texture feature vectors are generated, the inter-scale decision fusion is performed through clustering explained in previous section. As a result, the initial rock boundaries are achieved. Figure 7 shows rock detection result superimposed atop the original image. From the initial rock boundaries shown in Figure 7a, the final rock boundaries are extracted by contour evolution based on level set method with further refinement. In the contour evolution step, the edge flow vector field is first generated and the edge penalty function is computed considering the scale determined by the variance $\sigma^2$ of the Gaussian kernel $G_\sigma(x,y)$. We focus on rocks larger than 1/2500 of the entire image, i.e., about $20 \times 20$ pixels or more. Finally, after the boundary refinement, the rock segmentation results are yielded as shown in Figure 7b. Additionally, figure 8 shows segmentation results from the other MER PANCAM images, which demonstrate satisfactory performance. Figure 9 represent some failed cases. On the upper left corner in Figure 9, the structured soil region is misclassified into rocks. Also it shows difficulty to segment rocks partly covered by soil which have ambiguous boundaries. In figure 10 and 11, the implementation is extended to NAVCAM which has wider field of view (FOV: 45 degree) than PANCAM (FOV: 16 degree). Despite of more spatial resolution variation due to the wider FOV, the proposed method still shows satisfactory rock segmentation results, although figure 11 suffers from similar problems of figure 9.

## 5. CONCLUSION AND FUTURE WORK

Automated rock detection is necessary for the Mars Exploration Rover mission. This paper presents a framework to segment rocks from the MER images. In the two stage solution, rocks are firstly detected by texture-based image segmentation. For that purpose, three methods, wavelet based multi-resolution histograms, multi-channel approach, and inter-scale decision fusion are integrated. It yields reliable rock detection results but shows poor localization quality. To compensate this shortcoming, the rock boundary is refined by active contour algorithm based on the level set method in the second stage. The edge flow vector field is used as the external force to enforce the contour moving towards the edges and the stopping function is derived from the edge flow instead of the traditional gradient edge penalty function to warrant more robust results. This framework is applied to MER PANCAM and NAVCAM images to investigate its performance.

Experiments demonstrate satisfactory rock segmentation results through this fully automated process and give several worthy notes. First, the suggested framework can account for variations of rock size with no parameter tuning through the multi-scale

approach. Also, the results are robust to fault edges and edge leaking. This framework can be expected to be embedded into various rover image analysis applications such as path determination, rock analysis, and training data supplement for satellite remote sensing on Mars or other similar applications.

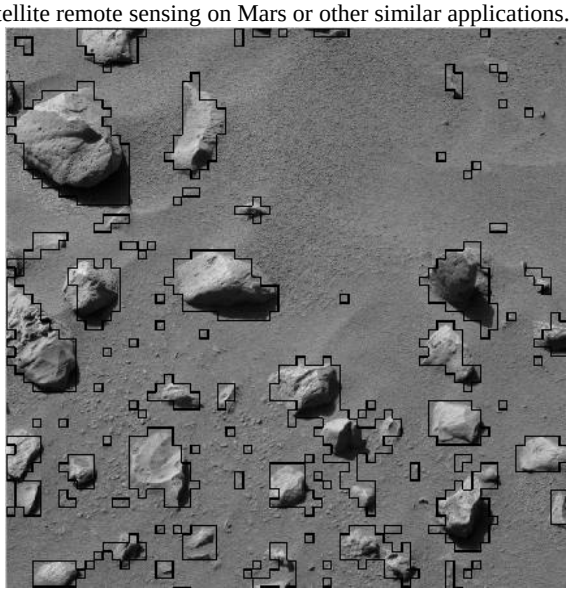


Figure 7a. Initial rock detection by texture-based segmentation

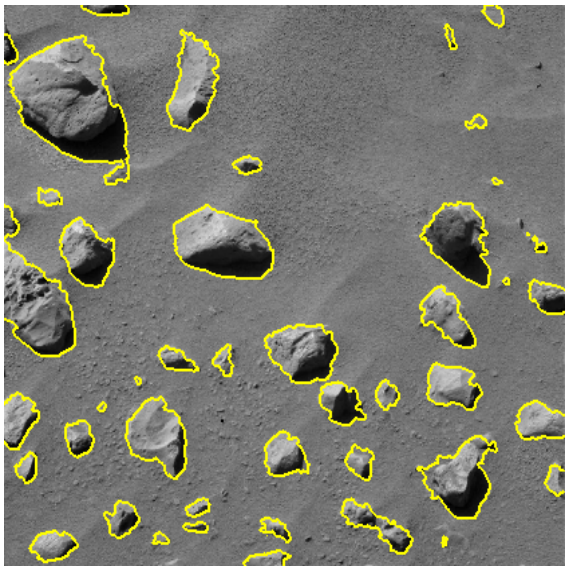


Figure 7b. Detected rocks [PANCAM, R1 filter (430nm)]

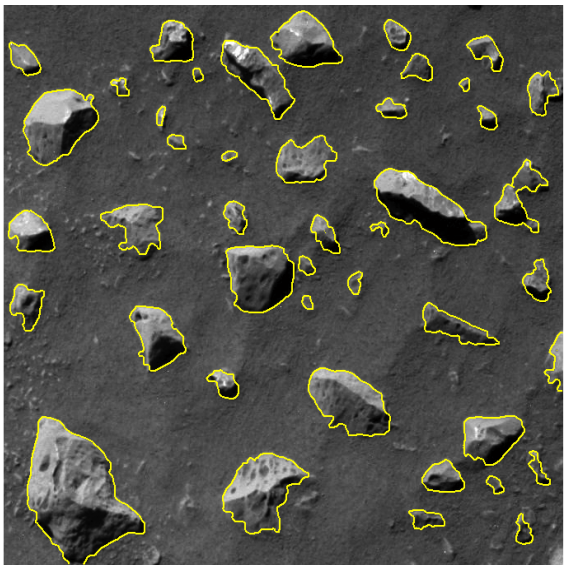


Figure 8. Detected rocks [PANCAM, R7 filter (980nm)]

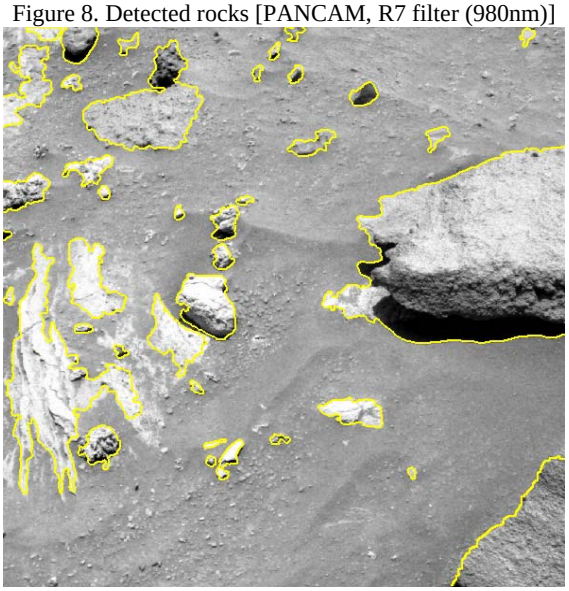


Figure 9. Detected rocks [PANCAM, L3 filter (670nm)]

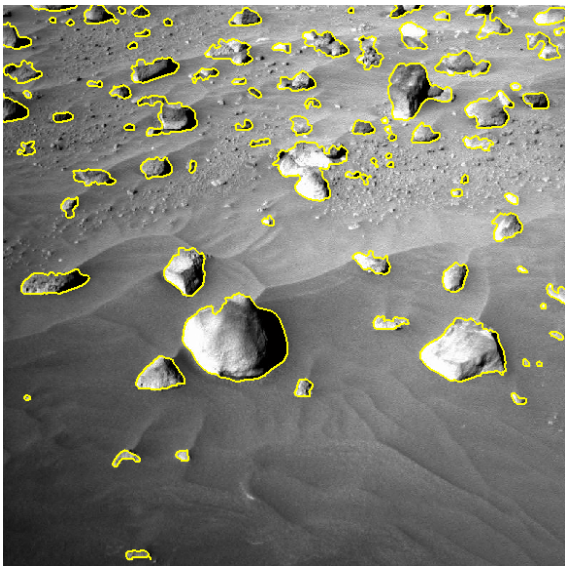


Figure 10. Detected rocks [NAVCAM image 1]

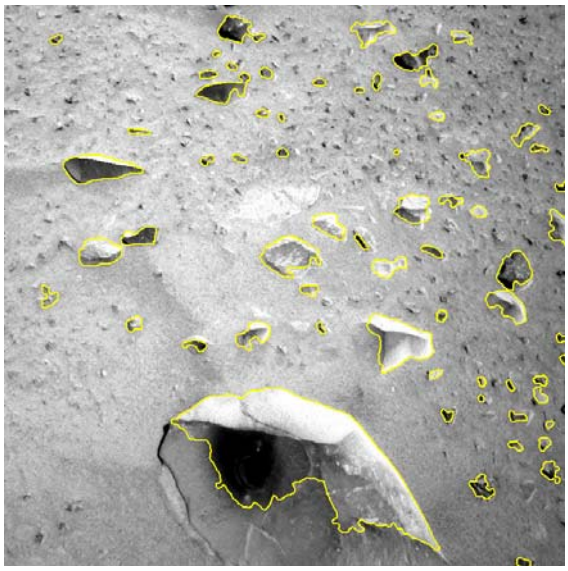


Figure 11. Detected rocks [NAVCAM image 2]

**REFERENCES**


Castãno, A., Anderson, R. C., Castãno R., Estlin, T., and Judd, M., 2004. Intensity-based rock detection for acquiring onboard rover science, Proceedings of *the 34th Lunar and Planetary Science Conference*, League City, USA

Crida, R.C. and Jager, G., 1994. Rock recognition using feature classification, Proceedings of the *IEEE South African Symposium on Communications and Signal Processing*, 4, pp. 152-157, Cape Town, South Africa

Dunlop, H., Thompson, D. R., Wettergreen, D.,2007. Multi-scale Features for Detection and Segmentation of Rocks in Mars Images, Proceedings of *IEEE Conference on Computer Vision and Pattern Recognition*, Minneapolis, USA pp.1-7.

Eisenman, A., Liebe, C. C., Maimone, M. W., Schwochert, M. A., and Willson, R. G., 2004. Mars Exploration Rover engineering cameras, NASA technical report 20060031265, Jet Propulsion Laboratory, California USA

Gilmore, M. S., Castãno, R., Mann, T., Anderson R. C., Mjolsness, E. D., Manduchi, R., and Saunders, R. S., 2000. Strategies for Autonomous Rovers at Mars, *Journal of Geophysical Research-Planets*, 105(E12), pp. 1-37.

Gor, V., Castãno R., Manduchi, R., Anderson, R. C., and Mjolsness, E., 2000. Autonomous rock detection for Mars terrain, Proceedings of *AIAA Space 2001*, Albuquerque, USA

Kass, M., Witkin, A., and Terzopoulos, D., 1988. Snakes: Active Contour Models, *International journal of Computer Vision*, 1, pp. 321-331.

Osher, S. A. and J. A. Sethian, 1988. Fronts Propagating with Curvature Dependent Speed: Algorithms based on Hamilton-Jacobi Formulation, *Journal of Computational Physics*, 79, pp. 12-49.

Roush, T. L., Gulick, V., Morris, R., Gazis, P., Benedix, G., Glymour, C., Ramsey, J., Pedersen, L., Ruzon, M., Buntine, W., and Oliver, J., 1999. Autonomous Science Decisions for Mars Sample Return, *Lunar and Planetary Science Conference XXX*, League City, USA

Sethian, J. A., 1990. A Review of Recent Numerical Algorithms for Hyper-surfaces Moving with Curvature Dependent Speed, *Journal of Differential Geometry*, 33, pp.131-161.

Sumengen, B., Manjunath, B. S., and Kenney, C., 2002. Image Segmentation using Curve Evolution and Flow Fields Proceeding of *IEEE International Conference on Image Processing* (ICIP), Rochester, USA,

Thompson D. R., Niekum, S., Smith, T., and Wettergreen, D., 2005. Automatic Detection and Classification of Geological Features of Interest, Proceedings of *IEEE Aerospace Conference*, Big sky, USA

Thompson, D. R. and Castano, R., 2007. A performance comparison of rock detection algorithms for autonomous planetary geology, Proceedings of the *IEEE Aerospace Conference*, Big sky, USA

W. Y. Ma and Manjunath, B. S., 1997. Edge flow: a framework of boundary detection and image segmentation. In Proceeding of. *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 744-749.