Automated Purgatoid Identification
Final Report

Mentor: Paolo Bellutta
Intern: Steven Wood

Introduction

Driving on Mars is hazardous: technical problems and unforeseen natural hazards can end a mission quickly at the worst, or result in long delays at best. This project is focused on helping to mitigate hazards posed to rovers by purgatoids: small (< 1 m high, < 10 m wide), ripple-like eolian bedforms commonly found scattered across the Meridiani Planum region of Mars. Due to the poorly consolidated nature of purgatoids and multiple past episodes of rovers getting stuck in them, identification and avoidance of these eolian bedforms is an important feature of rover path planning (NASA, 2011).

Background / Associated Projects

This project – to automate the identification of a specific ripple type -- is closely associated with the Ripple Detection Algorithm created by Yasuhiro Katayama (Japanese Aerospace Exploration Agency) in 2010, while a visiting scientist at JPL. That algorithm forms the core of the Mars Terrain Classifier, which utilizes high-resolution HiRISE images taken from the Mars Reconnaissance Orbiter as input, and creates terrain classification maps – to be used to assist rover path planning – as output. The Ripple Detection Algorithm currently classifies regions in HiRISE imagery into one of four categories: smooth, ripple, bedrock, or discontinuity (steep slopes, etc). However, it is unrealistic to have rovers avoid all ripples, especially when many ripples are not driving hazards, as a result purgatoid identification is a necessary addition to the Mars Terrain Classifier to make it fully useful.

Objective

This project, in support of the Mars Terrain Classifier, created and implemented an algorithm that uses Mars Reconnaissance Orbiter HiRISE images as data and automatically identifies the location of potential purgatoids in an input image. The output of this process is a color shaded HiRISE image functioning as a purgatoid density map. Additionally, the centroids of known purgatoids marked in the output.

Approach

The Automated Purgatoid Identification Algorithm (implemented as function ‘purg’ in Matlab version 2011a) is as follows:

<u>Step #0: Algorithm Parameters.</u>

The following inputs are expected from the user:

- Path to input image: image must be grayscale, specific format type is not required.
- Path to purgatoid template directory (‘shapes/’): This directory holds all the purgatoid training templates.
- Path to texton directory (‘textures/’): This directory holds all the textons

extracted from the training templates.

- Image file extension: any file extension supported by Matlab is allowed, in the format *.png or *.tiff or *.jpg, ext.
- Minimum size of possible purgatoids.
- Maximum size of possible purgatoids.
- Matching threshold for texton correlation coefficient. Any area being inspected that matches a texton above this coefficient is considered a hit and the appropriate bin incremented in the texture histogram.
- Matching threshold for moment invariants. Any area in the input image with a Euclidean distance below this threshold is considered a potential purgatoid.
- Matching threshold for gray-level co-occurrence matrix contrast.
- Matching threshold for gray-level co-occurrence matrix correlation.
- Matching threshold for gray-level co-occurrence matrix entropy.
- Matching threshold for texture histogram.
- Matching threshold for hits per area.

<u>Step #1: Preprocessing.</u>

The algorithm identifies potential purgatoids solely based on the spatial distribution of pixel intensities in the input image. Several steps are taken to enhance any waveforms found.

- Input image is smoothed (to reduce noise) using a 3x3 Gaussian filter.
- Smoothed image is run through a high-pass filter to keep details at the scale of purgatoids, while removing unneeded and potentially misleading large scale (low frequency) information.
- Image undergoes histogram equalization to enhance contrast.
- Optimum global threshold determined by Otsu's Method.
- Input image is then thresholded to produce a binary image. This binary image forms the basis for determining the connected components of the input image.
- Binary image is eroded by disk of radius size 1.
- Isolated single pixels in binary image are removed ('cleaned').
- Isolated single background pixels surrounded by foreground pixels are changed to foreground ('filled').
- Isolated 'spur' pixels (connected to a region only by an 8-connected path) are removed.
- Binary image is cleaned again.
- Connected components of binary image determined. Any connected component with an area less than 15 pixels is removed. This is to ensure that when the image is dilated, only the core shape of the potential purgatoids remain. (Purgatoids have fundamentally similar shapes, but are often warped when merging or overlapping with other bedforms. The ultimate goal of all preprocessing is to extract core ripple shapes from the raw input image).
- Binary image is dilated by disk of radius size 2.

- New connected components of binary image determined. The spatial distribution of each connected component in the binary image is then matched to the same pixel locations in the raw input image. The overlapping pixel values are then extracted from the raw input image, contrast enhanced, and placed in the final product of the preprocessing step. This preprocessing product is a hybrid of a binary and grayscale image, containing all 0's in the background, while each connected component consists of contrast enhanced grayscale values.
-

Step #2: Determine and Load Purgatoid Templates

Approximately 100 masks containing various properties of purgatoids are selected from the hybrid image created in step #2. This is done manually, and is an iterative process, i.e. through trial and error it can be determined that some masks are much better at representing the properties of purgatoids than others. The best masks will be retained. This selection process will not be required when the algorithm is complete, however, the following actions will occur:

- Each mask is loaded into a storage cell.
- The moment invariants for each template are determined.
- The gray-level co-occurrence contrast, correlation, and entropy for each template are determined.
- The texture histogram and hits per area for each template are determined.

Step #3: Pattern Matching

This step utilizes the following information to execute pattern matching to identify potential purgatoids.

- User supplied thresholds for matching.
- Hybrid image containing connected components that are purgatoid candidates.
- Array containing moment invariants for purgatoid templates.
- Cells containing gray-level co-occurrence matrix for each template.
- Cells containing raw purgatoid templates.

The matching sequence is as follows for each connected component in hybrid image:

Determine moment invariants (for equivalent binary area)
Start with first purgatoid template from training set
Calculate L2 norm between moment invariants
If within threshold
Determine gray-level co-occurrence matrix
Calculate L2 norm between contrasts
If within threshold
Calculate L2 norm between correlation
If within threshold
Calculate L2 norm between entropy
If within threshold
Create texture histogram for and hits per area for connected component

Calculate L2 norm between texture histograms
If within threshold
Compare hits per area
If within threshold
Label purgatoid

- If any threshold not passed merely continue loop until all connected components inspected.

Eighteen 2500 x 2506 pixel tiles from HiRISE images from the Meridiani Planum (16) and Eberswalde (2) regions of Mars were run through the purgatoid identification process. The training set consisted of 101 purgatoids manually selected from 2 tiles in Meridiani region, with the remaining 16 non-training tiles as a test to validate the method.

The optimal function parameters were found to be:

Upper\Lower limit cc:	1200\50 pixels
Correlation coefficient:	0.90
Moment Invariants:	1.75
Contrast:	25
Correlation:	0.35
Entropy:	10* *entropy found not to be useful
Texture histogram:	0.15
Hits per unit area:	0.25

Addtionally, the gray-level co-occurrence matrix neighbor offset was varied between 1 – 3. The purgatoid selection seemed to work best when comparing a reference pixel versus a neighbor with an offset of 3.
The computational savings of using the moment invariants and gray-level co-occurrence matrix prior to the texture histograms was not as great as expected or hoped. About 50 – 70 % of potential purgatoids were eliminated prior to requiring cross-correlation in the texture histogram step.

Results

At present, random checking indicates 50 – 70% of purgatoids are identified correctly in purgatoid dense regions. It is important to note accurate count of false negatives in purgatoid dense regions very difficult (see Figure 1). Purgatoids are on a smooth continuum of size and length – for smaller purgatoids it is a judgment call whether to include in count, or if the ripple is even a purgatoid.
However, rover planners have indicated that mapping the overall purgatoid density of area is still useful for hazard mitigation (Bellutta, 2011). To this end, the output images were divided into 100 x 100 pixel grid squares, and the number of purgatoids per grid square determined. Colors were assigned to indicate purgatoid amount, and density maps created for 18 tiles (for example see Figure 2).

Figure 1: Purgatoid Identification

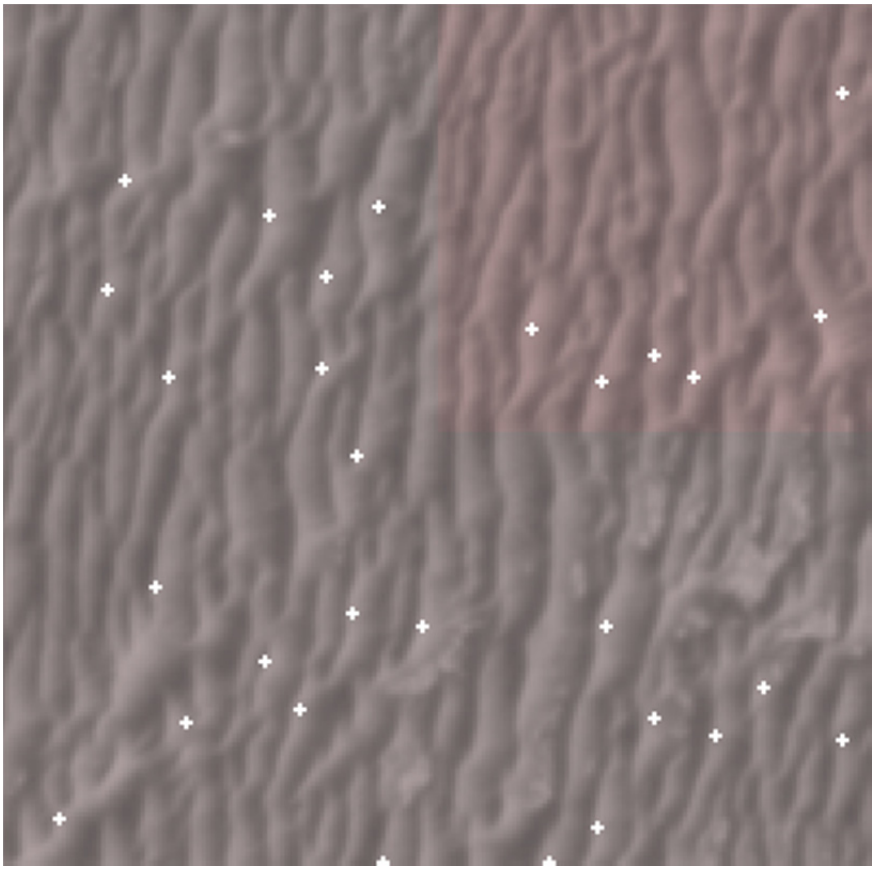
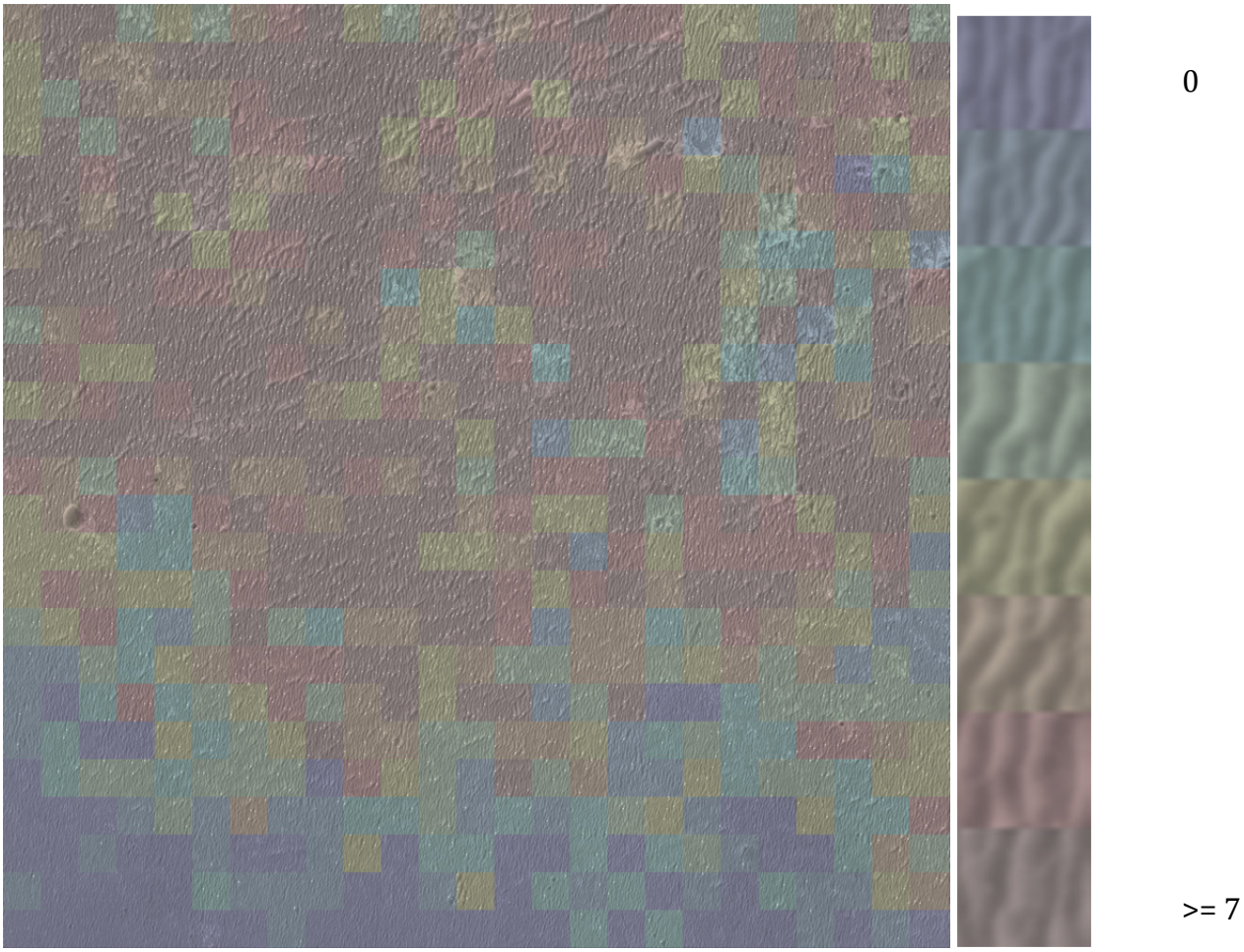


Figure 2: Purgatoid Density Map. Density levels range from 0 purgatoids per 100 x 100 pixel square, to >= 7.



Discussion

The purgatoid density maps show that automated purgatoid identification is possible, and useful, without requiring the centroid location of each purgatoid in an area. However, at present the process is still in an experimental phase and will need much more work done before being considered a vetted, reliable process. Better methods of purgatoid template selection, a smaller false negative rate (on average), and consistent purgatoid identification in disparate regions (separated by 10 – 100+ km) is required. Additionally, the algorithm may be implemented outside of MATLAB to allow for wider use.

Acknowledgments
This research was carried out at the Jet Propulsion Laboratory, California Institute of Technology, and was sponsored by the Jet Propulsion Laboratory Summer Internship Program and the National Aeronautics and Space Administration.

References

Bellutta, P. Personal Communication.

Katayama, Yasuhiro, “A Study of Ripple Detection Algorithm from HiRISE Image”, personally attended talk on August 17, 2010. Later received slides from Paolo Bellutta.

NASA, “Mars Exploration Rover Mission: Opportunity Update Archive”. Retrieved June 4, 2011 from Jet Propulsion Laboratory: http://marsrover.nasa.gov/mission/status_opportunity_2005.html