

# Machine Detection of Martian Impact Craters from Digital Topography Data

Brian D. Bue and Tomasz F. Stepinski

B.D. Bue is with the Department of Computer Science, Purdue University, West Lafayette, IN 47907 USA (e-mail: [bbue@cs.purdue.edu](mailto:bbue@cs.purdue.edu))

T.F. Stepinski is with the Lunar and Planetary Institute, Houston, TX 77058-1113 USA (e-mail: [tom@lpi.usra.edu](mailto:tom@lpi.usra.edu))

### Abstract

Research on automatic identification of impact craters on Mars and other planetary bodies has concentrated on detecting them from imagery data. We present a novel approach to crater detection that utilizes digital topography data instead of images. Craters are delineated by topographic curvature. Thresholding maps of curvature transforms topographic data into a binary image from which craters are identified using a combination of segmentation and detection algorithms. We apply our method to a large and technically demanding test site and compare the results to the existing catalog of manually identified craters. Our algorithm finds many small craters not listed in the manual catalog, but it fails to detect heavily degraded craters. A detailed quality assessment of the algorithm is presented. The topography-based crater detection algorithm offers a relatively simple and ready-to-use tool for identification and characterization of fresh impact craters with an adequate performance for the immediate application to Martian geomorphology.

### Index Terms

Automated extraction of features, digital topography, object recognition, segmentation, craters, Mars.

## I. INTRODUCTION

Impact craters are among the most studied features on Mars. They are structures formed by collisions of meteoroids with the Martian surface. Their importance stems from the wealth of information that detailed analysis of their distributions and morphology can bring forth. For example, statistics of crater sizes forms the basis for geological stratigraphy of Mars [8], [25], [29]. In addition, knowledge of crater morphologies enables studies of a number of outstanding issues in Martian geomorphology, such as: the nature of degradational processes [23], regional variations in geologic material [5], and distribution of subsurface volatiles [6]. Thus, surveying

Martian craters is an important task in planetary research.

Various researchers have manually collected datasets of crater locations and characteristics [2], [7], [13], [20], [21]. These datasets, some global while other regional in extent, are all based on visual inspection of imagery data. The most complete dataset, containing information on 42,283 craters distributed globally, is *The Catalog of Large Martian Impact Craters* [2] (hereafter referred to as the Barlow catalog). Building a comprehensive dataset of craters, like the Barlow catalog, is a laborious process, and the results are incomplete because of limitations of the visual detection method. These shortcomings increase the demand for automated crater detection techniques, as they would significantly speed up the rate of building the datasets and provide means for an objective and repeatable process for performing the analysis. In a parallel development, a need to develop an autonomous spacecraft navigation system also sparked an interest in automated crater detection algorithms [4], [14]

#### A. Previous research

An existing body of work focuses almost exclusively on researching methods for automatic detection of craters from images. This is a difficult task because of inherent limitations of imagery data and variety of crater structures. The “visibility” of an impact crater depends on the quality of an image, which, in turn, depends on illumination, surface properties, and atmospheric state. Craters on Mars, as seen on images, are often complex structures despite their overall circular appearance. Some craters are degraded by erosion and are barely distinguishable from the background data. In heavily cratered terrain, where automated detection of craters is most desirable, there is a significant degree of crater overlap. Finally, craters differ by orders of magnitude in size.

The researched image-based crater detection approaches could be divided into two general

categories, unsupervised, that are fully autonomous, and supervised, that require an input of domain expert. The unsupervised methods rely on pattern recognition techniques to identify crater rims in an image as circular or elliptical features [1], [4], [10], [12], [14]. The original image is pre-processed to enhance the edges of the rims and the actual detection is achieved by means of the Hough Transform [11] (hereafter referred to as HT), genetic algorithms [10], or the Radial Consistency algorithm [9] that identifies regions of rotational symmetry. In [4], [10], [14] the pre-processing consists of applying an edge detecting algorithm, in [1] it consists of calculating texture measures, and in [12] it consists of a combination of edge detection, texture measurement and edge direction analysis.

The supervised methods [3], [19], [26], [28] use machine learning concepts to train an algorithm into detecting craters. In a learning phase the training set of images containing craters labeled by domain experts is fed to an algorithm. In the detection phase, the previously trained algorithm detects craters in a new, unlabeled set of images. In [3], [26] a continuously-scalable template model technique is used to achieve detection. In [28] a number of algorithms are tested, and the support vector machines algorithm is shown to achieve the best rate of crater detection. In [19] genetic programming is used to generate a population of random detection algorithms whose performance is iteratively improved using a training set as selection criteria.

All of the image-based crater detection approaches involve complicated, multi-step algorithms to combat inherent limitations of imagery data. For example, in [10] the set of images from which craters are to be detected is first clustered with respect to image quality; a detection algorithm is then optimized for each cluster separately. In [12] detected craters are verified by template matching and neural networks are employed to remove false detections. In [15] unsupervised and supervised algorithms are combined to improve the detection rate. In spite of such sophistications,

image-based crater detection algorithms had only limited success. Supervised algorithms work well for small craters [19] and/or for relatively simple terrain, but their efficiency drops in proportion to the complexity of the terrain [26]. Unsupervised methods work well in the limited context of an autonomous spacecraft navigation system [4] because of relative simplicity of asteroid surfaces. However, in the context of cataloging craters on Mars, no image-based detection algorithm has been deemed adequate to be employed in a scientific study, although the recent advances [12] may change that.

### *B. Crater detection from topography data*

The purpose of this paper is to describe a topography-based crater detection algorithm as an alternative to image-based algorithms. Imagery data is well-suited for human visual interpretation but ill-suited for automated processing because images are skewed representations of the landscapes they portray. Fortunately, for planet Mars, in addition to imagery data, the digital elevation model (DEM) data is also available [22]. DEMs are much more fundamental descriptors of planetary surfaces than images. They are suitable for a quantitative geomorphic analysis and are well-suited for automated identification of craters.

Interestingly, no complete, DEM-based crater detection algorithm has been studied in the past. A constrained DEM-based crater detection algorithm was used [16] to co-register Mars Digital Image Mosaic (MDIM 1.0) with the more precise grid provided by the Martian DEMs. In this study, diameters and approximate locations of craters are already known from a manually collected catalog [21], and the role of the algorithm is restricted to adjusting these parameters using the DEM data. In [12] the DEM was used to supplement imagery data in detecting large craters. This lack of interest in DEM-based algorithms is probably due to scarcity and limited resolution of planetary topography data. Presently, Mars is the only extraterrestrial body for which

global topography data of sufficient resolution is available. However, such data will become available for Mercury once the MESSENGER spacecraft, currently en route, arrives there in 2011. Also, any future exploration of the Moon will require high resolution topography data [30].

Presently available Martian DEMs are constructed from the Mars Orbiter Laser Altimeter (MOLA) Mission Experiment Gridded Data Record (MEGDR) [22] that has resolution of  $1/128$  degree, or  $\sim 500$  m at the equator. The coarse resolution of the present DEM limits the size of craters that can be detected to  $\sim 3 - 5$  km in diameter. This is on par with the lowest size of craters in the Barlow catalog and small enough to extract a catalog of scientific significance. In the near future, significantly higher resolution DEMs of selected portions of Martian surface will be compiled from the Mars Express High Resolution Stereo Camera (HRSC) data, allowing for detection of smaller craters.

This paper begins with a detailed description of our DEM-based, unsupervised crater detection algorithm. Each step in the algorithm is illustrated by applying it to the large and challenging test site. The algorithm features three innovations that may have significance beyond the issue of crater detection. First, we establish that topographic curvature is the preferred terrain parameter to extract topographic rims. Second, we devise a method of segmenting the entire site into a large number of much smaller fragments without cutting through the craters. Third, we utilize a confirmation function that supplements the HT in identifying circular features in a noisy binary image. We apply our DEM-based detection algorithm to our test site and compare the results to the Barlow catalog. A detailed quality assessment of our algorithm is presented, followed by discussion and a roadmap for future improvements.

## II. OVERVIEW OF CRATER DETECTION ALGORITHM

Detecting craters from topographic data is, in principle, much easier than detecting them from imagery data because craters are landforms that can be defined in terms of terrain morphometric attributes calculated from the DEM. This eliminates the issue of “visibility” that impedes the performance of image-based algorithms. A DEM is a raster dataset,  $\mathcal{E}(x, y)$ , where each pixel (labeled by  $x$  and  $y$ ) is assigned an elevation value  $\mathcal{E}(x, y) = z(x, y)$ . Terrain morphometric attributes are another raster dataset where each pixel,  $(x_0, y_0)$  is assigned a value calculated from the  $z$  values. Often, only values of  $z$  in the immediate neighborhood of  $(x_0, y_0)$  are used in evaluating the value of an attribute at  $(x_0, y_0)$ . Examples of such attributes are slope, aspect, and topographic curvature. However, we also use a “flood” measurement, which is a terrain attribute that is not entirely determined by local data.

A slope measurement was used in [12], [16] as the terrain attribute defining the rims of the craters because crater rims are often associated with the steepest gradients in the Martian landscape. However, our analysis has shown that slope is an inferior rim indicator, especially for smaller or more degraded craters. In addition to slope, we have investigated terrain texture and profile curvature as rim indicators. Terrain texture is surrogated by standard deviation of  $z$  calculated over a circular moving window of fixed radius; radii of 5 to 20 pixels were used. Profile curvature,  $\kappa(x, y)$ , is the curvature in the gradient direction, it reflects the change of slope angle and thus is well-suited to indicate crater rims on the basis of their concavity. Indeed, we have found that  $\kappa$  is the best terrain attribute to outline a broad range of crater types, and we have selected it to define the rims of craters in our algorithm.

The scientific value of crater detection algorithm is enhanced if it can be applied to large sites on Mars. The computational cost of detecting craters from maps of  $\kappa$  can be prohibitively

expensive. In the case of the HT it is  $\mathcal{O}(n \times m \times \text{Min}[n, m])$ , where  $n$  and  $m$  are pixel dimensions of a site. This is because the site may contain craters as small as a single pixel, and as large as the smaller dimension of the site. To deal with this problem we segment the site into a large number of fragments without cutting through the craters using the flood attribute. The impact craters are topographic basins. We modify the elevation field  $z(x, y)$  so that all enclosed depressions are raised to the level of the lowest pour point around their edges, resulting in a modified (“flooded”) elevation field  $z_f(x, y)$ . A “flood” field  $\delta(x, y) = z_f(x, y) - z(x, y)$  has non-zero values only for pixels located inside topographic basins and is used for site segmentation. The largest segment is about the size of the largest crater in the site, and the majority of segments are approximately the size of the smallest craters. Applying the crater detection algorithm to each segment separately instead of the entire site detects all craters in the site at much reduced computational cost.

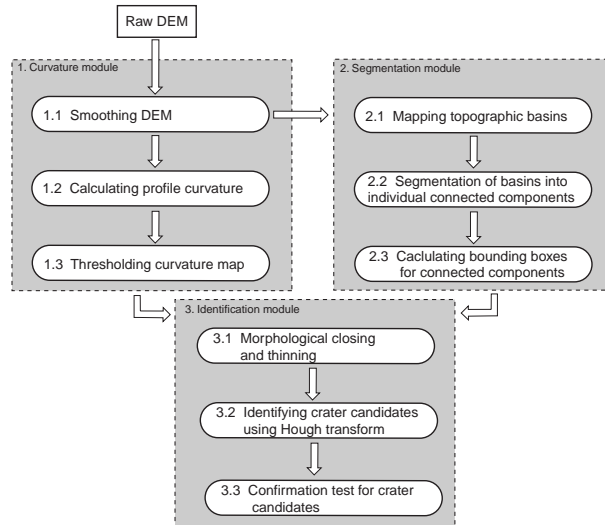


Fig. 1. Schematic diagram of overall organization and processing steps for our DEM-base crater detection algorithm.

We use the HT to identify craters in each segment. However, the HT by itself is not effective because of large number of false detections. We use the HT to produce a list of crater candidates



in each segment. That list is then fed to a confirmation algorithm designed to eliminate false detections. Fig.1 shows a schematic diagram of overall organization of our algorithm that consists of three major modules devoted to calculating profile curvature, site segmentation, and crater identification, respectively. Subsequent sections describe these modules in detail using our test site to illustrate how they work.

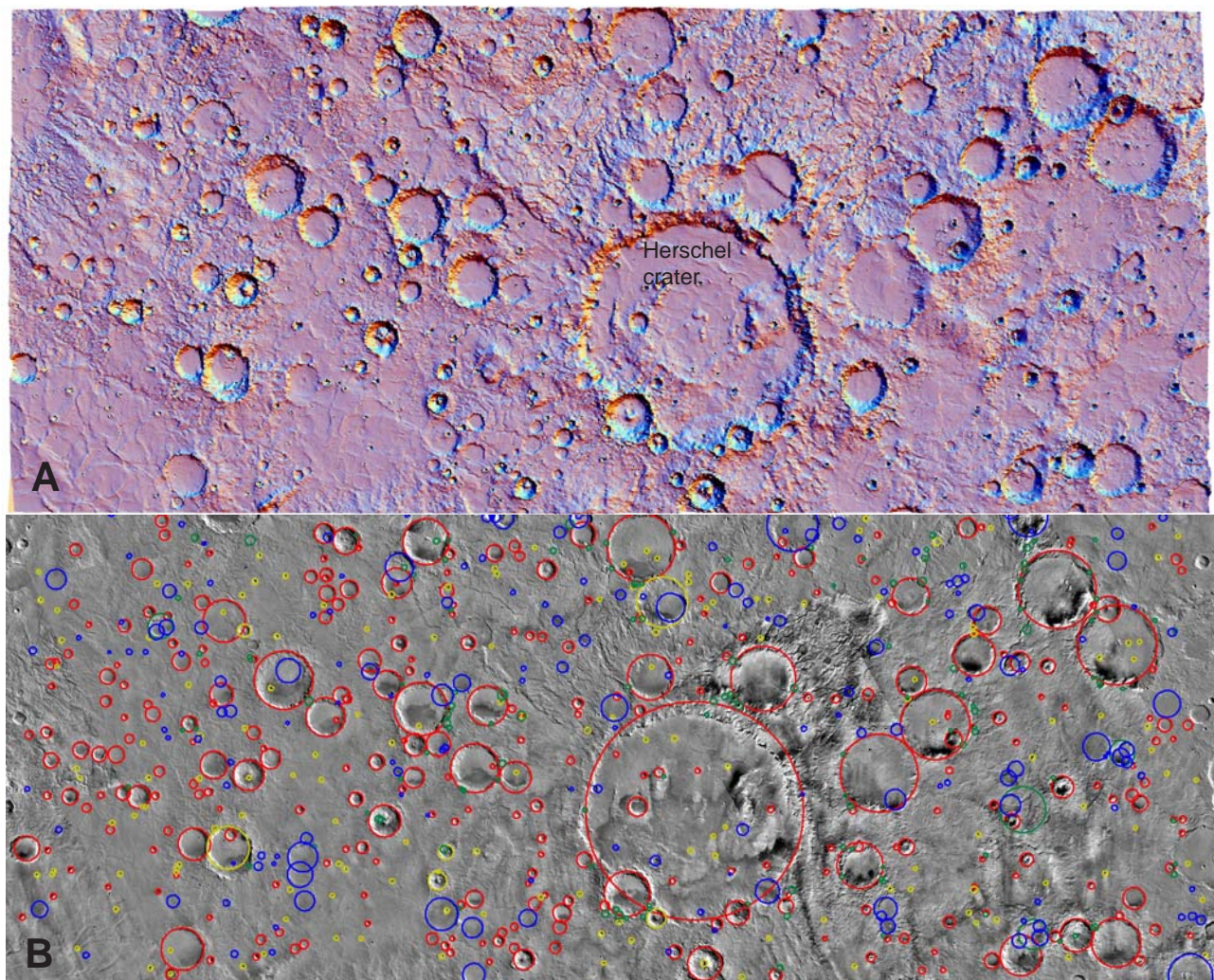


Fig. 2. (A) Visual rendering of topography for the test study site located around the Herschel crater. (B) Detected craters shown on top of image mosaic image of the test site. Red - detected craters also present in the Barlow catalog, yellow - detected craters absent from the Barlow catalog, blue - missed craters, green - false detections.

The heavily cratered test site (see Fig.2A) is located around the Herschel crater. It is bounded by the following coordinates (lon. E, lat. N) clockwise starting from the lower-left corner (114.00, -18.42), (114.60, -7.58), (141.40, -7.58), and (142.00, -18.42). The  $3143 \times 1282$  pixels DEM of this site is constructed by projecting [27] the MEGDR data into a rectilinear, 500-m resolution coordinate grid. It is a challenging site for any crater detection algorithm because of the range of crater sizes, degradation degrees, and overlapping geometries.

### III. IDENTIFYING CRATERS' RIMS USING PROFILE CURVATURE

Two dimensional surfaces, like the surface of Mars as represented by the DEM, have varying curvatures in different directions. In our application the profile curvature,  $\kappa$ , is of interest because it measures the change of slope angle. In a crater, the areas located on its walls tend to be convex in the gradient direction, and are identified by  $\kappa > 0$ , whereas the areas located on its rim tend to be concave in the gradient direction and are identified by  $\kappa < 0$ . This suggests using the values of  $\kappa$  to delineate craters from the DEM. However, the computation of curvature requires a reliable estimate of second order derivatives, which are sensitive to the smoothness of the surface. The DEM representation of the Martian surface lacks smoothness because of accuracy, resolution, interpolation, and the true roughness of Martian landscape. Calculating  $\kappa$  directly from the DEM would produce results too noisy for reliable delineation of craters. For this reason we smooth the elevation field  $z(x, y)$  using a mean value of  $z$  calculated over a circular moving window of constant radius. The radius equal to 5 pixels was determined to produce the best results. We calculate  $\kappa(x, y)$  from the smoothed elevation field  $z_s(x, y)$  using a formula given in [17].

The results of calculating  $\kappa$  for the test site are shown in Fig.3A. The grayscale gradient indicates values of  $\kappa$  between the maximum positive (+0.0309); which corresponds to most convex areas and is depicted by white, to the maximum negative (-0.0311), that corresponds to



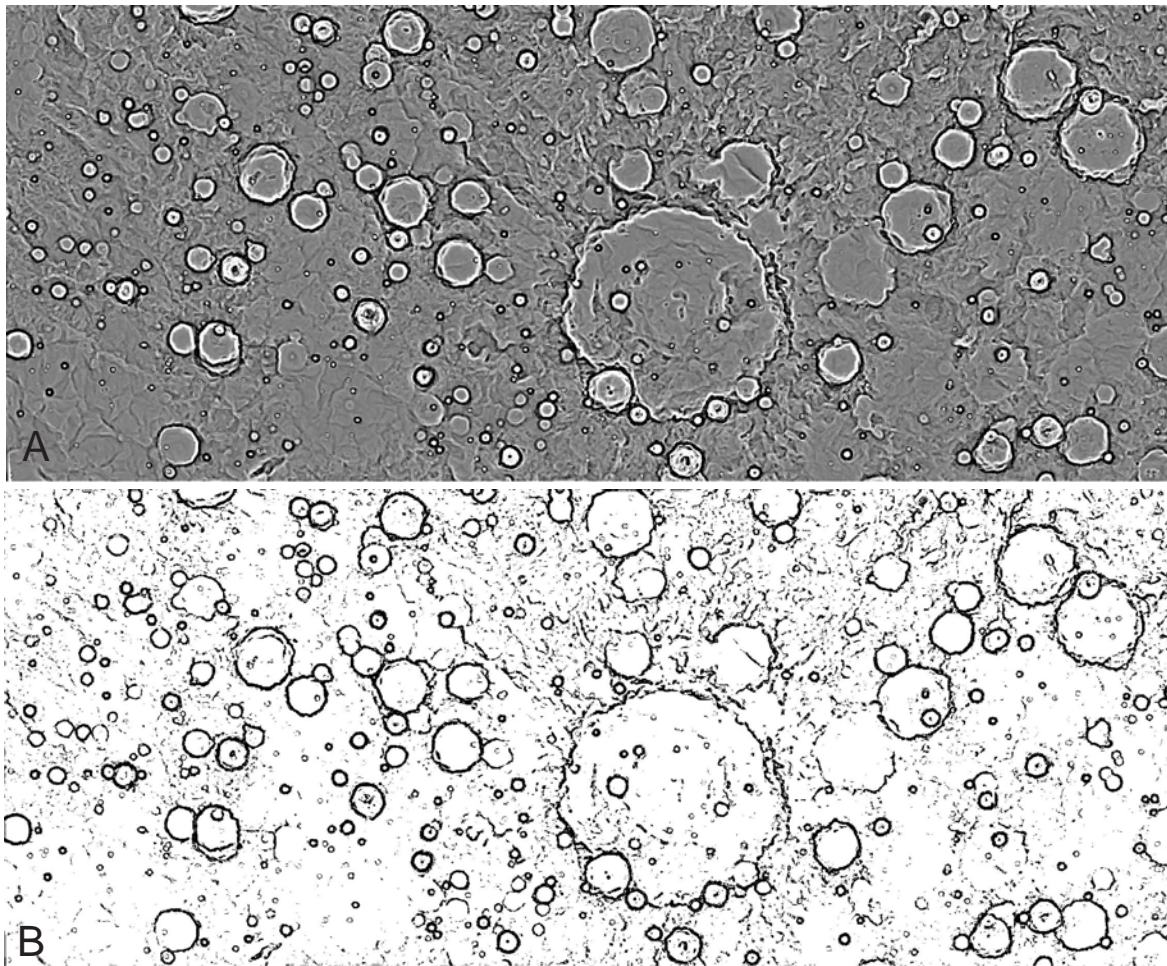


Fig. 3. (A) The grayscale gradient depicts values of profile curvature in the test site from large negative (concave areas), shown in black, to large positive (convex areas), shown in white. (B) The binary (black and white) image of the test site showing the areas with  $\kappa \leq -0.001$

the most concave areas and is depicted by black. Visual inspection of Fig.3A reveals that the majority of pixels are displayed in gray indicating near-zero curvature values. Craters appear as double (black and white) circles reflecting the rims and the walls, respectively.

We use the values of  $\kappa(x, y)$  to produce a binary (black and white) image,  $\mathcal{I}_\kappa$ , of the site

according to the following transformation,

$$\mathcal{I}_\kappa(x, y) = \begin{cases} 1 \text{ (black)}, & \text{for } \kappa(x, y) \leq \kappa_{\text{th}} \\ 0 \text{ (white)}, & \text{for } \kappa(x, y) > \kappa_{\text{th}} \end{cases} \quad (1)$$

Here,  $\kappa_{\text{th}}$  is a threshold value for concave areas. The chosen value of  $\kappa_{\text{th}}$  represents a tradeoff between selectivity and the presence of noise. Choosing  $\kappa_{\text{th}}$  close to  $\text{Min}[\kappa(x, y)]$  selects only areas with the highest concavity, eliminating noise, but also misses the rims of smaller or degraded craters. Choosing  $\kappa_{\text{th}}$  close to 0 selects areas with a broader range of concavity, including small and degraded craters, but also a fair amount of noise. We have chosen to use relatively large value of  $\kappa_{\text{th}} = -0.001$  in order to increase detection chances for small or degraded craters. Fig.3B shows the final binary image of thresholded profile curvature constructed for our test site. Comparing Fig.3B to Fig.2A indicates that profile curvature has delineated (to some degree) all craters in the DEM. Some noise is present, but does not present a major obstacle to crater detection since its spatial distribution does not display circular features.

#### IV. SITE SEGMENTATION

Craters are enclosed topographic basins, meaning that if they were subjected to unlimited rainfall they would collect water and become lakes. The size of the lake would be determined by crater's top off point. In an idealized situation this property could be used by itself to detect craters from the DEM. Applying the "flooding" algorithm [18] we produce a binary image,  $\mathcal{I}_f$  of the site according to the following transformation,

$$\mathcal{I}_f(x, y) = \begin{cases} 1 \text{ (black)}, & \text{for } \delta(x, y) > 0 \\ 0 \text{ (white)}, & \text{for } \delta(x, y) = 0 \end{cases} \quad (2)$$

where  $\delta(x, y)$  is the flood field described in section 2. Fig.4A shows the binary image,  $\mathcal{I}_f(x, y)$  constructed for the test site. Segmentation of  $\mathcal{I}_f$  into connected components produces numerous fragments. Comparison of Fig.4A with Fig.2A shows that indeed many of these fragments

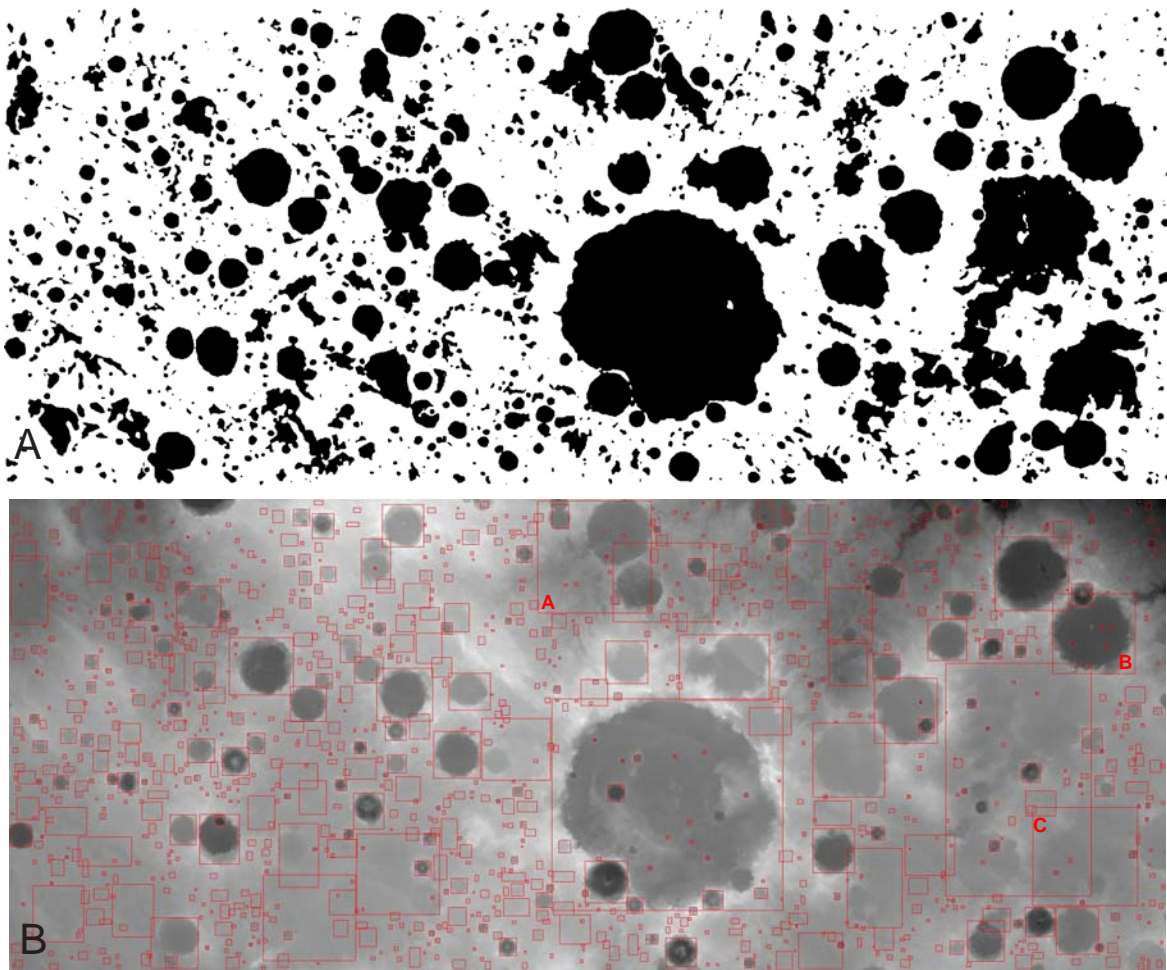


Fig. 4. (A) The binary image of the test site showing the spatial distribution of the flood,  $\delta(x, y)$ . (B) The background shows the values of elevation in the test site as depicted by the grayscale gradient; high areas are shown in white, low areas are shown in black. The red rectangles are bounding boxes of individual connected components in  $\mathcal{T}_f$ . Three bounding boxes are labeled A, B, and C for further discussion.

correspond to craters, however, the correspondence is not good enough to constitute a basis for a crater detection algorithm. This is because a real Martian landscape includes enclosed basins that are not craters - some craters are not basins due to degradation of their rims, and there are superimposed craters that form only a single fragment. These realities prevent flooding from serving as a stand-alone crater detection algorithm, however, we use it to reduce a computational

cost of detecting craters from  $\mathcal{I}_\kappa$ .

We use a connected components labeling algorithm to segment  $\mathcal{I}_f$  into  $k$  fragments; there are  $k = 1099$  fragments in the  $\mathcal{I}_f$  constructed for the test site. Each fragment contains some degree of topographic depression, but not all depressions are single craters, some are not craters at all, other contain multiple craters. For each fragment we calculate a bounding box  $B_i$ ,  $i = 1, \dots, k$ . To allow for situations where craters can be slightly larger than fragments due to low top off point caused by rim degradation, each  $B_i$  is slightly larger than the extent of its enclosed fragment. Fig.4B shows all 1099 bounding boxes overlaid on grayscale depiction of an elevation field in the test site. The bounding boxes divide the site into a large number of much smaller areas without cutting through the craters.

Because  $\mathcal{I}_f$  and  $\mathcal{I}_\kappa$  are coregistered images, applying bounding boxes  $B_i$  to  $\mathcal{I}_\kappa$  divides a binary image of thresholded profile curvature into  $k$  small images,  $\mathcal{I}_\kappa^i$ ,  $i = 1, \dots, k$  without cutting through the craters. We detect craters from each small image separately. This procedure guarantees that all detectable craters in the entire site are detected (possibly more than once). The benefit of site segmentation is the much reduced computational cost of crater detection as explained in section 2.

## V. CRATER DETECTION

Detection of craters in a small binary image  $\mathcal{I}_\kappa^i$  is achieved in three steps. In the first step  $\mathcal{I}_\kappa^i$  is modified by morphological closing operation followed by a thinning operation. We perform a morphological closing with a  $3 \times 3$  structuring element to smooth the edges of structures and to eliminate small holes in them. We use the thinning operation for skeletonization - reducing all lines to a single pixel thickness. Fig.5 shows a comparison of an original binary image of thresholded profile curvature confined to the bounding box labeled B on Fig.4B with the result

of applying the closing and thinning operations to that image. In the second step we use the circular HT to detect crater candidates. In the third step we examine each candidate using our confirmation algorithm. This algorithm either accepts a candidate as a crater and adds it to the catalog or rejects it. These three steps are performed for all fragment images  $\mathcal{I}_\kappa^i$ ,  $i = 1, \dots, k$  and the catalog is appended each time a new fragment is processed. Finally, we run an elimination algorithm on the entire catalog to remove duplicate detections.

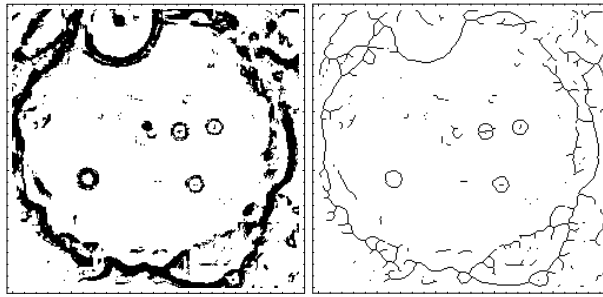


Fig. 5. (Left) Binary image of thresholded profile curvature confined to a bounding box labeled B on Fig.4B. (Right) The same image after modification by morphological closing and thinning operations.

#### A. Hough transform

We use the circular HT to detect crater candidates in binary images  $\mathcal{I}_\kappa^i$ ,  $i = 1, \dots, k$ . In previous work [4], [14] on image-based crater detection algorithms, done in the context of an autonomous spacecraft navigation system, the ellipse fitting HT was used. The much more computationally expensive elliptical HT make sense when craters are detected from images of asteroids, where they often appear as elliptical features because of image perspective. In the context of a rectilinear Martian DEM, the overwhelming majority of craters are circular, hence we have opted to use the less computationally expensive circular HT to detect crater candidates.

The circular HT is a simple transformation of a binary image to a numerical 2-D array of the same dimensions as the image. For every pixel in an image a corresponding cell in an array is assigned a total count of black pixels (indicating potential crater edges) on a circle centered at that pixel and having a radius equal to  $r$  pixels. Thus, the HT depends on the  $r$  parameter; the cells containing the higher counts are likely to be the centers of circles of radius  $r$ . In this paper we refer to this 2-D array as a layer,  $\mathcal{L}_r$ , and to the value of the count in each cell of a layer as correlation,  $\mathcal{L}_r(x, y)$ . For every image  $\mathcal{I}_\kappa^i$ , having dimensions of  $n_i \times m_i$  pixels, we are interested in finding craters as small as  $r_{\min}$  ( $= 5$  pixels in our calculations) up to as high as  $r_{\max}$  ( $= 1/2 \text{ Min}[n_i, m_i]$  in our calculations). Thus, we construct a 3-D array  $\mathcal{L}^i = (\mathcal{L}_{r_{\min}}^i, \dots, \mathcal{L}_j^i, \dots, \mathcal{L}_{r_{\max}}^i)$  of  $r_{\max} - r_{\min} + 1$  layers, each corresponding to a progressively larger crater radius.

In conventional applications of the circular HT the problem of finding circular structures from  $\mathcal{L}_j^i$  is tantamount to finding locations of cells with the highest correlation values. However, we have found that it is difficult to implement and inadequate to simply apply the conventional method to images of thresholded profile curvature. First, the distribution of correlation values in a layer is smooth and without obvious breaks, making it difficult to decide on selection criteria. Second, we have observed cases where cells with lower values of correlation correspond to centers of real craters whereas the cells corresponding to higher values of correlation do not correspond to craters at all. Clearly, in our context, the correlation value alone is not an adequate determinant of circular feature. Therefore, we have designed an unconventional procedure, that relies on values of correlation and “density” to obtain a list of crater candidates from  $\mathcal{L}^i$ .

The idea is to progressively narrow the selection of potential crater centers in  $\mathcal{L}_j^i$ . We start by selecting pixels (cells) with “high” values of correlation. Using only non-zero correlation



values, we calculate  $\mathcal{L}_{j,\text{th}}^i$ , the threshold correlation value. Only pixels with  $\mathcal{L}_j^i(x, y) \geq \mathcal{L}_{j,\text{th}}^i$  are kept for further consideration.  $\mathcal{L}_{j,\text{th}}^i$  is defined as the value of correlation for which the cumulative distribution function of all correlation values in a layer reaches a predetermined number  $q$ . Setting  $q$  just below 1 results in the selection of outliers - atypically high values of correlation. In our calculation we use  $q = 1 - 50/(\text{total number of non-zero correlation values in a layer})$ . This statistical approach is necessary because different layers have different distributions of correlation, and a single preset value of a threshold would not be appropriate for all layers.

To further narrow the selection of crater centers we calculate a density measurement,  $\rho_j^i$ , a 2-D array of the same dimensions as  $\mathcal{L}_j^i$ . First, the layer  $\mathcal{L}_j^i$  is modified by assigning a value of 0 to all pixels with  $\mathcal{L}_j^i(x, y) < \mathcal{L}_{j,\text{th}}^i$ . This modified layer is used to calculate the density values;  $\rho_j^i(x, y)$  is defined as a count of all non-zero pixels in a square,  $h \times h$  neighborhood centered on the  $(x, y)$  pixel. The high value of  $\rho_j^i(x, y)$  indicates that not only the  $(x, y)$  pixel, but also its neighboring pixels, are considered potential crater centers. In our context this increases the chances of  $(x, y)$  being the true crater center. Our second selection round is to keep only pixels with  $\rho_j^i(x, y) \geq \rho_{j,\text{th}}^i$ . The threshold is given by  $\rho_{j,\text{th}}^i = \text{Max}[\rho_{j,\text{max}}^i - \rho_0, \rho_{\text{alt}}]$ , where  $\rho_{j,\text{max}}^i$  is the maximum value of density in a layer,  $\rho_0$  is a predetermined density value, and  $\rho_{\text{alt}}$  is a predetermined, alternative density threshold value. The value of  $\rho_0$  regulates the range of acceptable densities, and the value of  $\rho_{\text{alt}}$  sets the smallest acceptable value of density.

The selection of craters centers during the HT calculation depends on three parameters,  $h$ ,  $\rho_0$ , and  $\rho_{\text{alt}}$ . The values of these parameters depends on index  $j$  (the size of craters to be detected); they are given in Table 1. To find a list of crater candidates for fragment  $\mathcal{I}_\kappa^i$  we repeat the procedure described above for all layers in  $\mathcal{L}^i$ , appending the list each time the next layer is processed. The  $l$ -th element of the final list is  $(r_l, x_{0,l}, y_{0,l}, \mathcal{L}_{j(l)}^i(x_{0,l}, y_{0,l}), \rho_{j(l)}^i(x_{0,l}, y_{0,l}))$ ,

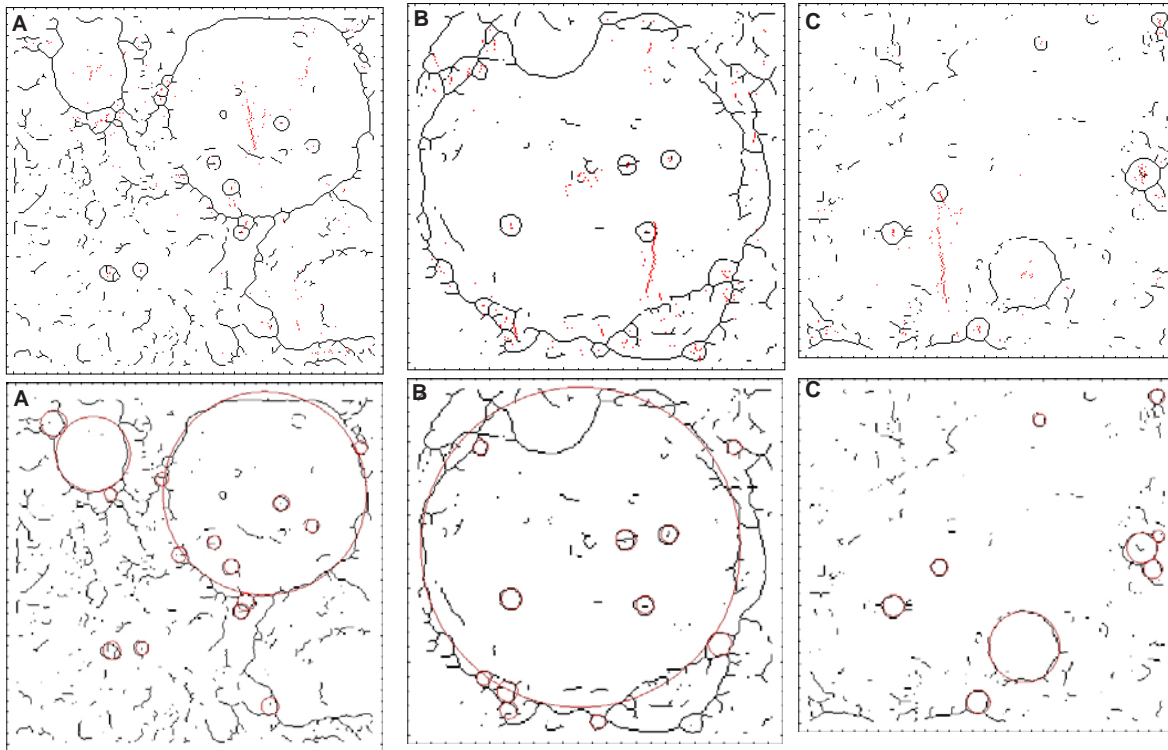


Fig. 6. (Top) Crater centers identified by the HT procedure are shown as red pixels in binary images of thresholded profile curvature confined to bounding boxes A, B, and C as labeled on Fig.4B. (Bottom) The final selection of craters by the confirmation algorithm is shown as red circles on top of binary images of curvature for the same three fragments.

corresponding to potential crater radius, x-coordinate of its center, y-coordinate of its center, the value of correlation, the value of density, respectively. Fig.6 (top three panels) shows examples of the results produced by the calculations described in this subsection. For three fragments denoted A, B, and C on Fig.4B, the calculations selected 241, 248, and 191 pixels, respectively, as potential crater centers. These centers are shown as red pixels on Fig.6. A crater radius is associated with each center but it's not shown on Fig.6. Note the relatively large numbers of selected centers - our priority here is not to miss a crater - nevertheless, the HT procedure reduces the number of potential crater centers by a factor of  $\sim 10^5$  from the original number of

pixels in  $\mathcal{I}_\kappa^i$ .

### B. Confirmation algorithm

In our context the HT is not capable of accurate crater detection, so it is used only for narrowing the number of possible candidates as described in the previous subsection. The actual detection is achieved by a confirmation algorithm which examines each candidate and determines whether it is a crater. We have developed this confirmation algorithm on the basis of our experience working with the DEM data. The input is the set of  $N_i^c$  craters candidates found by the HT in an image  $\mathcal{I}_\kappa^i$ . For the  $l$ -th candidate we select rim (black) pixels  $(x, y)$  in  $\mathcal{I}_\kappa^i$  that fulfills the relation  $(1 - \epsilon)^2 r_l^2 \leq (x - x_{0,l})^2 + (y - y_{0,l})^2 \leq (1 + \epsilon)^2 r_l^2$ , or pixels located within a ring of thickness  $2\epsilon$  centered on the candidate's perimeter. The idea of examining pixels in such a ring is that although craters' rims depart from exact circles, rim-defining pixels should be restricted to a fairly narrow ring. We also divide a ring into  $2\pi r_l$  angular segments and count a number of rim pixels within each segment. The pixels constituting a crater rim should form a fairly continuous string, a large number of empty angular segments indicates a landform other than a crater. We use four criteria to determine whether the  $l$ -th candidate is a crater. Fitness,  $f_l$ , measures how closely pixels in a ring fit to a circle. Circularity,  $C_l$  measures how circular is the shape of a figure obtained by connecting all pixels in a ring. Total "empty angle",  $\phi_l^{\text{emp}}$ , is an angular measure of all empty angular sectors, or a measure of discontinuity of a potential rim. Maximum gap,  $\phi_l^{\text{max}}$ , is an angular measure of the largest number of consecutive empty angular segments, or a measure of the largest gap in a potential rim.

All crater candidates are divided into three categories, small ( $5 < r < 13$ ), medium ( $14 < r < r_{\text{max}}/2$ ), and large ( $(r_{\text{max}}/2) + 1 < r < r_{\text{max}}$ ). Each category uses a slightly different crater detection algorithm. For large craters we first select candidates fulfilling the maximum  $\phi_l^{\text{emp}}$

criterion,  $\phi_l^{\text{emp}} \leq \phi_{\text{th}}$ . Candidates with rim discontinuities greater than  $\phi_{\text{th}}$  are rejected. For large craters this critical discontinuity measure depends also on crater size,  $\phi_{\text{th}} = \phi_{\text{th},1}$  if  $r_l > 30$  and  $\phi_{\text{th}} = \phi_{\text{th},2}$  if  $r_l \leq 30$ . The candidates that pass the maximum  $\phi_l^{\text{emp}}$  criterion are subjected to the fitness criterion. A cumulative distribution function of all candidates' fitness values is then constructed. A fitness threshold is the value of  $f_l$  for which the cumulative distribution function reaches a predetermined value of  $CDF_{\text{th}}$ . Candidates with values of  $f_l$  greater than the threshold are rejected, only the most fitting candidates are kept. This statistical criterion is necessary because a variety of binary rim images makes impossible to select a single threshold. We also use an absolute minimum value of threshold,  $f_{\text{min}}$ , and no statistically selected threshold is allowed to be smaller than  $f_{\text{min}}$ .

A detection algorithm for medium craters has three steps. The first step is the fitness criterion, identical to that described for the large craters. The second step is the maximum  $\phi_l^{\text{emp}}$  criterion identical to that described for the large craters but with a single value of  $\phi_{\text{crit}}$ . Finally, the third step is the maximum gap criterion,  $\phi_l^{\text{max}} < \varepsilon_g \phi_{\text{th},1}$ . Candidates with gaps in their rims that are larger than a given percentage of the total rim discontinuity are rejected.

Small craters require the most elaborate detection algorithm. We first select candidates that meet the fitness criterion. A fitness threshold,  $f_{\text{th}}$ , is statistically determined using the same method as for the large and medium craters. However, even this statistically customized threshold value, which is individually determined for each image, is not specific enough for small craters and explicit dependence of the threshold value on crater size is required. We use the value of  $f_{\text{th}}$  for candidates with the largest radii and the value of  $p_{f,1} f_{\text{th}} > f_{\text{th}}$  for candidates with the smallest radii. For all other candidates we determine the value of threshold via linear interpolation between these two extremes. This reflects an observation that smaller craters require a relaxed fitness

TABLE I  
PARAMETERS OF THE HT AND CONFIRMATION ALGORITHMS.

parameter	small	medium	large
$h$	5	3	3
$\rho_0$	4	0	0
$\rho_{\text{alt}}$	2	7	7
$\epsilon$	0.5	0.2	0.2
$CDF_{\text{th}}$	0.1	0.1	0.1
$f_{\text{min}}$	0.015	0.015	0.015
$p_{\text{f},1}$	1.5	–	–
$p_{\text{f},2}$	2.25	–	–
$C_{\text{th}}$	0.8	–	–
$\phi_{\text{th},1}$	150°	140°	100°
$\phi_{\text{th},2}$	165°	–	106°
$\varepsilon_{\text{g}}$	–	0.7	–
$\varphi_1$	45°	–	–
$\varphi_2$	130°	–	–
$\varphi_1$	230°	–	–

criterion. In parallel to the fitness criterion we also calculate a circularity criterion,  $C_l > C_{\text{th}}$  and  $f_l < p_{\text{f},2} f_{\text{th}}$ . The purpose of circularity criterion is to accept candidates that are very circular in shape under much more relaxed fitness criterion. The union of candidates that passed the fitness and circularity criteria are subjected to the maximum  $\phi_l^{\text{emp}}$  criterion. Again, small craters require individual thresholds of  $\phi_l^{\text{emp}}$ , and better fitting candidates are subject to more relaxed criterion than worse fitting candidates. We use the value of  $\phi_{\text{th},2}$  for the best fitting candidate and the value of  $\phi_{\text{th},1}$  for the worst fitting candidate. For all other candidates we determine the value

of  $\phi_{th}$  from linear interpolation between these two extremes. The candidates that passed the maximum  $\phi_l^{emp}$  criterion are subjected to the gaps criterion. Many candidates for small craters are terrain between two ridges. However, whereas a crater rim constitutes a more or less closed curve, the two ridges are discontinuous in two roughly opposite directions. We calculate the sizes of the two largest gaps,  $\phi_l^{max}$ ,  $\phi_l^{max-1}$  and an angular distance between them  $\Delta g$ . The candidate is rejected if  $\phi_l^{max-1} > \varphi_1$  and  $\varphi_2 < \Delta g < \varphi_3$ , as the fulfillment of these conditions indicates the existence of two separate ridges.

The output of the confirmation algorithm is a list of craters detected in a given image. For images A, B, C, shown on Fig.6 the confirmation algorithm has detected 16, 12, and 9 craters, respectively. These detected craters are shown as red circles on the bottom panels of Fig.6. Only about 5% of crater candidates suggested by the HT algorithm has been confirmed as craters by the confirmation algorithm.

## VI. APPLICATION TO THE TEST SITE.

Our test site, described at the end of section 2, is a heavily cratered terrain covering  $\sim 1.0 \times 10^6$  km<sup>2</sup>, the Barlow catalog lists 476 craters within its bounds. Our segmentation module has divided the test site into 1099 separate segments, and our crater detection module has identified 603 distinct craters in all the segments. The number of craters is smaller than a number of fragments because, although each fragment contains topographic depression, not all depressions are craters. In addition, many smaller craters are found in multiple fragments; the elimination algorithm removes duplicate detections.

Fig.2B shows a Viking-based, MDIM 2.1 image of the test site with detected craters indicated by circles of different colors. This image mosaic is coregistered with the DEM and has a resolution twice that of the DEM. We use an image instead of the DEM to display our results for

visual reasons, as small craters are more readily visible in an image than in the visual rendering of a DEM. The visual rendering of the DEM is shown on Fig.2A for side by side comparison with the image. The red circles denote craters that our algorithm has detected and that are present in the Barlow catalog; there are 307 such craters. Thus, we have detected 64% of all craters in the Barlow catalog. The blue circles denote craters in the Barlow catalog that our algorithm failed to detect; there are 169 such craters. The majority (112) of these missed craters are heavily degraded, showing no sharp rims. An additional 12 craters are only partially within the bounds of the site and are not detected because they are not full circles. The remaining 45 craters should have been detected but weren't. Thus, only 9% of craters in the Barlow catalog, capable of being identifying by our algorithm, were not detected. The yellow circles denote craters that have been identified by our algorithm but are absent from the Barlow catalog; there are 157 such craters. These are predominantly (but not exclusively) small craters that are easy to miss by manual survey. Interestingly, we have also detected 2 craters with diameter  $> 50$  km that are missing from the Barlow catalog. The green circles denote false positives, features identified as craters that aren't; there are 135 such features. They are mostly small features located on the rims of large craters. Fig.7 summarizes a comparison between the Barlow catalog and the catalog produced by our algorithm. In few cases of double craters, our algorithm has identified a single crater where in reality two similar size craters are superimposed on each other.

#### A. *Quality assessment.*

In order to compare different crater detection techniques it is important to use standard quantitative quality assessment factors. Following [12] we use factors developed in [24]: detection percentage,  $D = 100TP/(TP + FN)$ , branching factor,  $B = FP/TP$ , and quality percentage,  $Q = 100TP/(TP + FP + FN)$ . Here,  $TP$  stands for the number of true positive detections (de-

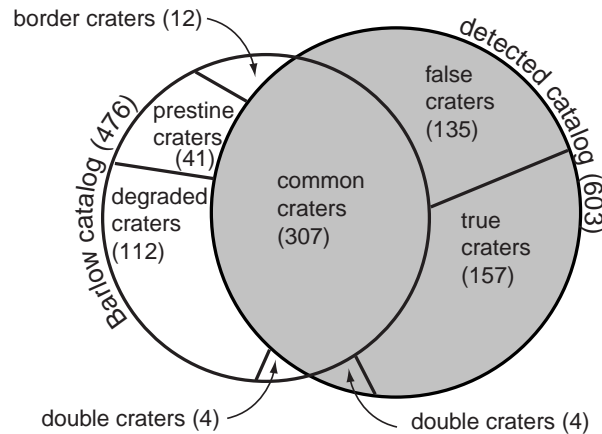


Fig. 7. Schematic comparison of craters cataloged by our detection algorithm with craters in the Barlow catalog.

tected craters are actual craters),  $FP$  stands for the number of false positive detections (detected craters are not actual craters), and  $FN$  stands for the number of false negative “detections” (nondetection of real craters).  $D$  can be treated as a measure of crater detection performance,  $B$  can be treated as a measure of delineation performance, and  $Q$  can be treated as an overall measure of algorithm performance. Calculation of these factors requires the existence of a “ground true” reference catalog. In our case there is no such reference catalog, so we assume that the union of Barlow catalog and all true craters in our catalog (633 craters altogether) constitutes the “ground true” catalog. With such assumption the factors for our algorithm are:  $D = 74\%$ ,  $B = 0.29$ , and  $Q = 61\%$ . If we exclude the craters that our method is not capable of detecting (heavily degraded and located on the edge) then  $D = 92\%$ ,  $B = 0.29$ , and  $Q = 73\%$ . The factors for the Barlow catalog are:  $D = 75\%$ ,  $B = 0$ ,  $Q = 75\%$ . These factors indicate that detection performance of our algorithm is on par with the performance of manually compiled Barlow catalog, although the manual catalog has a definitive edge in delineating craters from non-craters. Our algorithm significantly outperforms manual detection of nondegraded craters



but fails to detect heavily degraded craters.

## VII. DISCUSSION.

In this paper we have presented the first comprehensive method for detecting Martian craters from digital topography data. Although our input is the DEM data, the actual crater detection is performed from binary images resulting from thresholding image maps of profile curvature. Thus, the relative simplicity of our algorithm in comparison with image-based algorithms can be attributed to working with binary instead of grayscale images. The justification of our approach is twofold, first, detecting craters from DEMs is more efficient than detecting them from images; second, a DEM-based method offers a possibility of a seamless integration of detection and full characterization of craters. This second advantage of the DEM-based technique is of special interest for scientific studies where properties of craters are of particular interest. This paper focuses only on the detection portion of the technique, future research will address characterization. Our technique is suitable for large scale survey of Martian surface, in fact, the ultimate goal is to use this technique to compile a catalog of craters for the entire surface of Mars.

Ideally, performance of different crater detection algorithms should be compared by applying them to the same test site. Unfortunately, performance of each published detection algorithm is demonstrated on a different site or set of sites making a fair comparison impossible. With this caveat, we compare the performance of our algorithm to the performances of image-based algorithms [1], [12]. In [1] crater detection was applied to 26 images taken by the Mars Orbiter Camera (MOC). These images have resolution of 246 m/pixel or about twice the resolution of DEMs used in our study. They cover  $\sim 0.35 \times 10^6 \text{ km}^2$  or about a third of the area covered by our test site. Overall their algorithm has identified 546 craters with  $TP = 171$ ,  $FN = 93$  and

$FP = 282$ , resulting in quality assessment factors  $D = 64\%$ ,  $B = 1.65$ ,  $Q = 31\%$ . These results points to a significantly worse performance of their image-based algorithm in comparison to our DEM-based algorithm. Especially striking is the large number of false detections. Moreover, a visual review of original images identified by reference number in [1] reveals that they cover a significantly less complex terrain than our test site. In addition, we suspect that  $FN = 93$  is an undercount, judging by the large number of small undetected craters in two images for which detection results are shown in [1].

In [12] a comprehensive detection algorithm was applied to a number of images taken by Viking Orbiter, MOC, and High Resolution Stereo Camera. The overall performance for MOC images is given as  $D = 88\%$ ,  $B = 0.15$ ,  $Q = 78\%$ . This indicates a very impressive performance, however the images were chosen not to include heavily degraded craters because the algorithm is described as not capable of detecting such craters. This explains the high value of  $D$ . The low value of  $B$  is due to removal of false detections by the neural network, a post-detection supervised process. With degraded craters eliminated, the performance ( $D$ ) of our algorithm slightly surpasses the performance ( $D$ ) of algorithm presented in [12], keeping degraded craters decreases the performance of our algorithm.

The two shortcomings of our present algorithm are nondetection of degraded craters and a relatively large number of false detections. These problems are encountered by all crater detection algorithms. To address the issue of degraded craters one would have to find a terrain attribute capable of delineating them from a surrounding landscape. One possible solution is to perform a nonlinear transformation of elevation values in order to magnify curvature differences. The issue of false detections will be much reduced once we develop the characterization aspect of our algorithm. These objects would have a highly atypical character and could be removed on

that basis.

## ACKNOWLEDGEMENTS

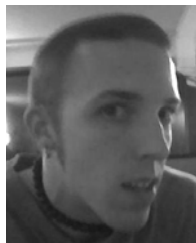
The authors would like to thank Nadine Barlow for providing the ASCII version of “The Catalog of Large Martian Impact Craters.” This research was supported by NSF under grant IIS-0430208 and was conducted at the Lunar and Planetary Institute, which is operated by the USRA under contract CAN-NCC5-679 with NASA. This is LPI Contribution No. 1304.

## REFERENCES

- [1] T. Barata, E. Ivo Alves, J. Saraiva, and P. Pina, “Automatic Recognition of Impact Craters on the Surface of Mars”, *Proc. Image Analysis and Recognition: International Conference ICIAR 2004, Porto, Portugal*, pp. 489-496, 2004.
- [2] N.G. Barlow, “Crater Size-distributions and a Revised Martian Relative Chronology”, *Icarus*, vol. 75(2), pp. 285-305, 1988.
- [3] M.C. Burl, T. Stough, W. Colwell, E.B. Bierhaus, W.J. Merline, and C. Chapman, “Automated Detection of Craters and Other Geological Features”, *Proc. Int. Symposium on Artificial Intelligence, Robotics and Automation for Space, Montreal, Canada*, 2001.
- [4] Y. Cheng, A.E. Johnson, L.H. Matthies, and C.F. Olsen, “Optical Landmark Detection for Spacecraft Navigation”, *Proc. 13th AAS/AIAA Space Flight Mechanics Meeting, Ponce, Puerto Rico, AAS 03-224*, pp. 1785-1803, 2003.
- [5] M.J. Cintala, J.W. Head, and T.A. Mutch “Martian Crater Depth/Diameter Relationship: Comparison with the Moon and Mercury”, In *Proc. Lunar Sci. Conf.*, vol. 7., pp 3375-3587, 1976.
- [6] M.J. Cintala, and P.J. Mouginis-Mark, “Martian Fresh Crater Depth: More Evidence for Subsurface Volatiles”, *Geophys. Res. Lett.*, vol. 7, pp. 329-332, 1980.
- [7] F.M. Costard, “The Spatial Distribution of Volatiles in the Martian Hydrolitosphere”, *Earth Moon and Planets*, vol. 45, pp. 265-290, 1989.
- [8] Crater Analysis Technique Working Group, “Standard Techniques for Presentation and Analysis of Crater Size-Frequency Data”, *Icarus*, vol. 37, pp. 467-474, 1979.
- [9] J. Earl, A. Chicarro, Ch. Koeberl, P. Giorgio Marchetti, and M. Milnes, “Automatic Recognition of Crater-like Structures in Terrestrial and Planetary Images”, In *Workshop on Role of Volatiles and Atmospheres on Martian Impact Craters*, #3053, Lunar and Planetary Institute, Houston, 2005.

- [10] R. Honda, Y. Iijima, and O. Konishi, "Mining of Topographic Feature from Heterogeneous Imagery and Its Application to Lunar Craters", *Progress in Discovery Science: Final Report of the Japanese Discovery Science Project*, Springer, p. 395, 2002.
- [11] P.V.C. Hough, "Method and Means for Recognizing Complex Patterns", *U.S. Patent 3069654*, 1962.
- [12] J.R. Kim, J-P. Muller, S. van Gasselt, J.G. Morley, G. Neukum, and the HRSC Col Team, "Automated Crater Detection, A New Tool for Mars Cartography and Chronology", *Photogrammetric Engineering & Remote Sensing*, vol. 71(10), pp. 1205-1217, 2005.
- [13] R.O. Kuzmin, N.N. Bobina, E.V. Zabalueva, and V.P. Shashkina, "Structural Inhomogeneities of the Martian Cryosphere", *Solar System Res.*, vol. 22, pp. 121-133, 1988.
- [14] B. Leroy, G. Medioni, A.E. Johnson, and L.H. Matthies, "Crater Detection for Autonomous Landing on Asteroids", *Image and Vision Computing*, vol. 19, pp. 787-792, 2001.
- [15] M. Magee, C.R. Chapman, S.W. Dellenback, B. Enke, W.J. Merline, and M.P. Rigney, "Automated Identification of Martian Craters Using Image Processing", In *Lunar and Planetary Science XXXIV*, #1756, Lunar and Planetary Institute, Houston (CD-ROM), 2003.
- [16] G.G. Michael, "Coordinate Registration by Automated Crater Recognition", *Planetary and Space Science*, vol. 51, pp. 563-569, 2003.
- [17] H. Mitsova, and J. Hofierka, "Interpolation by Regularized Spline with Tension: II. Application to Terrain Modeling and Surface Geometry Analysis", *Mathematical Geology*, vol. 25, pp. 657-669, 1993.
- [18] J.F. O'Callaghan and D.M. Mark, "The extraction of drainage networks from digital elevation data", *Computer Vision, Graphics and Image Processing*, vol. 28, pp.328-344, 1984.
- [19] C. Plesko, S. Brumby, E. Asphaug, D. Chamberlain, and T. Engel, "Automatic Crater Counts on Mars", In *Lunar and Planetary Science XXXV*, #1935, Lunar and Planetary Institute, Houston (CD-ROM), 2004.
- [20] D.J. Roddy, N.R. Isbell, C.L. Mardock, T.M. Hare, M.B. Wyatt, L.M. Soderblom, and J.M. Boyce, "I. Martian Impact Craters, Ejecta Blankets, and Related Morphologic Features: Computer Digital Inventory in Arc/Info and Arcview Format", In *Lunar and Planetary Science XXIX*, #1874, Lunar and Planetary Institute, Houston (CD-ROM), 1998.
- [21] J.F. Rodionova, K.I. Dekhtyareva, A.A. Khranchikhin, G.G. Michael, S.V. Ajukov, S. G. Pugacheva, V.V. Shevchenko, "Morphological Catalogue Of The Craters Of Mars", *ESA-ESTEC*, 2000.
- [22] D. Smith, G. Neumann, R.E. Arvidson, E.A. Guinness, and S. Slavney, "Mars Global Surveyor Laser Altimeter Mission Experiment Gridded Data Record", *NASA Planetary Data System*, MGS-M-MOLA-5-MEGDR-L3-V1.0, 2003.
- [23] L.A. Soderblom, C.D. Condit, R.A. West, B.M. Herman, and T.J. Kreidler, "Martian Planetwide Crater Distributions:

- Implications for Geologic History and Surface Processes”, *Icarus*, vol. 22, pp. 239-263, 1974.
- [24] J.A. Shufelt, “Performance Evaluation and Analysis of Monocular Building Extraction from Aerial Imagery”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21(4), pp. 311-326, 1999.
- [25] K.L. Tanaka, “The Stratigraphy of Mars”, *J. Geophys. Res.*, vol. 91, supplement, pp. E139-E158, 1986.
- [26] T. Vinogradova, M. Burl, and E. Mjosness, “Training of a Crater Detection Algorithm for Mars Crater Imagery”, *Aerospace Conference Proc. 2002, IEEE*, vol. 7, pp. 7-3201 - 7-3211, 2002.
- [27] P. Wessel, and H.F. Smith, “Free Software Helps map and Display Data”, *EOS Trans. AGU*, vol. 72, pp. 441, 1991.
- [28] P.G. Wetzler, B. Enke, W.J. Merline, C.R. Chapman, and M.C. Burl, “Learning to Detect Small Impact Craters”, *Seventh IEEE Workshops on Computer Vision (WACV/MOTION’05)*, vol. 1, pp. 178-184, 2005.
- [29] D.U. Wise and G. Minkowski, “Dating Methodology of Small, Homogeneous Crater Populations Applied to the Tempe-Utopia Trough Region on Mars”, *NASA Tech. Memo*, 81776, pp. 122-124, 1980.
- [30] M.T. Zuber, and I. Garrick-Bethell, “What Do We Need to Know to Land on the Moon Again?”, *Science*, vol. 310, no. 5750, pp. 983-985, 2005.



**Brian Bue** was an undergraduate intern at the Lunar and Planetary Institute (LPI) in Summer of 2004, where he worked with Dr. Stepinski on automated classification of Martian terrain. The following summer, he worked as a visiting graduate student at LPI towards extending the classification method to do automatic crater detection. He is currently finishing a MS in computer science at Purdue University, and plans to pursue a PhD in the near future. His E-mail address is [bbue@cs.purdue.edu](mailto:bbue@cs.purdue.edu)



**Tomasz Stepinski** is a staff scientist at the Lunar and Planetary Institute in Houston, TX. He received a MS in astrophysics from Warsaw Univesity in Warsaw, Poland in 1979, and the PhD in applied mathematics from the University of Arizona in 1986. Since 1990 he has been working at the Lunar and Planetary Institute pursuing research in planetary science and astrophysics. He has been working on topics ranging from magnetic field generation in accretion disks and evolution of solids in protoplanetary disks, to extrasolar planets and diversity of planetary systems. His current research interest is computational geomorphology with application to Martian valley networks and other landforms on Mars and Earth. His E-mail address is [tom@lpi.usra.edu](mailto:tom@lpi.usra.edu)