Crater Delineation by Dynamic Programming

Jorge S. Marques and Pedro Pina

***Abstract*—An algorithm to automatically delineate the contour of impact craters detected on remotely sensed images from the surface of Mars is presented in this letter. It processes the crater images in polar coordinates and is constituted by two main steps: *edge enhancement*, which is constructing an edge map based upon the intensity transitions along radial lines intersecting the center of the crater, and *crater delineation*, which is determining an optimal path from the minimization of an energy functional by Dynamic Programming. We obtained the performance of 96% of correct crater delineations evaluated in a data set of 1045 craters depicted from High Resolution Imaging Science Experiment (HiRISE) and Thermal Emission Imaging System (THEMIS) images (resolutions of 0.25–0.50 and 100 m/pixel, respectively) exhibiting a large diversity of terrains, crater dimensions, and degradations.**

***Index Terms*—Dynamic Programming, edge map, impact craters, Mars, rims.**

I. Introduction

CRATER detection algorithms (CDAs) have greatly evolved in the last decade, detecting much smaller structures with higher performances [1]–[5]. These performances are permitting their widespread use in the construction and upgrade of crater catalogs, e.g., from Mars [6] or Phobos [7]. CDA can still be improved, namely, for detecting tens of meters of craters, but the algorithms already available are robust enough to help detect impact craters at a large scale.

Thus, the focus of attention can now be moved toward crater characterization, namely, to establish a degree of preservation or erosion for each individual crater. Until now, this kind of study has been performed partially and at two different scales: first, using large craters (from tens to hundreds of kilometers in diameter) for detecting resurfacing events and to reconstruct past climates on Mars [8] and, second, using small craters (from few meters to few kilometers in diameter) to understand their processes of formation and evolution [9] or to define more accurate engineering models of the surfaces for reliable lander design and safe landings [10]. The measurements performed are based on a limited numbers of samples using approximate descriptions, obtained with specific tools in a manual mode [11]. This prevents any large-scale analysis and restrains the geographic area of study and the dimensional range of analysis.

Manuscript received November 11, 2014; revised February 10, 2015; accepted March 11, 2015. This work was developed in the frame of the projects PTDC/CTE-SPA/110909/2009, UID/EEA/50009/2013, and UID/ECI/04028/2013, funded by the Fundação para a Ciência e a Tecnologia (Portugal).

J. S. Marques is with the Institute for Systems and Robotics (ISR), Instituto Superior Técnico, University of Lisbon, 1049-001 Lisbon, Portugal (e-mail: jsm@isr.ist.utl.pt).

P. Pina is with the Centre for Natural Resources and Environment (CERENA), Instituto Superior Técnico, University of Lisbon, 1049-001 Lisbon, Portugal (e-mail: ppina@tecnico.ulisboa.pt).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/LGRS.2015.2413753

In order to establish a degree of erosion for each individual crater (of any size), a full analysis of the crater image and its surroundings is needed, encompassing not only parameters related to its shape or its ejecta but also descriptors of the irregularity of the contour and the identification of the broken or missing parts of the rim. The first step for establishing this issue consists in the delineation of the real contour of the crater. We remind that CDA outputs describe each crater by perfect circular shapes, no matter its degree of erosion, which are sufficient for chronostratigraphic studies [12] but not for full characterizations.

The delineation of the crater rim has been manually performed, and there are almost no algorithms in the literature addressing this problem. The exceptions are two recent approaches of ours: one based on a judicious sequence to find and link the crater edges in polar coordinates [13] and the other based on the watershed transform and other mathematical morphology operators [14]. The results obtained on a small data set from Mars achieved good rates, but when we enlarged the data set, encompassing a larger diversity of terrain ages and textures, the performances were not preserved. The evident degradation of the performance in the most difficult examples showed us that there was still room for improvement.

Therefore, the objective of this letter is to propose a novel algorithm able to deal with the automated delineation of impact craters of any size and degree of preservation, on a wide variety of terrains, and adequate for large-scale delineations.

II. Algorithm

A. Overview

We wish to estimate the crater contour, assuming that we know in advance its location and dimension, even though crater delineation is a challenging task since crater images often present low contrast between the rim and surroundings, making their detection very subtle. In addition, craters are often located in highly textured regions and superposed to each other, making the delineation of their borders even more difficult.

The first step of the algorithm relies on intensity variation and tries to detect its changes associated with the rim, while the second step tries to link the edges using geometric information. Unfortunately, simple edge detection and linking approaches fail in these images. Edge detection algorithms provide unreliable edges, most of them associated to the terrain irregularities. Most of the detected edges do not belong to the crater rim and cannot be connected by an edge-linking algorithm. To circumvent this difficulty, we define a continuous edge map, $e(\mathbf{x}) \in [0, 1]$, which measures the amount of directional intensity variation in the vicinity of each point $\mathbf{x}$. A value $e(\mathbf{x}) = 0$ is assigned to a pixel $\mathbf{x}$ if there is strong intensity variation in the vicinity of $\mathbf{x}$ in a direction orthogonal to the crater contour. On the contrary, a value $e(\mathbf{x}) = 1$ is assigned if the image is constant in such direction.

In the second step, we compute a closed contour, $\mathbf{x}(s)$, that minimizes an energy functional

$$E = \int e\left(\mathbf{x}(s)\right) ds + E_{int}(\mathbf{x}) \tag{1}$$

similar to the one used in the snake algorithm [15], [16]; $E_{int}(\mathbf{x})$ denotes the internal energy which measures deviations of the crater contour, $\mathbf{x}(s)$, with respect to a circle, and s denotes the arc length parameter of the curve.

It should be stressed that both operations become simpler and more effective if the image is converted to polar coordinates. These steps are addressed in the next sections.

B. Conversion to Polar Coordinates

Since craters have circular shape, they can be easily represented using polar coordinates, with the origin located at their center $\mathbf{c} = (c_1, c_2)$. The conversion from Cartesian coordinates $\mathbf{x} = (x_1, x_2)$ to polar coordinates (r, θ) is given by

$$x_1 = c_1 + r\cos\theta \quad x_2 = c_2 + r\sin\theta \tag{2}$$

where r denotes the distance of the point $\mathbf{x}$ to the crater center $\mathbf{c}$ and $\theta \in [0, 2\pi]$ denotes the angle of the vector $\mathbf{x} - \mathbf{c}$ with respect to the first axis. Given a crater image $I(\mathbf{x})$, we will define a polar image $P(r, \theta)$, assuming that corresponding points in both images have the same intensity, i.e., $P(r, \theta) = I(x_1, x_2)$ for all discrete values of (r, θ)

$$\theta = i\Delta\theta, \quad i = 0, \ldots, N-1 \tag{3}$$
$$r = r_{\min} + j\Delta r, \quad j = 0, \ldots, M-1 \tag{4}$$

where M denotes the number of radius values, N denotes the number of θ values, and $\Delta r = (r_{\max} - r_{\min})/M, \Delta\theta = 2\pi/N$. The computation of $P(r, \theta)$ requires the interpolation of the crater image $I(x)$ at noninteger values, e.g., using bilinear interpolation [17].

C. Edge Map

We wish to define an edge map in polar coordinates $e(r, \theta)$. This map should assign a low value to points which are likely to be edges and high values to points which are not. We will assume that edges are associated to intensity transitions along radial lines intersecting the crater center $\mathbf{c}$ (θ constant).

The radial gradient is defined as

$$g(r, \theta) = |P(r, \theta) * h(r)| \tag{5}$$

where $*$ denotes the convolution operation along the columns of P and $h(r)$ is the impulse response of a high-pass filter, defined by $h(r) = -u(r - T) + 2u(r) - u(r + T)$, where $u(r)$ is the unit step function. This convolution can be extremely fast if we compute the integral image along the columns of P [18].

After computing the gradient, the edge map is obtained using the logistic function

$$e(r, \theta) = \frac{2}{1 + e^{sg(r,\theta)}} \tag{6}$$

which maps the gradient intensity $g \in [0, +\infty[$ into an edge confidence $e \in [0, 1[$. Since we have a good estimation of the radius of the crater, R, we will restrict r to an interval $[r_{\min}, r_{\max}]$ centered on R.

Fig. 1. Image transformation. (Left) Original image and sampling points, (center) polar image, and (right) edge map.

Fig. 1 shows the conversion from Cartesian to polar coordinates, assuming that $r_{\min} = 0.8R, r_{\max} = 1.2R$, and the corresponding edge map. The first and last rows of the edge map are padded with high intensity values since the high-pass filtering results are unreliable.

D. Crater Delineation

The final step concerns crater delineation. We will assume that the edge map e has M lines and N columns. The crater boundary is characterized by a sequence of row indices $\mathbf{r} = (r_1, r_2, \ldots, r_N)$ such that $r_t \in \{1, \ldots, M\}$. These indices represent the crater radius for each direction. If the crater boundary was a circle centered at $\mathbf{c}$, then the index sequence would be constant. In practice, the radius r_t changes slowly and must obey the boundary condition $r_1 = r_N$ since it represents a closed contour.

The contour sequence $\mathbf{r}$ is chosen to minimize an energy functional

$$E(\mathbf{r}) = e(r_1 = k, 1) + \sum_{p=2}^{N} e(r_p, p) + c(r_{p-1}, r_p) \tag{7}$$

where $e(r_p, p)$ is the edge map and $c(r_{p-1}, r_p)$ denotes the cost associated to the transition from r_{p-1} to r_p. In this letter, we assume that $|r_p - r_{p-1}| \leq 1$ to enforce smooth transitions and the transition cost is defined by

$$c(r_{p-1}, r_p) = \begin{cases} 0 & \text{if } |r_p - r_{p-1}| = 0 \\ \alpha & \text{if } |r_p - r_{p-1}| = 1 \\ +\infty & \text{otherwise} \end{cases}. \tag{8}$$

The minimization of $E(\mathbf{r})$ under the constraint $r_1 = r_N = k$ can be solved by Dynamic Programming [19], [20]. Dynamic Programming has been used by several authors for the estimation of deformable contours, e.g., see [21]–[23].

Dynamic Programming minimizes $E(\mathbf{r})$ in two steps. The first step computes the optimal costs to go from column 1 and line k to column t and line j, defined as follows:

$$E_t(j) = \min_{r_2, \ldots, r_t : r_t = j} \left[e(r_1 = k, 1) + \sum_{p=2}^{t} e(r_p, p) + c(r_{p-1}, r_p) \right] \tag{9}$$

assuming an initial condition $r_1 = k$. These optimal costs can be efficiently computed by a forward recursion

$$E_t(j) = e(j, t) + \min_i \left[E_{t-1}(i) + c(i, j) \right]. \tag{10}$$

TABLE I
DYNAMIC PROGRAMMING ALGORITHM WITH BOUNDARY CONDITIONS $r_1 = r_N = k$

Forward recursion: computation of the optimal energies

$$E_1(j) = \begin{cases} e(k,1) & \text{if } j = k \\ +\infty & \text{otherwise} \end{cases}$$

$$E_t(j) = e(j,t) + \min_i \left[E_{t-1}(i) + c(i,j)\right], \qquad t = 2, \ldots, N$$

$$\psi_t(j) = \arg\min_i \left[E_{t-1}(i) + c(i,j)\right], \qquad t = 2, \ldots, N$$

Backward recursion: computation of the optimal contour

$$r_N^* = k$$

$$r_{t-1}^* = \psi_t(r_t^*) \qquad t = N, \ldots, 2\,.$$

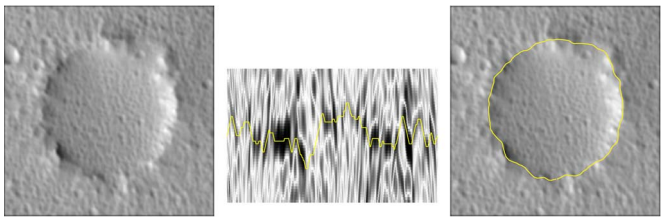

Fig. 2. Contour delineation. (Left) Original image, (center) edge map and optimal contour, and (right) transformed contour.

Since we want to retrieve the optimal path, it is important to store which value of i minimizes $[E_{t-1}(i) + c(i,j)]$ in (10). This information can be stored using a set of pointers

$$\psi_t(j) = \arg\min_i \left[E_{t-1}(i) + c(i,j)\right]. \tag{11}$$

After computing the optimal costs $E_t(j), t = 1, \ldots, N, j = 1, \ldots, M$, we know what is the minimum energy associated to an optimal path $r_1^*, \ldots, r_N^*$ ending in $r_N^* = k$. The optimal path $r^* = (r_1^*, r_2^*, \ldots, r_N^*)$ such that $r_N^* = k$ can be obtained by backtracking

$$r_{t-1}^* = \psi_t\left(r_t^*\right) \quad t = N, \ldots, 2. \tag{12}$$

The Dynamic Programming algorithm under the restriction $r_1 = r_N = k$ is summarized in Table I. This algorithm provides the optimal path assuming that we know the boundary conditions k. Since the optimal k is unknown, we repeat this procedure for all possible values of $k \in \{1, \ldots, M\}$ and choose the one which minimizes the energy.

Fig. 2 shows an example of an estimated contour plotted over the edge map (polar image) and the initial crater image.

III. EXPERIMENTAL RESULTS

A. Data Sets

We tested the algorithm on 1045 craters observed on remotely sensed images of Mars acquired by two cameras: 845 craters from High Resolution Imaging Science Experiment (HiRISE) images and 240 craters from a Thermal Emission Imaging System (THEMIS) global mosaic. Their distinct spatial resolutions permit analyzing a wide and complementary range of crater dimensions: subkilometric craters on the 0.25- and 0.50-m/pixel resolutions of the HiRISE camera and large kilometric craters (from few to up to several hundreds) on the 100-m/pixel resolution of the THEMIS Day IR 100m Global Mosaic.

In the construction of the data sets, we covered a large diversity of the Martian landscape. We selected regions in both hemispheres, with noticeable differences in crater densities, also exhibiting a wide variety of preservation, from pristine craters (with sharp rims) to degraded structures (with irregular, faint, or missing parts of the rim), and also examples of craters hardly noticeable.

The HiRISE images (data sets 1 to 8) comprise craters selected from eight distinct Martian regions, with a maximum diameter of 1.1 km and going down to 5 and 10 m for the smallest structures, i.e., 20 pixels for the resolutions of 0.25 and 0.50 m/pixel, respectively. For the THEMIS images (data set 9), the craters were extracted from the global mosaic all around the planet, with diameters below 200 km until a minimum of 4 km (or 40 pixels). Although craters smaller than those dimensions can, in many situations, be undoubtedly perceived, we decided to fix an inferior limit to avoid ambiguous situations.

B. Evaluation Method

We evaluated the algorithm through the comparison of the delineated contour with a manually created contour (ground-truth contour). Each crater was manually delineated, also estimating a contour in regions where the crater rim was absent, i.e., creating always one single closed contour for each impact structure. The ground-truth contours were delineated by an expert, and the ambiguous situations were decided by two more people. We believe this is enough to evaluate the method, although it would be interesting to obtain a statistical evaluation of the human errors produced by multiple experts.

The distortion between those pair of contours can be defined by several ways (e.g., [24]). In our case, we assume that the detected contour should be within a band of a given width Δ around the ground-truth contour. Points located outside this band are considered as errors, with their percentage used as a measure of distortion between both contours.

Let A denote the set of boundary points detected by the delineation algorithm, and let B denote the ground-truth boundary defined by an expert. We define the distance from a point $p \in A$ to the ground-truth contour B as the distance from p to the nearest point in B

$$d(p, B) = \min_{q \in B} \|p - q\|. \tag{13}$$

The amount of errors is given by

$$E_\Delta = \frac{1}{\#A} \sum_{p \in A} \delta\left(d(p,B) > \Delta\right) \tag{14}$$

where $\#A$ is the number of points of set A; $\delta(x) = 1$ if $x =$ true, and $\delta(x) = 0$ if otherwise. In this letter, we choose $\Delta = 0.05D$, where D is the crater diameter. If $d(p,B) = 0$, the point p is assumed to be correct, and there is no error (cp). If $0 < d(p,B) < \Delta$, the error in p is considered a small error (se), and if $d(p,B) \geq \Delta$, p is considered a gross error (ge).

C. Performances

The algorithm was tested assuming the following configuration for the parameters: $N = 360$ (angular step size 1°), $M = 61$ (radius step size), $\alpha = 0.02$ (transition cost), $s = 1$

TABLE II
AVERAGE PERFORMANCES (%) OF AUTOMATED CRATER DELINEATION
(*cp*—CORRECT POINTS; *ge*—GROSS ERRORS; *se*—SMALL ERRORS)

Dataset #	Craters Nb	Dynamic Prog.			Morphologic		
		cp	*se*	*ge*	*cp*	*se*	*ge*
1	60	71.9	25.6	2.5	49.2	41.2	9.6
2	100	42.4	49.2	8.4	34.9	51.6	13.6
3	110	71.0	26.3	2.7	47.8	42.8	9.4
4	65	44.1	44.0	11.9	30.0	50.1	19.9
5	100	67.1	29.1	3.8	47.5	43.9	8.6
6	120	54.6	40.2	5.2	49.4	41.8	8.9
7	135	67.5	29.3	3.3	49.4	41.0	9.6
8	115	59.0	36.2	4.8	50.8	40.6	8.6
9	240	72.1	26.3	1.5	42.0	46.9	11.1
1-8	805	60.1	34.9	5.0	45.8	43.7	10.5
1-9	1045	62.9	32.9	4.2	44.9	44.5	10.7

(edge map), and $T = 6$ (gradient). The computation time depends on the crater image size but can be considered as fast. The average processing time for the whole database of 1045 craters is 1.52 s per crater, running in the Matlab platform with an AMD Phenom 3.00-GHz processor.

Each crater was individually analyzed, and a closed contour was estimated by the current algorithm ("Dynamic Programming") and by one of the previous approaches ("Morphologic"). The "Polar" algorithm was also tested, but due to the high amount of craters for which it did not propose a contour (about 60%), we decided to not include it in our comparisons. The average performances obtained for each data set with those two algorithms are shown in Table II. Since, like in many applications, small errors are acceptable, we focus our analysis on gross errors (ge). The overall performance achieved by the proposed algorithm is high, with an error of only 4.2% of incorrect delineations. The average performance is better for THEMIS than for HiRISE images (errors of 1.5% and 5.0%, respectively). Also, the fluctuation within each HiRISE data set, due to the intrinsic characteristics of the different terrains, is acceptable. In comparison, the "Morphologic" algorithm obtained a lower global performance (overall error of 10.7%), which is also worse for each of the nine data sets.

The delineation outputs, provided in Fig. 3, are also a comprehensive illustration of the performances achieved since the level of accuracy attained by the algorithm is very precise. This is evident not only on craters where the rim is complete and clearly discernible but also in situations where the rim is obliterated by other craters or only partially visible. This thorough correctness is generally observed in the whole data set, independently of the dimension of the craters or of the sensor.

On the contrary, the difficulties in estimating the contours are very less frequent and only partially incorrect in some sections of the rim. They concern mainly elliptical craters with high eccentricity. The errors are due to the search of the optimal path by the Dynamic Programming algorithm within the circular mask, which does not permit to analyze the more distant regions of the rims of the craters of elliptical shape and consequently can never detect them. However, this geometric constraint is, on the contrary, a strong point in favor of the algorithm since the large majority of impact craters tend to be circular: The amount of elliptical craters in the inner solar system, in the range of 5–100 km in diameter, is estimated to be around 2–4% [25]. Nevertheless, the algorithm performs nicely in the large majority of irregularly shaped craters.

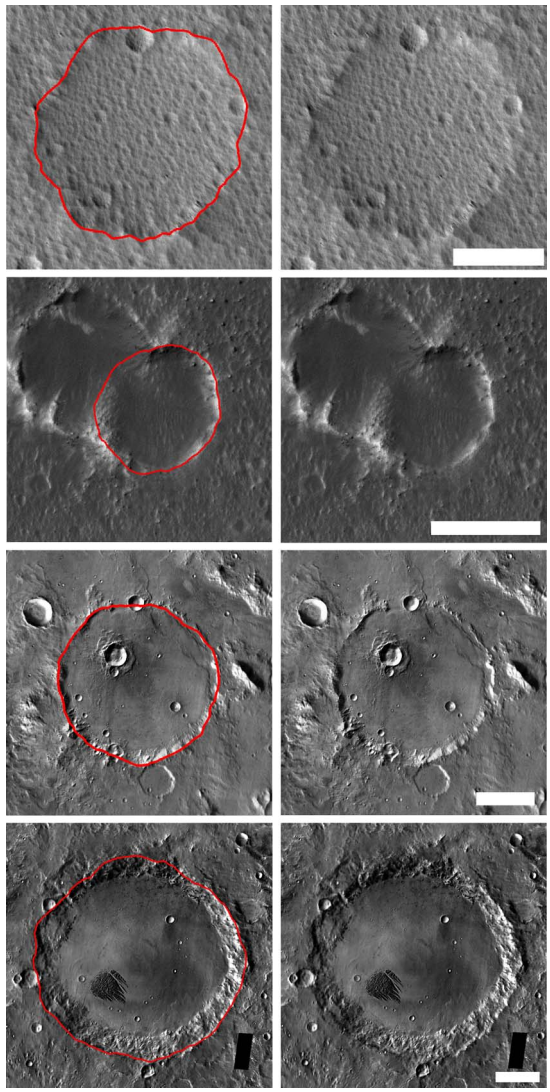

Fig. 3. Crater delineation examples. The white bars correspond to a scale of 50 m for HiRISE samples (first and second rows) and 20 km for THEMIS samples (third and fourth rows) [image credits: NASA/JPL/University of Arizona (HiRISE) and NASA/JPL-Caltech/Arizona State University (THEMIS)].

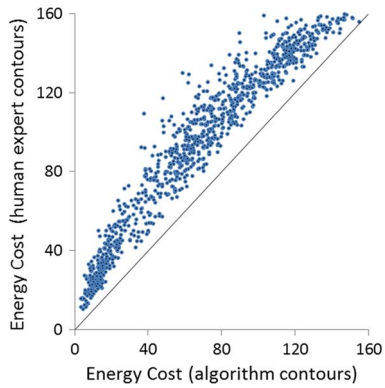

Fig. 4. Energy scatterplot between expert delineations and algorithm contours for the 1045 craters of the data set.

The ground-truth delineations and the estimated contours were obtained under different criteria. The former are obtained under human expert criteria, and the latter are obtained under

the optimization of an energy cost functional. Fig. 4 shows a comparison of the energies obtained by both approaches in 1045 craters. There is a good agreement between both methods, with the lowest energies obtained by the Dynamic Programming algorithm.

D. Sensitivity Analysis

It is important to characterize the performance of the algorithm if some of the parameters change or if the crater detection module presents significant displacements. We verified that displacements of the circular mask up to 20% can be recovered by the algorithm but larger displacements significantly change the estimated contour. This tolerance allows coping with relatively inaccurate estimates of the crater center and radius.

It is also interesting to check how the performance of the algorithm changes with the thickness of the mask characterized by $r_{\min}$ and $r_{\max}$. In the case of HiRISE images (data sets 1–8), the best results are achieved by the reference values of $r_{\min} = 0.8R$ and $r_{\max} = 1.2R$. In the case of THEMIS images (data set 9), the performance improves when the thickness is reduced. However, in this case, the algorithm becomes more dependent on the accuracy of the mask location and diameter.

Finally, we considered the influence of the discretization parameters M, N and the transition cost α on the algorithm performance. We evaluated the sensitivity of the algorithm with respect to the parameters of the reference configuration used in all experimental tests ($M = 61$, $N = 360$, and $\alpha = 0.02$), modifying each parameter p by the amounts $\pm 1/2p$. The results obtained indicate that the performance remains almost invariant to these changes.

IV. Conclusion

The algorithm achieved high delineation performances (average error of 4%) in a diversified data set of 1045 Martian craters with a large dimensional range (from 5 m to about 200 km), clearly outperforming the available alternatives, making it a good candidate for large-scale delineations.

The exploitation of the *a priori* knowledge about the problem, like the circular geometry and image intensity patterns of the craters, and its integration into an optimization procedure are the key features for the robustness of the algorithm. In particular, the geometry permits to adequately define a region of interest around the rim and constrain the space of search for edges of interest. Moreover, the enhancement of its edges by the edge map and the detection of the optimal path (the crater contour) with the Dynamic Programming algorithm are also strong points. Finally, converting and processing the crater images into polar coordinates greatly simplifies the algorithm.

In addition, this approach is not restricted to crater studies since it can be useful to delineate objects in other open application areas of remote sensing. Without being exhaustive, it can be used for individual tree delineation perceived on very high resolution images (for tree demography, growth modeling, and biomass estimation studies), in the delineation of the rotating circulation patterns observed in ocean surfaces (eddies) (for improving oceanic current characterization in relation with climatic phenomena), or in the delineation of circular infrastructures in oil and gas plants (for continuous risk monitoring and protection of the plants and surroundings).

References

[1] L. Bandeira, J. Saraiva, and P. Pina, "Impact crater recognition on Mars based on a probability volume created by template matching," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 12, pp. 4008–4015, Dec. 2007.

[2] E. R. Urbach and T. F. Stepinski, "Automatic detection of sub-km craters in high resolution planetary images," *Planet. Space Sci.*, vol. 57, no. 7, pp. 880–887, Jun. 2009.

[3] R. Martins, P. Pina, J. S. Marques, and M. Silveira, "Crater detection by a boosting approach," *IEEE Geosci. Remote Sens. Lett.*, vol. 6, no. 1, pp. 127–131, Jan. 2009.

[4] S. Vijayan, K. Vani, and S. Sanjeevi, "Crater detection, classification and contextual information extraction in lunar images using a novel algorithm," *Icarus*, vol. 226, no. 1, pp. 798–815, Sep./Oct. 2013.

[5] S. Jin and T. Zhang, "Automatic detection of impact craters on Mars using a modified adaboosting method," *Planet. Space Sci.*, vol. 99, pp. 112–117, Sep. 2014.

[6] G. Salamunićcar, S. Lončarić, P. Pina, L. Bandeira, and J. Saraiva, "MA130301GT catalogue of Martian impact craters and advanced evaluation of crater detection algorithms using diverse topography and image datasets," *Planet. Space Sci.*, vol. 59, no. 1, pp. 111–131, Jan. 2011.

[7] G. Salamunićcar, S. Lončarić, P. Pina, L. Bandeira, and J. Saraiva, "Integrated method for crater detection from topography and optical images and the new PH9224GT catalogue of Phobos impact craters," *Adv. Space Res.*, vol. 53, no. 12, pp. 1798–1809, Jun. 2014.

[8] J. Boyce and H. Garbeil, "Geometric relationships of pristine Martian complex impact craters, and their implications to Mars geologic history," *Geophys. Res. Lett.*, vol. 34, no. 16, Aug. 2007, Art. ID. L16201.

[9] W. A. Watters *et al.*, "Origin of the structure and planform of small impact craters in fractured targets: Endurance crater at Meridiani Planum, Mars," *Icarus*, vol. 211, no. 1, pp. 472–497, Jan. 2011.

[10] A. Basilevsky, M. Kreslavsky, I. Karachevtseva, and E. Gusakova, "Morphometry of small impact craters in the Lunokhod-1 and Lunokhod-2 study areas," *Planet. Space Sci.*, vol. 92, pp. 77–87, Mar. 2014.

[11] P. J. Mouginis-Mark, H. Garbeil, J. M. Boyce, C. S. E. Ui, and S. M. Baloga, "Geometry of Martian impact craters: First results from an interactive software package," *J. Geophys. Res.–Planet*, vol. 109, no. E8, Aug. 2004, Art. ID. E08006.

[12] W. K. Hartmann and G. Neukum, "Cratering chronology and the evolution of Mars," *Space Sci. Rev.*, vol. 96, no. 1–4, pp. 165–194, Apr. 2001.

[13] J. S. Marques and P. Pina, "An algorithm for the delineation of craters in very high resolution images of Mars surface," in *Pattern Recognition and Image Analysis*, vol. 7887, *Lecture Notes in Computer Science*. Berlin, Germany: Springer-Verlag, 2013, pp. 213–220.

[14] P. Pina and J. S. Marques, "Delineation of impact craters by a mathematical morphology based approach," in *Image Analysis and Recognition*, vol. 7950, *Lecture Notes in Computer Science*. Berlin, Germany: Springer-Verlag, 2013, pp. 717–725.

[15] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," *Int. J. Comput. Vis.*, vol. 1, no. 4, pp. 321–331, Jan. 1988.

[16] A. Blake, and M. Isard, *Active Contours*. New York, NY, USA: Springer-Verlag, 1998.

[17] R. Szeliski, *Computer Vision: Algorithms and Applications*. London, U.K.: Springer-Verlag, 2011.

[18] P. Viola and M. Jones,"Robust real-time face detection," *Int. J. Comput. Vis.*, vol. 57, no. 2, pp. 137–154, May 2004.

[19] R. E. Bellman, *The Bellman Continuum. A Collection of the Works of Richard E. Bellman*, R. S. Roth, Ed. Singapore: World Scientific, 1986.

[20] D. Bertsekas, *Dynamic Programming and Optimal Control*. Belmont, MA, USA: Athena Scientific, 2005.

[21] D. Geiger, A. Gupta, L. A. Costa, and J. Vlontzos, "Dynamic Programming for detecting, tracking, and matching deformable contours," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 17, no. 3, pp. 294–302, Mar. 1995.

[22] A. A. Amini, T. E. Weymouth, and R. C. Jain, "Using dynamic programming for solving variational problems in vision," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 12, no. 9, pp. 855–867, Sep. 1990.

[23] J. Cardoso, A. Capela, A. Rebelo, C. Guedes, and J. Pinto da Costa, "Staff detection with stable paths," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 6, pp. 1134–1139, Jun. 2009.

[24] K. Grauman and T. Darrell, "Fast contour matching using approximate earth mover's distance," in *Proc. IEEE CVPR*, 2004, pp. 220–227.

[25] G. S. Collins, D. Elbeshausen, T. M. Davison, S. J. Robbins, and B. M. Hynek, "The size-frequency distribution of elliptical impact craters," *Earth Planet. Sci. Lett.*, vol. 310, no. 1/2, pp. 1–8, Oct. 2011.