Feature Extraction Algorithms for Automatic Craters Identification

Nicole Christoff
[1]Technical University of Sofia
8 Kl. Ohridski Blvd, Sofia 1000
Bulgaria

[2]Aix-Marseille Université
CNRS, LSIS UMR 7296
France
{nicole.christoff@tu-sofia.bg}

ABSTRACT: *Recently the feature selection algorithms are extensively studied. Using 3D data, the features are drawn for automatic classification and identify craters. This will also help to text the performance of the classifiers. Our intention in this work is to observe the discriminative power of the original values, hereafter called "pure" values, of a minimal curvature by only converting them in the range of grey scale. We have tested the system and found that the five different classifiers show that better accuracy results are obtained over the features selected from the grey scale image. We also found that the method from computer vision is applied for the crater detection.*

Keywords: Mars Orbiter Laser Altimeter, 3D Mesh, Automatic Craters Detection, Machine Learning

Received: 27 August 2020, Revised 12 December 2020, Accepted 23 December 2020

DOI: 10.6025/jmpt/2021/12/1/1-8

1. Introduction

Observing and researching numbers and formation of craters, occuring on the surface of different celestial bodies is a big problem for astronomy. Its importance is connected to the estimation of age of the universe, the formation of stars and planets. The complexity of the problem arises when the impact crater zones are very heterogeneous due to the distribution and size of the craters. To be able to solve partially this problem, different methodologies of geometric image analysis are proposed.

Machine learning techniques have started being used on 2D surface images [3] and 2.5D [4], but there is no example of a method combining 3D mesh model as input and some type of classification method for recognition of crater rim. Some of the methods of impact crater detection, based on supervised algorithms, are neural network [5], support vector machine [6], Adaboost [2] and SparseBoost algorithm [7].

This paper provides an overview of three different classes of feature selection algorithms that can be employed in the task of automatic crater detection and an experimental setup with five different classifiers to test their discriminative power. The focus will be on proving that using only the minimal curvature information, calculated over the 3D meshes and converted to grey scale image, is enough to ensure good classification results.

This paper is organized as follows: Section 2 describes in detail the data processing algorithm, Section 3 gives a brief overview of the three classes of feature extraction algorithms, Section 4 present the classifiers that are going to be used in Section 5 for the experimental setup. The final section summarizes the key points of discussion and concludes the paper.

2. Data and Data Pre-processing

The 3D mesh data, employed in this research was generated from the Mars Orbiter Laser Altimeter (MOLA) with a resolution of 463.0836 m at the equator. The data is in equidistant cylindrical projection centered at (0°, 0°). The used sample is a rectangular region from Mars: Top: 13° N, Bottom: 0° N, Left: 25° W, Right: 0° W. It contains 5 328 000 vertices and 10 646 272 faces (Figure 1).

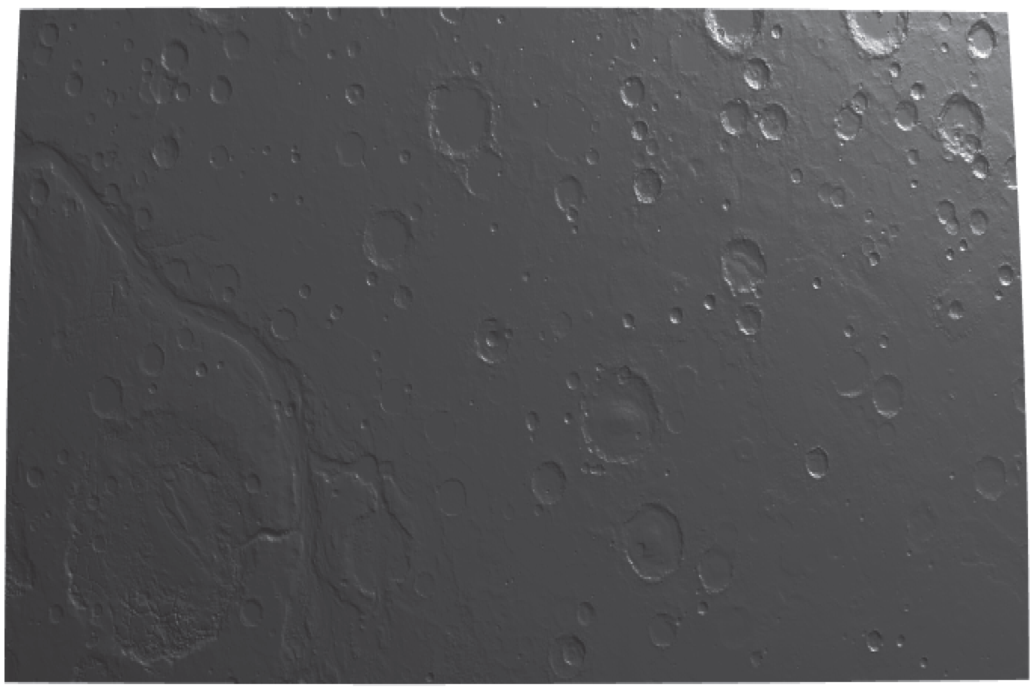
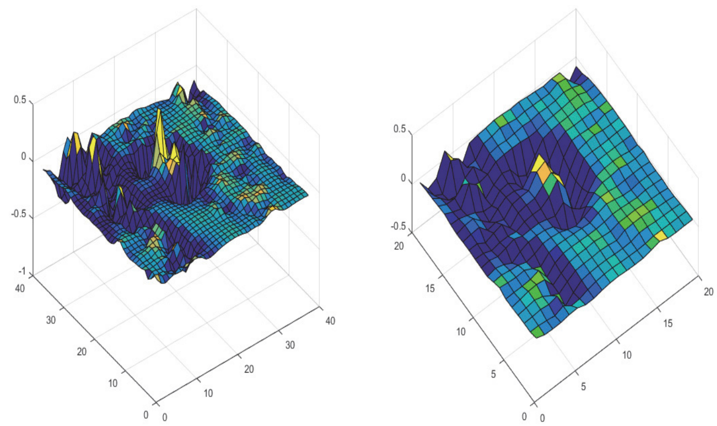


Figure 1. The piece of land of Mars

The first step of the sample data preparation is the computation of one of principal curvatures - the minimal curvature k_2. The two Principal Curvatures, k_1 and k_2 at a point $p \in S$, $S \subset R_3$ are the eigenvalues of the shape operator at that point. The k_1 and k_2 are respectively the maximum and minimum of the Second Fundamental Form. The principal curvatures measure how the surface bends by different amounts in different directions at that point [8].

The second step is to prepare a training set of positive and negative samples of craters. In this work, the Barlow catalogue [1] is used. It contains 459 craters for this area. Same number square blocks containing at least one crater, were extracted from the minimal curvature map as positive samples. The widths of the blocks are 1.5 times that of the crater diameter plus a constant, due to the calculation of error of displacement, equal to 10 pixels (Figure 2A). All the samples are resized to 20 x 20 pixels with the bilinear interpolation method (Figure 2B).

The same number of square blocks (459) are also randomly extracted containing no craters as negative samples. A second training set is prepared, where the values of minimal curvatures are converted into a quantified grayscale information k_{2G} between 0 and 255 using the minimum and maximum curvatures k_{2min} and $k_{2\max}$ (Figure 3 A and B).



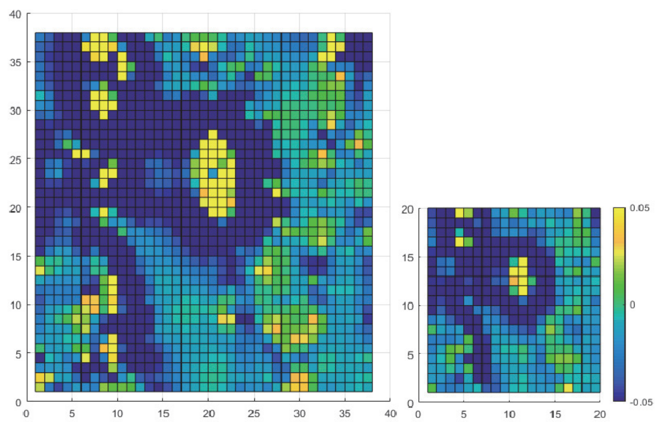


Figure 2. Positive sample “pure”: (A) Window with size equal to 1.5 times diameter of the crater plus constant equal to 10 px, (B) The same sample, resampled to 20x20 pixels

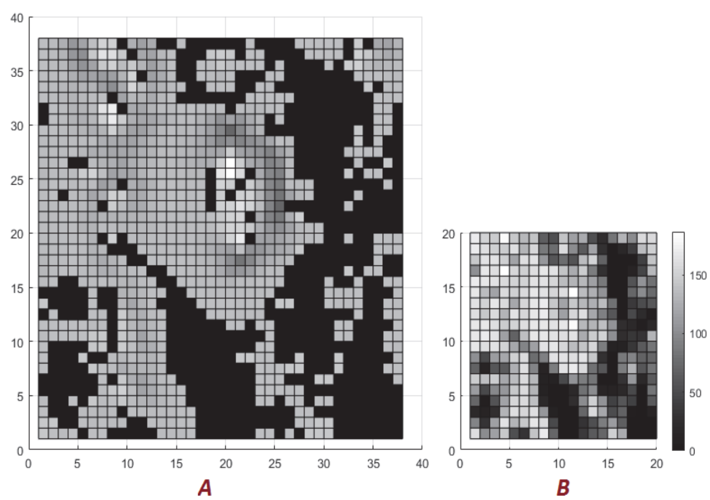


Figure 3. Positive sample in grey scale: (A) Window with size equal to 1.5 times diameter of the crater plus constant equal to 10 pixels, (B) The same sample, resampled to 20x20 pixels

3. Feature Extraction

Tree classes of features are calculated to train and test different classifiers: Local Binary Pattern operator (LBP), Haarlike and Scaled Haar-like feature. The LBP and Haar are an example of a typical computer vision computation pipeline for face recognition, used for dimension reduction preprocessing steps.

3.1. LBP Cascade Classifier

The Local Binary Pattern operator (LBP) was first introduced by [9] for byte adaptation of a previous study done by [10] and is a powerful texture descriptor. Texture is defined as a function of spatial variations in the pixel intensity of an image. The idea behind this operator is that common features, such as edges, lines, point, can be represented by a value in a particular numerical scale. As a result, 2124 features for each resized sample are obtained.

3.2. Haar-like Cascade Classifier

Haar-like features are attributes extracted from Region of Interest (ROI) used in pattern recognition. The utilization of these features instead of handling gray or color level of the pixels directly was proposed in [11]. First, the pixel values inside the black area are added together; then the values in the white area are summed. Only eight types of square masks are used to extract feature, as seen in Figure 4. Then the total value of the white area is subtracted from the total value of the black area. This result is used to categorize sub-regions. A total of 2048 features are obtained for each resized sample.

Figure 4. Eight type of masks are used for Haar-like feature extraction

The weak classifiers become strong classifiers when arranged in Haar-like cascade. They are able to detect structures despite illumination, color or scale variation. This method is one of the most popular techniques for face detection, firstly described by Viola and Jones [12].

3.3. Scaled Haar-like Feature

It is a variant of Haar-like feature. This feature is proposed by [2], which use the fact that large craters are usually deeper than small ones, and this means most of the Haar-like feature values of the large craters are larger than those of small ones. The scaled Haar-like feature is computed by division of the Haar-like feature value with a coefficient of the resolution of the sample to adjust the Haar-like features for large craters and small craters to the same scale. The number of scaled Haar-like features for all resized sample is the same as that of Haar-like features.

4. Training Different Classifiers

Five popular classifiers are used in this study.

4.1. K-Nearest Neighbours Algorithm (KNN)

Feature extraction is performed on raw data prior to applying KNN algorithm on the transformed data in feature space. A commonly used distance metric for continuous variables is Euclidean distance. Cosine similarity is very efficient to evaluate, especially for sparse vectors, as only the non-zero dimensions need to be considered.

4.2. Support Vector Machine (SVM)

SVM is a classier with good generalization power that uses the features extracted by the descriptor. SVM works basically by finding an optimal hyper plane that best separates the two classes. Kernel functions can be used with SVM in order to enable the classifier to deal with non-linearly separable classes. These functions modify the feature space trying to transform it into a linear separable problem. Several types of kernel functions are commonly used: uniform (linear), Gaussian (Radial Basis Function (RBF)) [13], quadratic [14] and cosine.

4.3. Decision Tree

Decision Trees are a non-parametric supervised learning method used for classification and regression. The goal is to create a model in the form of a tree structure. It breaks down a dataset into smaller and smaller subsets while at the same time an associated decision tree is incrementally developed. The final result is a tree with decision nodes and leaf nodes. A decision node has two or more branches. The leaf node represents a classification or decision [15].

4.4. AdaBoost

Boosted Trees incrementally builds an ensemble by training each new instance to emphasize the training instances previously mis-modeled. A typical example is AdaBoost. It is an approach to machine learning based on the idea of creating a highly accurate prediction rule by combining many relatively weak and inaccurate rules.

4.5. Bagged Tree

Bagging decision trees, an ensemble method, builds multiple decision trees by repeatedly resampling training data with re-placement, and voting the trees for a consensus prediction.

5. Results

For the tests the k-fold cross validation procedure is applied. 918 total samples of 918 are divided into 10 groups of equal size. Five different classifiers are trained each using 9 groups, holding out each of the groups. For each of the four classifiers, the group left out is tested. The 10 test results are averaged. All samples get to be used for both training and testing. The result is unbiased and with minimum variance.

The number of true positive detections is represented with TP (CR - detected craters, which are real craters and NCR – zones, with no craters), FP represents the number of false positive detections (CR – detected craters are not actual craters and NCR – detected no crater zones are not actual no craters zones), FN represents the number of false negative detections (CR – un-

10 fold cross validation			ACTUAL / PREDICTED: TP CR / TN CR	FN CR / FP NCR	TP NCR / TN NCR	FN NCR / FP CR
KNN	Euclidean	1	130	330	373	85
		3	63	397	449	9
		5	33	427	458	0
		7	25	435	457	1
		9	18	442	458	0
		11	14	446	458	0
	Cosine	1	188	272	369	89
		3	265	195	446	12
		5	262	198	454	4
		7	249	221	455	3
		9	245	215	456	2
		11	236	224	456	2
SVM	Linear		420	40	413	45
	Quadratic		418	42	427	31
	Cubic		401	59	431	27
	Gaussian		426	34	382	76
Decision tree			345	115	374	84
Ensemble classifiers	AdaBoost		407	53	429	29
	Bagged tree		402	58	406	52

Table 1. Confusion Matrix for Different Classifiers under “Pure”

detected real craters and NCR – undetected negative samples) and TN represents the number of true negative detections (CR – for positive samples and NCR – for no crater zones).

The best scenario is obtaining larger TP CR and TP NCR and smaller FP (CR and NCR). There are 459 positive and same number (459) negative samples. The best detection rate, for original values of minimal curvature, using SVM with quadratic kernel function respectively TP CR: 418, FN CR: 42, TP NCR: 427 and FN NCR: 31. A good TP CR results are obtained using SVM (Gaussian), but the number of FP CR increase two times (see Table 1).

10 fold cross validation			ACTUAL / PREDICTED			
			TP CR / TN CR	FN CR / FP NCR	TP NCR / TN NCR	FN NCR / FP CR
KNN	Euclidean	1	230	230	413	45
		3	229	231	439	19
		5	221	239	443	15
		7	216	244	443	15
		9	213	247	447	11
		11	209	251	445	13
	Cosine	1	275	185	391	67
		3	375	85	453	5
		5	389	71	450	8
		7	389	71	451	7
		9	394	66	451	7
		11	395	65	449	9
SVM	Linear		427	33	420	38
	Quadratic		452	8	442	16
	Cubic		455	5	440	18
	Gaussian		454	6	452	6
Decision tree			403	57	417	41
Ensemble classifiers	AdaBoost		428	32	251	207
	Bagged tree		458	2	445	13

Table 2. Confusion Matrix For Different Classifiers Under "Gray"

Using greyscale values of minimal curvature, better detection rate is obtained using SVM with Gaussian kernel function respectively TP CR: 454, FN CR: 6, TP NCR: 452 and FN NCR: 6. The best TP CR (458) is resulted, which is one of the challenges in astrophysics, using Bagged tree, but the number of FP CR increase two times (13) (see Table 2).

In Figure 5 is presented a comparative study with obtained accuracy results. The bins, colored in blue represent results for k_{2G} "grey" training set and with green – k_2 "pure" training set. Best accuracy detection rate for "pure" values of minimal curvature, we obtain using Bagged trees - 98.4%. For the greyscale values of k_2, best accuracy of 92.0% is obtained with SVM, using quadratic kernel function (Figure 5).

6. Conclusion and Future Work

In this paper, we have presented tree different methods for feature extraction from grey level image of minimal curvature, extracted from 3D mesh data, generated from the MOLA. We presented an overview of the ensemble of classifiers used to automatically detect craters on the surface of Mars. Thanks to the experimental results, we can conclude that the best performing classifier for crater detection is Bagged tree if we pass from RGB to grey scale image representation.

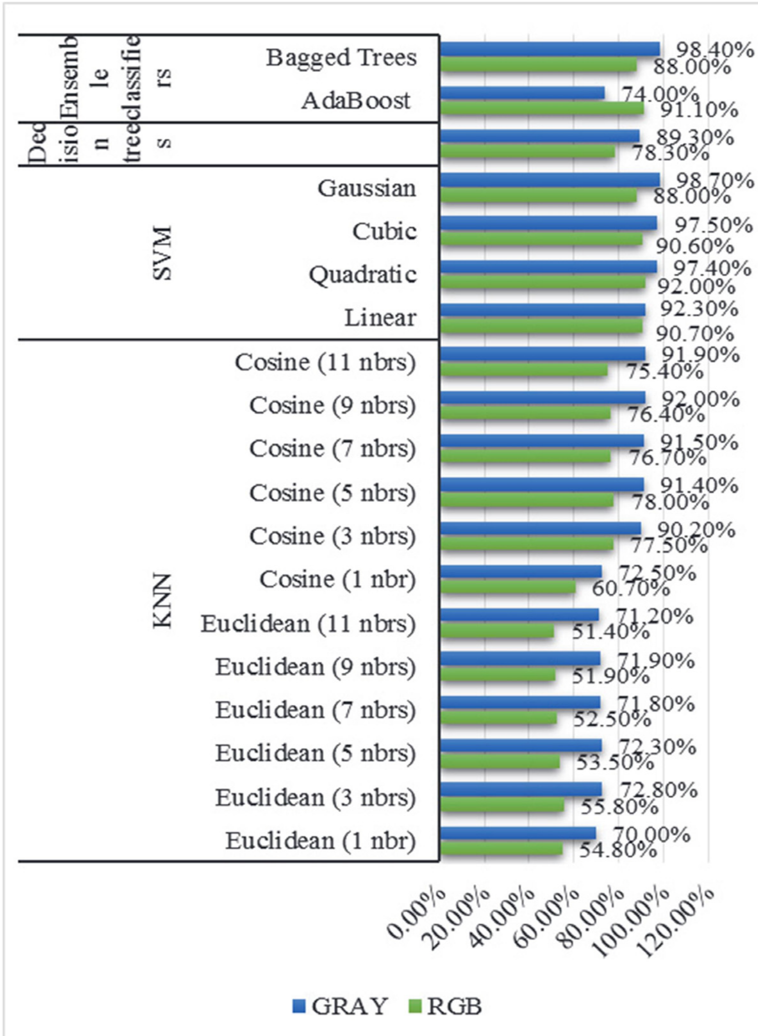


Figure 5. Accuracy

As future work, we plan to test those features on other types of calculated curvatures. Another step is to add the neural network back propagation of error classifier to the experimental setup.

Acknowledgement

This paper was supported by Contract No 162—0033-07 of Technical University-Sofia, Research Sector. Research project: «Segmentation and modeling of geometrical characteristics of 3D objects» - 2016.

References

[1] Barlow, N. G. (1988). Crater Size-Frequency Distributions and a Revised Martian Relative Chronology, *Icarus*, 75 (2) 285-305.

[2] Di, K., Li, W., Yue, Z., Sun, Y., Lui, Y. (2014). A Machine Learning Approach to Crater Detection From Topographic Data, *Advances in Space Research*, 54 (11) 2419-2429.

[3] Stepinski, T. F. (2012). Ding, W., Vilalta, R. (2012). *Detecting Impact Craters in Planetary Images Using Machine Learning*, p 146- 159.

[4] Vamshi, G. T., Martha, T. R., Kumar, K.V. (2016). An Object-Based Classification Method for Automatic Detection of Lunar Impact Craters From Topographic Data, *Advances in Space Research*, vol. 57, p 1978-1988.

[5] Kim, J. R., Muller, J., van Gasselt, S., Morley, J. G., Neukum, G. (2005). Automated Crater Detection, a New Tool for Mars Cartography and Chronology, *Photogramm. Eng. Remote Sens.*, 71 (10) 1205-1217.

[6] Wetzler, P. G., Honda, R., Enke, B., Merline, W. J., Chapman, C. R., Burl, M. C. (2005). Learning to Detect Small Impact Craters, *Proceedings 7th IEEE WACV/MOTION*, p 178-184.

[7] Wang, Y., Yang, G., Guo, L. (2015). A Novel Sparse Boosting Method for Crater Detection in the High Resolution Planetary Image, *Advances in Space Research*, 56 (5) 982- 991.

[8] Do Carmo, M. P. (1976). *Differential Geometry of Curves and Surfaces*, p 2-30, 1976.

[9] Ojala, T., Pietikainen, M., Harwood, D. (1994). Performance Evaluation of Texture Measures with Classification Based on Kullback Discrimination of Distributions, *International Conference on Pattern Recognition*, p 582-585, 1994.

[10] Wang, L., He, D. (1990). Texture Classification Using Texture Spectrum, *Pattern Recognition*, 23 (8), p 905-910, 1990.

[11] Viola, P., Jones, M. (2001). Rapid Object Detection Using a Boosted Cascade of Simple Features, *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, p 511-518, 2001.

[12] Viola, P., Jones, M. (2004). Robust Real-time Face Detection, *International Journal of Computer Vision*, 57, *p*137-154, 2004.

[13] Hsu, C.W., Chang, C. C., Lin, C. J. (2016). *A Practical Guide to Support Vector Classification*, 2016.

[14] Cleveland, W. S., Devlin. S. J. (1988). Locally Weighted Regression: An Approach to Regression Analysis by Local Fitting, *Journal of the American Statistical Association*, vol. 83, p 596-610, 1988.

[15] http://www.cs.princeton.edu/courses/archive/spr07/cos424/papers /mitchell-dectrees.pdf