AUTOMATED DEVELOPMENT OF FEATURE EXTRACTION TOOLS FOR PLANETARY SCIENCE IMAGE DATASETS. C. Plesko[1], S. Brumby[2], E. Asphaug[3], [1]University of California, Santa Cruz, cplesko@es.ucsc.edu, [2]Los Alamos National Laboratory, brumby@lanl.gov, [3]University of California, Santa Cruz, asphaug@es.ucsc.edu.

Introduction: There is more image data available to planetary scientists now than at any time in history. The problem now is how to best make use of it. It is impractical to analyze such large datasets manually, while development of handwritten feature extraction tools is expensive and laborious. This project explores the use of machine learning techniques to develop feature extraction algorithms for the Mars Orbiter Camera (MOC) narrow angle dataset using the Los Alamos GENIE machine learning software. GENIE, the GENetic Image Exploitation package, uses a genetic algorithm to assemble feature extraction algorithms from low-level image operators. Each algorithm is evaluated against user-provided training data, and the most accurate ones are allowed to "reproduce" to build the next generation. The algorithm population evolves until it converges to a solution or specified level of accuracy.

Mars Global Surveyor (MGS) [1] has been on orbit around Mars since 1997. It carries many scientific instruments including the Mars Orbiter Camera (MOC) [2]. The narrow angle dataset, the focus of this study, provides imagery with a spatial resolution of order 3 meters/pixel and is sensitive to light in a broad visible/near-infrared spectral range ($0.50\mu m$ – $0.90\mu m$). Since arriving at Mars, MOC has taken over 112,000 images, which have been used to study various planetary processes. Craters were selected as the feature of interest for this study because they are a discrete, easily recognizable feature that can be used to derive much information about a surface [3-4].

GENIE: Los Alamos National Laboratory's GENIE software [5] uses techniques from genetic algorithms (GA) [6-8] and genetic programming (GP) [9] to construct spatio-spectral feature extraction algorithms for multi-spectral remotely sensed imagery. Both the algorithm structure and the parameters of the individual image processing steps are learned by the system. GENIE has been described at length elsewhere [10-12], so we will only present a brief description here. In particular, the present work explores using GENIE on panchromatic imagery [13-14].

GENIE follows the paradigm of genetic programming: a population of candidate image-processing algorithms is randomly generated from a collection of low-level image processing operators. The fitness of each individual is assessed from its performance on training data provided by the human user via a graphical interface. Our fitness metric measures the total error rate (false positives and false negatives) on the feature extraction task. After a fitness value has been assigned to each candidate algorithm, the most fit members of the population reproduce with modification via the evolutionary operators of mutation and crossover. This process of fitness evaluation and reproduction with modification is iterated until the population converges, some desired level of classification performance is attained, or some user-specified limit on computational effort is reached. The final result is a gray-scale enhancement of the feature of interest, which is then converted into a final Boolean classification using a threshold [15].

Several different algorithms are evolved separately in this manner. Their individual results are then combined to form a final answer in a "voting" process. GENIE's genetic algorithm is a stochastic learning process, so individual results are likely to be highly variable in structure. The binary classification outputs of these classifiers is then summed and thresholded at whatever level we choose to define the joint decision. In general, a simple majority rule is adopted, though if the classifiers are noisy, a unanimous rule decision may also be adopted.

GENIE Results: We selected a training image, MOC image M0803054 [16], near Louros Valles (8.5S, 82.0W), to present a reasonably homogeneous terrain marked by a number of bowl shaped craters obvious to the human eye. GENIE was trained on the first 930 rows of pixels in the image (the image is 830 pixels wide) with a truth file based on manual analysis (Fig. 1) in which the analyst has marked some of the fresh, bowl shaped craters in the scene as true, and some of the protruding surface features and non-cratered terrain as false. The next 970 rows of this scene were also marked by hand, and kept back to serve as our Test Scene (Fig. 2).

GENIE was run 6 times, each time with a new population of 30 algorithms per generation, each run lasting for 50 generations. Running on standard Intel/Linux workstations, each run required 1 hour of wall-clock time. The best individual crater finding algorithm achieved a detection rate of 99% and a false alarm rate of 3%. On the test data, the performance of this algorithm dropped slightly.

The results of the various algorithms were then combined in three different voting schemes (Table 1)

[17]. Classifier "Vote 1" gives the results achieved by a majority vote with contributions from all six classifiers. As expected, the false alarm rate reported by the voting set is substantially lower than the false alarm rate reported by any individual algorithm on either scene. Results of this vote raise the question, how much of the behavior of this classifier is due to the good individual algorithm? Classifier "Vote 2" gives the results achieved by a majority vote with contributions from all classifiers except the strongest individual algorithm. Classifier "Vote 3" shows the result of adopting a "unanimous" voting decision rule for the all-component "Vote 1" classifier. For application where achieving low false alarm rate is paramount, this is a superior classifier. The extent to which this low-noise classifier can be used as the initial stage of a geometric-based extraction of craters as objects, will be the subject of future work.

	Training Scene		Test Scene	
Class.	D.R.	F.A.R.	D.R.	F.A.R.
Best ind.	98.94	3.00	94.76	2.74
Vote 1	97.31	1.24	94.33	2.34
Vote 2	97.48	2.97	94.77	7.59
Vote 3	84.83	0.15	84.38	0.52

Table 1. Results of best individual algorithm and voting on training and test data. D.R. is percent detection rate, and F.A.R. is percent false alarm rate.

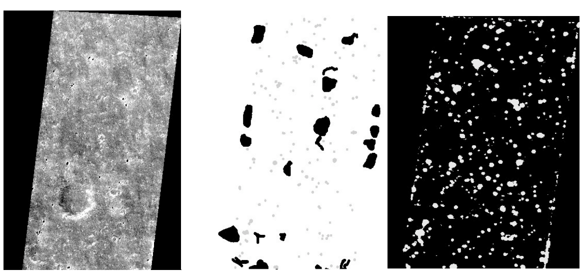

Figure 1. MOC Image M0803054: Training Scene, User-generated Training Data, result of first majority vote.

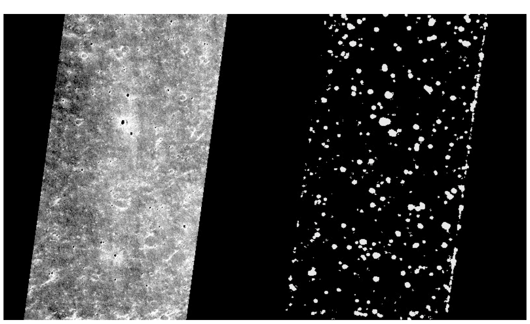

Figure 2. MOC Image M0803054: Test Scene 1, result of first majority vote.

Future Work: These results would be greatly improved by the ability to take parameters such as angle of illumination into account, and by the conversion of graphical detections into numerical results. We plan to experiment with a variety of post processing tools to this end. It would also be useful to compare GENIE results to manual results on comprehensive hand catalogued datasets.

This research presently focuses on Mars cratering as a testbed, but has obvious broader applications to other questions of planetary science, including analysis of cratering rates on satellites of Jupiter and Saturn, discernment of primary from secondary crater fluxes, recognition of apex-antapex impact asymmetries, and analysis of other landforms such as fractures in Europa's lithosphere, detection of changes over time on active planets such as Io, and location of landforms such as dune fields and mineral outcrops.

Conclusion: This study investigated the evolution of a voting set of crater finding algorithms for application to the Mars Orbiter Camera narrow angle dataset. We described the results on training and test images. The algorithms are successful at detecting craters within the images, and generalize well to an image that they have not seen before. We find these results to be encouraging for the application of GENIE to the MOC panchromatic dataset.

References: [1] Albee, A.L. et al. (1998) Science, 279, 13. [2] Malin, M.C., et al. (1992) JGR, 97(E5), 7699-7718. [3] Hartmann, W.K. et al. (1999) Nature, 397, 586-589. [4] Hartmann, W.K. et al. (2000) JGR, 105(E6), 15011-25. [5] Brumby, S.P. et al.(1999) Proc. SPIE, 3812, 24-31, (see http://www.daps.lanl.gov/genie). [6] Holland, J.H. (1975) Adaptation in Natural and Artificial Systems, U. Mich. [7] Rechenberg, I. (1973) Evolutionsstrategie: Optimierung technischer Systeme nach Prinzipien der biologischen Evolution, Fromman-Holzboog. [8] Fogel, L. et al. (1966) Artificial Intelligence through Simulated Evolution, Wiley. [9] Koza, J.R. (1992) Genetic Programming: On the Programming of Computers by Natural Selection, MIT. [10] Theiler, J. et al. (1999) Proc. SPIE, 3753, 416-425. [11] Harvey, N.R. et al. (2000) Proc. SPIE, 4132, 72-82. [12] Perkins, S. et al. (2000) Proc. SPIE, 4120, 52-62. [13] Plesko, C.S. et al. (2002) Proc. SPIE, 4480, 139-146. [14] Plesko, C.S. et al. (in press) Proc. SPIE, 4790. [15] For example, see Bishop, C.M (1995) Neural Networks for Pattern Recognition, Oxford Univ., 105 –112. [16] Malin, M.C. et al. (1999) http://photojournal.jpl.nasa.gov/, Catalog number M0803054, [17] R. O. Duda, R.O. et al. (2001) Pattern Classification, 2nd ed., Wiley-Interscience.