

Lunar Image Classification for Terrain Detection

Heng-Tze Cheng, Feng-Tso Sun, Senaka Buthpitiya, Ying Zhang, Ara V. Nefian*

Department of Electrical and Computer Engineering, Carnegie Mellon University

*Intelligent Robotics Group, NASA Ames Research Center

{hengtze.cheng, lucas.sun, senaka.buthpitiya, joy.zhang}@sv.cmu.edu,
ara.nefian@nasa.gov

Abstract. Terrain detection and classification are critical elements for NASA mission preparations and landing site selection. In this paper, we have investigated several image features and classifiers for lunar terrain classification. The proposed histogram of gradient orientation effectively discerns the characteristics of various terrain types. We further develop an open-source Lunar Image Labeling Toolkit to facilitate future research in planetary science. Experimental results show that the proposed system achieves 95% accuracy of classification evaluated on a dataset of 931 lunar image patches from NASA Apollo missions.

1 Introduction

Due to the rapid growth of image acquisition technology, large amount of planetary images from satellites are available for aerospace and planetary research. Understanding the semantic content in orbital images is therefore important for further analysis in planetary science. For example, terrain classification can facilitate landing site selection in robotic or manned mission on the Moon or Mars. In this work, we focus on lunar image classification in images collected by Apollo missions.

Lunar image classification is challenging because the shapes and colors of each type of terrain vary with locations, and the images are affected by dust and granular noise in the scanning process. Furthermore, the brightness and shadow caused by the angle of sunlight can significantly change the appearance of lunar surface. Inspired by previous work on face detection and object recognition [2]–[5], we combine both the techniques in image processing and the domain knowledge in planetary science to overcome the challenges.

In this paper, we present an automatic lunar image classification system with the following contributions. First, we have investigated multiple combinations of features and classifiers and compare their effectiveness in classifying terrains including craters, flat areas, and shadows. Second, we show that the proposed image feature—histogram of gradient orientation—is robust and effective for differentiating signatures of different terrains on the Moon. The system is one of the early attempts [6]–[8] to address the challenges in lunar terrain classification, and can be applied to images from the Moon, Mars, or other planets in space missions (e.g. LCROSS and HiRISE). Third, to facilitate future research in planetary science and imagery, we develop an

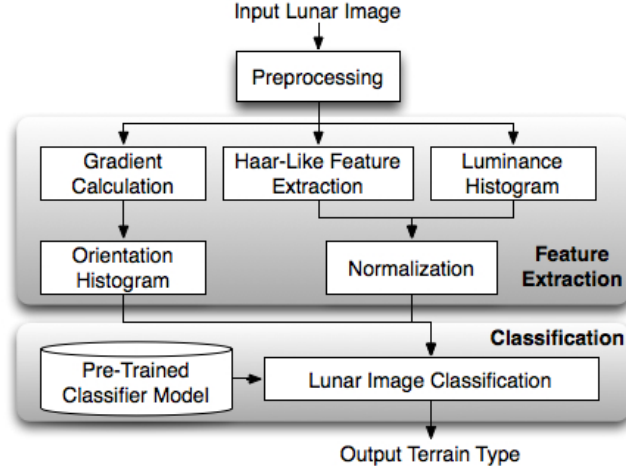


Fig. 1. System architecture of lunar image classification.

open-source Lunar Image Labeling Toolkit (LILT) with graphical user interface implemented in Java. We hope researchers can utilize the LILT toolkit to efficiently annotate large amount of planetary images for supervised learning.

The paper is organized as follows. We discuss related work in Section 2. In Section 3, we describe the proposed system architecture including feature extraction and classification algorithms. We present the lunar image labeling toolkit in Section 4, and show the experimental results in Section 5. We conclude the paper in Section 6.

2 Related Work

Our work is inspired by several previous attempts to crater or geological feature detection [6]–[8]. In [6], a feature template family is created by varying the resizing scale of continuously scalable detectors, and a testing image is classified as the most correlated template. The work has limitations when images are noisy or taken under varying sunlight angle. In [7], object-based analysis is used to extract textually or spectrally homogeneous regions. The method allows the identification of crater walls and crater floors separately; however, it requires high learning costs and is susceptible to noise. In [8], automatic crater detection is implemented using a boosting algorithm that select a small set of features characterizing the presence or absence of craters. The approach is validated by data sets of Mars surface captured by Mars Orbiter Camera.

There has also been extensive work on face, human, or object recognition [1]. In [3], rectangle features are used as a weak classifier initially, and then enhanced by incremental training using AdaBoost. A cascade filter is further used to discard the background of images rapidly and refine the detection results. In [4], object recognition is modeled as a machine translation problem. K-means is used for clustering

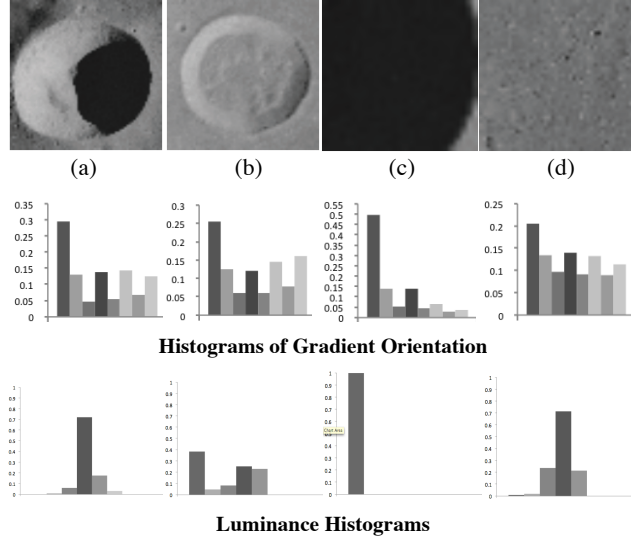


Fig. 2. Examples of lunar terrain types and the corresponding feature examples at the bottom: (a) crater with shadow (b) crater without shadow (c) shadow area (d) flat area.

followed by the EM algorithm for learning a lexicon of image words. Though some promising results are reported, it is mentioned that the approaches have certain limitations and does not work well on some objects. Locally normalized histogram of gradient orientations is adopted in [2] for person detection. The feature reduces false positive rates by an order of magnitude relative to the best Haar wavelet based detector from [5]. Inspired by these related works, we aim at lunar terrain classification and evaluate our system using various features and classifiers.

3 System Design and Methods

The system architecture is shown in Fig. 1. The goal of our system is to classify an input image patch to one of the four classes shown in Fig. 2. In this section, we describe each component of our approach in detail.

3.1 Preprocessing

First, all the images are converted to grayscale since lunar images mainly differ in luminance rather than color distributions. Then, for each input image, we apply Gaussian blurring filter to the entire image to reduce high-frequency noise and camera artifacts. Gaussian filtering is done by the convolution of each pixel in the input image with a Gaussian kernel and then summing to produce the output image. In experiments, we empirically use the 3-by-3 filter window as the Gaussian kernel.

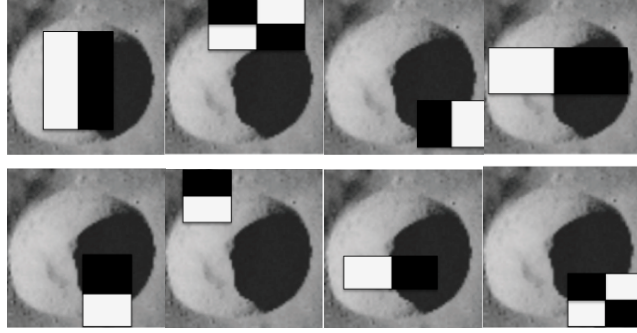


Fig. 3. The eight Haar-like masks used for feature extraction.

3.2 Image Feature Extraction

Histogram of Gradient Orientation. Histogram of gradient orientation has been used in the field of human detection [2]. While gradient is useful for edge detection, using the magnitude of gradient as a feature is less robust due to the variability of luminance contrast. In contrast with magnitude, the orientation of gradient is of lower granularity and thus less susceptible to high-frequency noises. The computation of gradient orientation can be formulated as:

$$\mathbf{v}^* = \underset{\mathbf{v}}{\operatorname{argmax}} |L(n(p_i, \mathbf{v})) - L(p_i)| \quad (1)$$

where $\mathbf{v} \in \{(i, j) \mid -1 \leq i \leq 1, -1 \leq j \leq 1, i, j \in \mathbb{Z}\}$ is the orientation of the gradient, which can be one of the following vectors: $(1,0), (1,1), (0,1), (-1,1), (-1,0), (-1,-1), (0,-1), (1,-1)$. p_i denotes the coordinate of a pixel, and $n(p_i, \mathbf{v})$ denotes the neighboring pixel given by $(p_i + \mathbf{v})$. $L(p_i)$ is the luminance of the pixel p_i . According to eq. (1), for each pixel p_i in the input image patch, we compute the difference of luminance between the pixel p_i and each of its eight neighboring pixels, $n(p_i, \mathbf{v})$. The direction with the maximum difference is the gradient orientation. After computing the orientation of gradient at each pixel, we sum up the occurrence of each orientation and form a statistical histogram. Since there are eight possible directions, the resulting histogram is an 8-dimensional vector as shown in Figure 2.

The intuition behind the use of this feature is inspired by the characteristics of lunar terrains. While craters are generally of circular shape, flat areas have no specific shape or edges, and shadow areas can have obvious edges from darkness to the area with sunlight. These characteristics can be distinguished by the gradient orientation histograms of image patches.

Haar-like Features. Haar-like features are widely used in object and face detection. The features are computed by probing each image patch P with a set of 8 different masks shown in Figure 3. In these masks, the black and white areas represent -1 and +1, respectively. When we apply a Haar-like feature mask on a specific image patch P , the computation of a single normalized feature value can be formulated as:

$$f(P) = \int_{[0,1]^2} P(x)\Omega(x)dx \quad (2)$$

where $\Omega(x)$ denotes one of the Haar-like masks. The image patch P and the mask are images defined in the interval $[0,1]^2$. The computation of these features is intensive but can be accelerated if the integral image is pre-computed. In our implementation, we applied the 8 distinct Haar-like feature masks and aggregated eight values into an 8-dimensional feature vector.

Luminance Histogram. The luminance histogram represents the luminance distribution of the image. The raw luminance histogram is of 256 bins. To reduce the dimension, we cluster every 32 successive luminance levels into one bin and obtain an 8-dimensional feature vector as shown in Figure 2.

3.3 Lunar Image Classification

K-Nearest Neighbor Classifier. In our work, we adopt k -NN as one of our classification algorithms. Specifically, for each testing image patch, we calculate the distance from each of the image patches in the training dataset. The distance is obtained by computing the L2-distance (i.e. Euclidean distance) between two feature vectors.

Decision Tree Classifier. In addition to k -NN classifier, we also adopt the decision tree classifier. Specifically, a decision tree is tiered into three layers. The topmost tier's decision tree decides if a particular patch belongs to the most common class in the training data. If positive, the classification is finished. Otherwise, the decision tree at the next tier is used to classify the patch as belonging to the second most common class in the training data or not. This process is repeated until a decision tree returns a positive result or all decision trees are exhausted. In cases where all decision trees are exhausted without obtaining a positive classification from any of the decision trees, the patch is assigned a default classification. The default classification in this case is the most common class in the training data.

4 Lunar Image Labeling Tool

Training image classifiers requires large amount of training data. Since there is no labeled lunar image database available, we decided to develop the Lunar Image Labeling Toolkit (LILT) that allow us to annotate terrain types and facilitate future research in the related field. The LILT toolkit is open-source and is available on our website¹.

¹ <http://www.ece.cmu.edu/~hengtzec/projects/lunar/LILT.zip>

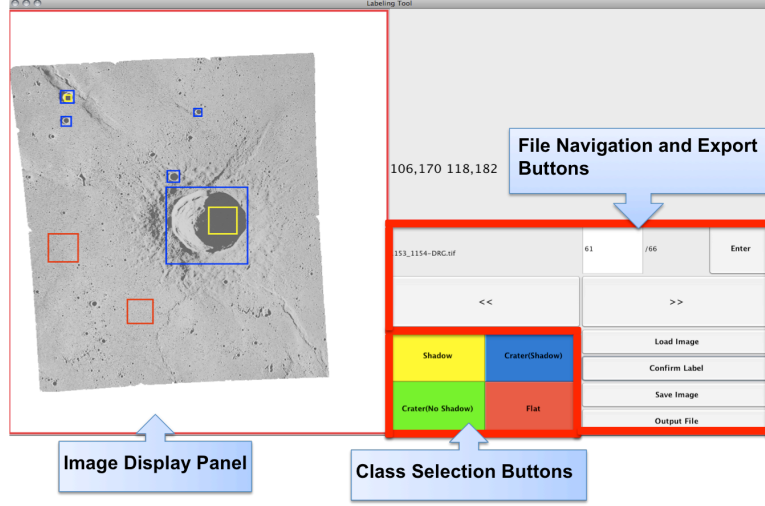


Fig. 4. User interface of the Lunar Image Labeling Toolkit.

The graphical user interface for terrain type labeling is developed in Java Swing. Figure 4 shows the user interface that consists of a file navigation section, a class labeling section, and a display panel. A user can intuitively choose a class, drag the desired zone, or undo operations. We use the LILT toolkit to label 931 lunar image patches of interest from 66 raw images. The toolkit supports various input image formats such as PNG, TIFF, or JPEG, and outputs a ground-truth file containing file name, patch coordinates, and the terrain type of the image patch.

5 Experimental Results

We evaluate the system on a dataset that consists of 931 lunar image patches from NASA Apollo missions.

Table 1. Accuracy with different feature and classifier combinations.

Method	Decision Tree	k -Nearest Neighbor
Luminance Histogram	59.47%	56.76%
Haar-like Feature	64.45%	68.47%
Gradient Orientation	85.73%	95.56%

5.1 Evaluation on Feature and Classifier Combinations

First, we evaluate the performance using different combinations of image features and classification algorithms to find which ones are more effective. As shown in Table 1, luminance histogram does not work well in general because the luminance

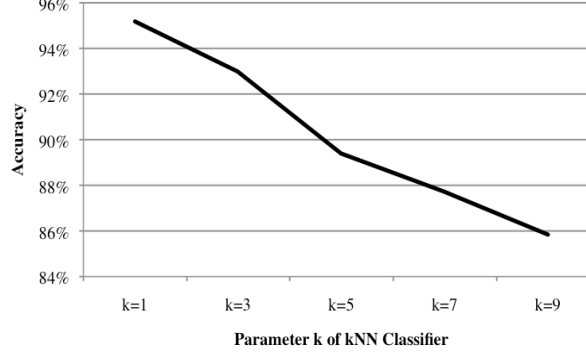


Fig. 5. Accuracy with different k in k -NN classifier.

distribution is not necessarily consistent across images of the same terrain type. One terrain type can be dark when the sunlight is blocked and bright otherwise. Haar-like features roughly capture the edges or texture changes and slightly improve the accuracy to around 65%, but fail to recognize terrains with highly varying sizes since the set of masks is fixed. Among the three features we have tested, histogram of gradient orientation yields the best performance, which is 85.73% using the decision tree, and 95.56% using the k -nearest neighbor classifier. The results show that the histogram of gradient orientation effectively learns the signature of shapes and edge directions of different terrain types. Furthermore, since only the distribution of gradient orientation (rather than magnitude) is computed, the feature is more robust and scale-invariant.

As for the comparison of classification algorithms, from Table I we can see that k -NN generally performs better than decision trees. When using histogram of gradient orientation as features, k -NN improves the overall accuracy by approximately 10%. Therefore, we choose gradient orientation histogram and k -NN as our proposed method for subsequent experiments.

Table 2. Classification accuracy using different percentage of training data.

Percentage of Training Data	10%	20%	30%	50%	90%
Accuracy	93.58	94.26	94.81	95.03	95.51

5.2 Cross Validation

We further conduct several experiments to test the best-performing system, which consists of gradient orientation histogram as feature and k -NN classifier, under different parameter settings. As shown in Figure 5, the k -NN classifier performs best when $k=1$, and the accuracy decreases as k increases. One reason is that some craters without shadows are of subtle edges and are more likely to be misclassified as flat areas; therefore, increasing k brings more potentially misleading patch candidates. As a result, we empirically set k to 1 for subsequent evaluations. Then, we test the performance under the condition that only a small amount of training data is available.

Table 3. Classification accuracy measured by cross-validation.

	10-fold Cross Validation	Leave-one-out Cross Validation
Accuracy	93.67	95.51

As shown in Table II, even using only 10% of training data (approximately 90 image patches), we still achieve 93.58% of accuracy. This shows the robustness of the proposed method. We also performed 10-fold and leave-one-out cross validation with the results shown in Table III. The overall accuracy is 93.67% and 95.7%, respectively.

6 Conclusions and Future Work

In this paper, we have investigated image feature and classifiers for lunar terrain classification. Evaluated on a large dataset of 931 lunar image patches from Apollo missions, the proposed system using histogram of gradient orientation and k -NN classifier achieved a high accuracy of 95.56%. We have also shown that the proposed approach outperforms Haar-like features and luminance histogram by around 30%. The system, along with the open-source Lunar Image Labeling Toolkit, can be applied to larger orbital image mining package for imagery from NASA space missions such as LCROSS and HiRISE.

For future work, we plan to focus on automatic terrain patch detection using scale-invariant image descriptors. Furthermore, we plan to extend the system architecture for recognizing a wider category of terrain types on the moon and other planets. We will also test the performance using multiple feature fusion and other generative or discriminative classifiers.

References

1. R. Datta, D. Joshi, J. Li, J. Z. Wang, "Image Retrieval: Ideas, Influences, and Trends of the New Age," *ACM Computing Surveys*, Vol. 40, No. 2, pp. 1–60, 2008.
2. N. Dalal and B. Triggs, "Histograms of Oriented Gradients for Human Detection," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp. 181–184, 2005.
3. P. Viola, M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp. 511–518, 2001.
4. P. Duygulu, K. Barnard, N. d. Freitas, P. Duygulu, K. Barnard, D. Forsyth, "Object Recognition as Machine Translation: Learning a Lexicon for a Fixed Image Vocabulary," in *Proc. European Conf. Computer Vision*, pp. 349–354, 2002.
5. A. Mohan, C. Papageorgiou, T. Poggio, "Example-based object detection in images by components," in *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 23, Issue 4, pp. 349–361, 2001.
6. M. C. Burl, W. J. Merline, E. B. Bierhaus, W. Colwell, C. R. Chapman, "Automated Detection of Craters and Other Geological Features," in *Proc. International Symp. Artificial Intelligence Robotics and Automation in Space*, 2001.
7. T. F. Stepinski, S. Ghosh, R. Vilalta, "Automatic Recognition of Landforms on Mars Using Terrain Segmentation and Classification," in *Proc. International Conf. Discovery Science*,

pp. 255–266, 2006.

8. R. Martins, P. Pina, J. Marques, M. Silveira, “Crater Detection by a Boosting Approach,” in *IEEE Geoscience and Remote Sensing Letters*, Vol. 6, No.1, pp. 127–131, Jan. 2009.