On Crater Verification Using Mislocalized Crater Regions

Ebrahim Emami, George Bebis
Department of Computer Science and Engineering
University of Nevada, Reno, USA
ebrahim@nevada.unr.edu, bebis@cse.unr.edu

Ara Nefian, Terry Fong
NASA Ames Research Center, USA
ara.nefian@nasa.gov, terry.fong@nasa.gov

Abstract

Automatic crater detection in planetary images is an important task with many applications in planetary science, spacecraft navigation, landing, and control. Typically, crater detection algorithms consist of two main steps: candidate crater region extraction and crater verification. Various methods have been proposed for extracting candidate crater regions, ranging from detecting circular/elliptical regions to detecting highlight and shadow regions. For crater verification, powerful feature extraction and machine learning techniques have been employed. While this two-step approach can be efficient and robust, inaccuracies in the candidate crater region extraction step can result in mislocalized crater regions which could affect verification performance. In this paper, we investigate the robustness of various feature extraction methods to mislocalized crater regions. Using features which are robust to localization errors but also choosing a more representative training set has yielded significant performance improvements on an extensive dataset from the Lunar Reconnaissance Orbiter (LRO).

1. Introduction

Craters are landmarks on planetary surfaces formed by the collisions of meteoroids. Craters contain important information about planets. For instance, the relative chronology of different planetary surfaces can be estimated by crater counting [1]. Craters can also be used as landmarks for spacecraft autonomous navigation and safe landing [2] [3]. Current crater catalogues are manually gathered and are usually small or specific to certain geographical areas. The increase of image data obtained from planetary surfaces requires more sophisticated crater detection approaches [4] [5]. Obviously, manual analysis of such amount of data is laborious, and automatic crater detection can be a practical solution for this problem [1].

A typical crater detection algorithm consists of two main steps: candidate crater region extraction and crater verification. In the first step, candidate crater regions are extracted using fast algorithms to reduce the load of the verification step. In the second step, machine learning techniques are typically used to solve a two-class pattern classification problem (i.e., crater vs non-crater) [6].

Automatic crater detection is a challenging problem due to variations in illumination, crater size, and planet surface features [1]. Another challenge comes from the fact that candidate crater regions are typically poorly localized which could increase the number of false positives and negatives during verification. One way to address this issue is by using features which are robust to localization errors. In this paper, we have performed a comprehensive comparison of state-of-the-art feature descriptors for crater verification using an extensive dataset and Support Vector Machines (SVMs) for classification. It should be emphasized that the focus of this paper is not on the details of feature descriptors or the classification algorithms. However, state-of-the-art feature descriptors along with simple crater detection/verification algorithms are used to test the performance of various feature descriptors in presence of localization errors.

To our knowledge, this is the first study comparing state-of-the-art descriptors for crater verification using an extensive dataset. The results of this study clarifies which type of feature descriptors and training data should be used alongside to obtain the best crater detection performance in presence of localization errors which crater detection algorithms suffer from. Our experimental results demonstrate that certain descriptors are significantly more robust than others when dealing with localization errors. Additional performance improvements can be obtained by augmenting the training set with mislocalized crater examples, generated in a systematic way from well localized crater examples.

The rest of this paper is organized as follows: Section 2 provides a literature review of crater detection approaches. In section 3, we briefly review the different types of features used our comparison. In section 4, the performance of different feature extraction techniques is discussed and analyzed. Finally, Section 5 contains our conclusions and directions of future work.

2. Background

Previous methods on crater detection can be classified into two main categories: supervised and unsupervised methods [6]. Unsupervised approaches mainly rely on image processing and pattern recognition techniques such as thresholding or circle and ellipse fitting [7] [8]. Supervised approaches discriminate between crater and non-crater regions using machine learning. These methods typically rely on a candidate region extraction step which feeds the candidate crater regions to a verifier. Supervised methods are usually more robust but dependent on the quality and number of training data [6]. In the remaining of this section, we review previous approaches for candidate region detection. Then, we review crater detection approaches with an emphasis on feature extraction techniques.

A simple sliding window approach was used in [9] to provide candidate crater regions for verification. Kim et al. [10] extract a set of candidate regions using edge detection and ellipse fitting. Ding et al. [11] extract candidate craters using a Kanade–Lucas–Tomasi (KLT) based detector. Urbach et al. [12] employed a method based on the simultaneous detection of the highlight and shadow parts of craters. Emami et al. [6] used convex grouping to extract possible candidate regions.

The simplest type of feature descriptor employed for crater detection is raw pixel intensity. Wetzler et al. [9] studied the performance of various classification techniques for small size crater detection. In their approach, each classifier was trained using pixel intensities and applied on test images using a sliding window approach. In a related work, Palafox et al. [13] compared the performance of SVMs and Convolutional Neural Networks (CNN), trained on pixel intensities, for detecting craters and volcanic rootless cones.

Ubrach et al. [12] used shape-based features to verify crater regions. Their feature vectors consisted of Hu`s seven moments invariants as well as other variables related to the shape and position of crater`s shadow and highlight regions.

Haar-like features have been very popular for crater detection. In [14], Martins el al. used Haar-like features from each image block and classified candidate regions using Adaboost [15]. In [1], Stepinski et. al. propose a method based on Haar-like features and Adaboost to detect sub-kilometer craters in High Resolution Stereoscopic Camera (HRSC) images. In [3], Liu et al. used Haar-like features with a fast classifier to extract candidate regions. The candidate regions were then verified using SVMs and a variant of Histogram of Oriented Gradient (HoG) features. Jin et. al. [16] also used a modified version of Adaboost with Haar-like features for crater detection.

3. Feature extraction and classification

We have considered two groups of features descriptors for evaluating the performance of crater verification assuming mislocalized crater regions. The first group consists of more robust but slower to compute descriptors including Haar [15], Local Binary Pattern (LBP) [18], Local Phase Quantization (LPQ) [19], and HoG [20] features. These features have been extensively studied in the computer vision community for various object detection problems. The second group consists of binary descriptors including Binary Robust Independent Elementary Features (BRIEF) [21], Oriented FAST and Rotated BRIEF (ORB) [22], Binary Robust Invariant Scalable Key points (BRISK) [23], and Fast Retina Keypoints (FREAK) [24] features. These descriptors were originally proposed for fast feature extraction assuming limited computational resources.

For classification, we decided to use SVMs although any other classifier could have been used. Since our goal was compare the robustness of different feature descriptors, the choice of the classifier is not as critical as far as a reasonable classifier is used. Given a set of examples from two classes, SVMs find the hyper-plane which keeps the largest possible portion of samples of the same class on the same side, while maximizing the distance of either class from the hyper-plane.

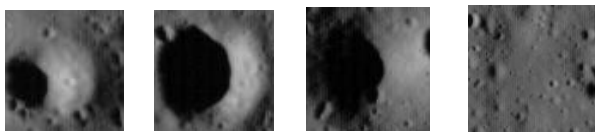
4. Experiments and results

4.1. Dataset

We used 578 planetary images in our experiments, each having a size of 600×400, captured by the Lunar Reconnaissance Orbiter (LRO) [25]. A random set of 428 of these images were used for training while the rest 150 images were used for testing. NASA scientists have partially labeled craters with size between 20x20 and 200x200 in all images. To study the effect of crater mislocalization, four training sets were generated from the training images. For each dataset, the positive samples (i.e., craters) were generated by randomly changing the location and size of the box enclosing the ground truth craters. The negative examples (i.e., non-craters) were selected randomly; the same negative examples were included in all datasets. The final training sets were as follows:

- *Training set 1:* Positive samples have between 90% and 100% overlap with ground truth craters.
- *Training set 2:* Positive samples have between 70% and 90% overlap with ground truth craters.
- *Training set 3:* Positive samples have between 50% and 70% overlap with ground truth craters.
- *Training set 4:* Positive samples have between 50% and 100% overlap with ground truth craters (referred as “mixed” in our experiments).

Each training set had 7320 positive samples, and an equal number of negative samples. All samples were normalized



(a) (b) (c) (d)

Figure 1: Sample training examples: (a) positive sample with 50%-70% overlap with ground truth, (b) positive sample with 70%-90% overlap with ground truth, (c) positive sample with more than 90% overlap with ground truth, (d) negative sample

to size 24×24. Figure 1 shows several training sample examples. The same procedure was used to generate four different test sets from the 150 test images. Each test set had 3542 positive examples and an equal number of negative samples.

4.2. Performance evaluation measures

Recall and precision rates were used to compare the performance of different methods as follows:

$$Recall = TP/(TP + FN), Precision = TP/(TP + FP) \quad (1)$$

where TP, FP, and FN are the number of true positives, false positives, false negatives respectively.

4.3. Performance comparisons on randomly generated test sets

The goal of this experiments is to evaluate the effect of localization errors on verification step without any assumption about a particular candidate crater region detection method. Clearly different candidate region detection methods produce different amount of localization error. While the study of candidate crater region detection methods is out of the scope of this paper, experiments on the four provided training sets, show how different feature descriptors work in presence of different amounts of localization error which can result from various candidate crater detection methods. In the next section, it is shown how this results can be extended to a specific candidate crater region detection method.

Using the feature extraction methods mentioned in section 3, several SVM classifiers were trained using the datasets mentioned in the previous section. For comparison purposes, we have also experimented with raw pixel intensity values. To test the robustness of different feature descriptors to crater localization errors, we performed several experiments assuming different amounts of overlap between craters in the training and test sets. For example, one of our experiments involved training using craters with over 90% overlap with ground truth in the training set and craters with 50%-70% overlap with ground truth in the test set.

Tables 1 and 2 show our experimental results using the first and second group of feature descriptors respectively. Several interesting observations can be made by examining these results. First of all, each classifier performs reasonably well when craters have been localized well both in the training and test sets (i.e., over 90% overlap with ground truth) with the best performance obtained using HOG features from the first group and FREAK features from second group. Given mislocalizations in the test sets, the performance of all classifiers degrades but not at the same rate for all of them. For example, LBP, Haar, and ORB features seem to degrade more gracefully compared to the other features. This demonstrates that certain features are more robust to localization errors compared to others. The best performance was obtained using ORB features (90.93% recall rate and 95.29% precision rate) assuming over 90% overlap with ground truth in the training set and 50%-70% overlap with ground truth in the test set. However, this is still lower than the best performance obtained using HOG features and well localized craters (99.04% recall rate and 99.32% precision rate).

One way to handle localization errors is by adding mislocalized craters in the training set. Therefore, we have performed additional experiments assuming 70%-90% and 50%-70% overlap with ground truth in the training set. Increasing localization error in the training set slightly increases classification error when mislocalization in the test set is small (i.e., over 90% overlap with ground truth); however, we have noticed significant performance improvements in most cases when mislocalization is also high in the test set which is the case in practice. Assuming 50%-70% overlap with ground truth both in the training and test sets, the best performance was achieved using HOG features (96.6% recall rate and 97.2% precision rate) from the first group and FREAK features (94.6% recall rate and 93.5% precision rate) from the second group. These rates compare favorably to the rates obtained assuming no localization error in the training set (i.e., 80% recall rate and 99.2% precision rate for HOG features and 75.5% recall rate and 97.9% precision rate for FREAK features).

Next, we performed a more realistic experiment assuming that some craters are well localized while others are poorly localized. We refer to these sets as “mixed” and contain craters with 50%-100% overlap with the ground truth. Figure 2 shows a side by side comparison of the different features tested in this case. The best performance was obtained using HOG features (97.8% recall rate and 98.3% precision rate) followed by that of FREAK features (95.6% recall rate and 95.5% precision rate). These rates again compare favorably with the rates obtained assuming no localization errors in the training and test sets or localization error only in the test set.

Overall, HOG features performed the best. FREAK features performed quite well; given their computational advantage, they could be viable alternative to HOG features

TABLE 1. PERFORMANCE OF VARIOUS FEATURE DESCRIPTORS ON MISLOCALIZED TRAIN/TEST DATA

	Train Data Overlap	Test Data overlap							
		50-70		70-90		90		Mixed	
		Recall	Precision	Recall	Precision	Recall	Precision	Recall	Precision
Raw pixels	50-70	0.7724	0.7341	0.8992	0.7627	0.9277	0.7683	0.8633	0.7552
	70-90	0.7871	0.9123	0.9322	0.9249	0.9489	0.9262	0.8890	0.9215
	90	0.8131	0.9584	0.9562	0.9644	0.9729	0.9650	0.9130	0.9130
	mixed	0.8532	0.9149	0.9653	0.9241	0.9800	0.9251	0.9322	0.9216
LBP	50-70	0.9034	0.9138	0.9433	0.9171	0.9543	0.9180	0.9362	0.9165
	70-90	0.8761	0.9372	0.9294	0.9406	0.9447	0.9415	0.9164	0.9398
	90	0.8642	0.9390	0.9229	0.9426	0.9393	0.9436	0.9096	0.9418
	mixed	0.8848	0.933	0.9373	0.9368	0.9526	0.9377	0.9249	0.9360
HOG	50-70	0.9661	0.9719	0.9966	0.9727	0.9989	0.9728	0.9879	0.9725
	70-90	0.8823	0.9896	0.9890	0.9907	0.9907	0.9963	0.9568	0.9904
	90	0.8058	0.9917	0.9805	0.9931	0.9904	0.9932	0.9252	0.9927
	mixed	0.9416	0.9828	0.9949	0.9835	0.9975	0.9836	0.9783	0.9833
LPQ	50-70	0.9060	0.9736	0.9489	0.9748	0.9543	0.9749	0.9359	0.9744
	70-90	0.8540	0.9815	0.9382	0.9831	0.9447	0.9833	0.9108	0.9826
	90	0.8286	0.9849	0.9314	0.9865	0.9424	0.9867	0.9003	0.9861
	mixed	0.8803	0.9805	0.9438	0.9818	0.9478	0.9819	0.9235	0.9814
HAAR	50-70	0.9859	0.5072	0.9966	0.5099	0.9977	0.5102	0.9941	0.5093
	70-90	0.1200	0.8551	0.1609	0.8879	0.1717	0.8941	0.1497	0.8804
	90	0.9255	0.7899	0.9853	0.8001	0.9893	0.8007	0.9655	0.7968
	mixed	0.3235	0.8389	0.4545	0.8798	0.4752	0.8844	0.4230	0.8719

TABLE 2. PERFORMANCE OF VARIOUS BINARY FEATURE DESCRIPTORS ON MISLOCALIZED TRAIN/TEST DATA

	Train Data Overlap	Test Data overlap							
		50-70		70-90		90		mixed	
		Recall	Precision	Recall	Precision	Recall	Precision	Recall	Precision
BRIEF	50-70	0.8232	0.7741	0.9067	0.7906	0.9194	0.7929	0.8864	0.7871
	70-90	0.7415	0.9252	0.9449	0.9403	0.9641	0.9414	0.8799	0.9366
	90	0.6505	0.9615	0.9305	0.9727	0.9646	0.9737	0.8449	0.9705
	mixed	0.8047	0.8392	0.9287	0.8576	0.9522	0.8607	0.8918	0.8539
FREAK	50-70	0.9458	0.9352	0.9718	0.9368	0.9783	0.9372	0.9636	0.9347
	70-90	0.8704	0.9713	0.9658	0.9741	0.9788	0.9744	0.9390	0.9734
	90	0.7546	0.9798	0.9509	0.9839	0.9715	0.9843	0.8873	0.9828
	mixed	0.9088	0.9524	0.9675	0.9551	0.9845	0.9559	0.9565	0.9546
ORB	50-70	0.8111	0.7787	0.6557	0.7400	0.8433	0.7854	0.7669	0.7689
	70-90	0.9266	0.9074	0.7300	0.8853	0.9438	0.9089	0.8650	0.9014
	90	90.93	95.29	68.34	93.83	93.84	95.43	0.8401	0.9492
	mixed	0.9054	0.8486	0.7698	0.8266	0.9274	0.8517	0.8681	0.8431
BRISK	50-70	0.3677	0.9307	0.4448	0.9420	0.4716	0.9451	0.4186	0.9386
	70-90	0.7775	0.8812	0.9526	0.9009	0.9701	0.9025	0.8952	0.8952
	90	0.2177	0.9948	0.5289	0.9979	0.6357	0.9982	0.4477	0.9975
	mixed	0.8099	0.4525	0.9130	0.4823	0.9291	0.4867	0.8884	0.4755

if time is an issue. Since it is difficult to ensure low localization error in practice, including mislocalized craters in the training set offers significant performance improvements. In general, we have noticed performance improvements when the localization errors in the training and test sets are comparable. For example, using the mixed set for training in the case of HOG features yielded a lower recall rate (i.e., 94.2% compared to 96.6%) when the localization error in the test set was higher (i.e., 50%-70% overlap with ground truth).

4.4. Performance comparisons using candidate crater region extraction

In our next set of experiments, we considered a more realistic scenario where the test data were extracted using a

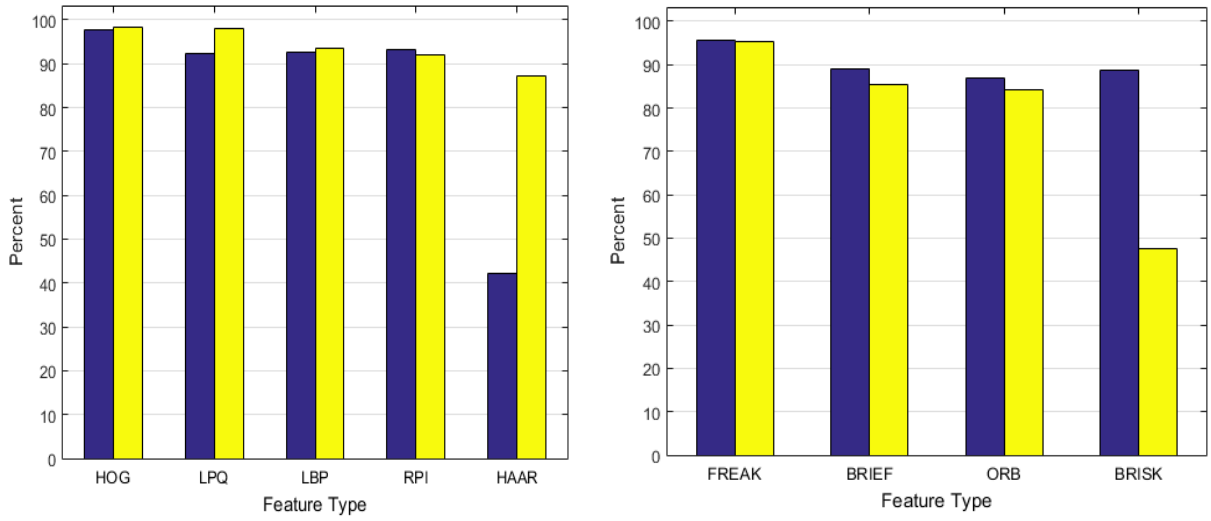


Figure 2: Recall (blue bar) and Precision (yellow bar) rates using the mixed dataset for training and test

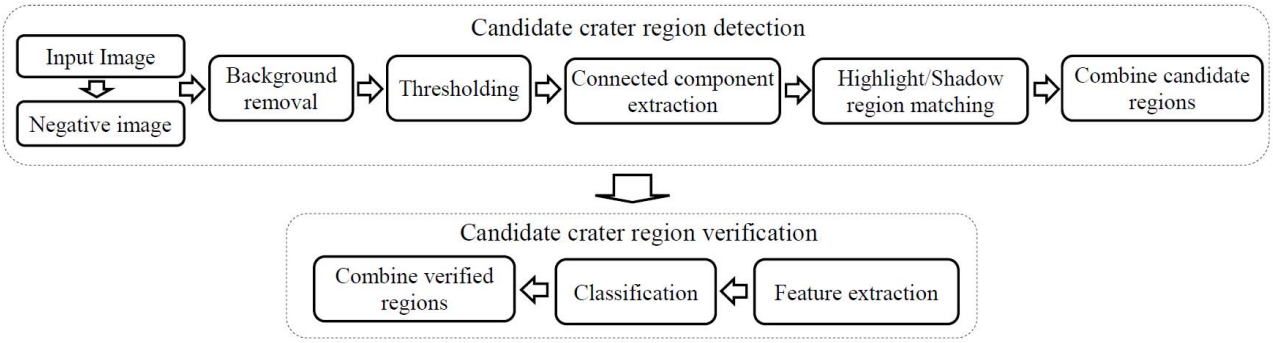


Figure 3: Main step automatic crater detection

candidate crater region detection algorithm. For this, we randomly select 20 LRO images. All craters greater than 20×20 are manually labeled in these images. Specifically, we implemented a simple method for extracting candidate crater regions using highlight and shadow regions inspired by the method proposed in [12]. It should be emphasized that our goal was not to implement a state-of-the-art candidate crater region detection algorithm but rather to test our classifiers using more realistically generated data. Specifically, our candidate crater detection algorithm works as follows:

1. *Obtain the negative image of the input.*

Apply steps 2-5 on the original and negative images:

2. *Apply median filter on the image.*
3. *Remove large background components, by subtracting the image from the median filter smoothed image.*
4. *Apply image thresholding.*
5. *Extract connected components in the image.*
6. *Match regions obtained from the original image with those obtained from the negative image.*
7. *Approximate matched regions by a square region.*
8. *Cluster candidate regions based on their overlap.*

In step 6, each highlight region is compared to the shadow regions and is matched with one of them if the distance between the regions` centroids is less than a threshold (relative to the regions' sizes). The candidate region is then defined by a box enclosing the two matched regions. In step 7, the regions are made square by enlarging the shorter side of the candidate region. Finally in step 8, the candidate regions are clustered in two steps. (1) Cluster the regions based on their overlap (2) Represent each cluster by averaging the bounding boxes. Two candidate regions represented by bounding boxes *box1*, and *box2* are clustered if their overlap defined as follows, is greater than a fixed threshold.

$$\frac{Area(box1 \cap box2)}{Area(box1 \cup box2)} \quad (2)$$

Figure 3 shows the main steps of our two step crater detection approach.

Table 3 provides a summary of our candidate crater region detections. While the candidate crater detection method can detect most of the craters (96.52% in average), the detected crater regions are not well localized (55.92% overlap with ground truth in average).

TABLE 3. CANDIDATE REGION DETECTION PERFORMANCE

Number of craters	230
Percent of detected true craters	96.52
Average number of detections corresposding to each ground truth crater	4.59 (std:5.15)
Average overlap between ground truth craters and their corresponding candidate regions	55.92 (std:12.13)
Average number of detected candidate regions in an image	620 (std:382)

The detected candidate regions were fed to the classifiers for verification. All the classifiers were trained using the mixed training set. Table 4 shows the verification performance of the different classifiers. In terms of recall rates, the HOG and LPQ classifiers performed the best from the first group while the FREAK and BRIEF classifiers performed the best from the second group. However, both the recall and precision rates were lower than those obtained in our previous experiments. This could be attributed to several reasons. First, almost 3.5% of ground truth craters were missed by our candidate crater region detection algorithm (i.e., not passed to the verifier). Also, the average overlap between candidate crater regions and ground truth is about 56%. In other words, the localization error in the test images is higher than that in the training set. As noted in the previous section, differences in the localization error between the training and test sets can degrade classification performance. Most importantly, for this experiment, the training samples were extracted randomly while the test samples are obtained from the candidate crater region extraction algorithm. We believe that this is the main reason for the low precision rates obtained in this experiment.

To test this hypothesis, a training set is obtained by extracting candidate regions from the train images. To increase the number and variation of training samples, the extracted candidate region locations are also slightly moved (the regions can slightly move, shrink or expand by small random amounts proportional to their size). Bootstrapping [26] is also used to improve the performance of the HOG classifier. First, the HOG classifier was trained using the same number of samples as before from the candidate crater region detection method. Next, candidate crater regions from the training images are classified and a percentage of false positives are added to the training set. Table 5 shows performance improvements after two iterations of bootstrapping. Figure 4 shows the detection results for a sample test image before and after bootstrapping. Our candidate crater region detection is first applied on this sample image. Candidate regions are then verified by the HOG classifier. The figure shows how the bootstrapping process improves the performance of the classifier in terms of reducing the number of false positives (red boxes).

TABLE 4. CANDIDATE REGION VERIFICATION PERFORMANCE

Classifier	Recall	Precision
HOG	88.08	44.8
LPQ	82.12	51.6
FREAK	83.13	33.86
BRIEF	84.31	31.99
LBP	78.72	30.62
BRISK	67.05	15.51
HAAR	29.36	35.35

TABLE 5. PERFORMANCE IMPROVEMENTS USING THE HOG CLASSIFIER AND BOOTSTRAPPING

Number of samples added	Recall	Precision
-	91.45%	60.00%
approximately 1400	89.59%	75.31%
approximately 800	89.59%	81.14%

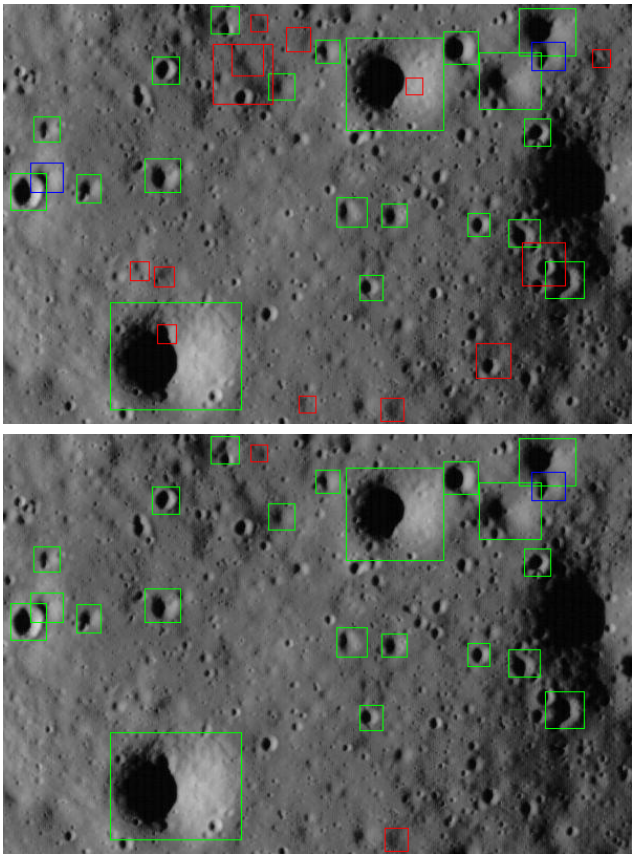


Figure 4: Crater detection results obtained using the HOG classifier before (top) and after bootstrapping (bottom). True positives are shown in green, while red and blue boxes show false positives and false negatives respectively. (Craters smaller than 20×20 are not considered in the detection rates)

5. Conclusions

In this paper, we investigated the problem of crater verification using mislocalized candidate crater regions. Our emphasis was on testing several state-of-the-art feature descriptors using the SVM classifier. To further improve verification performance, we proposed including mislocalized crater examples for training. Our experimental

results show that using HOG features while at the same time including mislocalized crater examples in the training set improves verification performance significantly. The proposed approach is not limited to the problem of crater detection and may be used to evaluate and improve the performance of various types of feature descriptors for other object classification problems.

Acknowledgements

This material is based upon work supported by NASA EPSCoR under cooperative agreement No. NNX11AM09A

References

[1] T. Stepinski, W. Ding, R. Vilalta, "Detecting Impact Craters in Planetary Images Using Machine Learning", Intelligent Data Analysis for Real-Life Applications: Theory and Practice, IGI Global, pp. 146-159, 2012.

[2] Z. Yu, , S. Zhu, P. Cui, "Sequence Detection of Planetary Surface Craters From DEM Data", World Congress on Intelligent Control and Automation, 2012.

[3] A. Maoyin, W. Pan, "Crater Detection Algorithm With Part PHOG Features For Safe Landing, International Conference on Systems and Informatics", pp 103-106. 2012.

[4] L. Bandeiraa, W. Ding, F. Tomasz,"Detection of Sub-kilometer Craters in High Resolution Planetary Images Using Shape and Texture Features", Advances in Space Research, Vol. 49. Issue 1. pp 64–74, 2012.

[5] G. Salamunićcara, S. Lončarić, "Open Framework For Objective Evaluation of Crater Detection Algorithms With First Test-field Subsystem Based on MOLA Data", Advances in Space Research, Vol. 42. Issue 1. pp 6–19, 2008.

[6] E. Emami, G. Bebis, A. Nefian, and T. Fong, "Automatic Crater Detection Using Convex Grouping and Convolutional Neural Networks", 11th International Symposium on Visual Computing, 2015.

[7] A. Smirnov, "Exploratory Study of Automated Crater Detection", 2012.

[8] G. Troglio, J. Moigne, A. Benediktsson, G. Moser, S. Serpico, "Automatic Extraction of Ellipsoidal Features for Planetary Image Registration", Geoscience and Remote Sensing Letters, Vol. 9. Issue 1. pp 95-99, 2012.

[9] P. Wetzler, R. Honda, B. Enke, W. Merline, C. Burl, "Learning to Detect Small Impact Craters", 7th IEEE Workshop on Application of Computer Vision, Vol. 1. pp178-184, 2005

[10] Kim, J., Muller, J. : Impact Crater Detection on Optical Images and DEMS, International Society for Photogrammetry and Remote Sensing, Working Group IV/9: Extraterrestrial Mapping Workshop, Advances in Planetary Mapping, 2003.

[11] Ding, M,. Caob, Y., Wub, Q. : Novel Approach of Crater Detection by Crater Candidate Region Selection and Matrix-pattern-oriented Least Squares Support Vector Machine, Chinese Journal of Aeronautics, Vol. 26. Issue 2. pp 385–39, 2013.

[12] E. Urbach, F. Stepinski, "Automatic detection of sub-km craters in high resolution planetary images", Planetary and Space Science, Volume 57, Issue 7, pp 880–887, 2009.

[13] Palafox, L., Alvarez, A., Hamilton, C. : Automated Detection of Impact Craters and Volcanic Rootless Cones in Mars Satellite Imagery Using Convolutional Neural Networks and Support Vector Machines, 46th Lunar and Planetary Science Conference (2015).

[14] R. Martins, P. Pina, J. Marques, M. Silveira, M. Silveira, "Crater Detection by a Boosting Approach", Geoscience and Remote Sensing Letters, Vol. 6. Issue 1. pp 127-131, 2009.

[15] P. Viola, M. Jones, "Robust Real-Time Face Detection", International Journal of Computer Vision, Vol. 57. Issue 2. pp 137-154, 2004.

[16] S. Jina, T. Zhanga," Automatic detection of impact craters on Mars using a modified adaboosting method ", Planetary and Space Science, Volume 99, pp 112–117, 2014.

[17] Z. Sun, G. Bebis and R. Miller, "Monocular precrash vehicle detection: features and classifiers," in IEEE Transactions on Image Processing, vol. 15, no. 7, pp. 2019-2034, 2006.

[18] T. Ojala, M. Pietikäinen, D. Harwood, "A comparative study of texture measures with classification based on featured distributions", Pattern Recognition - PR , vol. 29, no. 1, pp. 51-59, 1996.

[19] V. Ojansivu, J. Heikkilä, "Blur Insensitive Texture Classification Using Local Phase Quantization", Image and Signal Processing, Volume 5099 of the series Lecture Notes in Computer Science pp 236-243.

[20] N. Dalal and B. Triggs. "Histograms of oriented gradients for human detection". 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp 886-893, 2005.

[21] M. Calonder, V. Lepetit, C. Strecha, and P. Fua, "BRIEF: Binary Robust Independent Elementary Features", 11th European Conference on Computer Vision (ECCV), 2010.

[22] E. Rublee, V. Rabaud, K. Konolige and G. Bradski, "ORB: An efficient alternative to SIFT or SURF," Computer Vision (ICCV), 2011 IEEE International Conference on, , pp. 2564-2571, 2011.

[23] S. Leutenegger, M. Chli and R. Y. Siegwart, "BRISK: Binary Robust invariant scalable keypoints," Computer Vision (ICCV), 2011 IEEE International Conference on, pp. 2548-2555, 2011.

[24] Alahi, R. Ortiz, and P. Vandergheynst. "FREAK: Fast Retina Keypoint". IEEE Conference on Computer Vision and Pattern Recognition, 2012.

[25] http://www.nasa.gov/mission_pages/LRO/main/index.html.

[26] K. Sung, T. Poggio, "Example-based Learning For View-based Human Face Detection", IEEE Transactions on Pattern Analysis and Machine Intelligenc, Vol. 20. Issue 1. pp 39-51, 1998.