Automatic detection of Martian dark slope streaks by machine learning using HiRISE images

Yexin Wang, Kaichang Di *, Xin Xin, Wenhui Wan

State Key Laboratory of Remote Sensing Science, Institute of Remote Sensing and Digital Earth, Chinese Academy of Sciences, No. 20A, Datun Road, Chaoyang District, Beijing 100101, China

ARTICLE INFO

ABSTRACT

Dark slope streaks (DSSs) on the Martian surface are one of the active geologic features that can be observed on Mars nowadays. The detection of DSS is a prerequisite for studying its appearance, morphology, and distribution to reveal its underlying geological mechanisms. In addition, increasingly massive amounts of Mars high resolution data are now available. Hence, an automatic detection method for locating DSSs is highly desirable. In this research, we present an automatic DSS detection method by combining interest region extraction and machine learning techniques. The interest region extraction combines gradient and regional grayscale information. Moreover, a novel recognition strategy is proposed that takes the normalized minimum bounding rectangles (MBRs) of the extracted regions to calculate the Local Binary Pattern (LBP) feature and train a DSS classifier using the Adaboost machine learning algorithm. Comparative experiments using five different feature descriptors and three different machine learning algorithms show the superiority of the proposed method. Experimental results utilizing 888 extracted region samples from 28 HiRISE images show that the overall detection accuracy of our proposed method is 92.4%, with a true positive rate of 79.1% and false positive rate of 3.7%, which in particular indicates great performance of the method at eliminating non-DSS regions.

1. Introduction

Dark slope streaks (DSSs) are seen on the surface of Mars as dark, narrow, fan shaped features (e.g., Fig. 1) that appear on slopes covered by dust at low latitudes (Schörghofer, 2014). Observations from multiple coverage images show that fresh DSSs are darker than their surrounding area, then gradually fade away over time, finally merging with the background and disappearing (Carr, 2007; Gremminger, 2005–2006). The interest in investigating DSS arises because it is one of the active geologic features that can currently be observed on Mars. Study of its morphologies, distributions and changes, (Sullivan et al., 2001; Schorghofer and King, 2011) are important for revealing and understanding its formation mechanisms (Burleigh et al., 2012; King 2005–2006; Aharonson et al., 2003), inner geological structure (Gerstell et al., 2004), relations with atmospheric action (Cantor, 2007; Chuang et al., 2010; Baratoux et al., 2006), surface composition (Ojha et al., 2015),

and other fields (Schorghofer et al., 2002). Before these studies can be carried out, detection of DSSs is the first basic step.

According to Baratoux's statistics (Baratoux et al., 2006), the widths of DSSs mostly range from 25 to 125 m, and many streaks can only be clearly observable and confirmed on high resolution images. In the past, this work was mainly done manually (Schorghofer et al., 2002; Bergonio et al., 2013; Chuang et al., 2007). Currently, high resolution data from several missions are still being released each year. Nevertheless, the coverage of Mars at a resolution finer than 2 m/pixel is still low. In the near future, there will be more missions carrying high resolution sensors. With the increasingly massive amounts of Mars high resolution data available now and in the future, an automatic detection method is highly desirable.

Before the development of automatic approaches, manual extraction method was used: Schorghofer et al. (2007) studied slope steak activity in five areas by manually comparing Viking observations in 1977–1980 and Mars Orbiter Camera (MOC) observations 1998–2005; Aharonson et al. (2003) performed a global analysis of the slope streaks by manually examining 29,326 MOC narrow angle images and found slope streaks in 1386 images; Baratoux et al. (2006) manually identified and measured the
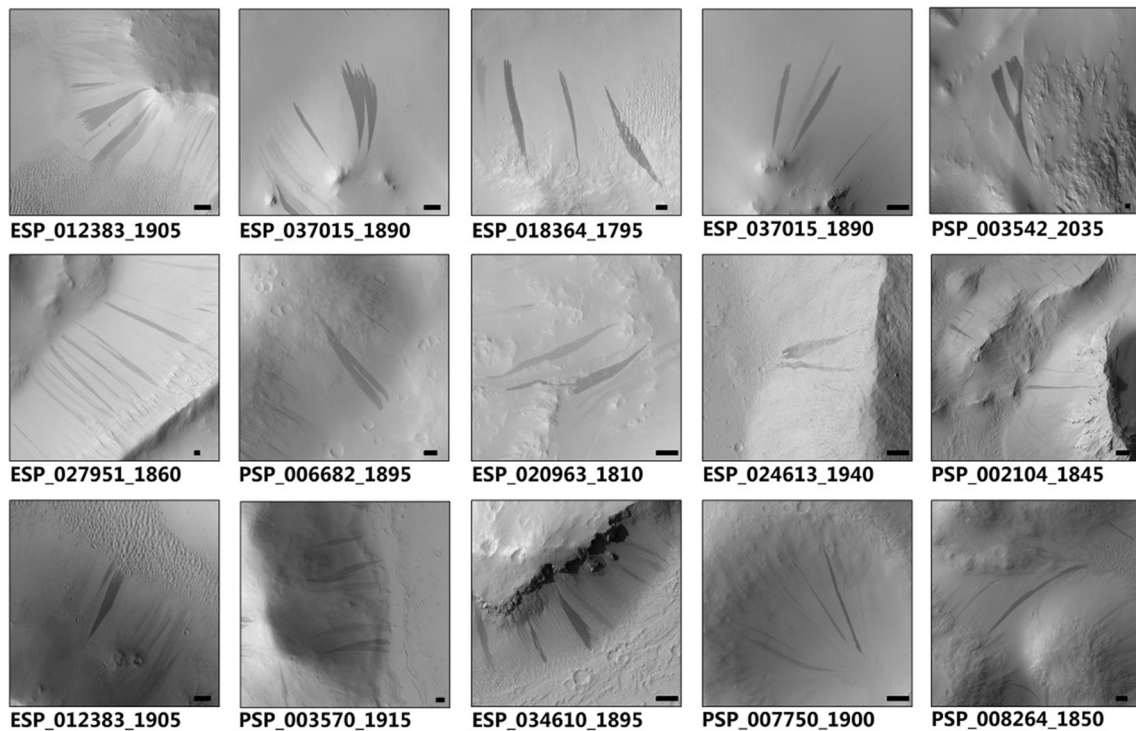

Fig. 1. Examples of DSSs on Mars. HiRISE observation IDs are given below each subfigure. All the images are stretched for contrast enhancement. The scale bar in each subfigure is 100 m, and north is up.

lengths, widths, and directions of DSSs in two regions and analyzed these properties with respect to factors such as terrain, wind direction, and surface roughness.

As manual work is time consuming, some automatic DSS extraction methods have been developed recently. Di et al. (2014a) used a context window that calculates the grayscale gradients for each pixel in the image, then set a threshold to choose the pixels that have high contrast against the background to form the DSS regions. However, the work did not include a recognition stage, so the extracted regions need the further confirmation of an expert. Wagstaff et al. (2012) computed the entropy in a local window for every pixel and picked the pixels with entropy higher than a certain threshold to form a region. They then used its shape properties and simple grayscale information to train a classifier. For DSSs that are located in more complicated backgrounds, these region descriptions are not sufficient to describe the target. Given the few previous works on DSS detection, machine learning is a promising approach that would save manual recognition time.

For other geology features, especially for craters, numerous machine learning detection methods have been proposed (Burl et al., 2001; Vinogradova et al., 2002; Martins et al., 2009; Ding et al., 2013; Cohen and Ding, 2014; Di et al., 2014b). Given crater characteristics, a window-based machine learning method is always employed. The typical way to implement a window-based machine learning approach is to use square windows that each includes a crater as positive samples, and windows without craters as negative samples. Then a feature description method is used to extract the characteristics of both positive and negative samples. With the extracted feature descriptions and the labels of the samples, a machine learning model is applied to train a classifier. When testing a new image, multiple scale sliding square windows are used to traverse the entire image, and the classifier is used to judge whether the current square window contains a crater. The final results are obtained by eliminating the reduplicative squares. If the contour of the crater is needed, edge extraction and ellipse fit-

ting should be done in the identified square (Kim and Muller, 2005). Because a DSS is slender and often appears very close to adjacent ones, the window-based method is not appropriate in this case. Using this window-based method, in some cases, a DSS may only cover a small part of the entire window, meanwhile the window may include portions of adjacent DSSs and other surface features (see Fig. 1). In this way, the features for describing a window containing a DSS as positive samples are indistinguishable enough such that an effective classification is not achievable (Cheng et al., 2014). Another mode of pattern recognition is to use object-based methods (Chen et al., 2014; Blaschke, 2010; Mallinis et al., 2008), i.e., first forming regions obtained by a certain segmentation algorithm (such as those described in Di et al. (2014a) and Wagstaff et al. (2012) and other methods: clustering (Maulik and Saha, 2010), region growing (Yu and Clausi, 2008), etc.), and then training the classifier using the features extracted from the regions. The segmentation step must be applied to the new images for identification before applying the classifier. However, object-based recognition methods only focus on grayscale or shape features within the contour, ignoring the adjacent grayscale information, which is the prominent characteristic of a DSS. A DSS lacks texture information on its surface, hence the grayscale difference between the inside and outside of the DSS is its distinguishable characteristic. If this information can be taken into account, it may improve the recognition accuracy.

In this study, we present an automatic DSS detection approach that combines region extraction and machine learning techniques. The data used in this study is briefly introduced in Section 2. In Section 3, we first present the interest region extraction process that combines gradient and regional grayscale information. A novel recognition strategy that considers the texture inside the extracted regions as well as the imaging characteristics of its surrounding area is then proposed. Section 4 presents the experiment results based on our proposed method. Finally, the conclusion is given in Section 5.

2. Data

As mentioned in the previous section, because of the small width of DSSs, they are only clearly observable on high resolution images. In contrast to two high resolution data sets obtained by the MOC narrow angle camera (resolution of 1.4 m/pixel, stop releasing new data since November 2006) and the High Resolution Stereo Camera (HRSC) (resolution up to 2 m/pixel), HiRISE images have a resolution up to 0.25 m/pixel, and large amounts of new data are being released every few days. Therefore, the work done in this paper is based on the HiRISE images. More specifically, the Reduced Data Record products of the HiRISE data, which have been radiometrically corrected, geometrically rectified, and map projected (downloadable from the HiRISE website http://www.uahirise.org/) are used in this study. We use 34 HiRISE images to test our method. The distribution of the images is shown in Fig. 2. The yellow dots represent the center locations of the HiRISE images. All the images are selected from dust cover area at low latitudes, which are the areas in which DSSs most likely lie.

3. Method

The automatic DSS detection method consists of interest region extraction, classifier training, and new data recognition. First, the interest region extraction process extracts region candidates that may include DSSs. Local Binary Pattern (LBP) features are then calculated from the normalized minimum bounding rectangles (MBRs) of the interest regions. In this way, features inside and outside the regions can both function. After that, a DSS classifier is trained based on these extracted features. When new data needs to be identified by the classifier, interest region extraction and LBP calculation from MBRs should be done beforehand.

3.1. Interest region extraction

Generally, DSSs appear in groups, i.e., there are fresh, slightly fading, and mostly faded DSSs clustered in the same area. Our

aim is to extract the contours and provide the location information of the fresh and slightly fading ones. In this subsection, we present an interest region extraction method to automatically extract regions that likely contain DSSs for both training the classifier and recognizing new data (described in the next section). The flowchart of the proposed region extraction method is shown in Fig. 3.

Note that the HiRISE image is large, therefore preprocessing is accomplished on divided sub-images. According to the size of the HiRISE images, we divide each image into sub-images with 50% overlapping area in each direction to ensure a DSS is not cut in half. The sub-images are preprocessed by successively using image adaptive enhancement, median filtering and mean filtering in order to increase contrast and remove noises. Then the images after preprocessing are recursively convolved by Deriche operator to extract the edges. The Deriche operator is based on Canny principles, and filters the images recursively in both x and y directions. The edge direction and amplitude are calculated based on the convolution results, then the amplitude is non-maximum suppressed in the computed direction and a hysteresis threshold is used to obtain the extracted edges. Further technical details can be found in Deriche (1987).

The Deriche operator provides excellent edge detection results by considering the local gradient information. When applying it on HiRISE images, some unwanted edges such as ridge lines and gullies cannot be removed by a simple principle such as length threshold. At the HiRISE resolution, DSSs are imaged as dark regions of length ranging from hundreds to thousands pixels. Given this feature, not just local but regional information in a large area should be considered. In this case, a sliding square window of width w is considered over the entire image, comparing the gray value of the center pixel $I(x, y)$ to the average gray value of the current window:

$$H(x, y) = I(x, y) - \frac{1}{w^2} \sum_{(m,n) \in W} I(m, n) \tag{1}$$

where $I(m, n)$ represents the gray value of each pixel in window W. Threshold D is set to preserve the pixels that have high contrast
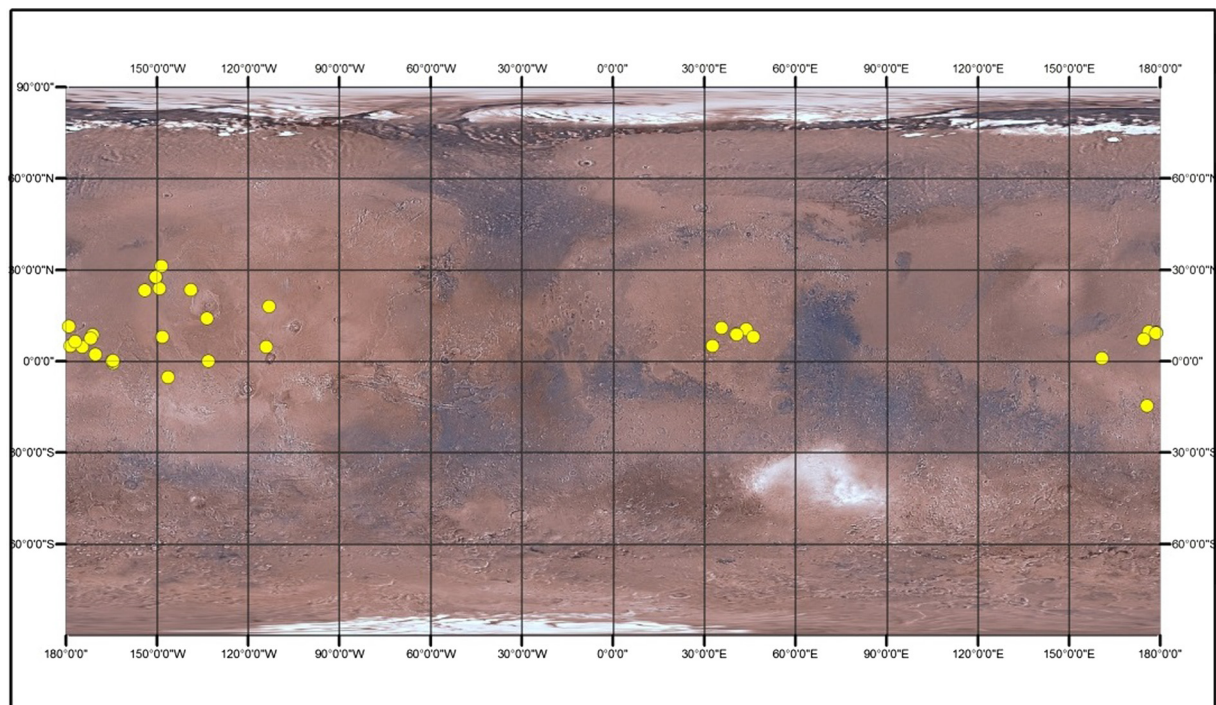

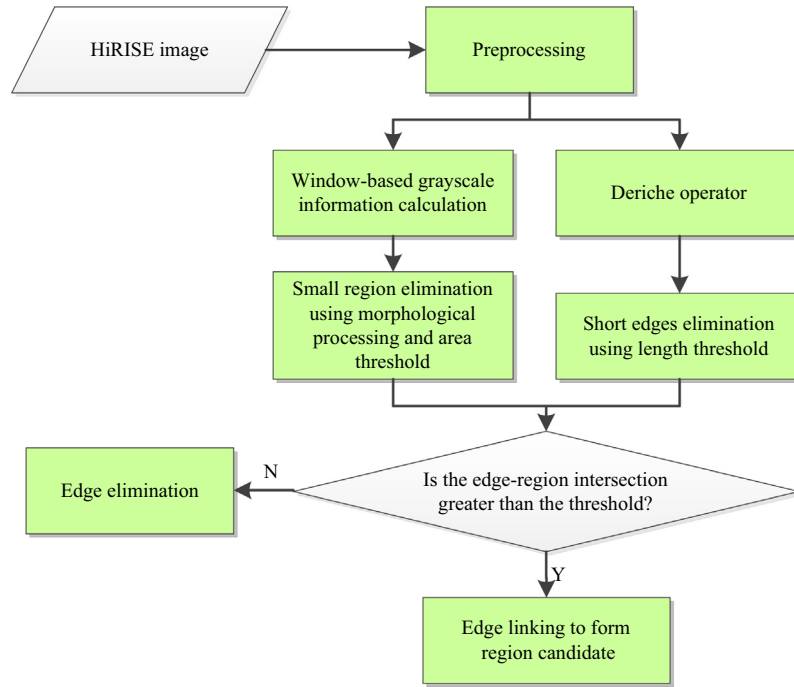**Fig. 2.** Distribution of the HiRISE images used in this study.

Fig. 3. Flowchart of the interest region extraction method.

with respect to the background to form dark regions. In practice, we adopt a 50×50 pixels window and D is set to 20 (for gray values ranging from 0 to 255 in our 8-bit images). In order to reduce calculation time, only pixels with gray values less than 120 are used in this window-based detection. Small noise regions extracted by the window-based method are eliminated by morphological opening and closing operations and small area thresholding. The remaining regions are expanded with a disk-shaped structure element of radius 3. If 70% of the length of an edge extracted by the Deriche method is in one of the regions, we keep the edge. Otherwise, the edge is eliminated. In this way, most undesired edges are removed. Finally, the remaining edge pieces are linked by adjacent and similar direction properties in order to form the interest regions for the subsequent processing.

Fig. 4 shows some of the extraction results. Fig. 4(a) shows several extracted DSSs. Fig. 4(b) shows a case where in addition to the DSSs, other areas are also extracted at the bottom of the image. The two images also show that some fading DSSs are merged with the background or other dark areas, hence they are not extracted. Because of the lighting, some terrain fluctuation regions such as ridges and ravines on the images may also be extracted, as shown in Fig. 4(c). As we can see from Fig. 4, the extracted interest regions

contain DSSs and other kinds of land features. Hence, pattern recognition should be performed to pick out the DSSs. This process is introduced in the next section.

3.2. LBP feature calculation in normalized MBRs

By extracting various region properties or calculating the grayscale statistics such as mean and variance, the region candidates derived in the previous subsection can be used directly for traditional object-based recognition method. However, as mentioned above, these properties are not sufficient and appropriate for a detailed description of DSS. Because of the various shapes of the interest regions, more complex description operators such as LBP (Ojala et al., 1996), Histogram of Oriented Gradient (HOG) Dalal and Triggs, 2005, and Haar-like (Lienhart and Maydt, 2002) features, which provide more precise description, are impractical to calculate. To solve this problem as well as utilize the grayscale difference both inside the DSS region and around it, we propose to calculate these complex features in the minimum bounding rectangle (MBR) of the extracted interest region. As the shapes and sizes of the interest regions vary and the directions of the rectangles are stochastic, we further propose to normalize the MBRs
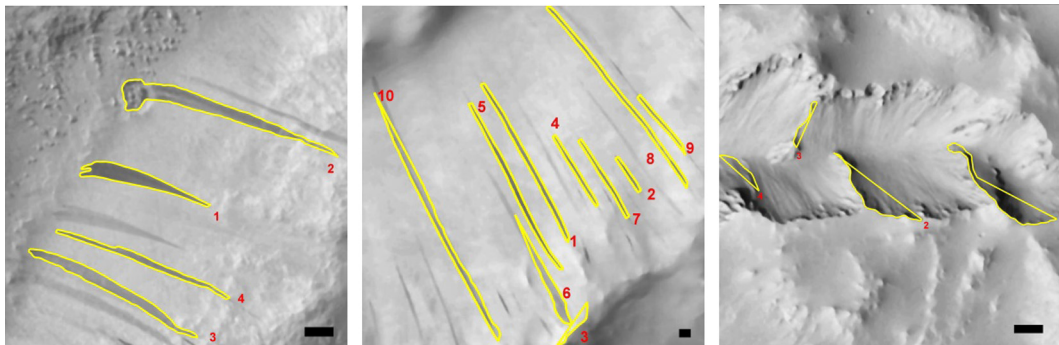

Fig. 4. Some examples of interest region extraction results. The scale bar in each subfigure is 100 m, and north is up.

before calculating complex description operators. Specifically, the MBRs are rotated to the horizontal direction and are resampled to the same size, as shown in rightmost column of Fig. 5. Fig. 5 shows several regions extracted from the HiRISE images, including both DSS and non-DSS regions. In this figure, it can be seen that there are distinct differences near the four corners of the rectangle, which indicate the usage of only shape and simple grayscale statistics cannot reflect the differences. Before applying the feature description operator, histogram equalization is applied to the normalized MBRs, as shown in the first rows of Fig. 6(a) and (b).

In this study, we use the LBP operator in the normalized MBRs of the interest regions to extract features for training the classifier and recognizing DSSs in new data. The LBP feature is a powerful descriptor for analyzing image texture that was first proposed by Ojala et al. (1996). The basic idea is to compare every pixel to each of its eight neighbors in the detection window (typically a square window). When the center pixel's value is greater than the neighbor's value, 1 is marked on that neighbor pixel, otherwise, it is set to 0. The 8-digit binary number is then converted to a decimal number, which is the value of the center pixel. In practice, circular neighbors and rotation invariance is usually applied (Ojala et al., 2002). After calculation of all the pixels in the detection window, the sub image framed by the detection window is transferred into a LBP code image whose value is this decimal number. The final uniform LBP descriptor is a vector that records the value of the normalized histogram computed from the LBP code image.

Fig. 6 illustrates the LBP feature calculation process and results of ten examples, including five DSS regions and five other regions. In Fig. 6(a) and (b), the first rows show the sample image after normalization, the second rows show the LBP code image, which consists of the LBP values with rotation invariance for each pixel in the first row images, and the third rows presents the uniform LBP value in histogram format. It is clear from Fig. 6 that the LBP features of DSSs in the normalized MBRs are distinct from other extracted surface features.

3.3. Classifier training and DSS recognition from new data

For the normalized MBR images, the ones that contain DSSs are manually labeled as positive samples, and the rest are labeled as negative samples. LBP features are calculated for all the samples,

and then each sample is described by a vector. Using these vectors and their corresponding labels as inputs, a DSS classifier is constructed by training with a machine learning algorithm. For DSS classifier training, we adopted the Adaboost machine learning algorithm (Freund and Schapire, 1997). The basic idea of this algorithm is to generate a weak classifier at each iteration and reassign the weight of each sample in favor of those that were misclassified by the previous weak classifiers. The final strong classifier is the combination of all the weak ones.

In our case, each normalized MBR image is a training sample consisting of a 1×59 vector using uniform LBP features with labels: $S = \{(x_i, y_i)\}$, $i = 1, 2, \ldots, N$, where N is the total number of training samples, $x_i = \{v_{i1}, v_{i2}, \ldots, v_{i59}\}$ is the normalized LBP description vector of the i-th sample, $y_i \in \{1, -1\}$ is the label of the i-th sample, in which 1 represents a sample containing DSS, and -1 represents samples containing other areas. The iteration number is set to T, which means that the final strong classifier is a combination of T weak classifiers. When $t = 1$, i.e., at the first iteration, weight of all samples is:

$$w_1(i) = \frac{1}{N} \tag{2}$$

For each following iteration, i.e., $t = 2, 3, \ldots, T$, a weak classifier $C_t(x)$ is obtained, whose accuracy rate may be little higher than a random classifier. Applying $C_t(x)$ to every training sample, the error rate of this classifier is calculated as:

$$e_t = P(C_t(x_i) \neq y_i) = \sum_{i=1}^{N} w_t(i) I(C_t(x_i) \neq y_i) \tag{3}$$

Coefficient α_t is defined as the weight of classifier C_t in the final strong classifier using the error rate by:

$$\alpha_t = \frac{1}{2} \ln \frac{1 - e_t}{e_t} \tag{4}$$

The weights of all the training samples are updated such that the weights of the wrongly classified samples are increased. The aim is to focus on the wrong samples at the next iteration. The weights of all the samples are reassigned as follows:

$$w_{t+1}(i) = \frac{w_t(i)}{Z_t} e^{-\alpha_t y_i C_t(x_i)} \tag{5}$$

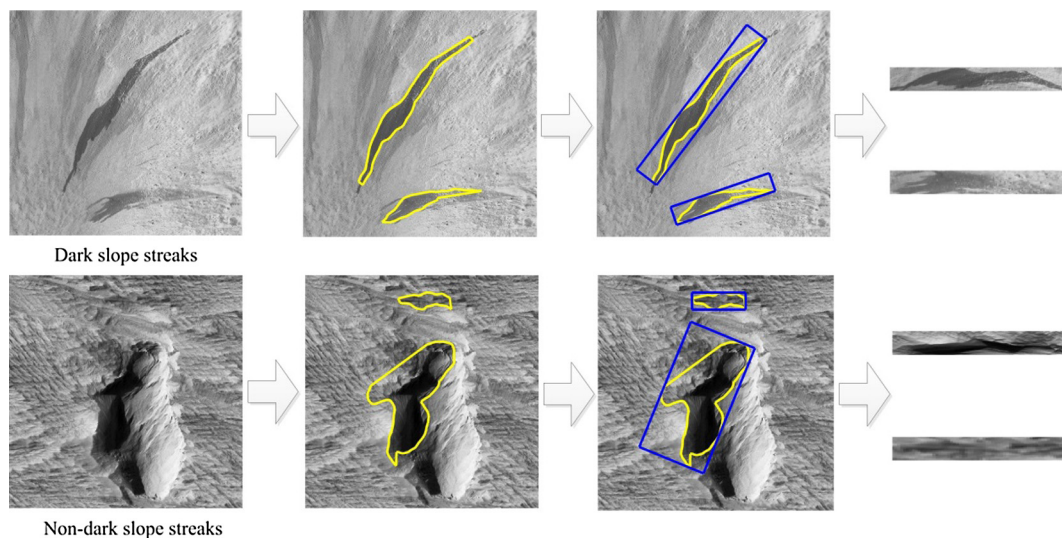Dark slope streaks

Non-dark slope streaks

Fig. 5. Extracted interest regions (yellow) and the corresponding minimum bounding rectangles (blue) are shown in the second and third columns, respectively. The rightmost column shows the MBRs after normalization. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)
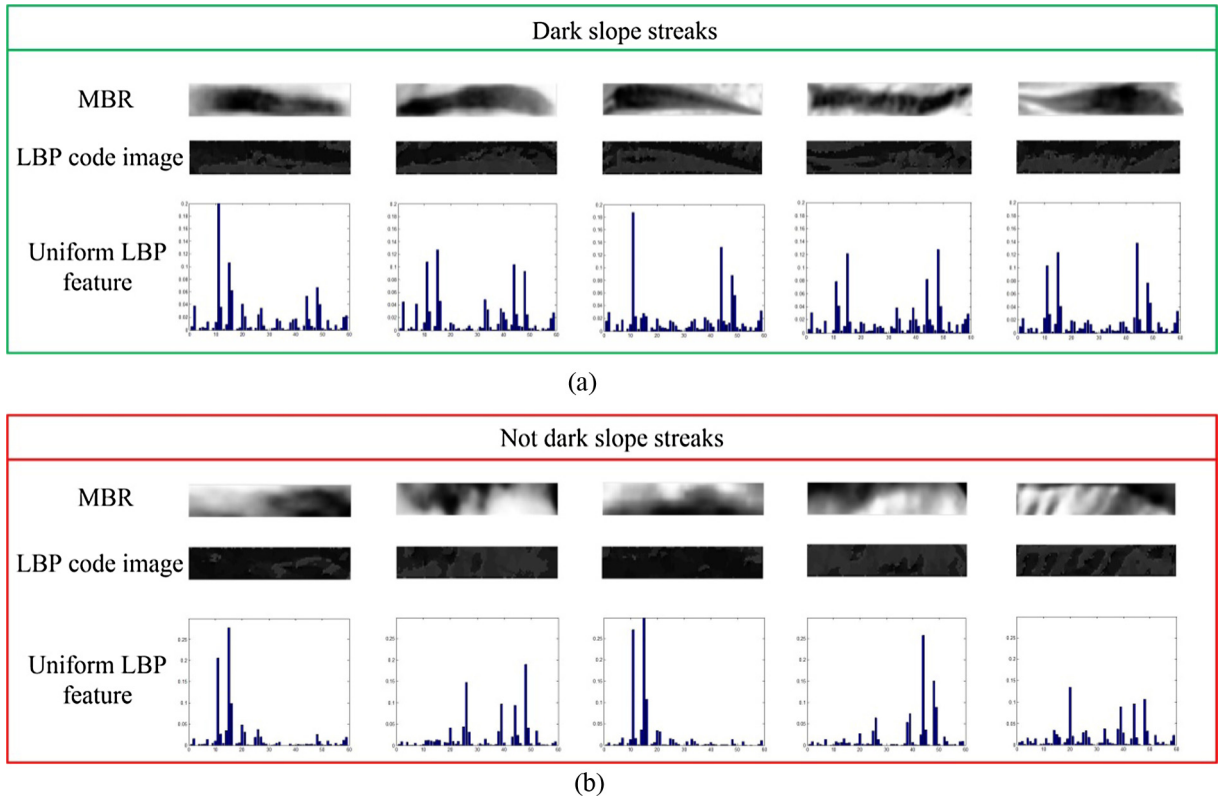
Fig. 6. Ten examples of the LBP feature extraction results for (a) DSSs and (b) other extracted regions in the normalized MBRs.

where $Z_t = \sum_{i=1}^{N} w_t(i)e^{-\alpha_t y_i C_t(x_i)}$.

Finally, the classifiers obtained at each iteration are combined to construct the strong classifier as follows:

$$C(x) = (\text{sign})\sum_{t=1}^{T} \alpha_t C_t(x) \qquad (6)$$

When the strong classifier is constructed, it can be applied to new data to identify DSSs. It should be stressed that new data should be preprocessed as mentioned in Sections 3.1 and 3.2. In other words, interest regions in new data images should be first extracted, and LBP features are calculated based on the normalized MBRs of the extracted regions. The input to the classifier is the LBP vector. As the labels are pre-defined as 1 and −1, the result of the identification is a value between [−1, 1]. Normally, a region with identification results greater than zero is considered to be one that contains a DSS if there is no specified threshold. The flowchart in Fig. 7 shows the processes of training using sample data (blue line) and recognition using new data (red line).

4. Experiments and discussion

To validate our methods, we use 34 HiRISE images, distributed as shown in Fig. 2. By applying the region extraction method introduced in Section 3, 1024 interest regions were detected. Of these regions, 888 regions extracted from 28 images were utilized for training the DSS classifier, and the remaining 136 interest regions from the other six HiRISE images were used for new data testing.

4.1. Comparative results of classifier training

All the extracted regions from the previous step were manually labeled in order to train the classifier and validate its recognition
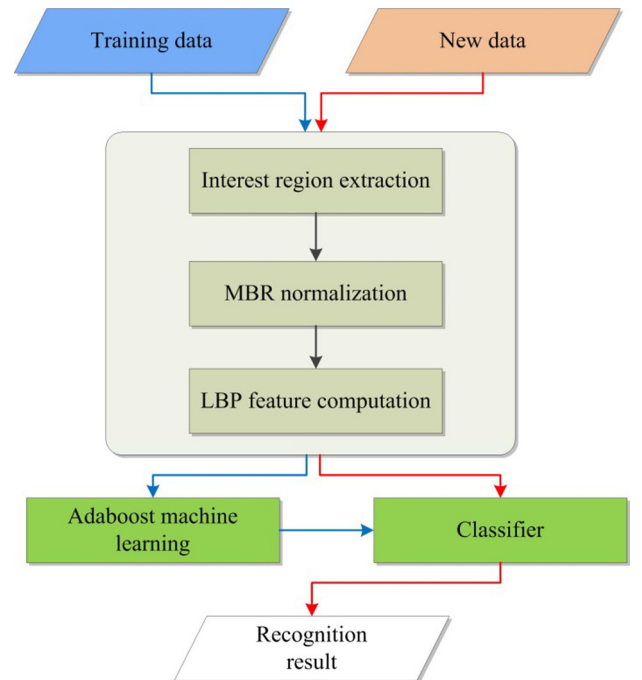

Fig. 7. Flowchart of both training (blue line) and recognition (red line) processes. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

results. We utilized 208 positive samples and 680 negative samples to train the DSS classifier using the strategy described in Section 3. About 11% of the training samples were used for a validation test.

During the training process, we deployed five different descriptors for comparison: Haar-like, HOG, region properties (RP), LBP, and LBP combined with RP (LBP + RP) features. Haar-like features were calculated on normalized 5×25 pixel samples, and the other descriptors (not including RP) were computed on 20×100 pixel samples. For the Haar-like descriptor, a total of 558 features were obtained for each normalized sample. In addition, the numbers of obtained HOG and LBP descriptors were 3456 and 59, respectively. RP here refers to the characteristics of the extracted regions instead of those of the MBRs. The RPs used for each sample consisted of 11 region shape features including area, the ratio of the region area and convex-hull area, fitting ellipse properties, compactness, and six grayscale features such as mean, standard deviation, smoothness, and entropy.

Fig. 8 shows the error decreasing curves of the five descriptors over 100 subsequent iterations. Note that for training, we set the number of iterations to 200, yet the fastest decline occurs from the 1st to the 50th iteration. After that, the curves decline gently; therefore, we only show the curve to the 100th iteration. In Fig. 8, the highest training error is from the method that used region properties. The other four are less than 5% after the 50th iteration. This is because the goal of each iteration is to assign a weight on one feature to obtain a weak classifier, and the number of features in RP is not sufficient to combine into a strong classifier. Haar-like and HOG features have lower errors than the other descriptors, which is unsurprising, because the number of features are large and Adaboost can build a classification tree by picking the most fitted features in training process to obtain a low error.

The performances of the descriptors are further evaluated using the receiver operating characteristic (ROC) curve, which shows the relation between the true positive rate (TPR) and false positive rate (FPR). Fig. 9 shows the ROC curves of five different descriptors obtained using the validation data. It can be seen that although the HOG and Haar-like features obtain high accuracy in the training phase, they fail to achieve a high TPR in the validation test. The LBP descriptor achieves a higher TPR than the other four when maintaining the same FPR. Moreover, Fig. 9 shows the addition of RP to LBP is not a significant help. Therefore, for describing and identifying DSS in normalized MBRs, the LBP operator is sufficient.

Using the LBP feature, we also compared the performances of three different machine learning algorithms: Adaboost, support vector machine (SVM) Mountrakis et al., 2011, and neural network (NN) (a ten-hidden-layer feed forward network was used) (Møller, 1993). The recognition results are shown in Table 1. The results
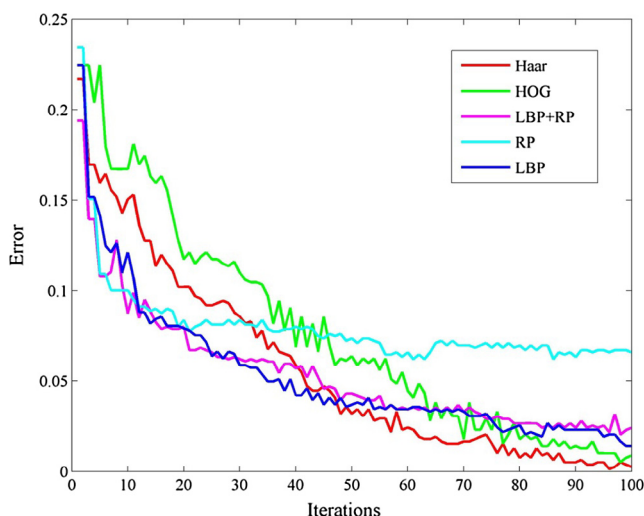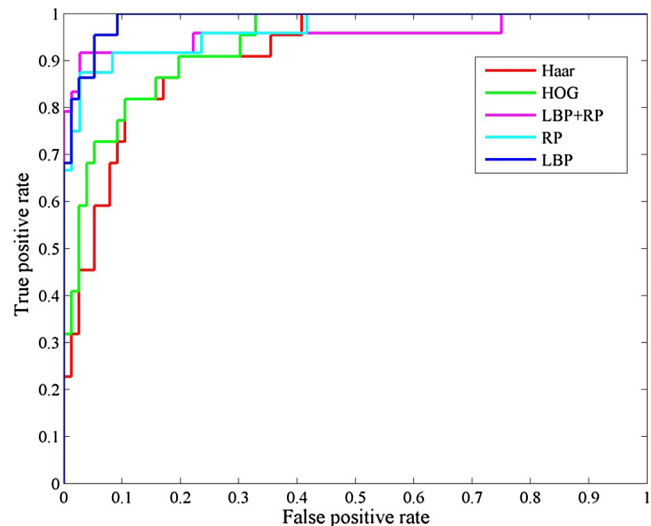

Fig. 9. ROC curves of five different features.

were calculated based on the validation data. Although samples for the validation test account for only 11% of all the samples, the final validation test results are actually based on all the samples. This was implemented by dividing all the samples into nine groups such that each group had almost the same number of positive and negative samples. Eight groups were used for training a classifier and the last one was used for validation. This process was repeated nine times so that each group had the chance to be in the validation test. Table 1 shows that compared to NN and Adaboost, SVM obtained the worst performance in every respect: highest FPR, lowest TPR and total accuracy. Although NN has a slight advantage over Adaboost with respect to TPR, its FPR is much larger than that of Adaboost. To evaluate the different machine learning algorithms in this study, FPR is crucial, as low FPR represents a low error for the elimination of non-DSS regions in all detected regions, which would significantly reduce labor requirement when the data volume is large. Based on this criterion, the Adaboost results have the lowest FPR and the highest total accuracy, and hence it is the most appropriate algorithm for identifying DSSs.

4.2. DSS recognition from new data

In order to test the classifier, we applied it on new data. After interest region detection and LBP feature calculation from the normalized MBRs, the classifier constructed in Section 4.1 takes the normalized LBP features as inputs and generates the recognition result. The preprocessing procedure on the new data before it is classified is the same as that of the training samples. Note also that all the training samples were treated like new data when they were chosen for the validation test. Hence, the detection accuracy of new data is similar to that listed in Table 1.

Fig. 10 shows the recognition results of a sub-image of a HiRISE image, in which the red rectangles are used to mark the MBRs that contain DSSs. Fig. 11 shows additional recognition results. When

Fig. 8. Recognition error of five features in each iteration of the training process.

Table 1
Performance comparison of three different classifiers trained using SVM, NN, and Adaboost.

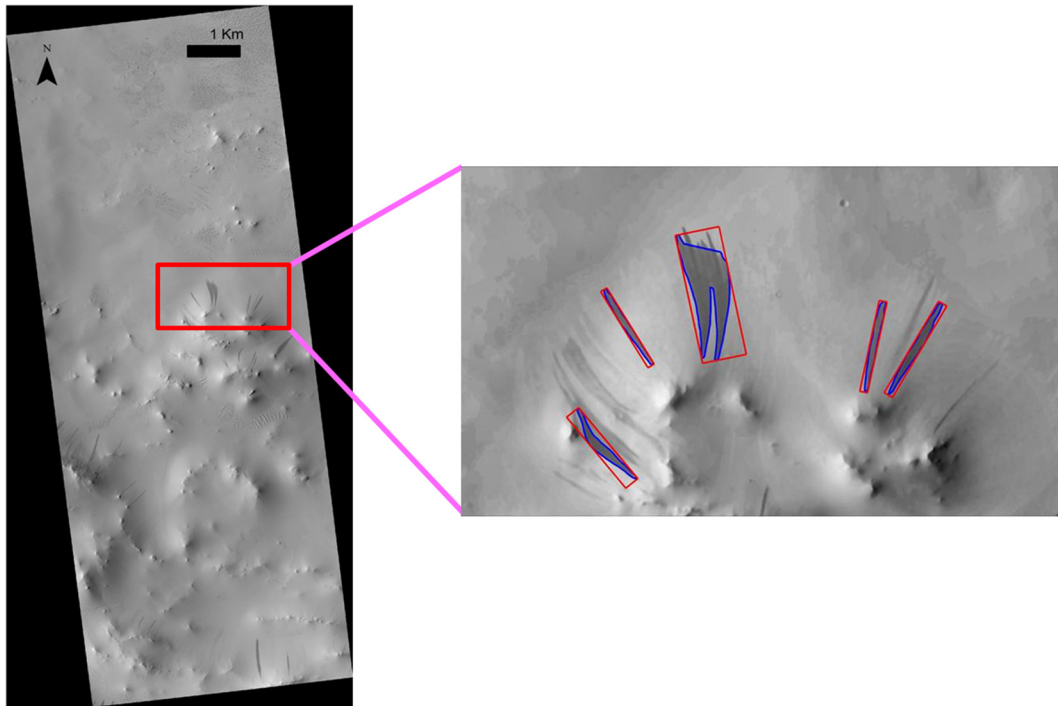| Evaluation criteria	Support vector machine (SVM)	Neural network (NN)	Adaboost
True positive rate (TPR) %	75.1	80.1	79.1
False positive rate (FPR) %	10.9	8.2	3.7
Total accuracy %	85.9	89.1	92.4

Fig. 10. Sub image (red rectangle on the left image) of a HiRISE image (ID: ESP_037015_1890) and its detection results (right). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)
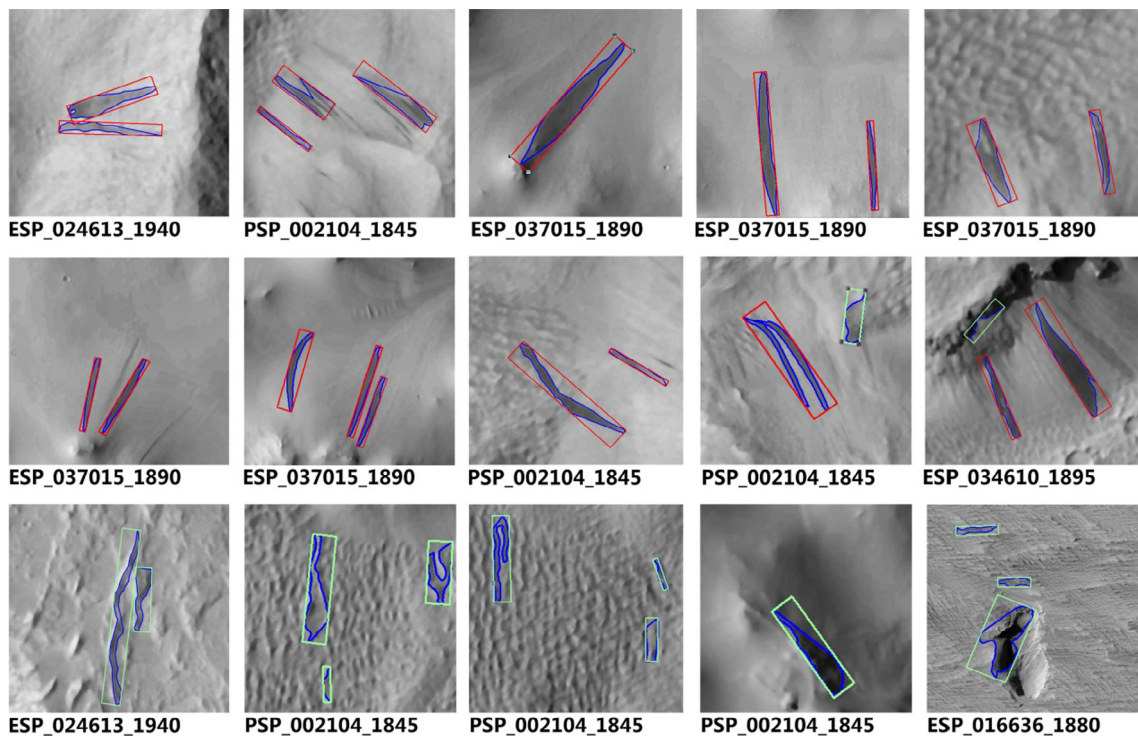

Fig. 11. Examples of the recognition results on sub-images of HiRISE images.

an MBR is recognized as other surface features, it is indicated with green. The ten images in rows 1 and 2 of Fig. 11 show that the DSSs are correctly identified, especially in the last two images of row 2, where there are MBRs containing non-DSSs that were successfully eliminated by the classifier. The five images in the last row show that in different terrain, various surface features such as ridge shadows and ravines are correctly identified and labeled.

5. Conclusion

In this study, an automatic detection method was proposed for extracting and identifying DSSs on the Martian surface using HiR-ISE images. The method consists of interest region extraction and machine learning techniques. First, interest regions are extracted from HiRISE images using a method that combines gradient and

regional grayscale information. A novel feature computation strategy is then used that normalizes the MBR of the extracted regions and calculates the LBP features. Finally, Adaboost machine learning algorithm is applied to train a DSS classifier using LBP features in order to identify DSSs in new HiRISE images. We experimentally compared five different feature descriptors and three different machine learning algorithms, and conclude that the classifier trained by Adaboost using LBP features has the highest total detection accuracy (92.4%) and lowest FPR (3.7%). The latter criterion indicates a good performance of the classifier at the task of eliminating non-DSS regions, which is especially advantageous for saving time when processing a large amount of data without focusing on the non-target areas. Future work may include research on training more complex and accurate classifiers to identify several of the most interesting Martian surface features.

References

Aharonson, O., Schorghofer, N., Gerstell, M.F., 2003. Slope streak formation and dust deposition rates on Mars. J. Geophys. Res. 108, 5138.

Baratoux, D., Mangold, N., Forget, F., Cord, A., Pinet, P., Daydou, Y., et al., 2006. The role of the wind-transported dust in slope streaks activity: Evidence from the HRSC data. Icarus 183, 30–45.

Bergonio, J.R., Rottas, K.M., Schorghofer, N., 2013. Properties of martian slope streak populations. Icarus 225, 194–199.

Blaschke, T., 2010. Object based image analysis for remote sensing. ISPRS J. Photogramm. Remote Sens. 65, 2–16.

Burl, M.C., Stough, T., Colwell, W., Bierhaus, E., Merline, W., Chapman, C., 2001. Automated detection of craters and other geological features. In: Proceedings of the Sixth International Symposium on Artificial Intelligence and Robotics & Automation in Space, Quebec, Canada, pp. 18–22.

Burleigh, K.J., Melosh, H.J., Tornabene, L.L., Ivanov, B., McEwen, A.S., Daubar, I.J., 2012. Impact airblast triggers dust avalanches on Mars. Icarus 217, 194–201.

Cantor, B.A., 2007. MOC observations of the 2001 Mars planet-encircling dust storm. Icarus 186, 60–96.

Carr, M.H., 2007. The Surface of Mars. Cambridge University Press.

Chen, G., Zhao, K., Powers, R., 2014. Assessment of the image misregistration effects on object-based change detection. ISPRS J. Photogramm. Remote Sens. 87, 19–27.

Cheng, G., Han, J., Zhou, P., Guo, L., 2014. Multi-class geospatial object detection and geographic image classification based on collection of part detectors. ISPRS J. Photogramm. Remote Sens. 98, 119–132.

Chuang, F.C., Beyer, R.A., McEwen, A.S., Thomson, B.J., 2007. HiRISE observations of slope streaks on Mars. Geophys. Res. Lett. 34, L20204.

Chuang, F.C., Beyer, R.A., Bridges, N.T., 2010. Modification of martian slope streaks by eolian processes. Icarus 205, 154–164.

Cohen, J.P., Ding, W., 2014. Crater detection via genetic search methods to reduce image features. Adv. Space Res. 53, 1768–1782.

Dalal, N., Triggs, B., 2005. Histograms of oriented gradients for human detection. Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on, vol. 1, pp. 886–893.

Deriche, R., 1987. Using Canny's criteria to derive a recursively implemented optimal edge detector. Int. J. Comput. Vision 1, 167–187.

Di, K., Liu, Y., Hu, W., Yue, Z., Liu, Z., 2014a. Mars surface change detection from multi-temporal orbital images. IOP Conference Series: Earth and Environmental Science, vol. 17, p. 012015.

Di, K., Li, W., Yue, Z., Sun, Y., Liu, Y., 2014b. A machine learning approach to crater detection from topographic data. Adv. Space Res. 54, 2419–2429.

Ding, M., Cao, Y., Wu, Q., 2013. Novel approach of crater detection by crater candidate region selection and matrix-pattern-oriented least squares support vector machine. Chin. J. Aeronaut. 26, 385–393.

Freund, Y., Schapire, R.E., 1997. A decision-theoretic generalization of on-line learning and an application to boosting. J. Comput. Syst. Sci. 55, 119–139.

Gerstell, M.F., Aharonson, O., Schorghofer, N., 2004. A distinct class of avalanche scars on Mars. Icarus 168, 122–130.

Gremminger, D., 2005–2006. Decadal variability in slope streak activity on mars HSGC Report 06–14, 39.

Kim, J.R., Muller, J.-P., Gasselt, S.V., Morley, J.G., Neukum, G., 2005. Automated crater detection, a new tool for Mars cartography and chronology. Photogramm. Eng. Remote Sens. 71, 1205–1217.

King, C., 2005–2006. Seasonality of slopes streak formation HSGC Report 10–19, 47.

Lienhart, R., Maydt, J., 2002. An extended set of Haar-like features for rapid object detection. Image Processing. 2002. Proceedings. 2002 International Conference on, vol. 1, pp. I-900-I-903.

Mallinis, G., Koutsias, N., Tsakiri-Strati, M., Karteris, M., 2008. Object-based classification using Quickbird imagery for delineating forest vegetation polygons in a Mediterranean test site. ISPRS J. Photogramm. Remote Sens. 63, 237–250.

Martins, R., Pina, P., Marques, J.S., Silveira, M., Silveira, M., 2009. Crater detection by a boosting approach. IEEE Geosci. Remote Sens. Lett. 6, 127–131.

Maulik, U., Saha, I., 2010. Automatic fuzzy clustering using modified differential evolution for image classification. IEEE Trans. Geosci. Remote Sens. 48, 3503–3510.

Møller, M.F., 1993. A scaled conjugate gradient algorithm for fast supervised learning. Neural Networks 6, 525–533.

Mountrakis, G., Im, J., Ogole, C., 2011. Support vector machines in remote sensing: A review. ISPRS J. Photogramm. Remote Sens. 66, 247–259.

Ojala, T., Pietikäinen, M., Harwood, D., 1996. A comparative study of texture measures with classification based on featured distributions. Pattern Recogn. 29, 51–59.

Ojala, T., Pietikainen, M., Maenpaa, T., 2002. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. IEEE Trans. Pattern Anal. Mach. Intell. 24, 971–987.

Ojha, L., Wilhelm, M.B., Murchie, S.L., McEwen, A.S., Wray, J.J., Hanley, J., et al., 2015. Spectral evidence for hydrated salts in recurring slope lineae on Mars. Nat. Geosci. 8, 829–832. 11//print.

Schörghofer, N., 2014. Slope streak (Mars). In: Encyclopedia of Planetary Landforms. Springer, New York, pp. 1–8.

Schorghofer, N., King, C.M., 2011. Sporadic formation of slope streaks on Mars. Icarus 216, 159–168.

Schorghofer, N., Aharonson, O., Khatiwala, S., 2002. Slope streaks on Mars: Correlations with surface properties and the potential role of water. Geophys. Res. Lett. 29, 2126.

Schorghofer, N., Aharonson, O., Gerstell, M.F., Tatsumi, L., 2007. Three decades of slope streak activity on Mars. Icarus 191, 132–140.

Sullivan, R., Thomas, P., Veverka, J., Malin, M., Edgett, K.S., 2001. Mass movement slope streaks imaged by the Mars Orbiter Camera. J. Geophys. Res. 106, 23607–23634.

Vinogradova, T., Burl, M., Mjolsness, E., 2002. Training of a crater detection algorithm for Mars crater imagery. Aerospace Conference Proceedings, 2002. IEEE, vol. 7, pp. 7-3201–7-3211.

Wagstaff, K.L., Panetta, J., Ansar, A., Greeley, R., Hoffer, M.P., Bunte, M., 2012. Dynamic landmarking for surface feature identification and change detection. ACM Trans. Intell. Syst. Technol. 3, 1–22.

Yu, Q., Clausi, D.A., 2008. IRGS: image segmentation using edge penalties and region growing. IEEE Trans. Pattern Anal. Mach. Intell. 30, 2126–2139.