Effective Feature Selection for Mars McMurdo Terrain Image Classification*

Changjing Shang, Dave Barnes and Qiang Shen
Department of Computer Science, Aberystwyth University, UK
{cns,dpb,qqs}@aber.ac.uk

Abstract

This paper presents a novel study of the classification of large-scale Mars McMurdo panorama image. Three dimensionality reduction techniques, based on fuzzy-rough sets, information gain ranking, and principal component analysis respectively, are each applied to this complicated image data set to support learning effective classifiers. The work allows the induction of low-dimensional feature subsets from feature patterns of a much higher dimensionality. To facilitate comparative investigations, two types of image classifier are employed here, namely multi-layer perceptrons and K-nearest neighbors. Experimental results demonstrate that feature selection helps to increase the classification efficiency by requiring considerably less features, while improving the classification accuracy by minimizing redundant and noisy features. This is of particular significance for on-board image classification in future Mars rover missions.

1 Introduction

There has been growing international interest in the exploration of the surface of Mars over the last decade [2]. In particular, the Panoramic Camera instruments mounted on the Mars Exploration Rovers have acquired many tens of thousands of high-resolution, stereo, multi-spectral images of rocks, soil, and sky from the landing sites. Automated segmentation and classification of such images has since become an important task, especially for surveying places, e.g. for geologic cues [9, 16]. This is because manual inspection and examination is extremely time intensive. Any progress towards automated detection and recognition of objects within Mars images, including different types of rocks and their surroundings, will make a significant contribution to the accomplishment of this task.

Mars images vary significantly in terms of intensity, scale and rotation, and are blurred with noise. This is mainly caused by rover motion, wavelength and resolution changes [4]. These factors make large-scale Mars image classification a very challenging problem. Although many approaches may be applied for classification of such images, it is difficult, if not impossible, to predict which technique would give the best result. Therefore, it is useful to build different classifiers and to validate their performance on a common data set, with respect to common criteria. For this purpose, part of the present work is set to investigate and compare the use of two potentially effective classifiers: Multi-layer perceptron (MLP) neural networks and K-nearest neighbors. Note that these well-developed image classification methods are intentionally used here in order to reduce potential mission risk. Flight projects normally opt to use existing mature technologies rather than totally new mechanisms that tend to have limited experimental performance data [9].

One critical step to successfully build an image classifier is to extract informative features from given images [5, 8, 10, 13]. Without explicit prior knowledge of what characteristics might best represent an original image, many features may have to be extracted. However, generating more features increases computational complexity (especially in light of on-board processing of large scale images concerned in this research), and not all such features may be useful to perform classification. Due to measurement noise the use of extra features may even reduce the overall representational potential of the feature set and hence, the classification accuracy. Thus, it is desirable to employ a method that can determine the most significant features, based on sample measurements, to simplify the classification process, while ensuring high classification performance.

This paper presents an integrated approach for performing large-scale Mars image classification, by exploiting feature selection mechanisms to ensure effective and efficient learning of classifiers. In particular, techniques based on fuzzy-rough sets [7] and information gain ranking [6, 12] are adopted. As a result, only those informative features are required to be generated in order to perform classification. This minimizes feature measurement noise and the computational complexity (of both feature extraction and feature pattern-based classification). The resulting systems gener-

*This work is funded by the Daphne Jackson Trust and the Royal Academy of Engineering.

978-0-7695-3872-3/09 $26.00 © 2009 IEEE
DOI 10.1109/ISDA.2009.105

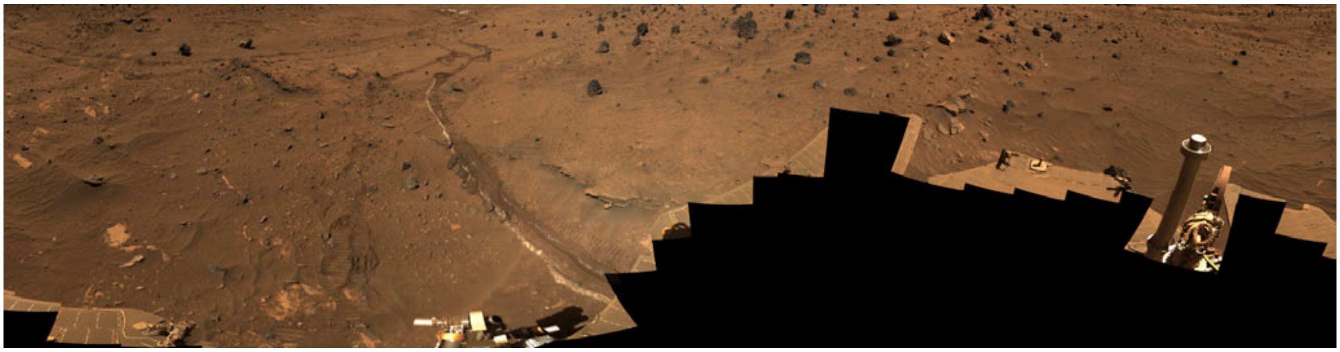


Figure 1. Mars McMurdo panorama image.

ally outperform those using more features or an equal number of features obtained by conventional dimensionality reduction techniques (e.g. principal component analysis [3]), without destroying the underlying semantics of the features. This is of great importance to on-board image classification in future Mars rover missions [1].

The rest of this paper is organized as follows. Section 2 introduces the Mars images under investigation. Sections 3, 4 and 5 outline the key component techniques used in this work, including feature extraction, feature selection and feature pattern classification. Section 6 shows the experimental results, supported by comparative studies. The paper is concluded in Section 7.

2 *McMurdo* panorama image

The images used in the present work are portions taken from the *McMurdo* panorama image, which is obtained from the panoramic camera on NASA's Mars Exploration Rover Spirit and presented in approximately true color [4]. *McMurdo* captures the view from Spirit's spot on the Columbia Hills, showing volcanic rocks around the rover, Husband Hill on the right, the El Dorado sand dunes near the hill and Home Plate below the dunes. As such, it reveals a tremendous amount of detail in part of Spirit's surroundings, including many dark, porous-textured volcanic, brighter and smoother-looking rocks, sand ripple, and gravel (mixture of small stones and sand).

Fig. 1 shows the most part of the original *McMurdo* image (of a size 20480×4124). This image, excluding the areas occupied by the instruments and their black shadows, is used for the work here, involving four major image types (i.e. classes) which are of particular interest. These image types are: grey or dark rock (rock1), orange colored rock (rock2), gravel, and sand, as illustrated in Fig. 2.

3 Feature extraction

Many techniques may be used to capture and represent the underlying characteristics of a given image [5, 11, 13]. In this work, local grey level histograms and the first and second order color statistics are exploited to produce a feature pattern for each individual pixel. This is due to the recognition that such features are effective in depicting the underlying image characteristics and are efficient to compute. The resulting features are robust to image translation and rotation and to scale and intensity variations.

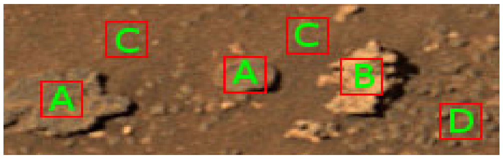


Figure 2. Image classes (A: rock1, B: rock2, C: sand, D: gravel).

3.1 Color statistics-based features

Color images originally given in the RGB (Red, Green and Blue) space are bijectively transformed to those in the HSV (Hue, Saturation and Value) color space [11, 15] (which is widely used in the literature). Six features are then generated per pixel, by computing the first order (mean) and the second order (standard deviation, denoted by STD) color statistics with respect to each of the H, S and V channels, from a neighborhood of the pixel [15]. The size of such neighborhoods is pre-selected by trial and error (which trades off between the computational efficiency in measuring the features and the representative power of the measured features).

3.2 Local histogram-based features

To reduce computational complexity, in extracting this type of feature, given color images are transformed to grey-level (GL) images. For each pixel, a set of histogram features $H_i, i = 1, 2, ..., B$, can be generated within a predefined neighborhood, with respect to a bin size B. Here, the neighborhood size is for convenience, set to the same as that used for color feature extraction, and H_i denotes the normalized frequency of the GL histogram in bin i. To balance between effectiveness and efficiency, B is empirically set to 16 in this work. In addition, two further GL statistic features are also generated, namely, the mean and STD.

4 Feature selection

Feature selection refers to the process of finding a subset of given features that are potentially most effective for use in solving a given problem. It is a particular form of dimensionality reduction which does not disrupt the underlying meaning of the selected features. Although many approaches exist for feature selection, the recently developed fuzzy-rough technique [7] and the popular information gain-based ranking (IGR) method [6, 12] are adopted here. Also employed as an alternative, is the conventional dimensionality reduction mechanism of principal component analysis (PCA) [3]. A brief introduction to these approaches is given below.

4.1 Fuzzy-rough feature selection

Let U be the set of pixels within a given image, P be a subset of features, and D be the set of possible image classes. Fuzzy-rough feature selection (FRFS) is based on the concept of fuzzy-rough dependency of D upon P. This dependency measure is defined by

$$\gamma_P(D) = \frac{\sum\limits_{x \in U} \mu_{POS_{R_P}(D)}(x)}{|U|} \tag{1}$$

where

$$\mu_{POS_{R_P}(D)}(x) = \sup_{X \in U/D} \mu_{\underline{R_P}X}(x) \tag{2}$$

$$\mu_{\underline{R_P}X}(x) = \inf_{y \in U} I(\mu_{R_P}(x,y), \mu_X(y)) \tag{3}$$

and U/D denotes the (equivalence class) partition of the image with respect to D, and I is a fuzzy implicator and T a t-norm. R_P is a fuzzy similarity relation induced by P:

$$\mu_{R_P}(x,y) = T_{A \in P}\{\mu_{R_{\{A\}}}(x,y)\} \tag{4}$$

That is, $\mu_{R_{\{A\}}}(x,y)$ is the degree to which pixels x and y are similar with regard to feature A. It may be defined in many ways, but in this work, the following commonly used similarity relation is adopted:

$$\mu_{R_{\{A\}}}(x,y) = 1 - \frac{|A(x) - A(y)|}{A_{max} - A_{min}} \tag{5}$$

where $A(x)$ and $A(y)$ stand for the value of feature $A \in P$ of pixel x and that of y, respectively, and A_{max} and A_{min} are the maximum and minimum feature values of feature A.

FRFS works by employing the above dependency measure to choose which features to add to the subset of the current best features. It terminates when the addition of any remaining feature does not increase the dependency.

4.2 Information-gain feature ranking

Let D_X be the value set of feature X and D_C be the label set of class variable C. The following equations define the entropy of the class before and after observing the feature X, respectively:

$$H(C) = -\sum_{c \in D_C} p(c) log_2 p(c) \tag{6}$$

$$H(C|X) = -\sum_{x \in D_X} p(x) \sum_{c \in D_C} p(c|x) log_2 p(c|x) \tag{7}$$

The amount by which the entropy of the class decreases after observing a certain feature reflects the additional information about the class provided by that feature and is called information gain: $IG = H(C) - H(C|X)$. It measures how well a given feature separates data points with respect to their underlying class labels. Suppose that there are N features: $X_i, i = 1, 2, ..., N$. Each of these can be assigned a score based on the information gain over the class entropy due to observing itself:

$$IG_i = H(C) - H(C|X_i) \tag{8}$$

The ranking of the features is then done with respect to the values of IG_i in a descending order, reflecting the intuition that the higher an IG value is, the more information the corresponding feature has to offer regarding the class. A subset of M features, $M \leq N$, can thus be selected by choosing the first M in the ranking list.

4.3 Principal component analysis

PCA can be used to reduce the dimensionality of a dataset [3] by projecting the data of a size N onto the eigenvectors of their covariance matrix, with the largest M eigenvalues taken, $M \leq N$. Formally, the principal component $PC_i, i \in \{1, 2, ..., M\}$, is obtained by

$$PC_i = \sum_{j=1}^{M} b_{ij} X_j \tag{9}$$

where X_j is the jth original feature, and b_{ij} are the linear factors (i.e. eigenvectors) that are chosen so as to make the variance of the corresponding PC_i as large as possible. The resulting PC_i are uncorrelated and can be ranked according to the amount of variation in the original data that they account for. Typically, the subset of those first several resultant principal features accounts for most of the variation in the data set and hence are retained, with the remainder discarded.

5 Image classifiers

Multi-layer perceptron neural networks [14] and K-nearest neighbors (KNN) [3] are used to accomplish image classification, by mapping input feature patterns onto the underlying image class labels. For learning such classifiers, a set of training data is selected from the typical parts (see Fig. 2) of the *McMurdo* image, with each represented by a feature pattern and assigned with an underlying class label.

6 Experimental results

From the *McMurdo* image of Fig. 1, a set of 270 sub-divided non-overlap images, of a size of 512 × 512 each, are used to perform this experiment. 948 pixels are selected from 16 of them for training and verification, which are each labeled with an identified class index (i.e. one of the four image types: rock1, rock2, sand and gravel). The rest of all these images are used as unseen data for classification. Each training pixel is represented by a pattern of 24 features (see Section 3), and each testing pixel by a smaller number of features selected by a given feature selection tool. The performance of each classifier is measured using classification accuracy, with ten-fold cross validation [12].

For easy cross-referencing, Table 1 lists the reference numbers of all the original features, where $i = 1, 2, ..., 16$. In the following, for KNN classification, the results are first obtained with K set to 1, 3, 5, 8, and 10. For the MLP classifiers, to limit simulation cost, only those of one hidden layer are considered here with the number of hidden nodes setting to 8, 12, 16, 20, or 24. The classifier which has the highest accuracy, with respect to a given feature pattern dimensionality and a certain number of nearest neighbors or hidden nodes is then taken for performance comparison.

Table 1. Feature meaning and reference

No.	Meaning	No.	Meaning	No.	Meaning
1	Mean(GL)	4	STD(H)	7	Mean(V)
2	STD(GL)	5	Mean(S)	8	STD(V)
3	Mean(H)	6	STD(S)	9-24	H_i

6.1 Use of selected or full features

It is important to show that at least, the use of a selected subset of features does not significantly reduce the classification accuracy as compared to the use of the full set of original features. For the given training set, IGR ranks the original 24 features in the following descending order: Mean(V), Mean(GL), H_{16}, STD(S), Mean(S), STD(H), Mean(H), STD(V), STD(GL), H_{15}, H_2, H_5, H_{10}, H_8, H_7, H_9, H_3, H_6, H_4, H_{11}, STD(H), H_1,H_{12},H_{14}, H_{13} (i.e. features 7, 1, 24, 6, 5, 3, 8, 2, 23, 10, 13, 18, 16, 15, 17, 11, 14, 12, 19, 4, 9, 20, 22, 21).

Fig. 3 shows the classification accuracy, in relation to the use of IGR-selected features. Each color box indicates the result of using a different combination of classifier and the number of selected features. The right-most case shows the results of using all of the 24 original features. Clearly, the use of different selected feature subsets significantly affects the classification performance. Table 2 lists the top performers (based on Fig. 3). The number (N) of hidden nodes and that (K) of the nearest neighbors used by these MLP and KNN classifiers are also provided in (the first column of) this table.

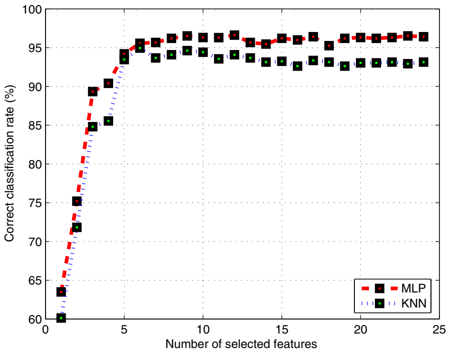


Figure 3. Performance of KNN and MLP in relation to the number of IGR-selected features.

Table 2. FRFS/IGR-selected vs. full features

Classifier	Set	Dim.	Feature No	Rate
MLP(N=20)	FRFS	9	$1, 2, 3, 4, 5, 6, 11, 16, 24$	97.00%
MLP(N=20)	IGR	9	$7, 1, 24, 6, 5, 3, 8, 2, 23$	96.52%
MLP(N=20)	IGR	12	$7, 1, 24, 6, 5, 3$ $8, 2, 23, 10, 13, 18$	96.62%
MLP(N=16)	Full	24	$1, 2, ..., 23, 24$	96.41%
KNN(K=8)	FRFS	9	$1, 2, 3, 4, 5, 6, 11, 16, 24$	94.10%
KNN(K=3)	IGR	9	$7, 1, 24, 6, 5, 3, 8, 2, 23$	94.62%
KNN(K=5)	IGR	5	$7, 1, 24, 6, 5$	93.46%
KNN(K=3)	Full	24	$1, 2, ..., 23, 24$	93.14%

The results demonstrate that the classifiers using IGR-selected features outperform those using the full set of original features. For instance, by employing just 5 or 9 top ranked features, the KNN has a classification accuracy of 93.46% or 94.62%, both beating the KNN that uses the full feature set (which has an accuracy of 93.14%). For MLP, the use of top 9 or 12 IGR-selected features can produce better results (96.52% or 96.62%) than that using the full set of features (96.41%), though only slightly. Importantly, such superior performance is achieved by much simpler classifier structures.

Using FRFS, the following 9 features are selected: Mean(GL), STD(GL), Mean(H), STD(H), Mean(S), STD(S), H_3, H_8, H_{16} (i.e. features $1, 2, 3, 4, 5, 6, 11, 16$

and 24 in Table 1), out of the original twenty-four. Table 2 lists the classification rates produced by the MLP and KNN classifiers that employ these 9 FRFS features, that is, 97.00% and 94.10% respectively. Both are higher than that of using the full set of original features. Again, such high performance is obtained by structurally much simpler classifiers.

6.2 FRFS/IGR/PCA-returned features

As one of the most popular methods for dimensionality reduction, PCA is adopted here as the benchmark for comparison. Fig. 4 shows the classification results of the KNN and MLP classifiers using a different number of principal features. Table 3 summarizes the top performers amongst the two types of classifiers, each using a certain number of PCA-returned features. For easy comparison, the results of those KNNs and MLPs which use 9 FRFS-selected or IGR-selected features are also included in this table, with the corresponding number (N) of hidden nodes and that (K) of nearest neighbors indicated.

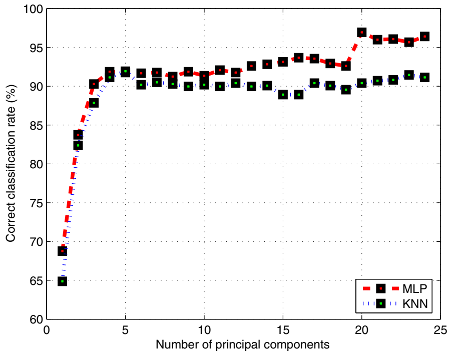


Figure 4. Performance of KNN and MLP vs. the number of principal components used.

Table 3. FRFS/IGR vs. PCA returned features

Classifier	Set	Dim.	Feature No	Rate
MLP(N=20)	FRFS	9	$1, 2, 3, 4, 5, 6, 11, 16, 24$	97.00%
MLP(N=20)	IGR	9	$7, 1, 24, 6, 5, 3, 8, 2, 23$	96.52%
MLP(N=12)	PCA	9	$1, 2, 3, 4, 5, 6, 7, 8, 9$	91.87%
MLP(N=16)	PCA	20	$1 - 20$	96.94%
MLP(N=20)	PCA	21	$1 - 21$	95.99%
MLP(N=20)	PCA	22	$1 - 22$	96.07%
MLP(N=20)	PCA	23	$1 - 23$	95.67%
MLP(N=20)	PCA	24	$1 - 24(full)$	96.41%
KNN(K=8)	FRFS	9	$1, 2, 3, 4, 5, 6, 11, 16, 24$	94.10%
KNN(K=3)	IGR	9	$7, 1, 24, 6, 5, 3, 8, 2, 23$	94.62%
KNN(K=8)	PCA	9	$1, 2, 3, 4, 5, 6, 7, 8, 9$	89.98%
KNN(K=3)	PCA	5	$1, 2, 3, 4, 5$	91.93%

These results show that, of the same dimensionality (i.e. 9), the MLP classifier that uses either FRFS or IGR selected features significantly outperforms its counterpart that uses PCA-returned features. Only when the number of principal features reaches 20 and 24, is the classification performance of MLPs comparable to that obtained by using the 9 FRFS or IGR selected features. Yet, this is at the expense of requiring many more feature measurements and much more complex classifier structures (in addition to the fact that implementing PCA itself incurs more computation than IGR and FRFS). As for KNN classifiers, those using 9 FRFS and IGR selected features outperform all of their corresponding counterparts which use any number of PCA-returned features (although Table 3 only presents the best results reachable by the latter).

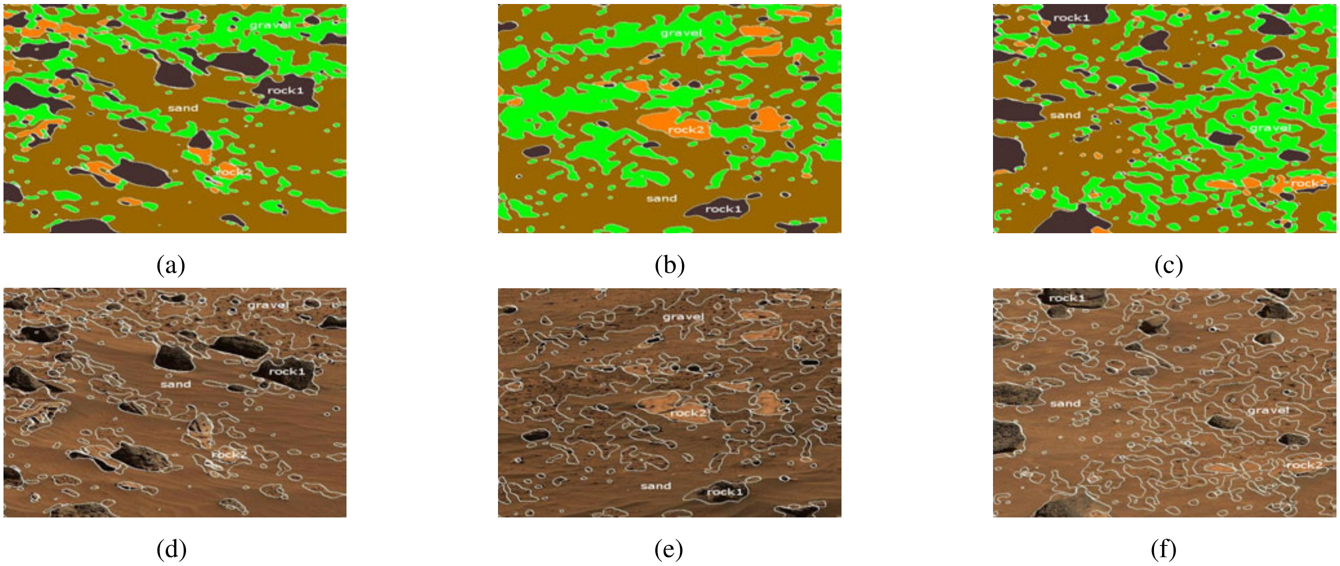
6.3 Classified and segmented images

The ultimate task of this research is to classify Mars panoramic camera images and to detect different objects or regions in such images. The above experimental evaluation provides a solid empirical grounding for the design of effective and efficient (MLP and KNN) classifiers. In particular, it is revealed that MLP performs the best and outstandingly. For instance, using 9 features selected by IGR and FRFS it can produce a classification rate of 96.52% and 97.00%, respectively. Based on this observation, the MLP classifier which employs the 9 FRFS-selected features is herein taken to accomplish the task of classifying the entire image of Fig. 1 (again, excluding the areas occupied by the instruments and their black shadows). As an illustration, three classified images are shown in Fig. 5, numbered by (a), (b) and (c) respectively, where four different colors represent the four image types (rock1, rock2, sand and gravel). From this, boundaries between different class regions can be identified and marked with white lines, resulting in the segmented images also given in Fig. 5, numbered by (d), (e) and (f) respectively.

From these classified images, it can be seen that all four image types vary in terms of their size, rotation, color, contrast, shapes, and texture. For human eyes it can be difficult to identify boundaries between certain different image regions, such as those between sand and gravel, and those between rock2 and sand. However, the classifier is able to perform under such circumstances, showing its robustness to image variations. Note that classification errors mainly occur within regions representing sand and gravel. This may be expected since gravel is itself a mixture of sand and small stones. Such errors are less important however, as the major attention for Mars image classification is to detect Martian rocks [16]. Almost all visible rocks on the image are correctly detected. Due to space limit, the segmented version of Fig. 1 is omitted here.

7 Conclusion

This paper has presented a study on Mars *McMurdo* panorama image classification. Effective feature selection mechanisms are employed in conjunction with MLP and



(a) (b) (c)

(d) (e) (f)

Figure 5. Classified and segmented images.

KNN classifiers to perform classification. Although the images encountered are complex, both types of classifier which use IGR or FRFS-selected features perform well (especially for the combined use of MLP and FRFS). This is supported with systematic comparative investigations, involving the use of more features or an equal number of features returned by principal component analysis. These results show the potential of feature selection in reducing redundant feature measures and also the noise associated with such measurement (as fewer features may even lead to higher classification accuracy). This, in combination with the observation that both IGR and FRFS preserve the underlying semantics of the selected features, also indicates that information loss can be minimized and even avoided in building the classifiers. Such work is of particular significance for on-board classification and analysis of large-scale images in future Mars rover missions [1].

References

[1] D. Barnes, S. Pugh, and L. Tyler. Autonomous science target identification and acquisition (astia) for planetary exploration. *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2009.

[2] Castano R. *et al.* Current results from a rover science data analysis system. *Proceedings of the IEEE Aerospace Conference*, 2006.

[3] R. Duda, P. Hart, and D. Stork. *Pattern Classification (2nd edition)*. Wiley & Sons, New York, 2001.

[4] http://marswatch.astro.cornell.edu/pancam_instrument/mcmurdo_v2.html.

[5] K. Huang and S. Aviyente. Wavelet feature selection for image classification. *IEEE Transactions on Image Processing*, 17:1709–1720, 2008.

[6] D. Iqbal and N. Lee. Efficient feature selection based on information gain criterion for face recognition. *Proceedings of International Conference on Information Acquisition*, pages 523–527, 2007.

[7] R. Jensen and Q. Shen. *Computational Intelligence and Feature Selection: Rough and Fuzzy Approaches*. IEEE Press and Wiley & Sons, 2008.

[8] T. Kachanubal and S. Udomhunsakul. Rock textures classification based on textural and spectral features. *Proceedings of World Academy of Science, Engineering and Technology*, 29:110–116, 2008.

[9] Kim W. *et al.* Rover-based visual target tracking validation and mission infusion. *AIAA Space 2005-6717*, 2005.

[10] L. Lepisto, I. Kunttu, and A. Visa. Rock image classification based on k-nearest neighbour voting. *Vision, Image and Signal Processing, IEE Proceedings*, 153(4):475–482, 2006.

[11] D. Martin, C. Fowlkes, and J. Malik. Learning to detect natural image boundaries using local brightness, color, and texture cues. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26:530–549, 2004.

[12] T. Mitchell. *Machine Learning*. McGraw Hill, 1997.

[13] D. Puig and M. Garcia. Automatic texture feature selection for image pixel classification. *Pattern Recognition*, 39:1996–2009, 2006.

[14] D. Rumelhart, E. Hinton, and R. Williams. Learning internal representations by error propagating. In D. Rumelhart and J. McClelland (Eds.). *Parallel Distributed Processing*, 1986.

[15] S. Sergyan. Color content-based image classification. *Proceedings of the 5th Slovakian-Hungarian Joint Symposium on Applied Machine Intelligence and Informatics*, pages 25–26, 2007.

[16] D. Thompson and R. Castano. Performance comparison of rock detection algorithms for autonomous planetary geology. *Proceedings of the IEEE Aerospace Conference*, paper no. 1251, 2007.