# Support Vector Machine-Based Classification of Rock Texture Images Aided by Efficient Feature Selection

Changjing Shang and Dave Barnes
Deptartment of Computer Science
Aberystwyth University, UK
Email: {cns, dpb}@aber.ac.uk

***Abstract*—This paper presents a study on rock texture image classification using support vector machines (and also K-nearest neighbours and decision trees) with the aid of feature selection techniques. It offers both unsupervised and supervised methods for feature selection, based on data reliability and information gain ranking respectively. Following this approach, the conventional classifiers which are sensitive to the dimensionality of feature patterns, become effective on classification of images whose pattern representation may otherwise involve a large number of features. The work is successfully applied to complex images. Classifiers built using features selected by either of these methods generally outperform their counterparts that employ the full set of original features which has a dimensionality several folds higher than that of the selected feature subset. This is confirmed by systematic experimental investigations. This study therefore, helps to accomplish challenging image classification tasks effectively and efficiently. In particular, the approach retains the underlying semantics of a selected feature subset. This is very important to ensure that the classification results are understandable by the user.**

## I. Introduction

Automated and accurate analysis of rock texture images is an important task, especially for surveying places (e.g. for geologic or life cues) on the Earth or in space [1], [19], [27], [29]. A key element of analysing rock images is to recognise the underlying rock types. However, rock images are in many cases non-homogenous and strongly directional. Also, the granular size and colour of rock textures may vary significantly in relation to their types, and may be blurred with measurement and transmission noise. These factors make rock texture image classification a challenging problem [18].

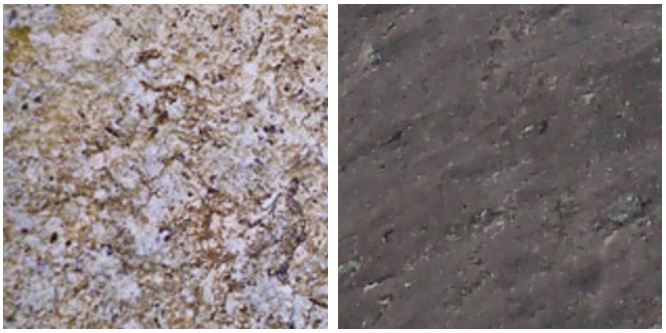
One critical step to successfully build an image classifier is to extract and use informative or predictive features from given images [9], [12], [17], [26]. To capture the essential characteristics of such images, many features may have to be extracted without explicit prior knowledge of what properties might best represent the underlying rock texture reflected by the original image. However, generating more features increases the computational complexity as well as the measurement noise. This may cause problems in many application domains such as on-board processing of Mars images where demand for computational memory and processing time must be minimised [27] (despite the nowadays generally available and relatively cheap computer power). In addition, not all such features may be useful to perform classification [12], [15], [23], [25]. 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 [16]. It is therefore, often necessary 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.

Recently, there have been significant advances in developing methodologies that are capable of minimising feature subsets in a noisy environment. In general, when decision labels are available, supervised feature selection methods usually outperform their unsupervised counterparts [2]. However, in many cases, the amount of labeled training samples may be limited as the thorough interpretation of the data available is infeasible. Therefore, it is necessary to develop unsupervised feature selection. Amongst the many techniques for unsupervised feature selection is the approach that is based on measuring data reliability (RFS) [4], [11]. This method uses the reliability measures to justify the relevance (or importance) of each feature to the problem at hand, and hence the possibility of being included in the selected feature subset. This has been shown to be a highly useful technique by which discrete or real-valued noisy data (or a mixture of both) can be effectively reduced. Yet, it has not been combined with learning mechanism to perform classification tasks in general and image analysis in particular. Of course, when information on underlying class labels is given, supervised feature selection may be employed to obtain better results.

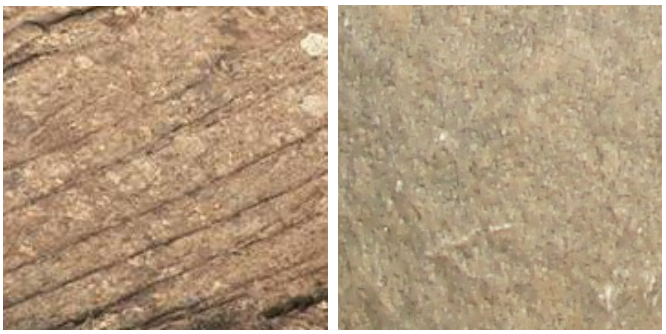
Inspired by this observation, this paper presents an integrated approach for performing rock texture image classification, by exploiting the potential of combining classification and feature selection techniques. In particular, support vector machines (SVMs) [30] are employed to accomplish rock classification. This is due to the recognition of their high generalisation performance in complex data sets [5], [30]. The (unsupervised) reliability measure based and (supervised) information gain based feature selection methods are utilised to ensure that classification is carried out with a substantially reduced subset of original features only. The resulting integrated approach helps to improve the effectiveness and efficiency of SVM-based image classifiers. This is because only those

informative features are required to be generated in improving the performance of the classifiers, minimising both the feature measurement noise and the computational complexity (of both feature extraction and feature pattern-based classification). Systematic comparative studies are carried out, including the use of alternative classification techniques (Decision Trees [22], and K-Nearest Neighbours [8]). Experimental results demonstrate that the proposed approach entails rapid and accurate learning of classifiers.

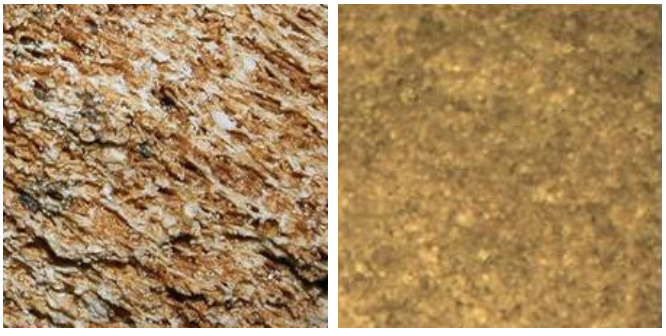
The rest of this paper is organised as follows. Section II introduces the rock texture images under investigation. Sections III, IV and V outline the key component techniques used in this work, covering feature extraction, feature selection and feature pattern classification, respectively. Section VI shows the experimental results, supported by comparative studies. The paper is concluded in Section VII.



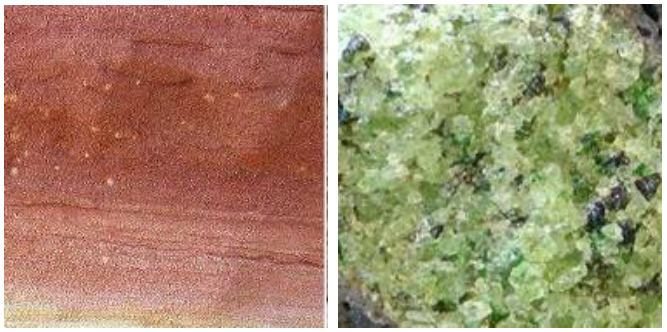
(1) rock-type 1 (2) rock-type 2



(3) rock-type 3 (4) rock-type 4



(5) rock-type 5 (6) rock-type 6



(7) rock-type 7 (8) rock-type 8

(9) rock-type 9 (10) rock-type 10

(11) rock-type 11 (12) rock-type 12

(13) rock-type 13 (14) rock-type 14

Fig. 1. Rock texture image types

## II. Rock Texture Images

The rock images considered are classified into 14 different texture types which are of practical significance, with each type having a representative as illustrated in Fig. 1.

These images reveal a tremendous amount of detail such as colour, structure, texture, grain size and orientation. The ultimate task of this research is to develop an image classifier that can recognise different image type or class in such texture image data, although the underlying approach taken is general.

## III. Feature Extraction

A variety of techniques may be used to capture and represent the underlying characteristics of a given image [12], [26]. In this work, low-level feature extraction approaches are employed due to their popularity and simplicity. In particular, local colour histograms and the first and second order colour statistics [5], [21] are exploited to produce a feature vector for each individual pixel. Such features are effective in depicting the underlying image characteristics and are efficient to compute. Also, the resulting features are robust to image translation and rotation, thereby potentially suitable for classification of the complex rock texture images concerned.

### A. Colour Statistics-Based Features

Images originally given in the RGB (Red, Green and Blue) colour space are first transformed to those in the HSV (Hue, Saturation and Value) space [21]. These spaces are in bijection with one another, and the HSV colour space is widely used in the literature. By computing the first order (mean) and the second order (standard deviation, denoted by STD) colour statistics with respect to each of the R, G, B, H, S and V channels, twelve features can be generated per pixel, from a certain neighbourhood of that pixel. For presentational simplicity, the resulting features are hereafter denoted as $MEAN_X$ and $STD_X$, $X \in \{\text{R, G, B, H, S, V}\}$, representing the first and second order statistics per colour space channel, respectively.

### B. Local Colore Histogram-Based Features

As the name indicates, such features are measured off the histograms computed from local regions of a given image [9]. In the present context, a histogram is a summary graph showing a count of grey levels falling in a number of resolution ranges (called bins), within a predefined neighbourhood. For a certain pixel, a set of histogram features $X_{hi}, i = 1, 2, ..., B$, where $X \in \{H, S, V\}$, are calculated (within the given neighbourhood), regarding a particular bin size $B$ (i.e. number of bins). Thus, the feature $X_{hi}$ represents the normalised frequency of the colour histogram in bin $i$ with respect to one of the H, S, and V colour channels. Here, for simplicity, individual bin widths are set equally, and the neighbourhood size is set to the same as that used in the above colour feature extraction. The bin size $B$ is empirically set to 8 in this work.

### C. Grey-level Statistics and Histogram-Based Features

Similarly, another set of local grey-level (GL) histogram features can be generated by first transforming colour images to GL ones. In this work, the bin size for computing GL histogram features is empirically set to 16. The resulting GL histogram-based features are denoted by $GL_{hi}$, $i = 1, 2, ..., 15, 16$. In addition, two further GL statistic features, mean and STD, which are denoted by $MEAN_{GL}$ and $STD_{GL}$, are also generated.

## IV. Feature Selection

Feature selection (FS) [7], [12], [15] addresses the problem of selecting amongst given features that are most informative or predictive. It forms a particular approach to the reduction of the number of features under consideration. Importantly, unlike conventional dimensionality reduction or general feature extraction methods, a feature selection algorithm preserves the original meaning of the selected features after reduction. The reminder of this section describes two approaches to implementing such a process of feature selection.

### A. Unsupervised Feature Selection Based on Data Reliability

*1) Nearest-Neighbour Guided Evaluation of Data Reliability:* The cluster-based approach to data reliability assessment has been popular in the literature [20], [28]. However, it requires high computational efforts: with the time and space complexity being $O(N^3)$ and $O(N^2)$, respectively (where $N$ is the number of input features) [4]. This problem can be alleviated by taking a recently proposed technique in measuring data reliability, as summarised below.

For a collection of data arguments $A = \{a_1, \ldots, a_N\}$, let $N^k_{a_i}$ be a set of $k$ nearest neighbours of an argument $a_i$, where $N^k_{a_i} \subset A, n_j \in N^k_{a_i}, n_j \neq a_i, j = 1 \ldots k$. The reliability of a specific argument can be determined by the distance of this argument to the members of its nearest neighbour set. Such a distance can be found using the *FindNearestNeighbor* algorithm given in Fig. 2. The higher this distance is, the less reliable that argument becomes. The following distance metric is used to measure the distance between two given arguments, for computational simplicity (any other distance metric may be applied if they do not incur too much overheads in computation):

$$d(a_i, a_j) = |a_i - a_j| \tag{1}$$

Given the distance metric, the reliability $R^k_{a_i}$ of argument $a_i$ depends on the average distance $D^k_{a_i}$ to its $k$ nearest neighbours (i.e. members of $N^k_{a_i}$), which is identified as

$$D^k_{a_i} = \frac{1}{k} \sum_{\forall n_t \in N^k_{a_i}} d(a_i, n_t) \tag{2}$$

Following this, the reliability measure $R^k_{a_i} \in [0, 1], i = 1 \ldots n$ can be obtained such that

$$R^k_{a_i} = 1 - \frac{D^k_{a_i}}{D_{max}} \tag{3}$$

where $D_{max} = \max_{a_p, a_q \in A, a_p \neq a_q} d(a_p, a_q)$.

This reliability measure is more efficient comparing to the existing cluster-based method since no data clustering process is required. The time and space complexities generally decrease by an order, to $O(N^2)$ and $O(N)$, respectively [4].

**FindNearestNeighbour**$(a_i, k, A)$

$A$, set of arguments, $a_i \in A, i = 1 \ldots N$;
$k$, number of nearest neighbours, $1 \leq k < N$;
$N_{a_i}^k$, set of $k$ nearest neighbours of $a_i$, $N_{a_i}^k \subset A$;
$|N_{a_i}^k|$, size of neighbour set, $0 \leq |N_{a_i}^k| \leq k$;
$d(a_i, a_j)$, distance between arguments $a_i, a_j \in A$;
$maxN$, neighbour $n_p \in N_{a_i}^k, d(a_i, n_p) = \max_{\forall n_t \in N_{a_i}^k} d(a_i, n_t)$;

(1) $N_{a_i}^k \leftarrow \emptyset$
(2) **for each** $a_j \in A$
(3) **if** $a_j \neq a_i$
(4) **if** $|N_{a_i}^k| < k$
(5) $N_{a_i}^k \leftarrow N_{a_i}^k \cup a_j$
(6) **else if** $d(a_i, a_j) < d(a_i, maxN)$
(7) $N_{a_i}^k \leftarrow (N_{a_i}^k - maxN) \cup a_j$
(8) **return** $N_{a_i}^k$

Fig. 2. The *FindNearestNeighbour* algorithm.

In the extreme case of $k = N - 1$, the time complexity becomes a linear function of $O(n)$ as well because no search for nearest neighbours is needed.

*2) Reliability-based Method for Feature Selection:* The above reliability measure can be applied to the problem of unsupervised feature selection. This is because it can be regarded as the discriminant factor to justify the relevance of each data feature to the problem at hand. The resulting method reflects the intuition that a feature is considered reliable (or relevance to the problem) if its values are tightly grouped together. In essence, with a dataset of $n$ samples $(x_1, \ldots, x_n)$, the reliability $FR_r$ of feature $f_r, r \in \{1, \ldots, N\}$ is estimated from the accumulative reliability measures generated for each of its values $f_{ir}, i = 1 \ldots n$:

$$FR_r = \sum_{i=1}^{n} R_{f_{ir}}^k \tag{4}$$

where the reliability measure $R_{f_{ir}}^k$ of each feature value $f_{ir}, i = 1 \ldots n$ is computed using Equations 2 and 3, given the set of $k$ nearest neighbours.

The higher the reliability, the more relevant the feature to the given problem. Thus, the original features can be ranked in accordance with their reliability degrees. A subset of $M$ most reliable features, $M \leq N$, can therefore be selected by choosing the first $M$ in the rank list. For simplicity, the reliability-based method for feature selection will be referred to as RFS hereafter.

### B. Information Gain-based Feature Ranking

Let $D_f$ be the value set of feature $f$ and $D_c$ be the label set of class variable $c$. The entropies of the class before and after observing $X$ are respectively defined by [13], [22]:

$$H(c) = -\sum_{u \in D_c} p(u) log_2 p(u) \tag{5}$$

$$H(c|f) = -\sum_{x \in D_f} p(x) \sum_{u \in D_c} p(u|x) log_2 p(u|x) \tag{6}$$

The amount by which the entropy measure over a certain class decreases after observing a certain feature reflects the additional information about the class that feature provides, and is called the information gain:

$$IG = H(c) - H(c|f) \tag{7}$$

It estimates how well a given feature separates data points with respect to their underlying class labels. Thus, all extracted features $f_i$, $i = 1, 2, ..., N$, can be ranked with regard to the $IG$ values of observing themselves:

$$IG_i = H(c) - H(c|f_i) \tag{8}$$

Such ranking can be arranged in descending order, reflecting the fact that the higher an $IG$ value is, the more information the corresponding feature has to offer regarding the class. A subset of $M$ most informative features, $M \leq N$, can therefore be selected by choosing the first $M$ in the rank list.

## V. Image Classifiers

### A. Support Vector Machine-based Classifiers

Support Vector Machines (SVMs) [30] are used to perform image classification here, mapping input feature vectors onto the underlying image class labels. Such a classifier seeks to find the optimal separating hyperplane among different classes by focusing on those training points (named support vectors), which are placed at the edge of the underlying features and whose removal would change the solution to be found.

More formally, SMVs construct a hyperplane in a space of a dimensionality higher than that of the original. The intuition is that by mapping the original data space into a much higher-dimensional space, the class separation between data points will become easier in that space. SVMs use a specific mapping such that the cross products of data points in the larger space are defined in terms of a kernel function [6] which is selected to suit the given problem. In so doing, the cross products may be computed in terms of the features in the original space, thereby minimising computational effort. In particular, a hyperplane in the higher dimensional space is defined as the set of points whose inner product with any vector in that space is constant.

A good hyperplane is learned over a training process such that the resulting hyperplane has the largest distance to the nearest training data points of any given class. This is in order to increase the discriminative power of the trained classifier. In the following, the Radial Basis function (RBF) kernel

is adopted to implement the SVM-based classifiers, and the popular sequential minimal optimisation algorithm is used to train the SVMs. Detailed SVM learning mechanism is omitted, but can be found in the literature (e.g. [24], [30]).

In order to increase the efficacy of the SVM classifiers, RFS is used to rank the extracted features and hence, to select those most reliable during the training phase. This is of practical significance as SVMs are proven high-performance classifiers, but their efficiency relies on effective minimisation of input features. Aiding SVMs with RFS-based feature selection helps to minimise the input dimensionality. For learning such classifiers, a set of training data is selected from the typical images (see Fig. 1) of identified rocks, with each pixel represented by a feature vector which is manually assigned an underlying class label.

### B. K-Nearest Neighbours (KNN)

K-nearest neighbours (KNN) algorithm [8], [18] is one of the simplest and most popular learning methods for building classifiers. To classify an unclassified feature vector $X$, KNN ranks the neighbours of $X$ amongst a given set of $P$ data $(X_i, c_i)$, $i = 1, 2, ..., P$, and uses the class labels $c_j$ $(j = 1, 2, ..., K)$ of the $K$ most similar neighbours to predict the class of the new vector $X$. In particular, the classes of these neighbours are weighted using the similarity between $X$ and each of its neighbours, where similarity is typically measured by the Euclidean distance metric (though any other distance metric may also do). Given such similarity measures, $X$ is assigned the class label with the greatest number of votes amongst the $K$ nearest class labels.

Note that the core of a KNN learning process bears close similarity to the *FindNearestNeighbour* algorithm, though it is the training vectors that are considered here rather than the features (or arguments) addressed there. KNN works without relying on prior probabilities, and that it is computationally efficient if the dimensionality of the input features is not very large. If however, the dimensionality is high, each distance calculation may become quite expensive. This reinforces in general, the need for employing a feature selection tool such as RFS to extend its capacity.

### C. Decision Trees

A decision tree (DTREE) based image classifier is learned from a set of training examples through an iterative process, of choosing a feature and splitting the given example set according to the values of that feature. The key question here is which of the features is the most influential in determining the classification and hence should be chosen first. Entropy measures or equivalently, information gains are used to select the most influential, which is intuitively deemed to be the feature of the lowest entropy (or of the highest information gain).

Essentially, the learning algorithm works by: a) computing the entropy measure for each feature, b) partitioning the set of examples according to the possible values of the feature that has the lowest entropy, and c) for each subset of examples repeating these steps until all features have been partitioned or other given termination conditions met. The C4.5 algorithm [22] is used to build decision trees in this research.

## VI. Experimental Results

### A. Experimental Setup

A set of 206 rock texture images with a size of $128 \times 128$ each are used to perform this experiment. 2744 pixel points are selected from 14 of these images for training and verification, 196 from each image type. Each of the pixels is labeled with an identified class index (i.e. one of the 14 image classes as illustrated in Fig. 1). Each training pixel is originally represented by a pattern vector of 54 features (see Section III). The size of a neighbourhood window used for extracting features is empirically set to $15 \times 15$. Of course, the actual classification process only uses subsets of selected features.

For evaluation, the performance of each classifier is measured using classification accuracy, with ten-fold cross validation. For easy cross-referencing, Table I lists the reference numbers of the original features that are extracted, where $i \in \{1, 2, ..., 7, 8\}$ (the empirically chosen bin size for colour channels) and $j \in \{1, 2, ..., 15, 16\}$ (the empirically chosen bin size for grey-level image). The SVM penalty parameter is set to 100, with standard Gaussian Radial Basis function (RBF) employed, for all experiments.

| No. | Meaning | No. | Meaning | No. | Meaning |
|---|---|---|---|---|---|
| 1 | $MEAN_R$ | 2 | $STD_R$ | 3 | $MEAN_G$ |
| 4 | $STD_G$ | 5 | $MEAN_B$ | 6 | $STD_B$ |
| 7 | $MEAN_H$ | 8 | $STD_H$ | 9 | $MEAN_S$ |
| 10 | $STD_S$ | 11 | $MEAN_V$ | 12 | $STD_V$ |
| 13-20 | $H_{hi}$ | 21-28 | $S_{hi}$ | 29-36 | $V_{hi}$ |
| 37 | $MEAN_{GL}$ | 38 | $STD_{GL}$ | 39-54 | $GL_{hj}$ |

TABLE I
Feature meaning and reference number

Note that in the following, for KNN classification, the results are first obtained with K set to 1, 3, 5, 8, and 10. Those classifiers which have the highest accuracy, with respect to a given feature pattern dimensionality and a certain number of nearest neighbours, are then taken to run for performance comparison.

### B. Comparison with the Use of Full Original Features

This subsection shows 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.

*1) Use of RFS-selected features:* For the given training data, RFS ranks the original 54 features in the following descending order as listed in Table II (see Table I and section III for the meaning of these features). Fig. 3 shows the classification accuracy, in relation to how many top-ranked features (based on RFS) are used. The right-most case is the result of using all of the 54 original features.

| Order | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| RFS | 54 | 53 | 2 | 12 | 10 | 5 | 21 | 3 | 9 | 29 | 11 |
| Order | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 |
| RFS | 13 | 4 | 37 | 1 | 39 | 38 | 6 | 40 | 7 | 20 | 45 |
| Order | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 |
| RFS | 36 | 8 | 49 | 22 | 28 | 41 | 46 | 52 | 33 | 42 | 18 |
| Order | 34 | 35 | 36 | 37 | 38 | 39 | 40 | 41 | 42 | 43 | 44 |
| RFS | 43 | 25 | 23 | 32 | 35 | 44 | 31 | 47 | 34 | 30 | 19 |
| Order | 45 | 46 | 47 | 48 | 49 | 50 | 51 | 52 | 53 | 54 | |
| RFS | 14 | 26 | 48 | 17 | 51 | 50 | 24 | 15 | 27 | 16 | |

TABLE II
RANKING ORDER OF RFS-SELECTED FEATURES

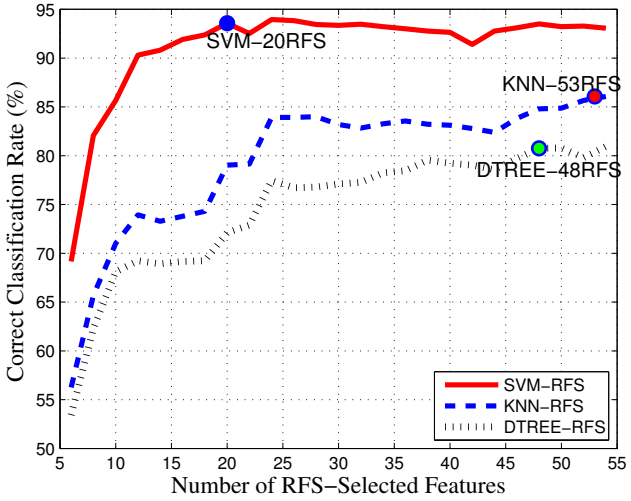



Fig. 3. Performance of SVMs, KNNs and DTREEs vs. the number of RFS-selected features.

Table III summarises the correct classification rates produced by the SVM, DTREE and KNN classifiers, all with 10-fold cross validation, where the number of the nearest neighbours K used by these KNN classifiers are also provided (in the first column).

| Classifier | Set | Dim. | Feature No | Rate |
|---|---|---|---|---|
| SVM | RFS | 20 | First 20 RFS | 93.58% |
| SVM | RFS | 24 | First 24 RFS | 93.95% |
| SVM | Full | 54 | 1, 2, ..., 53, 54 | 93.07% |
| KNN(K=8) | RFS | 53 | First 53 RFS | 86.07% |
| KNN(K=5) | Full | 54 | 1, 2, ..., 53, 54 | 86.07% |
| DTREE | RFS | 48 | First 48 RFS | 80.78% |
| DTREE | Full | 54 | 1, 2, ..., 35, 54 | 80.94% |

TABLE III
RFS-SELECTED VS FULL ORIGINAL FEATURES

For SVM classifiers, the classification accuracy of using the first 20 RFS-selected features is higher than that of using all 54 original features (93.58% vs. 93.07%). For DTREE classifiers, the accuracy resulting from using the first 48 RFS-selected features is very close to that from using the full set of original features (80.78% vs. 80.94%). However, for KNN classifiers, only when 53 original features are used, i.e. only one out of 54 features may be ignored, can the accuracy reach that is achievable when the full set of original features are employed (80.07% vs. 80.07%). Overall, the combined use of SVM and RFS techniques offers the best performance, with a classification rate of 93.95% using 24 RFS-selected features. This is indicative of the potential of RFS in reducing not only redundant feature measurements but also the noise associated with such measurements, improving both effectiveness and efficiency of the classification process.

Table IV presents detailed break-downs of classification results per class when the SVM classifiers are employed, with respect to the two popular performance indices: precision and recall. For a given class $Class$, precision and recall are defined as follows:

$$Prec. = \frac{|Ps\ Correctly\ Classified\ into\ Class|}{|Ps\ Correctly/Incorrectly\ Classified\ into\ Class|} \quad (9)$$

$$Recall = \frac{|Ps\ Correctly\ Classified\ into\ Class|}{|Ps\ in\ Class|} \quad (10)$$

where $Ps$ stands for pixels and $|X|$ for the number of $X$. These measures are important to check whether the proposed approach works if the distribution of individual classes is not uniform, as with the present problem. These results are again based on 10-fold cross validation. They show that the SVM classifier that uses only 20 RFS-selected features has similar rates as the SVM which uses all 54 original features, regarding these performance indices. For the two most difficult classes: C1 and C2, both classifiers attain reasonable results. Although the performance of the SVM that uses the subset of RFS-selected features drops a little on these two classes overall, the recall rate for C1 actually improves.

| | Use of Full Features | | Use of RFS-selected | |
|---|---|---|---|---|
| Class | Precision | Recall | Precision | Recall |
| C1 | 80.6% | 82.7% | 78.1% | 85.7% |
| C2 | 84.6% | 84.2% | 82.5% | 74.5% |
| C3 | 90.1% | 88.3% | 92.1% | 95.4% |
| C4 | 84.9% | 83.2% | 87.2% | 83.2% |
| C5 | 90.0% | 91.8% | 96.4% | 94.9% |
| C6 | 98.0% | 98.5% | 98.5% | 99.0% |
| C7 | 99.5% | 99.5% | 100.0% | 100.0% |
| C8 | 97.9% | 95.4% | 96.9% | 95.9% |
| C9 | 99.0% | 99.0% | 99.5% | 100.0% |
| C10 | 100.0% | 99.0% | 99.5% | 99.0% |
| C11 | 89.8% | 89.8% | 91.3% | 91.3% |
| C12 | 94.5% | 96.4% | 93.7% | 98.0% |
| C13 | 100.0% | 99.5% | 100.0% | 99.5% |
| C14 | 94.5% | 95.9% | 94.8% | 93.9% |

TABLE IV
PRECISION AND RECALL PER CLASS: RFS-SELECTED VS FULL ORIGINAL FEATURES

The results of Tables III and IV conjunctively form a significant case in support of the proposed approach. They are

also indicative of the potential of RFS in reducing not only redundant feature measurements but also the noise associated with such measurements, improving both effectiveness and efficiency of the classification process in general.

*2) Use of IGR-selected features:* IGR ranks the original 54 features in the descending order as listed in Table V (see Table I for the meaning of these features). Fig. 4 shows the classification accuracy in relation to how many top-ranked features (by IGR) are used. Three plots are presented, in correspondence to the use of three different types of classifier: SVM, KNN and DTREE. The right-most cases are the results of using all of the 54 original features. Clearly, the use of feature subsets of a different cardinality significantly affects the classification performance. For comparison, Fig. 4 also includes the accuracy rates achieved by the three types of classifier that each uses 20 (SVM), 53 (KNN) and 48 (DTREE) RFS-selected features, respectively.

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Order | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 |
| IGR | 10 | 9 | 8 | 7 | 24 | 16 | 27 | 25 | 6 | 18 | 17 |
| Order | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 |
| IGR | 14 | 4 | 3 | 38 | 5 | 15 | 37 | 23 | 11 | 30 | 12 |
| Order | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 |
| IGR | 2 | 1 | 19 | 42 | 26 | 31 | 35 | 51 | 22 | 34 | 50 |
| Order | 34 | 35 | 36 | 37 | 38 | 39 | 40 | 41 | 42 | 43 | 44 |
| IGR | 43 | 52 | 41 | 49 | 28 | 44 | 32 | 46 | 47 | 48 | 45 |
| Order | 45 | 46 | 47 | 48 | 49 | 50 | 51 | 52 | 53 | 54 | |
| IGR | 33 | 53 | 40 | 21 | 29 | 36 | 13 | 20 | 54 | 39 | |

TABLE V
RANKING ORDER OF IGR-SELECTED FEATURES

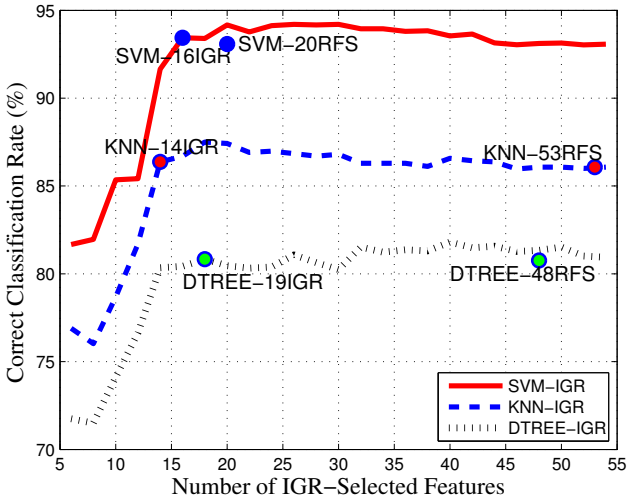



Fig. 4. Performance of SVMs, KNNs and DTREEs vs. the number of IGR-selected features.

Table VI summaries the correct classification rates produced by the classifiers, based on the results of Fig. 4. Also included in this table are the numbers of IGR-returned features that are necessary to be used for the corresponding classifiers to reach a comparative outomce as per those classifiers which use RFS-selected features. As with the case of using features returned by RFS, SVM-based classifiers using IGR-selected features perform the best.

| Classifier | Set | Dim. | Feature No | Rate |
|---|---|---|---|---|
| SVM | RFS | 20 | First 20 RFS | 93.58% |
| SVM | IGR | 16 | First 16 IGR | 93.44% |
| SVM | Full | 54 | 1, 2, ..., 53, 54 | 93.07% |
| KNN(K8) | RFS | 53 | First 53 RFS | 86.07% |
| KNN(K8) | IGR | 14 | First 14 IGR | 86.37% |
| KNN(K5) | Full | 54 | 1, 2, ..., 53, 54 | 86.07% |
| DTREE | RFS | 48 | First 48 RFS | 80.78% |
| DTREE | IGR | 18 | First 18 IGR | 80.83% |
| DTREE | Full | 54 | 1, 2, ..., 35, 54 | 80.94% |

TABLE VI
RFS AND IGR-SELECTED VS FULL ORIGINAL FEATURES

Collectively, these results show that although far less features are employed, the classifiers using reduced feature subsets outperform their counterparts which use the full features. The employment of IGR in these classifiers helps not only to reduce redundant feature measurements (thereby simplifying classification process), but also to minimise the noise associated with such measurements (thereby improving the classification accuracy).

### C. Comparison between RFS and IGR

The performances of using IGR or RFS to support the different types of classifier are herein further compared. It can be seen from the results shown in the preceding section that the SVM classifier with the first 16 IGR-selected features has similar performance (93.44%) as the SVM with the first 20 RFS-selected features (93.58%). Both beat that which is achievable by the use of the full set of original features (93.07%). Thus, both approaches can significantly reduce the number of features that are required for SVM to perform.

Interestingly, the KNN and DTREE classifiers that employ IGR-selected features require a much smaller number of features than their counterparts which use RFS-selected features, in order to reach the classification accuracy that is close to what is achievable using full features. Yet, neither type of classifier is able to obtain a classification rate so high as the SVM-based. Overall, for the present image dataset, the three types of classifier that employ supervised IGR provide similar (for SVM) or better (for KNN and DTREE) results in terms of classification rate and number of features used, comparing to those using the unsupervised RFS.

These results indicate that if it is feasible to make use of supervised feature selection then IGR should be used; otherwise, use RFS. This can be expected, as in general, when decision labels are available during the training phase, supervised feature selection methods usually outperform their unsupervised counterparts [2]. Yet, in many cases where the thorough interpretation of a large data is infeasible, the amount of labeled training samples is often limited. In such circumstances, most conventional supervised techniques may fail on

the 'small labeled-sample problem' [14]. Besides, information required about the underlying classes of the image patterns may not be available for certain applications, it will therefore be necessary to employ the unsupervised feature selection techniques such as RFS. It is very encouraging to note that for the problem at hand, when SVM-based classifiers are used with RFS-selected features the best performer is very similar to that when IGR-selected features are employed, both in classification accuracy and in number of features required.

## VII. Conclusion

This paper has presented a study on rock texture image classification using support vector machines (and also K-nearest neighbours and decision trees) with the aid of feature selection techniques. It has offered both unsupervised and supervised methods for feature selection, based on data reliability and information gain ranking respectively.

Following this approach, the conventional classifiers which are sensitive to the dimensionality of feature patterns, become effective on classification of images whose pattern representation may otherwise involve a large number of features. Although the images encountered are complex, the resulting feature pattern dimensionality of selected features is manageable. Classifiers built using such selected features generally outperform their counterparts that employ the full set of original features which has a dimensionality several folds higher than that of the selected feature subsets. This is confirmed by systematic experimental investigations.

This work helps to accomplish challenging image classification tasks effectively and efficiently. Unlike transformation-based dimensionality reduction techniques [15], the approach retains the underlying semantics of the selected feature subset. This is very important to ensure that the classification results are understandable by the user. It is of particular significance for classification and analysis of real-world images in the areas of medical analysis and space engineering [27]. Indeed, such applications remain as active research.

The present research suggests the use of SVMs to carry out classification. However, such classifiers are known to be effective. It would be useful to further compare their performance with other well-known systems such as multi-layer feedforward neural networks, in addition to the K-nearest neighbours and decision tree based classifiers. It is also very interesting to compare the present work with the approach that performs classification tasks by directly using data reliability measures [3].

## References


[1] J. Autio, L. Rantanen, A. Visa and S. Lukkarinen, *The Classification of Rock Texture Analysis by Co-occurrence Matrices and the Hough Transform*, Proc. of International Symposium on Imaging Applications in Geology, pp. 5-8, 1999.

[2] C. Bishop, *Neural Networks for Pattern Recognition*, Oxford,U.K.: Oxford Univ. Press, 1995.

[3] T. Boongoen, C. Shang, N. Iam-On and Q. Shen, *Extending Data Reliability Measure to a Filter Approach for Soft Subspace Clustering*, IEEE Transactions on Systems, Man and Cybernetics, Part B: Cybernetics, Vol. 41(6), pp. 1705-1714, 2011.

[4] T. Boongoen and Q. Shen, *Nearest-neighbor Guided Evaluation of Data Reliability and Its Applications*, IEEE Transactions on Systems, Man and Cybernetics, Part B: Cybernetics, Vol. 40(6), pp. 1622-1633, 2010.

[5] O. Chapelle, P. Haffner and V. Vapnik, *Support Vector Machines for Histogram-based Image Classification*, IEEE Tran. on Neural Networks, Vol. 10, pp. 1055-1064, 2001.

[6] N. Cristianini and J. Shawe-Taylor, *An Introduction to Support Vector Machines and other Kernel-based Learning Methods*, Cambridge University Press, 2000.

[7] M. Dash, H. Liu, and J. Yao, *Dimensionality Reduction of Unsupervised Data*, in Proceedings of the 9th IEEE International Conference on Tools with Artificial Intelligence, pp. 532-539, 1997.

[8] R. Duda, P. Hart and D. Stork, *Pattern Classification*. (2nd edition). Wiley & Sons, New York. 1997.

[9] R. Gonzalez, R. Woods, *Digital Image Processing and Scene Analysis* (3rd Edition), Prentice Hall, 2008.

[10] J. Handl and J. Knowles, *Feature Subset Selection in Unsupervised Learning via Multiobjective Optimization,* Int. J. Comput. Intell. Res., Vol. 2, no. 3, pp. 217-238, 2006.

[11] Y. Hong, S. Kwong, Y. Chang and Q. Ren, *Consensus Unsupervised Feature Ranking from Multiple Views,* Pattern Recognit. Lett., Vol. 29, no. 5, pp. 595-602, 2008.

[12] K. Huang and S. Aviyente, *Wavelet Feature Selection for Image Classification*, IEEE Trans. Image Proc, Vol. 17, pp. 1709-1720, 2008.

[13] D. Iqbal and N. Lee , *Efficient Feature Selection Based on Information Gain Criterion for Face Recognition,* Proc. Int. Conf. on Information Acquisition. pp523-527, 2007

[14] A. Jain and D. Zongker, *Feature Selection: Evaluation, Application, and Small Sample Performance,* IEEE Trans. Pattern Anal. Mach. Intell., Vol. 19, pp. 153-158, 1997.

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

[16] R. Jensen and Q. Shen, Are More Features Better? *IEEE Transactions on Fuzzy Systems*. Vol 17, no. 6, pp1456-1458, 2009.

[17] T. Kachanubal and S. Udomhunsakul, *Rock Textures Classification Based on Textural and Spectral Features,* Proc. of World Academy of Science, Eng. and Tech, Vol. 29, pp110-116, 2008.

[18] L. Lepisto, I. Kunttu and A. Visa, *Rock Image Classification Based on K-nearest Neighbour Voting*, Vis. Im. and Sig. Proc., IEE Proc. Vol. 153(4), pp. 475-482, 2006.

[19] L Lepist, I. Kunttu and A. Visa , *Rock Image Classification Using Color Features in Gabor Space*, Journal of Electronic Imaging , Vol. 14 (4), 2005

[20] H. Liu and L. Yu, *Toward integrating feature selection algorithms for classification and clustering,* IEEE Trans. Knowl. Data Eng., Vol. 17, no. 4, pp. 491C502, Apr. 2005.

[21] D. Martin, C. Fowlkes and J. Malik, *Learning to Detect Natural Image Boundaries Using Local Brightness, Color, and Texture Cues*, IEEE Trans. Patt. Anal. and Mach. Inte, Vol. 26, pp. 530-549, 2004.

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

[23] C. Perez, P. Estevez, P. Vera, L. Castillo, C. Aravena, D. Schulz and L. Medina, *Ore Grade by Feature Selection and Voting Using Boundary Detection in Digital Image Analysis,* Int. J. Mineral Processing., Vol. 101, pp. 28-36, 2011.

[24] J. Platt, *Macines Using Sequential Minimal Optimisation. in Advances in Kernel Methods: Support Vector Learning*. MIT Press Cambridge, USA. pp185-208, 1999.

[25] D. Puig and M. Garcia, *Automatic Texture Feature Selection for Image Pixel Classification*, Pattern Recognition, Vol. 39, pp. 1996-2009, 2006.

[26] T. Randen, J. Husoy and S. Stavanger, *Filtering for Texture Classification: A Comparative Study*, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 21 pp. 291 - 310, 1999.

[27] C. Shang, D. Barnes and Q. Shen. *Facilitating Efficient Mars Terrain Image Classification with Fuzzy-rough Feature Selection*, International Journal of Hybrid Intelligent Systems, Vol. 8(1), pp. 3-13, 2011

[28] L. Talavera, *Feature Selection as Retrospective Pruning in Hierarchical Clustering,* in Proceedings of the 3rd International Symposium on Intelligent Data Analysis. Springer, Amsterdam, The Netherlands, 1999.

[29] D. Thompson and R. Castano, *Performance Comparison of Rock Detection Algorithms for Autonomous Planetary Geology*, in Proc. of IEEE Aerospace Conf. 2007

[30] V. Vapnik, it Statistical Learning. New Yark, Wiley, 1998.