Computers & Geosciences 106 (2017) 10–17

Contents lists available at ScienceDirect

Computers & Geosciences

journal homepage: www.elsevier.com/locate/cageo

Research paper

Unsupervised feature learning for autonomous rock image classification

Lei Shu[a,b,*], Kenneth McIsaac[a,b], Gordon R. Osinski[b,c], Raymond Francis[d]

[a] Department of Electrical and Computer Engineering, University of Western Ontario, 1151 Richmond St, London, ON, Canada N6A 5B7
[b] Centre for Planetary Science and Exploration, University of Western Ontario, 1151 Richmond St, London, ON, Canada N6A 5B7
[c] Department of Earth Science, Physics and Astronomy, University of Western Ontario, 1151 Richmond St, London, ON, Canada N6A 5B7
[d] Jet Propulsion Laboratory, California Institute of Technology, 4800 Oak Grove Dr, Pasadena, CA 91109 United States

ARTICLE INFO

Keywords:
Unsupervised feature learning
Self-taught learning
Autonomous rock classification

ABSTRACT

Autonomous rock image classification can enhance the capability of robots for geological detection and enlarge the scientific returns, both in investigation on Earth and planetary surface exploration on Mars. Since rock textural images are usually inhomogeneous and manually hand-crafting features is not always reliable, we propose an unsupervised feature learning method to autonomously learn the feature representation for rock images. In our tests, rock image classification using the learned features shows that the learned features can outperform manually selected features. Self-taught learning is also proposed to learn the feature representation from a large database of unlabelled rock images of mixed class. The learned features can then be used repeatedly for classification of any subclass. This takes advantage of the large dataset of unlabelled rock images and learns a general feature representation for many kinds of rocks. We show experimental results supporting the feasibility of self-taught learning on rock images.

1. Introduction

1.1. Background and motivation

Autonomous geological detection is becoming an increasingly important technique for robotic platforms exploring remote environments such as Mars (e.g. Francis et al. (2014a), (2014b)). It can maximize the scientific return and reduce the need for human involvement. In the case of Mars specifically, the bandwidth limit and large time delay (3–22 min one-way travel time) of data transmission makes autonomous techniques even more critical and valuable. The past two decades have seen tremendous achievements in Mars exploration. Among them are Mars Exploration Rovers (MER) and Mars Science Laboratory (MSL) missions. Both missions sent rovers to the surface of Mars and explored their respective regions of interest with various scientific instruments. Two autonomous onboard systems have been developed for these rovers: the Onboard Autonomous Science Investigation System(OASIS) (Castano et al., 2004, 2007, 2008), and the Autonomous Exploration for Gathering Increased Science(AEGIS) system (Estlin et al., 2009, 2012). Both systems are actively used and have enabled the rovers to autonomously identify and react to serendipitous science opportunities by analyzing imagery onboard with computer vision techniques. Tasks included locating rocks in the images, analyzing rock properties, and identifying rocks that merit further investigation through autonomous selection and sequencing of targeted observations. However, the rovers still heavily rely on explicit instructions given by scientists on Earth, which requires extensive communication and frequent command cycles. As such, there is still a long way to go before rovers will possess sufficient "intelligence" to reason about science goals, make informed decisions, and respond to discoveries autonomously (Francis et al., 2014b).

An alternative approach to AEGIS and OASIS is increasingly being used in geosciences in the form of computer vision. For example, Chanou et al. (2014) and Pittarello and Koeberl (2013) developed and applied quantitative image analysis methods to analyze the images of individual rock samples. In these approaches, components or particles of a rock image are first segmented, which then allows the measurement and quantification of various properties, such as shape complexity, preferred orientation, size-frequency, and so on. A different advanced technique that we focus on here is rock image classification (Shang and Barnes, 2012). Instead of the exact quantitative measurement of particles in rock images, the approach of rock image classification is to identify the specific type of rock(s) based on visual appearance. The identification of rock type is important as this provides information as to the environment in which the rock was created and its subsequently geological history (Gor et al., 2001). For example, the size of crystals in igneous rocks can be used to estimate cooling rates and provides constraints on the depth of formation; the

[*] Corresponding author at: Department of Electrical and Computer Engineering, University of Western Ontario, 1151 Richmond St, London, ON, Canada N6A 5B7.
E-mail address: lshu2@uwo.ca (L. Shu).

http://dx.doi.org/10.1016/j.cageo.2017.05.010
Received 26 March 2016; Received in revised form 3 February 2017; Accepted 18 May 2017
Available online 25 May 2017

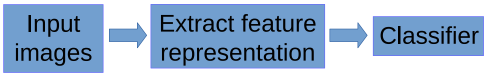

Fig. 1. The typical framework of image classification.

grain size and shape of sedimentary rocks provides information as to the mode of deposition; and the properties of rocks formed by meteorite impact craters reflects the pressure and temperature of formation and of the environment prior to impact. As such, autonomous rock classification has the potential to provide valuable information about the origin and evolution of rocky planetary bodies throughout the Solar System.

1.2. Related work

A typical framework of image classification (see Fig. 1) includes extracting feature representation for input images and feeding the feature representation into a classifier. In general, the performance of image classifiers is heavily dependent on the selection of a feature representation. Unfortunately, rock textures are seldom homogeneous. As a result, the design of a feature representation is difficult, which makes rock image classification extremely challenging. There have been a few attempts at developing feature representation for rock image classification to date. All these previous works use either hand-engineered features manually selected for the specific application, or automatically selected features chosen using time-consuming methods.

Prior works mostly involve manually selected features. In order to reduce the time-consuming process of manual identification of rock samples, Ślipek and Młynarczuk (2013) and Młynarczuk and Górszczyk (2013) conducted autonomous classification of microscopic images of rocks by four pattern recognition methods - nearest neighbour, k-nearest neighbours (k-NN), nearest mode, and optimal spherical neighbourhoods. Sharif et al. (2015) built a small library of grayscale images from a total of 30 hand samples, and used Bayesian analysis to classify them with selected Haralick textural features (Haralick et al., 1973). In order to distinguish adjacent outcrops, Francis et al. (2014a) started with some fundamental visual “channels” such as colour and difference between colour channels, then utilized multi-class linear discriminant analysis (MDA) to identify the principal visual components. Harinie et al. (2012) utilized Tamura features (Tamura et al., 1978) to classify hand samples of rocks into the three major categories, namely, igneous, sedimentary and metamorphic. Dunlop (2006) studied features such as shape, albedo, colour and textures, then conducted rock classification with different feature combinations. Singh et al. (2004) compared 7 well-established image texture analysis algorithms for rocks classification and the results suggested that Law's masks (Laws, 1980) and co-occurrence matrices (Haralick et al., 1973) were best. Lepistö et al. (2003) classified rock images by methods based on textural and spectral features. The spectral features are some colour parameters and the textural features are calculated from the co-occurrence matrix. In order to improve the classification accuracy, Lepistö et al. (2005) combined colour information in Gabor space (Tou et al., 2007) to the texture description. Given that various visual descriptors extracted from images are often high dimensional and non-homogenous, Lepistö et al. (2006b) conducted rock images classification based on k-nearest neighbour voting, which combined k-NN base classifiers for different descriptors by voting. A similar idea of combining base classifiers came to Lepistö et al. (2006a). Each feature descriptor had a corresponding separate base classifier, and better classification accuracy can be achieved by combining opinions provided by each base classifier.

Other works have concentrated on feature selection. Chatterjee (2013) used the genetic algorithm to select features, and then classified limestone with multi-class SVM (Support Vector Machine). Shang and Barnes (2012) utilized a reliability-based method and mutual information to select features, then classified rocks images in a more general dataset. Both works showed that their own feature selection methods worked well in their dataset, but feature selection itself is time-consuming. When the dataset becomes complicated, one might have to think of what kind of feature pool to select from, or even devising a brand new feature representation.

All the previous representations used for rock images consist either of an entirely manually crafted feature set or a set of features automatically selected from a set of manually crafted features. These manual features are not good enough to represent inhomogeneous rock images and are time-consuming to get. Our proposed methods address this deficiency by automatically learning the feature representations. Our experimental results demonstrate that the learned feature representations have the potential to be more flexible and powerful.

1.3. Introduction to this study

We have approached the problem of feature selection for geological classification in two ways in this paper. First, we propose an unsupervised feature learning technique (Coates et al., 2011) to extract features for rock images. The approach is to autonomously learn the feature representation from a large amount of data rather than manually choosing the features. This has the benefit of making the feature representation much more flexible when using different datasets. The feature learning method we utilized is based on K-means (Coates and Ng, 2012), which is fast and easily implemented. We applied this method to the classification of rock images with SVM (Support Vector Machine). (Both K-means and SVM are described below).

The second autonomous feature selection method we propose in this paper is called self-taught learning (Raina et al., 2007; Wang et al., 2013). The concept behind self-taught learning is to learn a feature representation from unlabelled images of *mixed-class* and then train a classifier on a *subset of the data* that has been labelled to identify certain subclasses represented within the original data set. For image classification, having enough labelled images is important. Basically, the more images you have, the better learning you get. However, it is usually difficult and expensive to label images. Though researchers have resorted to tools such as AMT (Amazon Mechanical Turk) to have a large number of people help with labelling, there are still financial costs and concerns about the quality of labelling. Thus the ability to use *unlabelled images* would greatly enhance an autonomous feature identification technique. In addition, it is highly unlikely that a particular dataset will only contain the classes of the images we are interested in. It is much more likely that a dataset will comprise a mix of all kinds of possible rock classes. As such, we utilized self-taught learning to directly learn feature representation from unlabelled rock images of mixed-class and then applied the feature representation to labelled rock images which we are interested in for classification. In such an approach, the unlabelled images do not have to follow the same distribution as the labelled images, and the labelled images for classification can belong to merely subclasses of the unlabelled images (Raina, 2009). This attribute is particularly important for applications such as planetary exploration where the potential rock types will be uncertain.

Below, we first present the rock image dataset. Next we provide background on the set of manually selected features, the K-means feature learning approach and the self-taught learning approach. Finally, we show the effects of parameter selection for the feature learning methods as well as the results of classification with both the manual features and both types of learned features.

2. Rock image dataset

We photographed 9 different types of rock hand samples to generate a rock image dataset. The samples are provided by Department of Earth Science in Western University. These rocks are

Table 1
9 types of rocks used for classification in this study.

Rock types	Description
Limestone	It is a sedimentary rock consisting largely of calcium carbonate. It is light grey and smooth to touch.
Volcanic breccia	It is formed from angular gravel and boulder-sized clasts cemented together in a matrix. The angular nature of the clasts indicates that they have not been transported very far from their source. The texture is coarse-grained. Clasts are poorly sorted.
Oolitic limestone	It is made up mostly of ooliths which are sand-sized carbonate particles that have concentric rings of calcium carbonate. The colour is grey and texture is fine grained and porous.
Dolostone	It is a sedimentary carbonate rock that contains a high percentage of the mineral dolomite. It has a stoichiometric ratio of nearly equal amounts of magnesium and calcium.
Rhyolite	It is a silica-rich volcanic rock. Its texture is porphyritic and very compact. The groundmass with varying amounts of glass is also dense and fine grained. The colour is light reddish.
Granite	It is a felsic plutonic rock. It contains high percentage of light coloured constituents and low percentage of dark minerals. So the colour is basically light and texture is phaneritic. The size of the individual constituents is very varied.
Andesite	It is an extrusive rock intermediate in composition between rhyolite and basalt. It is basically grey and lighter coloured than basalt. Texture is porphyritic and interweaved. The groundmass is fine grained and glassy.
Peridotite	It is a very dense and coarse-grained igneous rock. The colour is generally dark greenish-grey and the texture is phaneritic. It is olivine-rich and has low silica content and very little feldspar.
Red granite	It has an equigranular texture with much pink orthoclase, grey quartz and biotite. It is coarse grained and the grains are developed enough to be recognised by the naked eye.

randomly selected. Table 1 lists all these rocks and the brief descriptions. Each type of rock reveals different appearance such as colour, structure, texture and grain size.

A dataset of approximately 700 textural images was generated from these 9 different types of rocks. There are roughly 80 images in total for each type of rock. Each image has size of 128 × 128 × 3 pixels and is between 1 and 2 cm across in reality (Fig. 2). Note that, the scale of the rocks was not accurately measured. Thus, we didn't research on how different scales will affect the classification in this study.

3. Methods

Our first set of experiments compares the behaviour of a Support Vector Machine (SVM) classifier using two different feature sets: Manually selected features, and autonomously selected features. To provide the needed background, in the first two parts of this section, we present the two competing feature sets (manual features and features learned based on K-means) and the background of the SVM classifier.

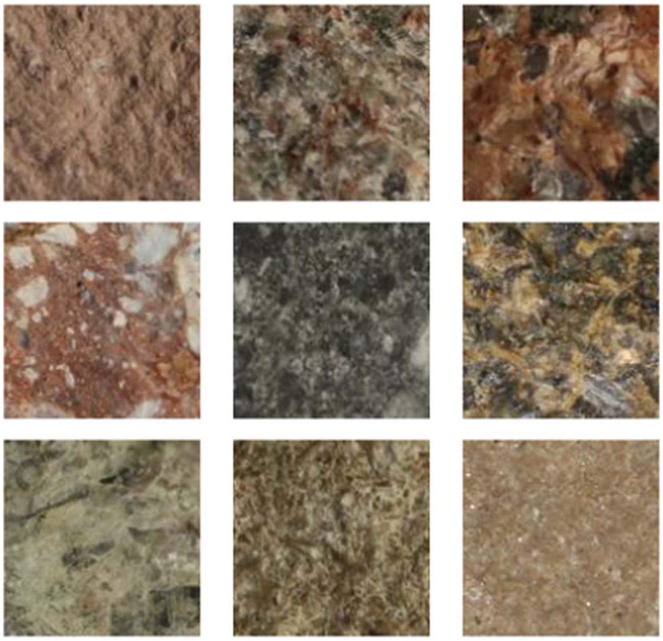

Fig. 2. Sample images from dataset. Each sample stands for one class. All images have size of 128 × 128 × 3 pixels and are between 1 and 2 cm across. From top to bottom, the first column – rhyolite, volcanic breccia, limestone; the second column – granite, andesite, oolitic limestone; the third column – red granite, peridotite, dolostone.

The final part of this section presents the needed background to the concept of self-taught learning, which is used in our second set of experiments.

3.1. Feature representations

3.1.1. Manual features

Among manual features, texture is commonly used to describe and represent the rock images (Lepistö et al., 2003; Paclík et al., 2005). It is determined by the way in which the grey levels are distributed over the pixels and describes an image as orderly or coarse, smooth or irregular, homogeneous or inhomogeneous (Shang and Barnes, 2012). It has been shown that first and second order statistics of texture can reasonably provide a small number of relevant and distinguishable features (Aggarwal and Agrawal, 2012). Note that, these features do not have any geological meaning. They are carefully hand-crafted by computer scientists.

First-order statistics describes the pixel intensity distribution of the image. If $I(i)$ stands for intensity of pixel i in image, and N is the number of pixels in the whole image, then five of the first-order statistics can be represented as

- Mean

$$\bar{I} = \frac{1}{N}\sum_{i=1}^{N} I(i) \tag{1}$$

- Median – intensity value which separates the higher half of pixel intensity from the lower half
- Standard deviation

$$I_{sd} = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(I(i)-\bar{I})^2} \tag{2}$$

- Skewness

$$skewness = \frac{\frac{1}{N}\sum_{i=1}^{N}(I(i)-\bar{I})^3}{\left(\sqrt{\frac{1}{N}\sum_{i=1}^{N}(I(i)-\bar{I})^2}\right)^3} \tag{3}$$

- Kurtosis

$$kurtosis = \frac{\frac{1}{N}\sum_{i=1}^{N}(I(i) - \bar{I})^4}{\left(\sqrt{\frac{1}{N}\sum_{i=1}^{N}(I(i) - \bar{I})^2}\right)^4} \tag{4}$$

Here, *skewness* is used to measure the symmetry of the histogram distribution and *kurtosis* is used to describe the flatness of the histogram distribution.

Second-order Statistics describes the information about relative positions of the various intensities. It is calculated from a grey level co-occurrence matrix (GLCM), which describes how frequently two grey levels of pixels appear (Haralick et al., 1973). For two-dimensional images, there are four grey level co-occurrence matrices in total, by orientation of 0°, 45°, 90°, 135° respectively. Among the second-order statistics calculated from GLCM, four of them are used in this paper,

- Angular Second Moment (ASM)

$$ASM = \sum_{I_1, I_2} P(I_1, I_2)^2 \tag{5}$$

- Entropy

$$Entropy = -\sum_{I_1, I_2} P(I_1, I_2) logP(I_1, I_2) \tag{6}$$

- Contrast

$$Contrast = \sum_{I_1, I_2} |I_1 - I_2|^2 logP(I_1, I_2) \tag{7}$$

- Correlation

$$Correlation = \sum_{I_1, I_2} \frac{(I_1 - \mu_1)(I_2 - \mu_2)P(I_1, I_2)}{\delta_1\delta_2} \tag{8}$$

Here, $P(I_1, I_2)$ is the frequency of co-occurrence matrix (Haralick et al., 1973). Four directions(i.e., 0°, 45°, 90°, 135°) are averaged out, which could make it rotate invariantly. *ASM* is to measure the smoothness or uniformity of the image region. *Entropy* is to measure the disorderliness. *Contrast* is a measure of local level variations which takes high values for image of high contrast. *Correlation* is a measure of correlation between pixels in two different directions.

With both first and second order statistics, there will be 9 textural features. If calculated in all colour channels, there will be 27 features in total for each image. In order to show how different feature configurations affect classification results, both the whole feature set and 4 subsets were used to represent the images during our experiments. The configurations of these feature sets are denoted as following.

- MF I – all first and second order statistics;
- MF II – only first-order statistics;
- MF III – only second-order statistics;
- MF IV – 5 features including Mean, Skewness, Kurtosis, Entropy, Correlation;
- MF V – 5 features including Skewness, Kurtosis, ASM, Entropy, Correlation.

3.1.2. Unsupervised feature learning based on K-means

Among various feature learning methods (Lee et al., 2006; Le et al., 2011), an approach based on K-means has previously been identified as fast and easily implemented (Coates and Ng, 2012). The basic framework is as follows. First, n random sub-patches (i.e., small red squares on *Input image* in Fig. 4) are extracted from the unlabelled dataset. Each sub-patch has a size of $w \times w \times d$ pixels, where w refers to receptive field size (width of sub-patch) and d is the number of colour channels. Hence, in terms of pixel intensity, the extracted patches can be represented as vectors $(x_1, \ldots, x_n)$ in $\mathbb{R}^N$, with $N = w \cdot w \cdot d$. Next, K-means (Kanungo et al., 2002) algorithm is used to generate K centroids $C_1, \ldots, C_K$, where each centroid C_i is also a vector in $\mathbb{R}^N$. All these centroids are the feature filters for the whole dataset and represent a basis set for all images. Fig. 3 shows the 60 centroids learned from training dataset of rock images. The parameter configuration is provided in Table 2. It is worthwhile to note here that these extracted features may not intuitively make much sense, and do not necessarily represent geologically meaningful properties. However, on visual inspection, some of them (e.g., the one in first row and third column) are clearly to detect edges, which are fundamental elements for rock textural images.

With K learned centroids, each patch x_j can be mapped to a $\mathbb{R}^K$ vector $f(x_j)$, and each element of the vector can be represented as

$$f_i(x_j) = \max\{0, \mu - z_i\}, \ 1 \le i \le K \tag{9}$$

where, $z_i = \| x_j - C_i \|_2$, μ is the mean of all z_i. Essentially, this operation represents all image patches as a weighted combination of the set of identified features. The max operation is used so patches "too far" from a particular centroid are treated as independent of that centroid.

The above steps explain how to transform an input patch from $\mathbb{R}^N$ to $\mathbb{R}^K$. With this transformation complete, we can now extract a representation of an entire image by applying the transformation to many sub-patches in the whole image. The framework of this process is shown in Fig. 4. The whole image is first cropped into many sub-patches of the same size ($w \times w \times d$) with stride (step size) s. If we assume that the entire image has a size of $a \times b \times d$ and stride s is 1, then there will be $(a - w + 1)*(b - w + 1)$ sub-patches in total mapped to $\mathbb{R}^K$ for each image. After reducing the feature dimensionality with pooling (e.g., split the y^{ij} into four equal-sized quadrants and compute the sum of the y^{ij} in each quadrant), each image will be represented as a feature vector $[\phi_1, \phi_2, \ldots, \phi_{4K}]^T$ in the same feature space.

3.2. Classification method

Support Vector Machine(SVM) (Cortes and Vapnik, 1995; Burges, 1998) was used to classify rock images with both manual features and learned features. It is one of the most powerful and widely used methods for classification. It learns a hyperplane or set of hyperplanes in high-dimensional space, which separate data points with the largest margin.

The rock classification here is a multi-class classification problem. Thus, we used one-vs-all linear SVM in the experiments. The objective function is L2-SVM as shown in formula (10), where w defines the classifier and ξ_i is the slack variable to deal with outliers of data. We used L2-SVM regularization term because it is differentiable and imposes a bigger loss for data which violate the margin (Koshiba and Abe, 2003).

$$\begin{aligned} &\min \frac{1}{2}\| w \|^2 + \frac{C}{2}\sum_{i=1}^{n} \xi_i^2 \\ &\quad s.t.\ \ y_i(w^T x_i + b) \ge 1 - \xi_i, \ i = 1, \ldots, n \\ &\quad \xi_i \ge 0, \ i = 1, \ldots, n \end{aligned} \tag{10}$$

The regularization parameter C was determined by 5-fold cross-validation. In 5-fold cross-validation, we first divided the training set into 5 subsets of equal size. Sequentially each subset was tested using the classifier trained on the remaining 4 subsets. Thus, the cross-validation accuracy is the percentage of images which are correctly classified in all of the subsets. Various C values were tried and the one with the best cross-validation accuracy was picked. The training is based on the demonstration code from Coates et al. (2011).

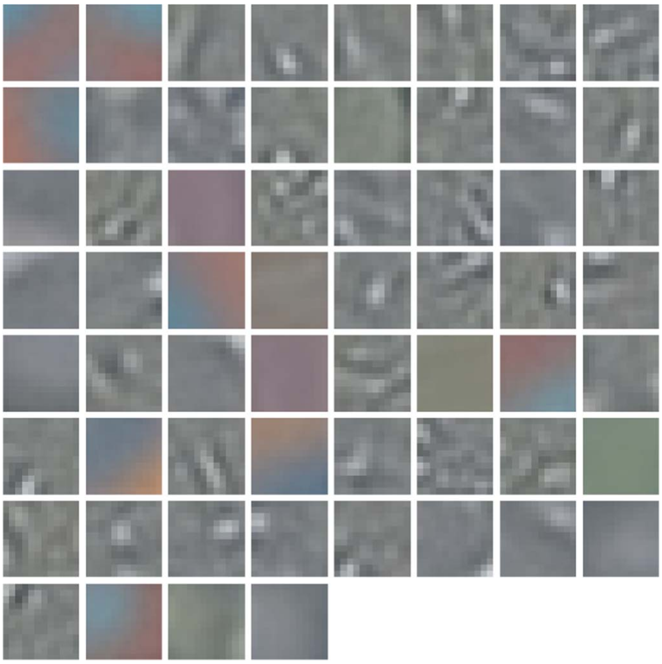

Fig. 3. 60 centroids learned from training dataset, each centroid has size of 12 × 12 × 3, configuration of parameters refers to Table 2.

Table 2
Parameter configuration for feature learning. stride – step size between two adjacent sub-patches, rfsize – receptive field size, K – number of centroids, numPatches – number of sub-patches extracted for training.

stride	1
rfsize	12
K	60
numPatches	50,000

3.3. Self-taught learning

The method discussed above, using K-means clustering and SVM, requires a fully labelled data set. Unfortunately, in machine learning settings, unlabelled datasets are significantly easier to obtain than labelled ones (Duda and Hart et al., 1973; Blum and Mitchell, 1998). Thus, our goal in this second part of the work was to utilize an algorithm (known as *self-taught learning*) that is able to take as much advantage of unlabelled data as possible. Self-taught learning consists of two stages (Raina et al., 2007). In the first step, a feature representation is learned from *unlabelled images*. In the second step, the learned features are then used to train a classifier on a *smaller, labelled data set*. Once the general feature representation has been learned in the first stage, it can be used repeatedly for different classification tasks. Unlike semi-supervised learning (Zhu et al., 2003), we do not assume that the unlabelled images were drawn from the same distribution as labelled images. Hence, self-taught learning is more general and powerful (Wang et al., 2013).

As an example, suppose the task is to identify basalt and limestone. It is time-consuming to gather a large dataset which consists only of basalt and limestone. A more common case and an easier approach is to collect a dataset containing various types of rocks (e.g., breccia, gneiss, sandstone, etc.) including basalt and limestone, be it from the internet or manually photography. Given that labelling images is costly (requires time of a trained geologist), it would be inefficient to label all images in the dataset if the goal is only the identification of basalt and limestone. The self-taught learning approach would be to use the entire large initial data set for feature learning. The large size of the initial data set means that we learn a generic set of features about rocks of all types. Then, we apply the learned feature representation to another small labelled dataset containing *only* basalt and limestone to train a classifier. The reason why this approach works is that the other types of rocks contain some basic visual patterns ("basic elements" as mentioned in Raina et al. (2007)) similar to ones in basalt and limestone, such as edges. Therefore, self-taught learning learns how to represent images in terms of these basic elements. By applying this learned representation to labelled images, we can obtain a higher level representation of labelled data as well, thus an easier supervised learning task.

The formalism of self-taught learning is as follows. We have unlabelled dataset X_u of n examples drawn from k classes, $X_u = \{x_1^{C_1}, \ldots, x_i^{C_j}, \ldots, x_n^{C_k}\}$. In addition, there is also a set of m labelled examples $X_l = \{x_1^{C_u}, \ldots, x_i^{C_v}, \ldots, x_m^{C_w}\}$. These labelled examples come from classes $\{C_u, \ldots, C_v, \ldots, C_w\}$, which is just a subset of original classes, i.e. $\{C_u, \ldots, C_v, \ldots, C_w\} \in \{C_1, \ldots, C_j, \ldots, C_k\}$. The task is to learn feature representation from X_u, and then apply the feature representation to X_l for further classification. Fig. 5 shows the general framework.

4. Experimental design

We conducted experiments to test both approaches discussed in this paper. To test the performance of unsupervised feature learning, we first compared classification performance using various combinations of manual features (i.e. first and second order statistics) and with classification performance using a feature set learned using the K-means approach. Fig. 6 shows how we separated the dataset into a training set and a testing set. For all manual features and unsupervised feature learning, the whole dataset was split into 70% as training data and 30% as testing data.

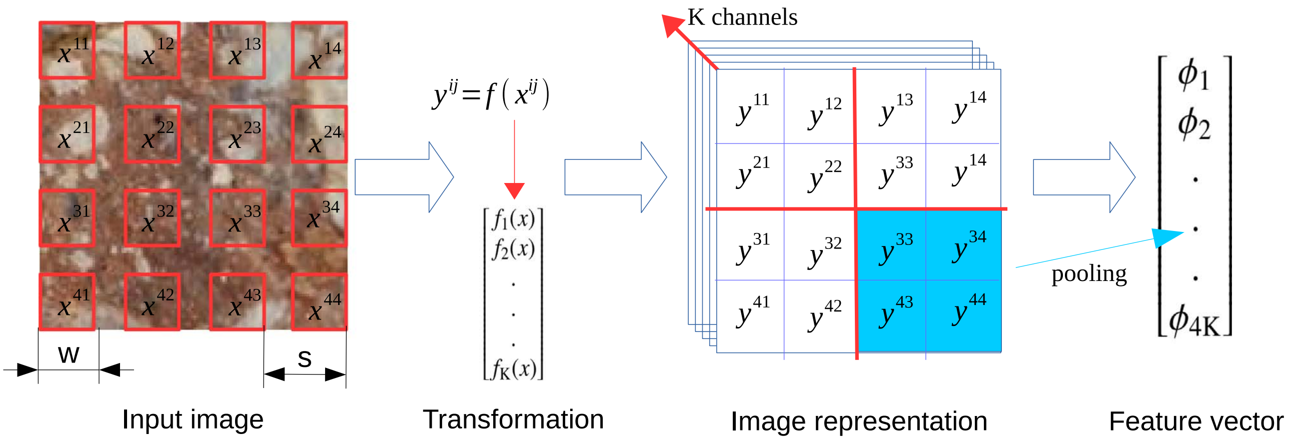
x^{11} x^{12} x^{13} x^{14}
x^{21} x^{22} x^{23} x^{24}
x^{31} x^{32} x^{33} x^{34}
x^{41} x^{42} x^{43} x^{44}
W
S
$y^{ij} = f(x^{ij})$
$f_1(x)$
$f_2(x)$
$f_K(x)$
K channels
y^{11} y^{12} y^{13} y^{14}
y^{21} y^{22} y^{33} y^{14}
y^{31} y^{32} y^{33} y^{34}
y^{41} y^{42} y^{43} y^{44}
pooling
ϕ_1
ϕ_2
ϕ_{4K}
Input image
Transformation
Image representation
Feature vector

Fig. 4. Framework of representing a rock image with learned features.

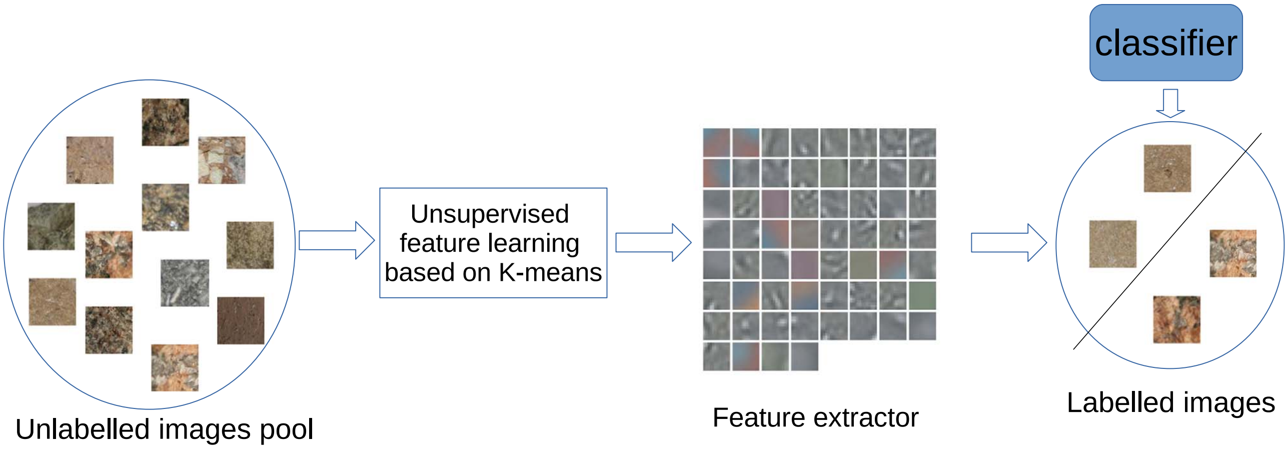

Fig. 5. Framework of self-taught learning.

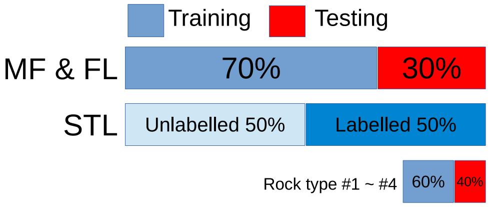

Fig. 6. Dataset separation. MF-manual features, FL-feature learning, STL-self-taught learning.

In the second part of the work, we explore the concept of self-taught learning. In this case, the whole initial dataset was split half-and-half into a labelled part and an unlabelled part (“unlabelled” here means we ignored the labels). We used the unlabelled half of the data to perform the feature identification step. In the labelled half, only samples from rock type #1 ~rock type #4 (which are rhyolite, volcanic breccia, limestone and granite) were picked to classify, and this sub-dataset was separated as 60% for training and 40% for testing.

5. Results and discussion

5.1. Parameters for unsupervised feature learning

There are several tunable parameters for unsupervised feature learning, such as the receptive field size (rfsize), the step size (stride) and the number of centroids (K) (Coates and Ng, 2012). Below, we present the results of experiments that investigated how these parameters affect the performance and how to choose these parameters.

5.1.1. Number of centroids

We extracted feature representations with 10, 15, 20, 30, 45 and 60 centroids and fixed the receptive field size (6 pixels) and stride (1 pixel). Fig. 7 clearly shows that the test accuracy generally goes up as the number of the centroids (K) increases. This is reasonable because a larger dictionary of feature bases is usually better able to capture structures and patterns inherent in the images. This is consistent with the work of Van Gemert et al. (2008) and Coates et al. (2011), who also observed that learning large numbers of features can substantially improve supervised classification results. As such, it is best to set K as large as computing resources will allow.

5.1.2. Stride

Stride is the space between sub-patches where features will be extracted (See Fig. 4). In this experiment, we fixed the number of centroids (60) and receptive field size (6 pixels), and then chose the stride over 1, 2, 4 and 8 pixels (Fig. 8). There is a clear downward trend in performance with increasing step size as expected. The smaller stride is able to cover more details in the images, so it will provide a better representation of the images. However, a small stride is also computationally more expensive. Thus, as with our recommendations for the

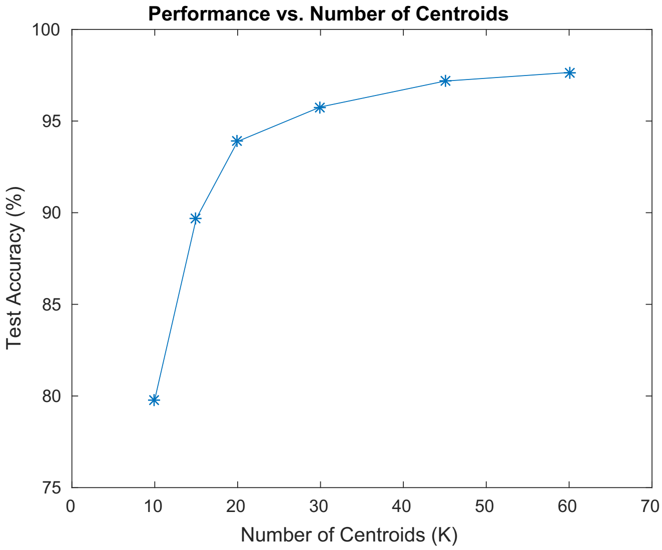

Fig. 7. Performance vs. number of centroids, rfsize=6, stride=1.

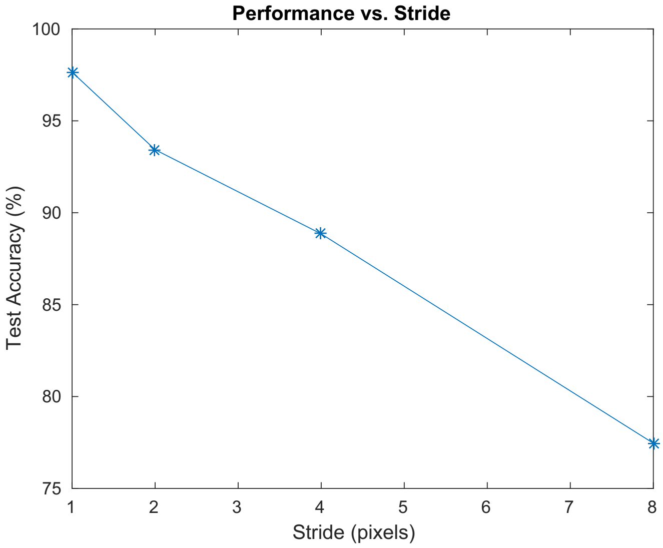

Fig. 8. Performance vs. stride, K=60, rfsize=6.

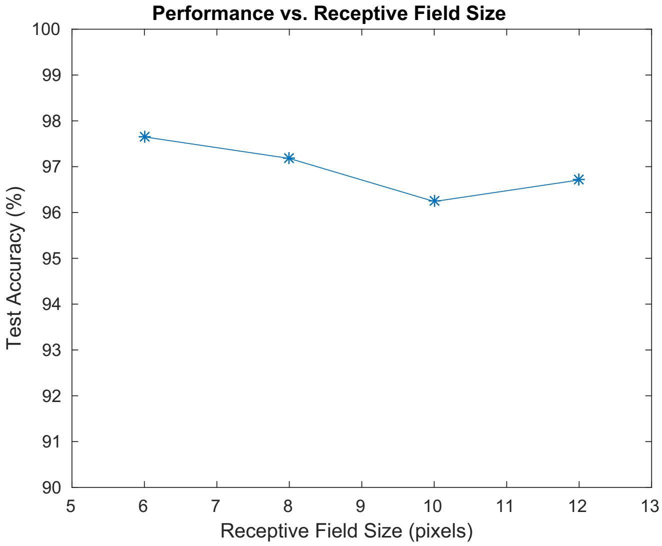

Fig. 9. Performance vs. receptive field size, K=60, stride=1.

number of centroids, it is best to set stride as small as compute resources will allow.

5.1.3. *Receptive field size*

Receptive field size represents the size of an area in an image from which the features are extracted. In general, a larger receptive field size should result in the learning of more complex features that cover a larger region of the images. However, this will increase the dimensionality of the feature space (see Section 3.1.2) and may require the learning of more feature bases or require more images in order to get the same performance. We evaluated the effect of receptive field size by testing it on 6, 8, 10 and 12 pixels. For the other parameters, we used stride of 1 pixel and 60 centroids. Fig. 9 shows that all the numbers performed similarly and the 6 pixels outperformed others slightly. It is unclear as to whether there is, or could be, a general rule for choosing the receptive field size. However, given that the small receptive field size produces a low dimensionality of the feature space, which in turn reduces computation and also that our experiment showed the small size can work reasonably well, it is suggested to use a small receptive field size if the computing resources allow a large K and a small stride.

5.2. *Performance comparison*

We used one group of parameters (Table 2) for unsupervised feature learning, and compared the classification result based on it with the results based on using manual features. Among the parameters, numPatches is the number of extracted sub-patches for learning features. The value of it depends on the size of dataset and the size of individual image sample.

Table 6 shows the testing accuracy for all the methods. Note that accuracy varies considerably for different combinations of manual features. MF I has accuracy as high as 96.24%, while MF III only achieves 66.20%. It is apparent that, for our dataset, pure first order statistics (MF II) outperforms second order statistics (MF III) considerably in representing rock images – 95.77% VS 66.20%. However, adding second order statistics to first order statistics can further improve the performance (from 95.77% (MF II)) to 96.24% (MF I). The goal, however, was not to compare the manual feature combinations and see which one provides the best result, but rather to show how much variability there is in the results depending on the features used. As it turns out, even slight changes in feature combinations may cause large difference in performance, such as MF IV (92.02%) and MF V (74.18%). So, it clearly indicates that manually selecting appropriate features is difficult.

Table 3
Performance of different methods. MF-manual features, FL-feature learning, STL-self-taught learning.

Features	Test accuracy
MF I	**96.24%**
MF II	95.77%
MF III	66.20%
MF IV	92.02%
MF V	74.18%
FL	**96.71%**
STL	**90.32%**

In this case, one may resort to automatic feature selection methods such as filter and wrapper methods (Chandrashekar and Sahin, 2014). With these feature selection methods, one may know what features contribute much to representing images. However, there still exist limitations such as what feature selection method to use and what features to choose from. In addition, there is no guarantee that well-selected features for one image dataset A can be applied to another dataset B. For example, the dataset we used here appears to yield good results with the first-order statistics, but there may be other datasets that instead require the second-order statistics. Therefore, one has to conduct different feature selections for different datasets and to make sure the pool containing the features is large enough. If not, hand-crafting new and complicated features such as SIFT (Lowe, 1999) might be needed.

While selecting manual features is time-consuming, unsupervised feature learning is more straightforward. The feature learning based on K-means we implemented in this paper can autonomously learn feature representation from training data, and get a relatively higher testing accuracy as high as 96.71%. Although we cannot guarantee this feature learning method would outperform any manual feature setting other than first and second order statistics, this flexible and easily implemented method is capable of working well.

Self-taught learning gets test accuracy as high as 90.32% (Table 3). Features are learned from “unlabelled” data (first half of the whole dataset) with the same feature learning method and parameter configuration as in Table 2. The reason why the accuracy for this approach is not as good as FL is that we are using fewer data to both learn feature representation and train the classifier. This is not unexpected, because the more data available for learning features will result in a more generalized representation and more data for training will also result in a better classifier. Another reason is we are applying the feature representation learned from one subset (“unlabelled”) to another subset (“labelled”), rather than to the same subset as in FL, where learning feature and training classifier share the same subset (70% of the whole dataset). So, just as classifier typically performs better with the training data than testing data, feature representation performs better on the same training subset in FL.

6. Conclusion

In the first part of the work, we conducted rock image classification with various combinations of manual features as well as unsupervised feature learning. The results of these experiments show that different combinations of manual features affected classification substantially; whereas unsupervised feature learning based on K-means performed pretty well. While there is no guarantee that this feature learning method can absolutely outperform any manual features configuration, it is easily implemented and more flexible than the manual features.

We also explored the use of self-taught learning based on unsupervised feature learning for classification of rock images. The approach proved promising. It can learn the feature representation directly from unlabelled images of mixed rock types, and then repeatedly apply the feature representation to different sub-classes of rocks. We suggest that the fundamental reason as to why this approach works is that rock images share some basic visual patterns or elements. As such, as long as these basic patterns can be learned from the whole mixed dataset, they can be well utilized for representing the new groups of images belonging to the sub-class.

This autonomous rock image classification with learned features can enhance the capability of robots on planetary exploration, and enlarge the scientific returns. This technique can also be applied to geological image archive (e.g. autonomous labelling) or image retrieval etc. Future work will be to improve and test this technique on a larger and more general rock image dataset.

Acknowledgements

The authors would like to thank Dr. Steve Hicock for providing rock samples.

References

Aggarwal, N., Agrawal, R., 2012. First and second order statistics features for classification of magnetic resonance brain images. J. Signal Inf. Process. 3 (2), 146–153.

Blum, A., Mitchell, T., 1998. Combining labeled and unlabeled data with co-training. In: Proceedings of the eleventh annual conference on Computational learning theory, pages 92-100. ACM.

Burges, C.J., 1998. A tutorial on support vector machines for pattern recognition. Data Min. Knowl. Discov. 2 (2), 121–167.

Castano, R., Estlin, T., Anderson, R.C., Gaines, D.M., Castano, A., Bornstein, B., Chouinard, C., Judd, M., 2007. Oasis: onboard autonomous science investigation system for opportunistic rover science. J. Field Robot. 24 (5), 379–397.

Castano, R. et al., 2008. Experiments in onboard rover traverse science. In IEEE Aerospace Conference, 2008 IEEE, pages 1-11.

Castano, R. et al., 2004. Autonomous onboard traverse science system. In Aerospace Conference, 2004. Proceedings. 2004 IEEE, volume 1. IEEE.

Chandrashekar, G., Sahin, F., 2014. A survey on feature selection methods. Comput. Electr. Eng. 40 (1), 16–28.

Chanou, a., Osinski, G.R., Grieve, R.a.F., 2014. A methodology for the semi-automatic digital image analysis of fragmental impactites. Meteorit. Planet. Sci. 49 (4), 621–635.

Chatterjee, S., 2013. Vision-based rock-type classification of limestone using multi-class support vector machine. Appl. Intell. 39 (1), 14–27.

Coates, A., Ng, A.Y., 2012. Learning feature representations with k-means. Neural Netw.: Tricks Trade, 561–580, (Springer).

Coates, A., Ng, A. Y., Lee, H., 2011. An analysis of single-layer networks in unsupervised feature learning. International conference on artificial intelligence and statistics, pp. 215–223.

Cortes, C., Vapnik, V., 1995. Support-vector networks. Mach. Learn. 20 (3), 273–297.

Duda, R.O., Hart, P.E., et al., 1973. Pattern Classification and Scene Analysis 3. Wiley, New York.

Dunlop, H., 2006. Automatic Rock Detection and Classification in Natural Scenes. (Ph.D. thesis), Carnegie Mellon University.

Estlin, T. et al., 2009. Automated targeting for the mer rovers. SMC-IT 2009. In: Proceedings of the Third IEEE International Conference on Space Mission Challenges for Information Technology, 2009, pp. 257-263. IEEE.

Estlin, T.A., Bornstein, B.J., Gaines, D.M., Anderson, R.C., Thompson, D.R., Burl, M., Castano, R., Judd, M., 2012. Aegis automated science targeting for the mer opportunity rover. ACM Trans. Intell. Syst. Technol. (TIST) 3 (3), 50.

Francis, R., McIsaac, K., Thompson, D., Osinski, G., 2014a. Autonomous mapping of outcrops using multiclass linear discriminant analysis. In Proceedings of the International Symposium on Artificial Intelligence, Robotics and Automation in Space, Montreal, Canada.

Francis, R., McIsaac, K., Thompson, D.R., Osinski, G.R., 2014b. Autonomous rock outcrop segmentation as a tool for science and exploration tasks in surface operations. SpaceOps 2014 Conference, 1798.

Gor, V., Castano, R., Manduchi, R., Anderson, R., Mjolsness, E., 2001. Autonomous rock detection for mars terrain. Space, 1–14.

Haralick, R.M., Shanmugam, K., Dinstein, I.H., 1973. Textural features for image classification. IEEE Trans. Syst. Man Cybern. 6, 610–621.

Harinie, T., Chellam, I. J., Bama, S. S., Raju, S., Abhaikumar, V., 2012. Classification of rock textures. In Proceedings of the International Conference on Information Systems Design and Intelligent Applications 2012 (INDIA 2012) held in Visakhapatnam, India, January 2012, Springer. pp. 887–895.

Kanungo, T., Mount, D.M., Netanyahu, N.S., Piatko, C.D., Silverman, R., Wu, A.Y., 2002. An efficient k-means clustering algorithm: analysis and implementation. IEEE Trans. Pattern Anal. Mach. Intell. 24 (7), 881–892.

Koshiba, Y. Abe, S., 2003. Comparison of l1 and l2 support vector machines. In Neural Networks, 2003. In: Proceedings of the International Joint Conference on, volume 3, pp. 2054–2059. IEEE.

Laws, K.I., 1980. Textured image segmentation. Technical report, DTIC Document.

Le, Q.V., Karpenko, A., Ngiam, J., Ng, A.Y., 2011. Ica with reconstruction cost for efficient overcomplete feature learning. Adv. Neural Inf. Process. Syst., 1017–1025.

Lee, H., Battle, A., Raina, R., Ng, A.Y., 2006. Efficient sparse coding algorithms. Adv. Neural Inf. Process. Syst., 801–808.

Lepistö, L., Kunttu, I., Autio, J., Visa, A., 2003. Rock image classification using non-homogenous textures and spectral imaging.

Lepistö, L., Kunttu, I., Visa, A., 2005. Rock image classification using color features in gabor space. J. Electron. Imaging 14 (4), (040503–040503).

Lepistö, L., Kunttu, I., Visa, A., 2006a. Classification of natural rock images using classifier combinations. Opt. Eng. 45 (9), (097201–097201).

Lepistö, L., Kunttu, I., Visa, A., 2006b. Rock image classification based on k-nearest neighbour voting. IEE Proc. Vision. Image Signal Process. 153 (4), 475–482.

Lowe, D.G., 1999. Object recognition from local scale-invariant features.In: Proceedings of the Seventh IEEE International Conference on Computer vision, 1999, vol. 2, pp. 1150–1157. IEEE.

Młynarczuk, M., Górszczyk, A., Ślipek, B., 2013. The application of pattern recognition in the automatic classification of microscopic rock images. Comput. Geosci. 60, 126–133.

Paclík, P., Verzakov, S., Duin, R.P., 2005. Improving the Maximum-likelihood Co-occurrence Classifier: A Study on Classification of Inhomogeneous Rock ImagesImage Analysis. Springer, 998–1008.

Pittarello, L., Koeberl, C., 2013. Clast size distribution and quantitative petrography of shocked and unshocked rocks from the el'gygytgyn impact structure. Meteorit. Planet. Sci. 48 (7), 1325–1338.

Raina, R., 2009. Self-taught Learning. Stanford University.

Raina, R., Battle, A., Lee, H., Packer, B., Ng, A.Y., 2007. Self-taught learning: transfer learning from unlabeled data. In: Proceedings of the 24th international conference on Machine learning, pages 759-766. ACM.

Shang, C. Barnes, D., 2012. Support vector machine-based classification of rock texture images aided by efficient feature selection. In: Proceedings of the 2012 International Joint Conference on Neural Networks (IJCNN), pp. 1-8. IEEE.

Sharif, H., Ralchenko, M., Samson, C., Ellery, A., 2015. Autonomous rock classification using bayesian image analysis for rover-based planetary exploration. Comput. Geosci. 83, 153–167.

Singh, M., Javadi, A., Singh, S., 2004. A comparison of texture teatures for the classification of rock imagesIntelligent Data Engineering and Automated Learning-IDEAL. Springer, 179–184.

Ślipek, B., Młynarczuk, M., 2013. Application of pattern recognition methods to automatic identification of microscopic images of rocks registered under different polarization and lighting conditions. Geology, Geophysics and Environment, 39.

Tamura, H., Mori, S., Yamawaki, T., 1978. Textural features corresponding to visual perception. IEEE Trans. Syst. Man Cybern. 8 (6), 460–473.

Tou, J.Y., Tay, Y.H., Lau, P.Y., 2007. Gabor filters and grey-level co-occurrence matrices in texture classification. In: Proceedings of the MMU International Symposium on Information and Communications Technologies, pp. 197–202.

Van Gemert, J.C., Geusebroek, J.-M., Veenman, C.J., Smeulders, A.W., 2008. Kernel codebooks for scene categorization. European conference on computer vision. Springer, pp. 696-709.

Wang, H., Nie, F., Huang, H., 2013. Robust and discriminative self-taught learning. In: Proceedings of the 30th International Conference on Machine Learning, pp. 298-306.

Zhu, X., Ghahramani, Z., Lafferty, J., et al., 2003. Semi-supervised learning using gaussian fields and harmonic functions. ICML 3, 912–919.