Automated Kerogen Classification in Microscope Images of Dispersed Kerogen Preparation[1]

by
L.I. Kuncheva[2*], J.J. Charles[2], N. Miles[3], A. Collins[4], B. Wells[4] and I.S. Lim[2]
[2]School of Computer Science, Bangor University, LL57 1UT, UNITED KINGDOM
[3]PetroStrat Limited, Llandudno, UNITED KINGDOM
[4]Conwy Valley Systems Ltd, UNITED KINGDOM

Abstract

We develop the classification part of a system that analyses transmitted light microscope images of dispersed kerogen preparation. The system automatically extracts kerogen pieces from the image and labels each piece as either inertinite or vitrinite. The image pre-processing consists of: background removal, identification of kerogen material, object segmentation, object extraction (individual images of pieces of kerogen) and feature calculation for each object. An expert palynologist was asked to label the objects into categories inertinite and vitrinite, which provided the ground truth for the classification experiment. Ten state-of-the-art classifiers and classifier ensembles were compared: Naïve Bayes, decision tree, nearest neighbour, the logistic classifier, multilayer perceptron (MLP), support vector machines (SVM), AdaBoost, Bagging, LogitBoost and Random Forest. The logistic classifier was singled out as the most accurate classifier (accuracy $> 90\%$). Using a 10 times 10-fold cross-validation provided within the Weka software, we found that the logistic classifier was significantly better than five classifiers ($p < 0.05$) and indistinguishable from the other four classifiers. The initial set of 32 features was subsequently reduced to 6 features without compromising the classification accuracy. A further evaluation of the system alerted us to the possible sensitivity of the classification to the ground truth that might vary from one human expert to another. The analysis also revealed that the logistic classifier made most of the correct classifications with high certainty.

KEY WORDS: Kerogen recognition, machine learning, image processing, transmitted light microscopy.

1. Introduction

Kerogen is organic matter which can yield hydrocarbons on heating. Anglicising the kerogen content of sedimentary rocks, using images of material seen under the microscope, is therefore a vital component of oil and gas exploration (Tyson 1990). The kerogen pieces found in images of palynofacies can be broadly classified as inertinite, vitrinite and other. Vitrinite and inertinite macerals are derived from mainly higher plant material and include cell walls and gelified tissues. Figure 1 shows a typical image of a slide prepared for transmitted light examination containing vitrinite, inertinite, microfossils and other debris. Distinguishing between inertinite and vitrinite, based on their appearance in the image, is difficult. Inertinite differs from vitrinite in being derived from plant material that has been more strongly altered and degraded. Vitrinite appears a slightly lighter shade of brown, typically with more raggedy edges, and with more compact shape than inertinite. Vitrinite and inertinite are more easily identified using reflected light but the ready availability of transmitted light images from palynological and geochemical studies, presents the opportunity for a much increased collection of useful data.

In this study we describe a software system that extracts the kerogen objects from the image

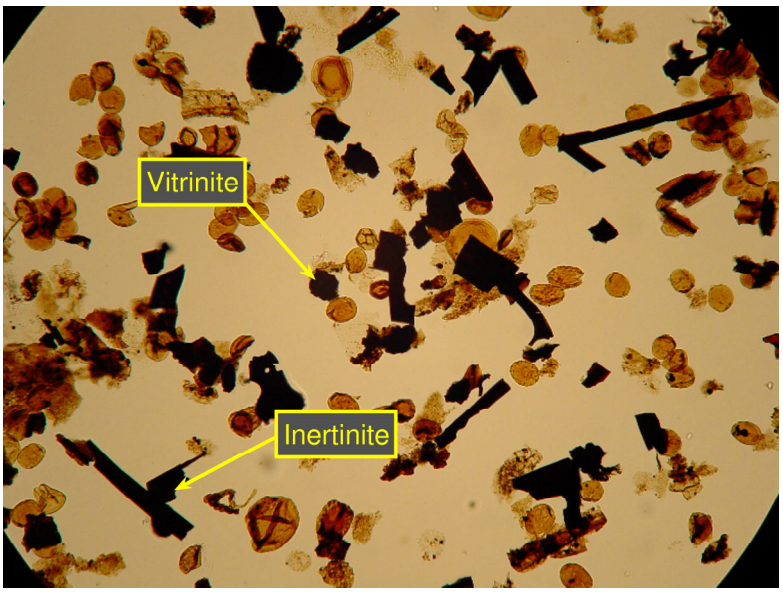

Figure 1: Dispersed kerogen preparation slide with indicated examples of inertinite and vitrinite.

and labels them in one of two classes: inertinite or vitrinite. Automated classification of fossilised material has been a topic of interest for decades (Athersuch et al. 1994; Bollmann et al. 2004; Hills 1988; Swaby 1992; Weller et al. 2005). While earlier object recognition systems used rule-based approaches (Athersuch et al. 1994; Liu et al. 1994), neural networks have been the preferred classifier in the more recent works (Bollmann et al. 2004; Weller et al. 2005, 2007). The competitive accuracy of neural networks across various distributions of the data compensates for their lack of transparency and their sensitivity to design choices. Neural networks have been the favourite classification model in other object recognition domains, e.g. classification of pollen types (France et al. 2000) and polyplankton species (Jonker et al. 2000; Wilkins et al. 1999). A combination of a statistical approach and a knowledge-based approach to classification of pollen types has also been suggested (Bonton et al. 2001; Boucher et al. 2002). A common theme in all these works is the imperative need for automatic extraction of the objects and their features from the original image. Plankton and pollen objects have distinct elliptical shapes, hence a number of image processing algorithms can be applied to crop a single object from an image. Kerogen objects, however, vary dramatically in shapes and sizes. Also, it is sometimes difficult even for the trained eye, to determine whether a cluster of dark-coloured forms contains several objects (touching or overlapping) or represents a single piece of kerogen. Thus the extraction of the individual kerogen pieces from an image poses a severe challenge itself. The accuracy and relevance of the features extracted from each object determines to a large extent the success of the classification part of an object recognition system. It has been demonstrated that if a careful extraction followed by a selection of features is carried out, even simple classification models will lead to good results (Flesche et al. 2000; Wang 1995). We propose an automatic system for information extraction from palynological images. The first three stages of the system are illustrated in Figure 2. The images of dispersed kerogen preparation slides are captured using a digital camera attached to a microscope. A stepping device is mounted on the microscope so that only a small portion of the slide is viewed at a time. A digital firewire camera and control box are used to capture high resolution images and transfer them to the computer. The setup is shown in Figure 2 A, and a typical captured image

is shown in Figure 2 B. To counter the problem of uneven background illumination due to the transmitted light, the background is equalised by fitting parabolas to horizontal and vertical straps of the image (Charles et al. 2008b) The resultant image is shown in Figure 2 C. The dark material (kerogen) is found next in the image (Fig. 2 D). The recently proposed Centre Supported Segmentation (CSS) algorithm (Charles et al., in press) is applied next to find the supposed centres of the kerogen objects (Fig. 2 D). These centres are subsequently used as markers for the watershed algorithm (Vincent and Soille 1991) so that individual pieces of kerogen are identified and catalogue (a random sample of 9 pieces is shown in Fig. 2 F). The automatic classification of the identified kerogen objects into inertinite and vitrinite constitutes the fourth stage of the system, which is the topic of this study.

2. The Classification Experiment

2.1 The features

609 objects were extracted from 7 microscope images of dispersed kerogen preparation. An expert palynologist was asked to label the objects into classes inertinite, vitrinite and other. The 8 objects labelled as "other" were not used in the subsequent training of the system, so the final labelled data set consisted of 601 objects in two classes. Using Matlab, 32 features were extracted from the image of each object accounting for its size, shape, texture and colour. Most of these are standard features included in the image processing package Halcon [1] and used in previous studies of palynological images (Weller et al. 2005) and object recognition in general (Wang 1995; Boucher et al. 2002). Along with the standard features we added equant/lath ratio and the variability of the colour at the periphery of the object, as these were assumed to be important for distinguishing between inertinite and vitrinite. Table 1 shows the grouping of the features and a short comment for each. Figure 3 displays "prototype" objects from the two classes. These objects were selected as the closest to the respective class means using the 32 (normalised) features.

2.2 The classifiers

The classifiers for our problem, shown in Table 2 (columns 1 and 2), come from three groups: standard (simple) classifiers, neural networks and classifier ensembles (Duda et al. 2001; Bishop 1995; Kuncheva 2004). All classifiers are well known and have been shown to perform well across a variety of datasets. A brief description follows, with the logistic classifier relegated to the end, and presented with more detail.

Simple classifiers. In the *Naïve Bayes classifier* the features are considered to be conditionally independent. Hence the class-conditional probability density functions (pdf) are approximated as products of marginal probabilities. The priors and the marginal probabilities are estimated from training data. The object is assigned to the class with the largest posterior probability, calculated using these estimates. The *decision tree classifier* works by making a sequence of decisions at the nodes of a decision tree until a leaf is reached. Each leaf has a class label associated with it, and this label is assigned to the object. Typically the decisions at the nodes are made by comparing a feature with a threshold value, both computed through training. In the *k-nearest neighbour classifier (k-nn)*

[1] `http://www.mvtec.com/halcon/`

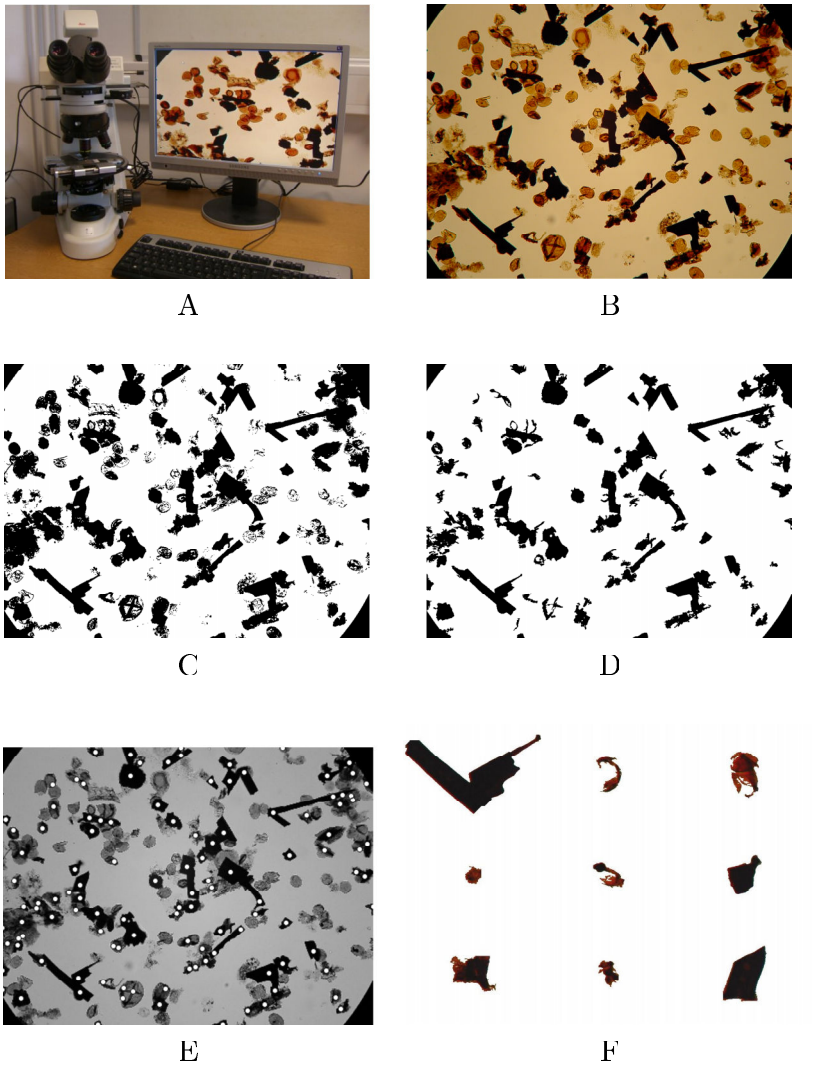

Figure 2: The first three stages of the kerogen recognition system. 1. Image acquisition: microscope setup (A) and an aqcuired image (B); 2. Background removal (C) and identification of kerogen (D); 3. Segmentation through the Centre Support Segmentation algorithm (centres overlaid on the original grey-level image (E)) and 9 randomly chosen examples of extracted objects (F).

Figure 3: The central objects (prototypes) of the two classes: Inertinite (A) and Vitrinite (B)

Table 1: Groups of features

Type	Feature	Notation	Explanation
Colour	mean red		
	mean blue		
	mean green		
	mean gray		
Size	inner radius	r_i	Radius of the largest circle contained within the object
	outer radius	r_o	Radius of the smallest circle containing the object
	diameter	d	Maximum distance between 2 contour pixels
	perimeter	p	Number of pixels on the object's border
	circle difference	$r_o - r_i$	Difference between the outer and inner radii
	area	a	Total number of object's pixels
	distance	δ	Mean distance from centre of gravity to contour pixels
Texture	(All features are extracted from the grey-level image)		
	entropy		Entropy of the histogram
	anisotropy		Symmetry of the histogram about its median
	correlation		Correlation between intensity of neighbouring pixels
	homogeneity		Homogeneity of neighbouring pixels
	contrast		Contrast of neighbouring pixels
	energy		Energy of neighbouring pixels
	rim variability		Variance of intensity at the border ($r_i/5$ pixels)
Shape	anisometry	e_+/e_-	Ratio of major to minor elliptic semi-axes
	eccentricity	d_-/d	Ratio of minor axis to diameter
	rectangularity	a/a_b	Ratio of areas: object/smallest bounding rectangle
	bulkiness	$\pi(e_+)(e_-)/a$	Ratio of areas: ellipse/object
	convexity	a/a_c	Ratio of areas: object/convex hull
	variance x		Variance across x-axis with respect to centre of gravity
	variance y		Variance across y-axis with respect to centre of gravity
	covariance		
	compactness	$4\pi a/p^2$	Ratio of areas: object/circle with the same perimeter
	sigma	σ	Standard deviation of distances from centre to contour
	roundness	$1 - \sigma/\delta$	
	sides	$1.41(\delta/\sigma)^{0.4724}$	Number of pieces of a regular polygon
	equant/lath	r_i/d	Equant/lath ratio
	structure factor	$\pi(e_+)^2/a - 1$	anisometry×bulkiness - 1

the object receives the label most represented among its k nearest neighbours. The neighbours are sought from a labelled reference set. The most popular variant of k-nn is the single nearest neighbour classifier (1-nn) where the object is assigned to the class of its nearest neighbour. This classifier does not require any prior training apart from specifying the reference set.

Neural network classifiers. The *MultiLayer Perceptron (MLP)* is a feedforward structure with one input layer, one output layer and a desired number of hidden layers. The n features in the problem are fed to the n input nodes, and the c discriminant functions (one per class) are obtained from the nodes at the output layer. The output with the largest value determines the class label of the object. The weights of the MLP are trained through backpropagation. The *Support Vector Machines classifier (SVM)* transforms the input space spanned by the n features, $\Re^n$, into an infinite-dimensional Hilbert space, where a linear classifier is trained. The linear boundary translates back to a highly accurate nonlinear boundary in the original space. By design SVM discriminates between two classes only.

Classifier ensembles. Combining the decisions of several classifiers has been shown to be better than taking forward the decision of a single classifier (Kuncheva 2004). Included in this category are top ranked methods that have shown robustness and accuracy across disciplines. They are often chosen as strawmen, against which other classifiers are compared. The classifier ensemble in *AdaBoost* is build sequentially. Each new classifier is trained on a part of the data that has proven to be difficult for the classifiers trained thus far. This approach induces diversity in the ensemble. The output class is determined by weighted majority voting where the more accurate classifiers have stronger votes. In *Bagging*, the classifiers are trained independently on bootstrap samples from the training data. The class is determined by majority vote (or plurality vote, if there are more than two classes). Theoretical studies led to developing *LogitBoost* so as to remedy AdaBoost's adverse sensitivity to noise in the data. Finally, *Random Forest* is a version of Bagging where diversity is sought by randomising the construction of the decision tree classifiers used as the ensemble members.

The Logistic Classifier. Let $\mathbf{x}$ be the object to be classified, where $\mathbf{x} = [x_1, \ldots, x_n]^T \in \Re^n$. Let $\omega_1, \ldots, \omega_c$ be the class labels, $P(\omega_i)$ be the prior probabilities and $P(\omega_i|\mathbf{x})$ be the posterior probabilities for the classes, $i = 1, \ldots, c$. The Logistic classifier relies on the assumption that the log-odds of the posterior probabilities for any two classes can be approximated as a linear function. Without loss of generality, we can pick class ω_c and fix its discriminant function to be $g_c(\mathbf{x}) = 0$ for any $\mathbf{x}$. The remaining $c - 1$ discriminant functions are calculated as

$$g_i(\mathbf{x}) = \log \frac{P(\omega_i|\mathbf{x})}{P(\omega_c|\mathbf{x})} = \beta_{i0} + \sum_{j=1}^{n} \beta_{ij} x_j, \quad i = 1, \ldots, c-1,$$

where β_{ij}, are coefficients obtained through training the classifier. The training is done by the *Iterative Reweighted Least Squares (IRLS)* method using *Newton-Raphson* updates (Bishop 2006; Hastie et al. 2001). The logistic classifier provides linear discriminant functions.

For each classifier, a 10-fold cross-validation was carried out 10 times. Thus the classification accuracy was estimated as the average of 100 testing sets of size 60 objects each. All classifiers are implemented in Weka, a free software environment for machine learning and data mining (Witten and Frank 2005). The default parameter settings were used in all experiments.

Table 2: Accuracy of the ten classifiers for the kerogen data set (10 times 10-fold cross-validation).

Type	Classifier	Accuracy[%] ± std	
		All features	Top 6 features
Standard classifiers	Naive Bayes (Hand and Yu 2001)	81.38± 5.36 •	85.11± 4.36 •
	Decision tree (Breiman et al. 1984)	84.01± 4.53 •	86.94± 4.20 •
	Logistic* (Hastie et al. 2001)	89.07± 3.44	90.62± 3.87
	Nearest Neighbour (Duda et al. 2001)	82.58± 4.41 •	85.84± 4.38 •
Neural networks	MLP[1] (Bishop 1995)	87.95± 4.07	89.29± 3.66
	SVM[2] (Cristianini and Shawe-Taylor 2000)	88.17± 4.26	90.07± 3.95
Classifier ensembles	AdaBoost (Freund and Schapire 1997)	84.51± 4.59 •	85.09± 4.44 •
	Bagging (Breiman 1996)	87.32± 4.47	88.06± 4.29
	LogitBoost (Friedman et al. 2000)	85.96± 4.38 •	87.14± 4.21 •
	Random Forest (Breiman 2001)	86.71± 4.57	88.07± 4.15 •

*Chosen as the base for comparison

• The classifier is significantly worse than the chosen classifier ($\alpha = 0.05$)

[1] MLP stands for Multi Layer Perceptron

[2] SVM stands for Support Vector Machines

3. Results

3.1 Classification accuracy

The averaged classification accuracy and the standard deviations are shown in Table 2. The Logistic classifier performs best among the chosen group. To evaluate the significance of this result, paired t-test was carried out between the Logistic classifier and all other classifiers. The classifiers which were significantly worse than the Logistic classifier according to the paired t-test are marked by a bullet in Table 2.[2] To reinforce the finding that the Logistic classifier is the best for our data, we counted the Win/Draw/Loss score for each classifier. For a given classifier D, Win is the number of classifiers significantly worse than D in the paired t-tests, Loss is the number of classifiers significantly better than D, and Draw is the remaining number of classifiers where no significant difference has been detected. A measure of total performance of D is therefore Total = Win−Loss. Table 3 shows the results for the ten classifiers in this study sorted by the Total measure. The Logistic classifier is again the best one among the selected 10 classifiers.

3.2 Feature selection

It is well known in pattern recognition that classifiers might work better if only a subset of the original features are used. This seemingly counterintuitive phenomenon is called "peak effect" (also "curse of dimensionality") and is due to the fact that no classifier is perfect. The ideal classifier, optimal in Bayesian sense, would not suffer from the peak effect, and its accuracy will not be adversely affected if noisy features are used. Classifiers trained on data are affected in different degrees. Classifiers based on features extracted from images are particularly vulnerable because there is no limit to the number of features that may be extracted. This stands as a challenge in classification of sedimentary organic matter from image analysis (Weller et al. (2006)).

[2] We note that the significance cannot be re-confirmed from the accuracies and the standard deviations shown in the table because the t-tests were *paired* across the 100 testing results.

Table 3: Number of statistically significant Wins, Losses and Wins−Losses for the 10 classifiers on the kerogen data

Classifier	Total score Win−Loss	Win	Loss
Logistic	5	5	0
Bagging	4	4	0
SVM	4	4	0
MLP	4	4	0
Random Forest	2	2	0
LogitBoost	1	2	1
AdaBoost	−3	1	4
Decision tree	−4	0	4
Nearest neighbour	−6	0	6
Naïve Bayes	−7	0	7

Table 4: Features ranked by the greedy stepwise selection procedure with the logistic classifier (average of 10-fold cross-validation).

Feature		Accuracy±std	Rank±std
1	mean red	0.816 ± 0.055	1.20 ± 0.60
2	mean blue	0.875 ± 0.044	2.00 ± 0.00
3	entropy	0.888 ± 0.040	2.80 ± 0.60
4	inner radius	0.897 ± 0.038	4.80 ± 1.60
5	rectangularity	0.900 ± 0.036	9.60 ± 4.22
6	diameter	0.902 ± 0.036	10.30 ± 4.75

The Logistic classifier favoured in our study might work better with a subset of the original features. To test this, we ran feature selection using Weka with the following protocol. A greedy stepwise feature selection was applied within a 10-fold cross-validation. Each feature was assigned a rank based on when it was selected. For example, the feature selected first (deemed to be the most relevant one) will get rank 1, while the feature selected last will get rank 32. By averaging the 10 ranks for each feature, obtained from the 10 folds, we have a measure of "importance" for that feature. Table 4 gives the best 6 features according to their ranks.

After finding the ranks, we carried out an experiment to evaluate the contribution of each feature. We split the data randomly into 90% for training and 10% for testing and repeated the experiment 100 times. Figure 4 displays the averaged (testing) classification accuracy of the logistic classifier as a function of the number of selected features. A horizontal line is placed at the accuracy with 6 features to show that beyond that point the accuracy levels off and then declines, which signifies overtraining of the classifier. Therefore we propose to proceed with the top 6 features: mean red, mean blue, entropy, inner radius, rectangularity and diameter. It was surprising that the bespoke features, equant/lath ratio and rim variability were often picked towards the end of the procedure which gave them a poor rank score. Since there are two classes in the problem, only one discriminant function is needed. The

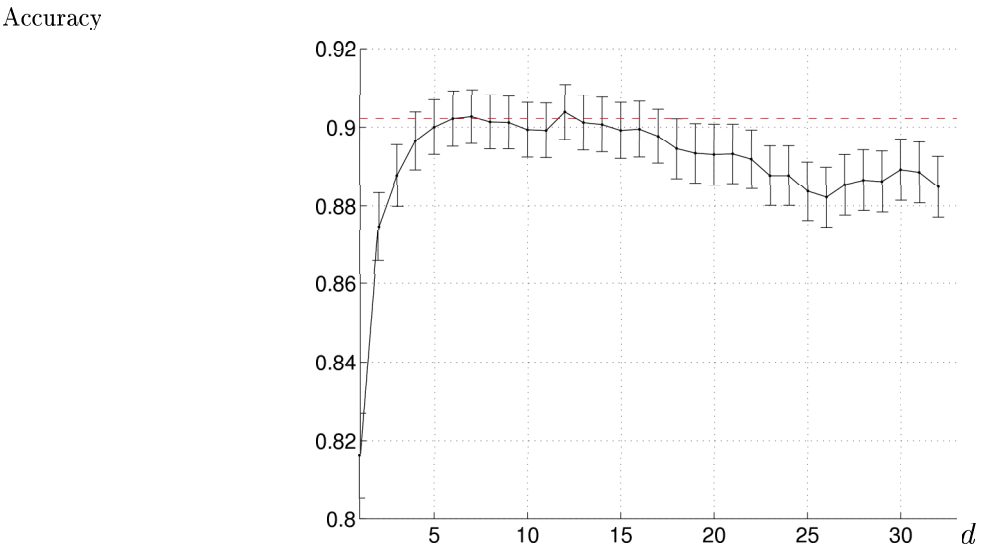

Figure 4: Classification accuracy versus size, d, of the selected feature subset. The bars indicate the 95% confidence intervals.

coefficients of this linear function when the classifier was trained on the whole data set (normalised) are given in Table 5. If the expression is positive, the predicted class is inertinite, if negative or zero, the predicted class is vitrinite. Also shown in the table is the confusion matrix on the whole data set and the resubstitution accuracy (note that that is not much different from the accuracy estimated through 10 times 10-fold cross-validation in Table 2).

Table 5: Coefficients and confusion matrix for the logistic classifier with the best 6 features trained on the whole data set (I stands for "inertinite" and V stands for "vitrinite")

(a) Coefficients

meanred	−3.033
meanblue	1.9054
entropy	−2.086
innerradius	−0.5118
rectangularity	−0.3388
diameter	−0.3873
intercept	−2.1066

(b) Confusion matrix

	Assigned	
	I	V
I (true)	210	28
V (true)	26	337

Accuracy = 91.01%

4. Further Evaluation of the System

It is interesting to find out how certain the classifier is in its prediction and whether the misclassified objects have been labelled with lower certainty. It is also curious whether the human expert and the classifier are uncertain about the same objects. The *certainty* of the classification decision can be measured by the posterior probability for the winning class. The larger the probability, the higher the certainty (confidence). Let $g(\mathbf{x})$ be the output of the discriminant function produced by the logistic

classifier. An estimate of the posterior probability $P(\text{inertinite}|\mathbf{x})$ is $P(\text{inertinite}|\mathbf{x}) = \frac{1}{1+\exp(-g(\mathbf{x}))}$, and, respectively, $P(\text{vitrinite}|\mathbf{x}) = 1 - P(\text{inertinite}|\mathbf{x})$. Figure 5 plots a histogram of $P(\text{inertinite}|\mathbf{x})$ across the 601 objects in the data set. Values close to 0 and to 1 signify high certainty while values near 0.5 signify low certainty. The shape of the histogram shows that most objects are classified with high certainty.

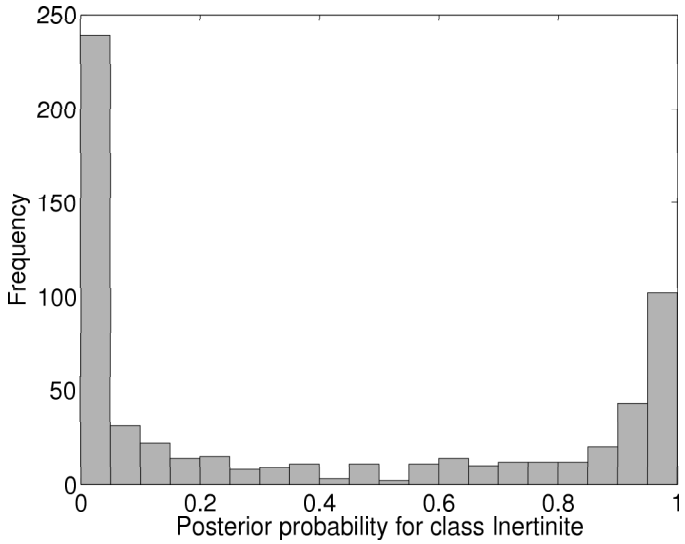

Figure 5: Histogram of $P(\text{inertinite}|\mathbf{x})$ for the whole data set.

A look at the 54 misclassified objects reveals that these are not necessarily the objects labelled with the lowest certainty. The posterior probability for the wrong class spans the whole interval from 0.5 to 1. This suggests that setting up a certainty threshold and allowing the system to abstain from making a decision will not necessarily improve the system's accuracy.

The 5 objects labelled in the wrong class with the highest certainty are shown in Figure 6. A second opinion of a palynologist expert was sought on the misclassified objects. Out of the 28 objects mislabelled as inertinite, the expert agreed with the system's decision on 9 objects (32%). Out of the 26 objects mislabelled as vitrinite, the expert agreed with the system's decision on 13 objects (50%) and placed another 7 objects (27%) in class "other". In total, the second expert accepted the system's (supposedly wrong) decision for 22 objects out the 54 (41%) and labelled further 7 of these objects (13%) into class "other".

The disagreement between the expert opinions prompts the following two comments about designing an automatic system for kerogen classification. First, the ground truth on which the system is trained may vary considerably from one expert to another. This makes the system *expert-specific*. This concern has been flagged before by Athersuch et al. (1994) pointing out that fossil identification is often a personal science. While an expert system relies on rules which are generally accepted, a system based on expert labelling of objects may only be as good as that one expert. Therefore it is important to seek consensus between experts on a set of objects in order to prepare the data. Second, the fact that the experts may disagree means that the two classes do not have a clear-cut definition in the first place. Thus labelling borderline cases in either of the two classes should not be considered a severe mistake on the system's part.

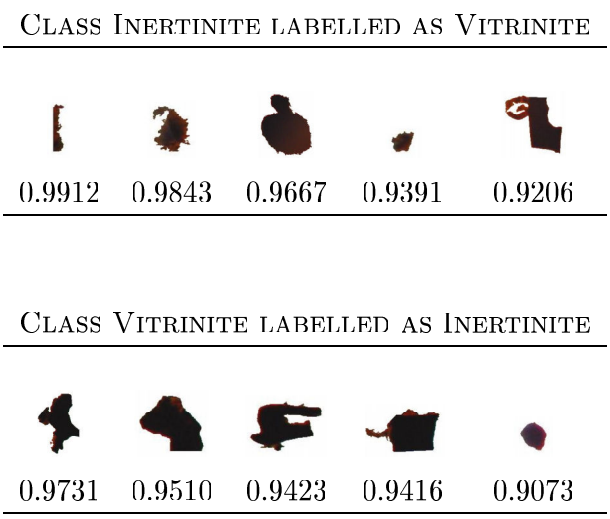

Figure 6: The five misclassified objects from each class, labelled with largest certainty (shown underneath).

5. Conclusions

We present an approach to automatic classification of inertinite and vitrinite pieces in images of dispersed kerogen preparation. To our knowledge, this is the first attempt at a completely automatic classification system for kerogen types from microscope images of this type. Such a system can be a first step towards remote palynology analysis concerning oil exploration. Palynologists are often required to be present on oil rigs to help identify geological horizons as they are drilled. There are currently not enough palynologists to cover all this work, which is geographically widespread. If palynological slides could be scanned by an automated system on the oil rig and pictures of all the palynomorphs made available to a remote palynologist then there would be considerable advantages to the oil industry.

The experiments singled out the Logistic classifier as the best choice among a selection of state-of-the-art classifier models. This classifier is simple and robust, and outperformed both simple models such as Naïve Bayes and Nearest Neighbour, as well as more sophisticated classifiers such as Support Vector Machines (SVM) and a standard Multilayer Perceptron (MLP). Surprisingly, the logistic classifier fared better than all the ensemble classification methods which are believed to be better than single classifiers. This result may be explained by possible overtraining of the more sophisticated classifiers.

The results from the experiments suggested that the problem can be solved at equal accuracy (even slightly better) using only 6 of the 32 features. Most of the features are derived from one another, and this creates redundancy. The most important features were found to be from the colour group (mean red and mean blue), as vitrinite pieces are generally lighter than inertinite pieces. Importance of colour was also noted by Weller et al. (2006) in regard to classification of palynomorphs by an artificial neural network classifier. However, we note that within the top 6 features, there is a representative from each of the four groups: colour (mean red and mean blue), size (inner radius and diameter), texture

(entropy) and shape (rectangularity).

In all the analyses presented above we assumed that the expert's decision on each piece is precise and final. Thus, all our classifier is meant to do is match the expert's decision as close as possible. In this study we left the learning completely to the training algorithm of the chosen classifier model, and have not attempted to mimic the expert's logic in reaching the decision. The system will be further extended to include palynomorph extraction and recognition. After the kerogen pieces have been classified, they will be removed from the image, and the remaining objects (palynomorphs, amorphous matter) will be extracted and subsequently classified. Our future research will follow two paths. First, we will evaluate the stability of the classification results with respect to small and large alterations of the object extraction method. Second, we plan to solicit another expert's opinion on the whole data set and carry out experiments to evaluate concordance between the experts themselves and also with respect to the systems designed on their individual and collective opinions.

Aknowledgements

The EPSRC CASE grant Number CASE/CNA/05/18 is acknowledged with gratitude.

References

Athersuch J, Banner FT, Higgins AC, Howarth RJ and Swaby PA (1994). The application of expert systems to the identification and use of microfossils in the petroleum industry. Mathematical Geology, 26(4):483–489

Bishop CM (1995). Neural Networks for Pattern Recognition. Clarendon Press, Oxford

Bishop CM (2006). Pattern Recognition and Machine Learning. Springer, NY

Bollmann J, Quinn P, Vela M, Brabec B, Brechner S, Cortés M, Hilbrecht H, Schmidt DN, Schiebel R and Thierstein HR (2004). Automated particle analysis: Calcareous microfossils. In P Francus, editor, Image Analysis, Sediments and Paleoenvironments, pages 229–252. Kluwer Academic Publishers, Dordrecht, The Netherlands

Bonton P, Boucher A, Thonnat M, Tomczak R, Hidalgo P, Belmonte J and Galan C (2001). Colour image in 2d and 3d microscopy for the automation of pollen rate measurement. Image Analysis and Stereology, 20:527–532

Boucher A, Hidalgo P, Thonnat M, Belmonte J, Galan C, Bonton P and Tomczak R (2002). Development of a semi-automatic system for pollen recognition. Aerobiologia, 18(3-4):195–201

Breiman L (1996). Bagging predictors. Machine Learning, 26(2):123–140

Breiman L (2001). Random forests. Machine Learning, 45:5–32

Breiman L, Friedman J, Olshen R and Stone C (1984). Classification and Regression Trees. Wadsworth International, Belmont, Calif.

Charles JJ, Kuncheva L, Wells B and Lim I (2008a). Object segmentation within microscope images of palynofacies. Computers & Geosciences. (accepted, http://dx.doi.org/10.1016/j.cageo.2007.09.014)

Charles JJ, Kuncheva LI, Wells B and Lim I (2008b). Background segmentation in microscope images. In Proc 3rd International Conference on Computer Vision Theory and Applications VISAPP08. Madeira, Portugal

Cristianini N and Shawe-Taylor J (2000). An Introduction to Support Vector Machines. Cambridge University Press, Cambridge, UK

Duda RO, Hart PE and Stork DG (2001). Pattern Classification. John Wiley & Sons, NY, second edition

Flesche H, Nielsen AA and Larsen R (2000). Supervised mineral classification with semiautomatic training and validation set generation in scanning electron microscope energy dispersive spectroscopy images of thin sections. Mathematical Geology, 32(3)

France I, Duller A, Duller G and Lamb H (2000). A new approach to automated pollen analysis. Quaternary Science Reviews, 19(6):537–546

Freund Y and Schapire RE (1997). A decision-theoretic generalization of on-line learning and an application to boosting. Journal of Computer and System Sciences, 55(1):119–139

Friedman J, Hastie T and Tibshirani R (2000). Additive logistic regression: a statistical view of boosting. Annals of Statistics, 28(2):337–374

Hand DJ and Yu K (2001). Idiot's Bayes - not so stupid after all? International Statistical Review, 69:385 – 398

Hastie T, Tibshirani R and Friedman J (2001). The Elements of Statistical Learning. Springer, New York

Hills S (1988). Outline extraction of microfossils in reflected light images. Camputers & Geosciences, 14(4):481–488

Jonker R, Groben R, Tarran G, Medlin L, Wilkins M, Garcia L, Zabala L and Boddy L (2000). Automated identification and characterisation of microbial populations using flow cytometry: the aims project. Scientia Marina, 64:225–234

Kuncheva LI (2004). Combining Pattern Classifiers. Methods and Algorithms. John Wiley and Sons, N.Y.

Liu S, Thonnat M and Berthod M (1994). Automatic classification of planktonic foraminifera by a knowledge-based system. In Proceedings of the 10th Conference on Artificial Intelligence for Applications, pages 358–364. IEEE Computer Society Press, San Antonio, Texas

Swaby PA (1992). VIDES: An expert system for visually identifying microfossils. IEEE Expert: Intelligent Systems and Their Applications, 7(2):36–42

Tyson RV (1990). Automated transmitted light kerogen typing by image analysis. Mededelingen Rijks Geologische Dienst, 45:139–149

Vincent L and Soille P (1991). Watersheds in digital spaces: an efficient algorithm based on immersion simulations. IEEE Transactions on Pattern Analysis and Machine Intelligence, 13(6):583–598

Wang L (1995). Automatic identification of rocks in thin sections using texture analysis. Mathematical Geology, 27(7):847–865

Weller AF, Corcoran J, Harris AJ and Ware JA (2005). The semi-automated classification of sedimentary organic matter in palynological preparations. Computers & Geosciences, 31(10):1213–1223

Weller AF, Harris AJ and Ware JA (2007). Two supervised neural networks for classification of sedimentary organic matter images from palynological preparations. Mathematical Geology, 39(7):657–671

Weller AF, Harris AJ, Ware JA and Jarvis PS (2006). Determining the saliency of feature measurements obtained from images of sedimentary organic matter for use in its classification. Computers & Geosciences, 32(9):1357–1367

Wilkins MF, Boddy L, Morris CW and Jonker RR (1999). Identification of phytoplankton from flow cytometry data by using radial basis function neural networks. Applied and Environmental Microbiology, 65(10):4404–4410

Witten IH and Frank E (2005). Data Mining: Practical Machine Learning Tools and Techniques. Morgan Kaufmann, 2nd edition