PLANT LEAF CLASSIFICATION USING PROBABILISTIC INTEGRATION OF SHAPE, TEXTURE AND MARGIN FEATURES

Charles Mallah
Department of Science, Engineering & Computing
Kingston University London
Kingston upon Thames, Surrey, United Kingdom
email: charles.mallah@kingston.ac.uk

James Cope and James Orwell
Department of Science, Engineering & Computing
Kingston University London
email: J.Cope@kingston.ac.uk; james@kingston.ac.uk

ABSTRACT
Plant species classification using leaf samples is a challenging and important problem to solve. This paper introduces a new data set of sixteen samples each of one-hundred plant species; and describes a method designed to work in conditions of small training set size and possibly incomplete extraction of features. This motivates a separate processing of three feature types: shape, texture, and margin; combined using a probabilistic framework. The texture and margin features use histogram accumulation, while a normalised description of contour is used for the shape. Two previously published methods are used to generate separate posterior probability vectors for each feature, using data associated with the k-Nearest Neighbour apparatus. The combined posterior estimates produce the final classification (where missing features could be omitted). We show that both density estimators achieved a 96% mean accuracy of classification when combining the three features in this way (training on 15 samples with unseen cross validation). In addition, the framework can provide an upper bound on the Bayes Risk of the classification problem, and thereby assess the accuracy of the density estimators. Lastly, the high performance of the method is demonstrated for small training set sizes: 91% accuracy is observed with only four training samples.

KEY WORDS
Pattern Recognition, Plant Leaves Classification, k-Nearest Neighbours, Density Estimators, Combining Features.

1 Introduction

Plant species are not only vast in number, but also in their use; estimates of the number of species of flowering plants range from 220,000 [1] to 420,000 [2]. Automatic recognition of plants is useful for a variety of industries such as foodstuff and medicine [3], reduction of chemical wastage during crop spraying [4, 5], and also for species identification and preservation [6].

Plant taxonomy suggests that a species can be successfully inferred from the leaves; the leaves are more readily available, easily found, and collected than other parts of the plant. The literature suggests a variety of work has been documented in the machine vision field using data sets of leaves and image processing techniques, see Section 2.

The investigation is conducted using an existing and new data set. The new data set consists of 1,600 images of leaf specimens (16 samples each of one-hundred species). Estimation of the leaf class (species) uses three features, which are analysed separately: a shape descriptor, an interior texture histogram, and a fine-scale margin histogram. These are then combined to provide an overall indication of the species (and associated probability).

The 'k-Nearest Neighbour' (K-NN) classifier is a fundamental tool in pattern analysis, providing a straightforward means of determining class membership for an unseen sample vector, given a finite training set. For a set of unseen vectors, the resulting precision and recall provide an an insight into the separability of the class labels, and easily generalises to problems with multiple classes.

The K-NN classifier is extended to output information for all classes in the data set; in turn this is used to estimate the posterior probability of the leaf species; or, can be combined with the posterior probability calculated using a different feature or a set of multiple features.

We propose that the density estimates given from the standard [7] and the recent extension [8] are used in a framework to enable multiple features to be combined in a straightforward approach. A separate K-NN density estimator is generated for each feature, separately. Then, a simple product of the three different density estimations will provide the final estimate.

Evaluation of discrete classifier performance is reasonably straightforward, combining the precision and recall into a single 'mean accuracy' characteristic. This approach can also be used to indirectly evaluate the per-channel probabilistic outputs, which are integrated into a final discrete output. The direct evaluation of probabilistic output uses an information theoretic metric, i.e. the 'information gain', to measure the mutual information between the actual posterior and the estimates of this distribution provided by the classifier.

The results show that, using all evaluation metrics, the proposed approach of combined density estimates from multiple features provides improved performance. The robustness of the approach is demonstrated on the challenging new one-hundred leaves data set, where the classification performance remains relatively accurate given a drastic

reduction in training size.

The paper is organised as the following: Section 2 discusses the previous work related to plant leaf features and density estimation with K-NN classifiers. The two data sets, consisting of the Iris plant and one-hundred plant species, are described in Section 3. The three extracted features from the one-hundred species data set are defined in Section 4. Section 5 explores the two K-NN density estimators tested in this paper, and Section 6 discusses the evaluation method. Section 7 contains the results and discussion of the two density estimators with the two aforementioned data sets. Finally, the paper is concluded in Section 8.

2 Previous Work

In addition to discussing the features, classifier, and possible framework proposed for leaf species identification. In the literature, several approaches are explored in order to solve the outlining problem of automatic species recognition of leaves. We discuss the various literature related to shape, interior texture and fine-scale margin features with images of leaves. Section 2.4 explores classification methods using with density estimates.

2.1 Shape Features

Various different shape extraction techniques have been applied to leaves. J.-X. Du et al. [3] explore leaf classification with a data set of 20 species of leaf and 20 samples of each. Fifteen conventional digital morphological shape features were calculated per sample and used to classify the plant species automatically. Contrary to this approach, Z. Wang et al [9] explores the classification of leaf species using a single relatively complex feature: the Centroid Contour Distance Curve (CCDC). The object eccentricity is also used as a rejection criterion to reduce computation time. Both of these features are scale, translation and rotation invariant after the appropriate normalisations are applied.

Y. Shen et al [10] also use the CCDC. Their approach gives a precision rate of 72%, given this many sample points. A direct comparison is made to the method suggest by A. Hong et al [6], who uses the CCDC curve in addition to an Angle Code Histogram of the shape's contour. A dissimilarity measure is given by combining a weighted summation of the two features. It is noted on a data set of flower images that a recall rate of 56% is achieved using these shape features, compared to a set of colour features which archives 67% (both over a set of 80 images). A combination of shape and colour gives a recall rate of 80% over 80 images.

2.2 Texture Features

A number of both traditional and novel texture analysis techniques have been applied to leaves. Backes et al. have applied multi-scale fractal dimensions [11] and deterministic tourist walks [12] to plant species identification by leaf texture, although their experiments involved very limited data sets which makes them hard to evaluate. Casanova et al. [13] used an array of Gabor filters on a larger data set, calculating the energy for the response of each filter applied, and achieved reasonable results, whilst Liu et al. have presented a method based on wavelet transforms and support vector machines [14]. Cope et al. [15] achieved an 85% identification rate on 32 species of a single plant species using the co-occurrences of different scale Gabor filters.

Whilst these studies were all performed on texture windows acquired using traditional imaging techniques (i.e. cameras and scanners), Ramos [16] used images acquired using a scanning electron microscope (SEM), and Backes [17] used magnified cross-sections of the leaf surface epidermis. While these provide interesting results, such images are not available on a large scale.

2.3 Fine Margin Features

Although the leaf margin is of great importance to botanists, there has been surprisingly little work towards its automated analysis. This may be due to the fact that not all plant species leaves feature teeth, although in these cases classification could still be aided by quantitative descriptors of the margin, which in many of these cases are still not entirely smooth. Another reason could be the difficulty in successfully acquiring meaningful measurements, particularly if it is assumed that these descriptors should match those currently used by botanists.

McLellan and Endler [18] used a single value measure of margin roughness, calculated by summing the angles between the lines connecting adjacent points around the contour. This was then combined with a selection of single-parameter shape descriptors.

2.4 Probability Density Estimation

There are several important motivations for obtaining an accurate estimate of the class membership probabilities for a test sample. Most generally, it allows the calculation of the Bayes risk, which is a more informative indicator of the problem complexity than the precision and recall. Secondly, there are specific applications of pattern analysis in which the posterior probabilities are required as outputs, in addition to the indication of which class label is the most probable. Thirdly, for pattern analysis applications in which several diverse channels of information are processed separately, and integrated to produce an overall estimate, a Bayesian approach is the most appropriate, for which accurate probability estimates from each channel is required.

There are good reasons for treating these channels separately: it is possible that they may not all always be available, or with caveats, e.g. with increased noise.

Also, the separation enables ongoing analysis to one channel in isolation, without affecting the analysis of the other channels. All these considerations particularly apply when the training set is relatively small: the performance of any given classifier at fixed training set size is an important characteristic. Homogeneity between different feature types is also a valid issue. Classifiers such as LMNN [19], adjust the weighting of the features in order to improve classification. Additionally, a K-NN classification rule based on Dempster-Shafer theory [20] can addresses this issue. However, in many cases, it is shown that the straightforward and simplistic nature of K-NN is desirable because of ease of implementation and greater understanding of the classification reasoning.

Given there are often more than one feature type available, a combination method or rule can be used to improve the classification performance. Often in multiple class problems the product of two probability vectors provides the better combination rule, as opposed to other rules such as the mean value [21]. In this situation, there is assumption of independence. It is also noted that density estimates from a K-NN classifier are often only reliable for large training sets. Therefore, in response to this, we hope to further evaluate the density framework using various training sizes.

3 Data Sets

A known plant data set is used to validate the methods outlined in the paper, see Section 3.1. In addition, a new one-hundred species leaves data set is included, which is a relatively difficult classification domain consisting of major/minor species of plant leaves, see Section 3.2.

3.1 Iris Plants

A well-known data set in the pattern recognition literature is Fisher's iris data [22, 23]. It consists of 50 samples from 3 varieties of Iris plant. The features of the iris data consist of four scalar values; these refer to the Length and Width of the Petal (PL and PW), and the Length and Width of the Sepal (SL and SW).

Guvenir et al, [24], use the iris data set (amongst others) to validate the performance of their own weighted K-NN classifier. In the case of $K = 3$, using a ten-fold cross-validation evaluation, the mean classification accuracy was reported to be 90.7% for the standard classifier definition. The authors report an accuracy of 94% using their weighted K-NN metric (with the same $K = 3$ parameter). We create four separate feature vectors per iris sample, which relate to each of the petal and sepal dimensions (rather than use all values in one feature vector).

3.2 Leaves

In this paper, we introduce a new leaves data set, which consists of one-hundred varieties of leaves. For each variety, 16 examples of leaves were collected; a colour image was captured of each example placed on a white background. It is worth noting that the problem inherently consists of having a wide set of classes, with a low number of examples. As such, the approach and selected methods may be effected by this constraint. However, this problem expands the study to comment on the accuracies and effectiveness of multi-class classification given a small training sample.

Image segmentation was completed via a semi-automated method. This consisted of using global thresholding [25] on grey scale images, followed by manual correction for any spurious segmentation. The binary images are used as the input for the feature extraction processed. Figure 1 contains a sample set of images from the new data set.

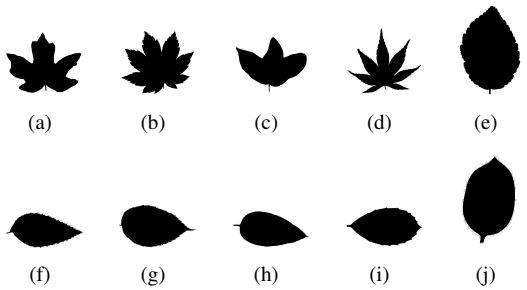


Figure 1. A small variety of plant species that are part of the challenging one-hundred species leaves data set: (a) Acer Campestre. (b) Acer Circinatum. (c) Acer Mono. (d) Acer Palmatum. (e) Alnus Rubra. (f) Alnus Sieboldiana. (g) Betula Austrosinensis. (h) Eucalyptus Urnigera. (i) Ilex Aquifolium. (j) Ilex Cornuta.

4 Leaf Classification Method

Three different leaf features are used. These consist of a shape signature, an interior texture feature histogram, and a fine-scale margin feature histogram; described in Sections 4.1 to 4.3, respectively. We intend on using the information gained from the selected K-NN density estimators, which in turn are used as prior estimates for predicting the most likely class. Each feature channel is processed in this way and the posterior distribution is estimated by combining all prior estimates.

This probabilistic framework allows multiple feature vectors, or modalities, to be combined as independent estimations of the class probability. Thus, there is an expectation of a positive impact when combining information from multiple sources. Two K-NN density estimation methods are explored to produce posterior distributions of the plant species for each feature channel, see Section 5. The poste-

rior estimates are combined to provide the final classification estimation.

4.1 Leaf Shape Feature

A shape descriptor is a 1D function that represents a 2D area or boundary. The Centroid Contour Distance Curve (CCDC) [6, 10, 9] is shape descriptor implemented in this experiment because it is scale, translation and rotation invariant.

Let $\{x_i, y_i : x_1, y_1, x_2, y_2, \ldots, x_s, y_s\}$ be the set of points about the contour of a shape where s is the maximum number of points, i.e. the perimeter points. Depending on the size and complexity of the leaf, s may naturally vary between one-hundred and 3000 elements.

An ordering function is required to create a chain of connectivity of the perimeter pixels, since the signal will only retain the pertinent shape information by tracing the contour in a contiguous fashion. A point reduction is made with linear interpolation. Let $\bar{x}$ and $\bar{y}$ be the centroid of the shape, computed as:

$$\bar{x} = \frac{\sum_{i=1}^{s} x_i}{s}, \bar{y} = \frac{\sum_{i=1}^{s} y_i}{s} \tag{1}$$

The shape descriptor, d_i, can be described with the following equation:

$$d_i = \sqrt{(x_i - \bar{x})^2 + (y_i - \bar{y})^2} \tag{2}$$

The CCDC is currently given about an arbitrary starting point, such as whichever point happens to be at $i = 1$. However, it is necessary to order the signature with a suitable and repeatable starting point. The alignment method used was based on the principal axis of the shape (the 2nd moment of inertia). The perimeter of the shape is projected onto the principal axis and the two points at either end are matched to two perimeter pixels. Out of these two pixels, one is selected to be the starting point based on which is farthest of the two from the centroid.

The d_i signature is translation invariant, due to the distance metric from the centroid. For scale invariance, the signature is normalised, such that:

$$d'_i = \frac{d_i}{\sum_{i=1}^{s} d_i} \tag{3}$$

Finally, the signature vector can then be scaled down to an appropriate size with linear interpolation, ensuring that all signatures have the same number of elements, as well as decreasing computation times by using smaller data sets. In this study, the vector sizes are fixed to $s = 64$ elements.

4.2 Leaf Texture Feature

Texture descriptors are generated, based on the method of [15]. For each leaf image, 1024 small windows from the surface of the leaf are randomly selected. Then, For each window, 20 features based on the responses from different filters applied to all the pixels in the window.

The filters used are a rotationally invariant version of the Gabor filter:

$$g(x, y) = \exp \frac{r^2}{2\sigma^2} \cos \frac{2\pi r}{\lambda}$$

where $r = \sqrt{x^2 + y^2}$ is the distance from the centre of the filter, σ is the standard deviation, and λ is the wavelength, set to be $\lambda = 3\sigma$. Five different scale filters are used, produced by varying σ.

Each filter is convolved with the window and four features are then calculated for that filter for the window:

1. Average positive value
$$\sum_{\substack{(i,j)\in W \\ s_j \geq 0}} \frac{f_{ij}}{|W|}$$
2. Average negative value
$$\sum_{\substack{(i,j)\in W \\ s_j \leq 0}} \frac{f_{ij}}{|W|}$$
3. Energy
$$\sum_{(i,j)\in W} \frac{{f_{ij}}^2}{|W|}$$
4. Entropy
$$-\sum_{(i,j)\in W} \frac{|f_{ij}|}{|W|} \log \frac{|f_{ij}|}{|W|}$$

Where W is the current window, f_{ij} is the response for the current filter at pixel (i, j), and $|W|$ is the size of the window.

Each of the 1024 20-feature vectors is then quantized to one of 64 predetermined vectors. A histogram is then built from the number of the 1024 vectors assigned to each of the 64 quantisation vectors, giving a 64-feature vector for describing each leaf.

4.3 Leaf Margin Feature

For the leaf margins we use the descriptors described in [26]. To extract the margin of a leaf, first, a median filter is applied to the binary leaf image to acquire a smoothed version of the leaf's shape. From this, m evenly spaced points are calculated, encompassing the entire outline. For each of these points, a corresponding point on the original outline is calculated.

This is done by first estimating the line that is normal to the edge of the leaf at this point as being perpendicular to the line which runs between the two points at distance k either side of the current point. The sub-pixel point at which this line intersects the original leaf's outline is then found. The distance between this point and the current point is then calculated, and these distances for all the points in the smoothed outline are combined in order to produce a margin signature, $\mathbf{s} = \langle s_1, ..., s_m \rangle$.

The extracted margin is partitioned into n overlapping windows, $\mathbf{x} = \langle x_1, ..., x_n \rangle$, of equal size and spacing (in this case $n = \frac{m}{8}$ and the window size used is $\frac{m}{128}$).

For each point within a window, x_i, 3 values are calculated:

1. Magnitude - This is the signed distance between the smoothed margin point and its corresponding point in the original margin, where the sign is determined by whether the original margin point lines inside or outside of the smoothed margin.

2. Gradient - The signed difference between the current point in the margin signature and the next point.

3. Curvature - The angle at the current point between the previous point and the next point.

For each of these, 2 features are then calculated for the window, producing a 6-feature descriptor for each window:

- Average positive value

$$\sum_{\substack{s_j \in x_i \\ s_j \geq 0}} \frac{s_j}{|x_i|}$$

- Average negative value

$$\sum_{\substack{s_j \in x_i \\ s_j \leq 0}} \frac{s_j}{|x_i|}$$

Where x_i is the current window, s_j is the value at a point within the signature, and $|x_i|$ is the size of the window.

The feature-vectors for the margin windows are used to produce a 64-feature vector describing the margin, using the same quantisation and histogram method as for the textures in section 4.2.

5 Density Estimation from a K-NN Classifier

In this section, we describe the methods that output probability information from a K-NN classifier [20, 3], to be used for classification with multiple feature types (if required).

Consider each feature has an independent probability vector $\{P_i^o : o = 1, o = 2, \ldots, o = f\}$, where f is the number of features to be combined. The combined probability vector, C, is simply the product of two of more vectors, such that:

$$C_i = P_i^o \cdot P_i^1 \ldots \cdot P_i^f \tag{4}$$

In turn, a combined probability vector C_i can be produced for all possible combinations of features. In our case, the best combination is selected from the one with the highest ACC. The boosting process is achieved by selecting the best performing combination. It is also possible to analyse the effectiveness of each individual feature vector, and its contribution to the classification solution.

The density estimation process will output a single posterior probability vector, P, per feature type. The probability vector consists of a single probabilistic estimation for each category, such that $\{P_i : P_1, P_2, \ldots, P_o\}$, where o is the number of categories (or classes). Each element of P, is a non-zero density estimate where $\sum P_i = 1$.

Two density estimators are explored in this paper; the well known standard by Fukunaga [7, 27]; and an extension of this by Atiya [8].

The estimate by Fukunaga is given as:

$$\hat{P}(L_o|x) = \frac{K_o}{K} \tag{5}$$

where K_o is the number of classes from the K nearest neighbors, to sample x, that belong to class L_o.

Atiya [8] describes a method for applying a suitable optimised set of weights to the neighbours K_o. The estimate is given as:

$$\hat{P}(L_o|x) = \frac{\sum_{i=1}^{K} B_{oi} e^{w_i}}{\sum_{i=1}^{K} e^{w_j}} \tag{6}$$

where B is a matrix relating to the neighbours of x, such that $x(n)$ relates to $B(n)$, and w_i is the unconstrained optimal weight.

The optimisation of the weights w_i is completed by maximising the log likelihood of the set of weights (by using the steepest ascent algorithm).

In both density methods, an additional 'neighbour' is required, essentially of which the identity is not known. This means that this particular neighbour belongs to each class with equal probability. The effect of this is to eliminate zero probability estimates from the solution (which would compound to a minus infinity error log value later on). Alternatively, a small 'tax' is taken from the non-zero elements to soften the zeros in the density vector with a constant tax level $t = 0.01$. Both non-zero and zero elements are raised by splitting the collected sum equally amongst them all, such that:

$$P_i' = P_i(1 - t) + \frac{t}{o} \tag{7}$$

where o is the number of classes (or elements in P).

6 Performance Evaluation

The density estimation framework is evaluated through unseen cross-validation. Due to the sparse training size of the leaves data set, it is necessary to measure multiple trials of the proposed classification system.

Initially a similar method to the 'ten-fold' evaluation [24] is used. Here we extract one leaf example for each of the one-hundred species to be used as query data. The remaining 15 examples per species are used for training. This allows $n = 16$ different trials to be simulated, by alternating the query leaf through all available 16 examples, hence referred to as 'sixteen-fold' evaluation. The ACC is the performance metric of choice, simply the mean result of the n trials.

In each trial, the prior density statistics are calculated from a subset of the training data. In this method, we ensure that the trial query data is completely unseen. A series of $m = n - 1$ sub trials are computed in a similar ten-fold manor; the aggregate statistics of which are used to create the density estimation model. In this way, the query set remains unseen during the statistic gathering stage of the framework.

In addition to monitoring the ACC as the primary figure of merit, we implement a probabilistic evaluation method related to information theory and the Bayes Risk [28]. The evaluation metric is given as the mean log of the probability estimation for the true class, such that:

$$\mathrm{ELL} = \frac{1}{n}\sum_{j=1}^{n}\log\left(W_j\right) \tag{8}$$

where W_j is the density estimation value for for the correct class index for trial j.

The Bayes risk is minimised as the estimator maximises the expected log value, in this case as ELL approaches $\log(1)$.

The various density estimation models can be compared through the above evaluation processes. Further investigation is carried out with the leading methods in order to monitor classification accuracy as a function of training size.

7 Results & Discussion

In this section, the classification accuracy results from the two data sets are reported. First, the relatively simpler iris data set, in Section 7.1; secondly the one-hundred species leaves data set, in Section 7.2.

7.1 Iris

The performance results using the iris data set are shown in Table 1. The PROP and WPROP density estimation methods were tested in this case with training size set to 49 samples. The PW is the strongest feature that performs almost equally as well when compared to the PL and PW combination; as well as the SW and PW combination, i.e any other feature performs stronger when in combination with PW. Curiously, but perhaps because of the low level feature types, two of the single features perform better on their own then the combined set. The WPROP methods performance comparably to PROP with regards to mean classification accuracy, both reported to have 96%. However, with regards to the ELL performance metric, WPROP does indeed give an improved density estimation. Comparatively, Guvenir et al [24] report a classification accuracy of 90.7% for the standard K-NN definition on the iris data set (using a similar $k = 3$ parameter and a ten-fold evaluation). The authors also test their weighted K-NN metric, and produce an accuracy of 94%. In both cases, using a feature vector consisting of all four of the PW, PL, SW and SL scalar values.

Method	PL	PW	SL	SW	ELL	ACC
PROP			✓	✓	-0.402	50.00
				✓	-0.473	53.33
		✓		✓	-0.244	58.00
	✓			✓	-0.241	60.00
			✓		-0.378	60.00
		✓	✓	✓	-0.230	78.67
	✓		✓		-0.190	79.33
	✓		✓	✓	-0.229	80.00
		✓	✓		-0.187	80.67
	✓	✓	✓	✓	-0.144	86.00
	✓	✓		✓	-0.128	92.00
	✓	✓			-0.098	92.00
	✓				-0.213	92.67
		✓			-0.213	96.00
WPROP				✓	-0.517	53.33
			✓	✓	-0.446	55.33
		✓		✓	-0.237	57.33
			✓		-0.388	60.00
	✓			✓	-0.233	62.00
	✓		✓		-0.181	76.67
	✓		✓	✓	-0.245	79.33
		✓	✓	✓	-0.242	80.00
		✓	✓		-0.174	80.67
	✓	✓	✓	✓	-0.156	86.00
	✓				-0.171	92.67
	✓	✓		✓	-0.118	93.33
	✓	✓			-0.080	94.67
		✓			-0.127	96.00

Table 1. Classification results of the iris data. Check marks in the PL, PW, SL and SW fields indicate which features were combined.

7.2 Leaves

A contrast to the Iris data set results is observed, shown in Table 2. Here, the combination of the complex features are in aid of improving the classification accuracy.

It is observed that both the density methods perform comparably, with 96% mean classification accuracy. However again, the WPROP method is dramatically improved with regards to the ELL (Expected Log Likelihood) metric with -0.554 from -0.814 for WPROP and PROP, respectively, (using the SHA+TEX+MAR features).

The SHA feature performs worse than the other two feature types, perhaps due to the 'misalignment' problem indicated previously, see Section 4.1; where an occasional

misaligned shape signal is stored for a sample, thus making that sample not quite rotation invariant as assumed.

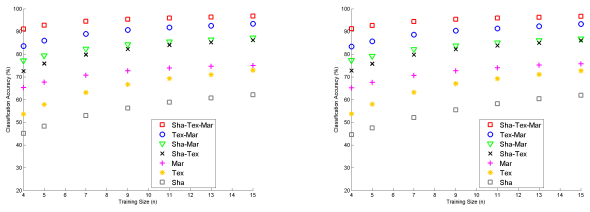
Method	SHA	TEX	MAR	ELL	ACC
PROP	✓			-1.626	62.13
		✓		-1.570	72.94
			✓	-1.553	75.00
	✓	✓		-1.210	86.19
	✓		✓	-1.193	87.19
		✓	✓	-1.139	93.38
	✓	✓	✓	-1.685	96.81
WPROP	✓			-1.491	61.88
		✓		-1.636	72.75
			✓	-1.240	75.75
	✓	✓		-1.330	86.06
	✓		✓	-1.287	86.75
		✓	✓	-1.238	93.31
	✓	✓	✓	-0.956	96.69

Table 2. Classification results of the leaves data set using the combination of the SHA, TEX and MAR features.

The classification accuracy as a function of training size is shown in Figure 2. Similarly, the Expected Log Likelihood metric as a function of training size is shown in Figure 3. These results both use the various combinations of the SHA, TEX and MAR features.

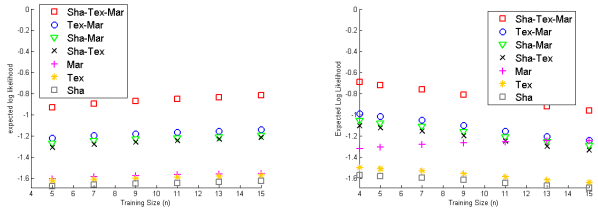
With regards to the new one-hundred species leaf data set presented in this paper, the highest performing density method tested is the WPROP method, with regards to the Expected Log Likelihood performance metric (using 15 samples per species). However, the PROP method has shown a 96.81% mean classification accuracy, 0.12% higher than the WPROP method. This result had shown the optimal combination of features was of the three available, i.e. the shape, texture and margin features.

Regarding the PROP density estimation method and the performance as a function of training size, we report that a steady decrease in classification accuracy is noted as training size is decreased. The lowest score at 91% with training size 4 and using all three available features, see Figure 2. With the K-NN classifier fixed at $k = 3$, a training size of 3 or lower produces severe errors and remains unmeasured.



(a) Using PROP. (b) Using WPROP.

Figure 2. Mean classification accuracy results of the leaves data set as a function of training size.



(a) Using PROP. (b) Using WPROP.

Figure 3. Expected Log Likelihood results of the leaves data set as a function of training size.

8 Conclusion

This paper has presented a K-NN density estimation framework with three features for species classification. It has has been shown to be effective in this particular task, coping with low training samples and a relatively large variety of categories. Confirmation of results were obtained from testing the framework with the well known iris data set and a ten-fold cross-validation evaluation.

With regards to the new one-hundred species leaf data set presented in this paper, both test density estimators performed equally well for mean classification accuracy. With regards to the density performance metric (Expected Log Likelihood), the top performing method is the PROP density estimator. The results show that, using all evaluation metrics, the proposed approach of combined density estimates from multiple features provides improved performance.

8.1 Further Work

We consider that future work on the density estimation framework using K-NN classifiers may include adapting the framework to cope with data loss and noisy feature vectors e.g. object occlusion and missing parts of the sample leaf.

We believe that many other features pertaining to shape, texture or fine-scale edge could be implanted in the suggested framework. Given that each channel is individually processed, the evaluation of the various combinations does indeed reveal which features are useful for describing the leaves.

It is entirely feasible to utilise a variety of conventional morphological shape features [3] as additional input vectors into the density estimation framework. Further work on density estimations and various other methods in this domain could be explored, in particular in attempting to maximise the performance metric of the posterior estimates.

References

[1] R. W. Scotland and A. H. Wortley. How many species of seed plants are there? *Taxon*, 52(1):101–104, 2003.

[2] R. Govaerts. How many species of seed plants are there? *Taxon*, 50(4):1085–1090, 2001.

[3] J. X. Du, X. F. Wang, and G. J. Zhang. Leaf shape based plant species recognition. *Applied mathematics and computation*, 185(2):883–893, 2007.

[4] Joao Camargo Neto et al. Plant species identification using elliptic fourier leaf shape analysis. *Computers and Electronics in Agriculture*, 50(2):121–134, 2 2006.

[5] Maria Persson and Bjorn Astrand. Classification of crops and weeds extracted by active shape models. *Biosystems Engineering*, 100(4):484–497, 8 2008.

[6] A. Hong, G. Chen, J. Li, Z. Chi, and D. Zhang. A flower image retrieval method based on roi feature. *Journal of Zhejiang University-Science A*, 5(7):764–772, 2004.

[7] K. Fukunaga and L. Hostetler. K-nearest-neighbor bayes-risk estimation. *Information Theory, IEEE Transactions on*, 21(3):285–293, 1975.

[8] A. F. Atiya. Estimating the posterior probabilities using the k-nearest neighbor rule. *Neural computation*, 17(3):731–740, 2005.

[9] Z. Wang, Z. Chi, D. Feng, and Q. Wang. Leaf image retrieval with shape features. *Advances in Visual Information Systems*, pages 41–52, 2000.

[10] Y. Shen, C. Zhou, and K. Lin. Leaf image retrieval using a shape based method. *Artificial Intelligence Applications And Innovations*, pages 711–719, 2005.

[11] André R. Backes and Odemir M. Bruno. Plant leaf identification using multi-scale fractal dimension. In *International Conference On Image Analysis And Processing*, pages 143–150. Springer Berlin / Heidelberg, 2009.

[12] André R. Backes, Wesley N. Gonçalves, Alexandre S. Martinez, and Odemir M. Bruno. Texture analysis and classification using deterministic tourist walk. *Pattern Recognition*, 43:685–694, 2010.

[13] Dalcimar Casanova, Jarbas Joaci de Mesquita Sá Junior, and Odemir M. Bruno. Plant leaf identification using Gabor wavelets. *International Journal of Imaging Systems and Technology*, 19:236–243, 2009.

[14] Jiandu Liu, Shanwen Zhang, and Shengli Deng. A method of plant classification based on wavelet transforms and support vector machines. *Emerging Intelligent Computing Technology and Applications*, 5754:253–260, 2009.

[15] J. Cope, P. Remagnino, S. Barman, and P. Wilkin. Plant texture classification using gabor co-occurrences. *Advances in Visual Computing*, pages 669–677, 2010.

[16] Elio Ramos and Denny S. Fernandez. Classification of leaf epidermis microphotographs using texture features. *Ecological Informatics*, 4:177–181, 2009.

[17] André R. Backes, Jarbas Joac de Mesquita Sá Junior, Rosana M. Kolb, and Odemir M. Bruno. Plant species identification using multi-scale fractal dimension applied to images of adaxial surface epidermis. *Computer Analysis Of Images And Patterns*, 5702:680–688, 2009.

[18] Tracy McLellan and John A. Endler. The relative success of some methods for measuring and describing the shape of complex objects. *Systematic Biology*, 47:264–281, 1998.

[19] K. Q. Weinberger and L. K. Saul. Distance metric learning for large margin nearest neighbor classification. *The Journal of Machine Learning Research*, 10:207–244, 2009.

[20] T. Denoeux. A k-nearest neighbor classification rule based on dempster-shafer theory. *Classic Works of the Dempster-Shafer Theory of Belief Functions*, pages 737–760, 2008.

[21] D. M. J. Tax, M. Van Breukelen, R. P. W. Duin, and J. Kittler. Combining multiple classifiers by averaging or by multiplying? *Pattern Recognition*, 33(9):1475–1485, 2000.

[22] R. A. Fisher. The use of multiple measurements in taxonomic problems. *Annals of Human Genetics*, 7(2):179–188, 1936.

[23] R. O. Duda and P. E. Hart. Pattern recognition and scene analysis. 1973.

[24] H. A. Guvenir and A. Akkus. Weighted k nearest neighbor classification on feature projections 1. 1997.

[25] N. Otsu. A threshold selection method from gray-level histograms. *Automatica*, 11:285–296, 1975.

[26] T. Beghin, J. Cope, P. Remagnino, and S. Barman. Shape and texture based plant leaf classification. In *Advanced Concepts for Intelligent Vision Systems*, pages 345–353. Springer, 2010.

[27] L. Kanal. Patterns in pattern recognition: 1968-1974. *Information Theory, IEEE Transactions on*, 20(6):697–722, 1974.

[28] T. M. Cover, J. A. Thomas, and J. Wiley. *Elements of information theory*, volume 6. Wiley Online Library, 1991.