Leaf Classification from Local Boundary Analysis

Anne Jorstad, Applied Mathematics and Scientific Computation, University of Maryland
David Jacobs, Department of Computer Science, University of Maryland

AMSC 664
Final Report
Spring 2008

1 Abstract

We have developed an algorithm for classifying leaves based on local analysis of their boundary curves. The wavelet transform is used to generate a coefficient vector at each boundary point over several scales. The distributions of these vectors are compared to determine similarity between leaves. This method is meant to complement an already-implemented leaf classification system whose decisions rely entirely on global shape information, and the combined results of the local and global models are presented.

2 Background

There is an ongoing project between members of the University of Maryland, Columbia University, and the National Museum of Natural History Smithsonian Institution to create an electronic field guide for plants [1]. The ultimate goal of this project is to develop a system where a user in the field can take a picture of an unknown plant, feed it into the system carried on a portable computer, and have the system classify the species and display sample images of the closest matches in near real-time. A working implementation has been developed for the leaves of woody plants of the Baltimore-Washington, DC area, a database contains 7481 leaves in 245 species. This system uses the Inner-Distance Shape Context (IDSC) [4] to generate a distance between each pair of leaves. The IDSC measures the shortest distance between two points on a path contained entirely within a figure. It is very useful for detecting similarities between deformable structures (see Figure 1).

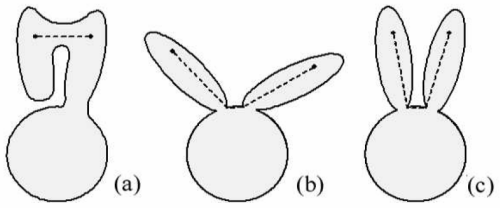
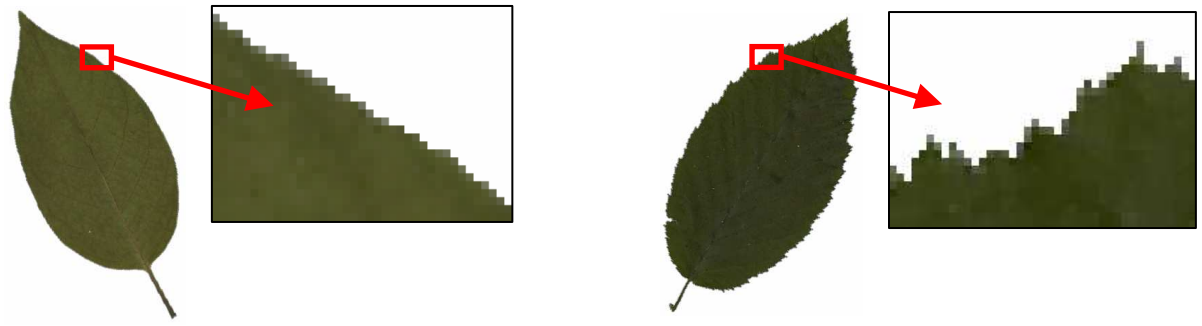


Figure 1: The Inner-Distance between two equivalent points is very similar when the bunny ears change position (b) and (c), but not when the actual structure of the object changes (a).

The current system projects points evenly around the boundary of each leaf and measures the relative Inner-Distance values between leaves, and this provides reasonable predictions of species classification for most examples. However, this system has trouble when a pair of leaves has very similar global features, even when their local serrations are quite distinct (see Figure 2).



(a) *(b)*

Figure 2: Globally similar leaves with distinct local features.
(a) Cephalanthus occidentalis (smooth boundary)
(b) Carpinus caroliniana (serrated boundary)

Leaf edges can be completely smooth, or have varying amounts of serration, and although these characteristics do affect the overall shape description of the leaves, the specific amount of variation is not taken into account in the IDSC-based system. Here, we develop a classification algorithm based exclusively on local information extracted from the leaf edges. This information is then combined with the original global shape information, and a more accurate overall classifier is trained.

3 Algorithm

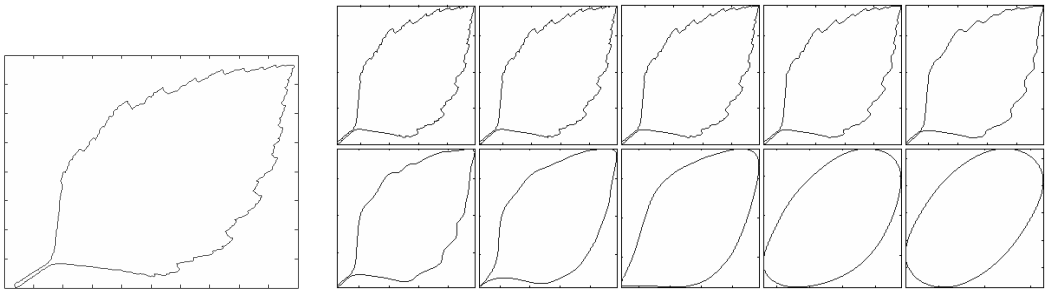
3.1 Wavelet Decomposition

The wavelet transform is used to construct a vector of local information for each boundary point on the boundary of a leaf. The boundary of each leaf is given as input, as a set of approximately 2000 discrete *(x,y)* points. The 1-D discrete wavelet transform is then applied to the boundary vector for each dimension separately. The wavelet transform converts a vector of *n* points into two vectors of *n*/2 points: a vector of approximation coefficients, φ, which provide the best approximation for the original vector given only half as many points, and a second vector of detail coefficients, ψ, which provide the extra local detail information needed to reconstruct the original vector from the approximation [2] (see Figure 3). Because the goal of this work is to make decisions based on local information, it is the detail coefficients that will be used to classify each leaf. Reapplying the wavelet transform to the approximation coefficients generates the coefficients at the next coarser scale. For the general case in one dimension:

$$\begin{aligned} f(t) &= \sum_n [d_n \psi_{1n}(t) + c_n \phi_{1n}(t)] \\ &= \sum_n [d_n \psi_{1n}(t) + c_n \sum_m [d_m \psi_{2m}(t) + c_m \phi_{2m}(t)]] \\ &= \ldots \end{aligned}$$

$$\text{where } \psi_{in}(t) := n^{th} \text{ point of } i^{th} \text{ scale}$$

In this project the wavelet coefficients are calculated via the stationary discrete wavelet transform using the 4-tap Daubechies-2 wavelet basis. Empirically it was determined that using the detail coefficients for the first three scales provides the most relevant information. After three scales, too much local information is lost to be able to provide meaningful serration classification.



(a) *(b)*

Figure 3: (a) Original leaf boundary. (b) First 10 scales of approximation curves after detail information has been removed.

For analysis, each of the original boundaries is represented by a 6-dimensional coefficient vector: the *x* and *y* detail coefficient values for each of the first three scales. It is necessary that two leaves which are the same up to a rotation should be recognized as identical, so in order to enforce this rotation invariance, one more processing step must be applied to the coefficients before they can be classified. As each boundary point is now being considered as a separate entity, we are really attempting to classify the smoothness of each point. The coarsest scale of coefficient is rotated to lie strictly on the x-axis, and all other coefficients of that point vector are rotated by this same angle:

$$\theta = -tan^{-1}\left(\frac{y_{d3}}{x_{d3}}\right)$$

$$\begin{bmatrix} x'_{di} \\ y'_{di} \end{bmatrix} \leftarrow \begin{bmatrix} cos\ \theta & -sin\ \theta \\ sin\ \theta & cos\ \theta \end{bmatrix} \begin{bmatrix} x_{di} \\ y_{di} \end{bmatrix}$$

where (x_{di}, y_{di}) is the pair of detail coefficients at the ith scale

This transformation provides a uniform orientation to all points, imposing the desired rotation invariance, and effectively reduces the degrees of freedom by one, to 5:

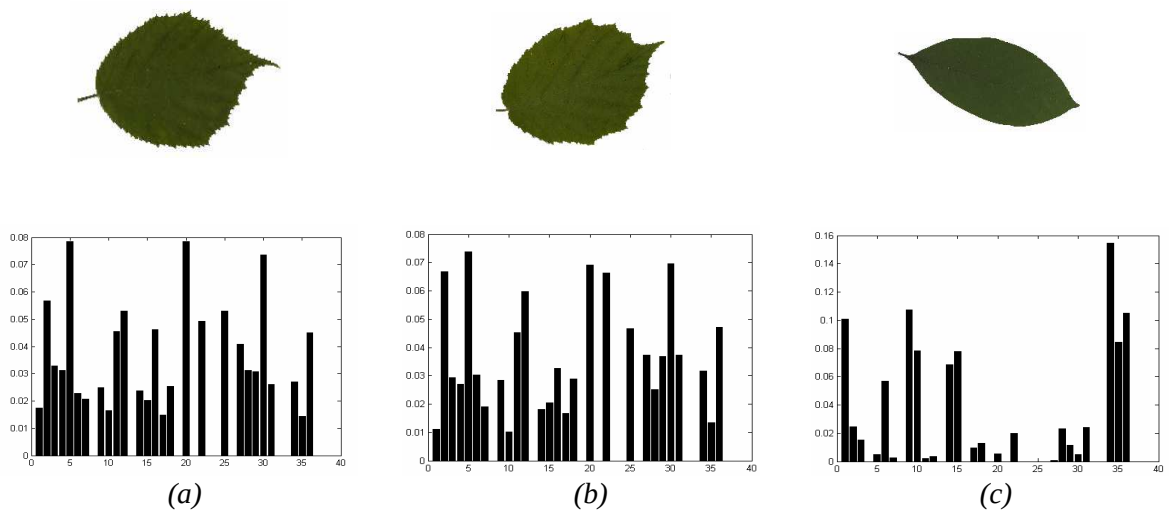
$$(x, y) \rightarrow [x'_{d1},\ y'_{d1},\ x'_{d2},\ y'_{d2},\ x'_{d3},\ 0]$$

Each leaf is now represented as approximately 2000 unordered 5-dimensional vectors, one for each boundary point. These collections of vectors must now be classified. In order to have a meaningful basis for comparison, a set of representative boundary point types must be defined, so that each leaf can be characterized by the distribution of its boundary points over these clusters.

3.2 Clustering

The K-Means clustering scheme was chosen to generate the representative clusters. In this algorithm, k initial cluster centers are chosen at random from the input data, and each point in the full data set is assigned to its closest cluster center. The cluster centers are then redefined to be the mean value of all data points associated with each cluster. This process is iterated until the cluster centers stop changing locations. In this experiment, the number of clusters, k, was chosen to be 36. Empirically, this value

should be between about 25 and 50, as using fewer than this causes all the distributions look about the same, while much more and the exact boundaries between the clusters start to play too large a role. The value of 36 has been used successfully as the number of clusters for texture analysis in the past [6]. All boundary points from one leaf of each species were fed into the algorithm, approximately 500,000 points total, and the 36 representative cluster centers were obtained. For each individual leaf, a distribution of its boundary point coefficient vectors can now be found over the 36 cluster centers, and the resulting histograms can be compared.



(a) *(b)* *(c)*

Figure 4: Leaf image and corresponding histogram for (a) Corylus americana, (b) Corylus americana, different example, (c) Asimina triloba.

The distribution between each pair of leaves can now be compared during the testing phase, using the chi-squared distance:

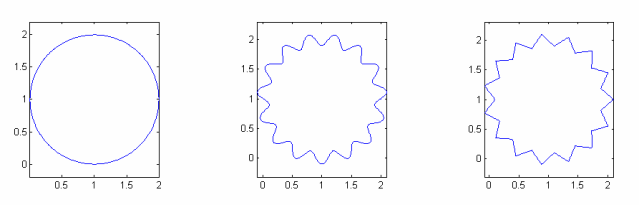
$$d = \sum_{n=1}^{36} [\ell_1(n) - \ell_2(n)]^2$$

where $\ell_i(n) = \%$ of distribution of leaf i in cluster n

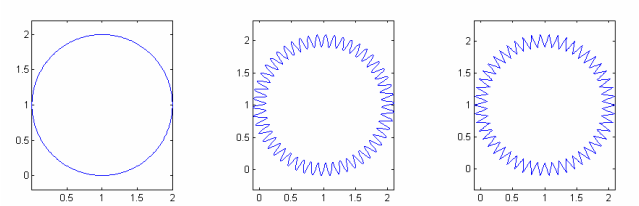
Nearest neighbor classification is used to assign an unknown leaf to the species of the leaf with the most similar distribution.

4 Validation

To demonstrate that a clustering of wavelet decompositions is actually able to capture and distinguish between local serration information, a very simple test case is processed. Three curves were constructed: a circle, a circle plus a sine curve, and a set of straight lines joined into a circular star (see Figure 5).



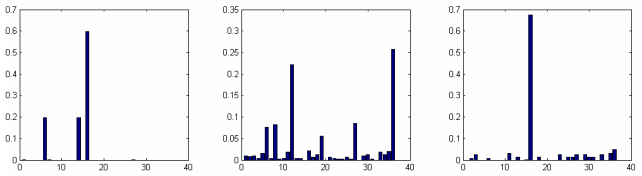
(a)



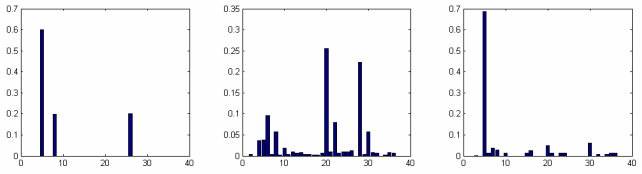
(b)

Figure 5: Simple validation curves. (a) Visual to understand the properties of each curve, with 15 peaks in each circle, (b) the actual curves tested, with 50 peaks in each circle.

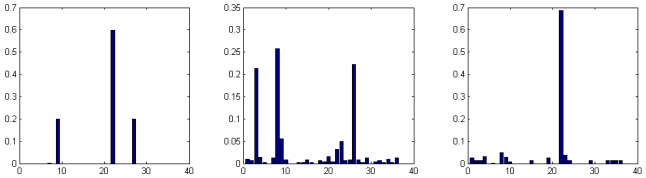
The wavelet detail coefficient vectors are generated for each curve, and then all boundary point vectors are provided as input to the K-means clustering. The 36 points that start the K-Means iterations are picked at random from the input data, so it is expected that the final cluster centers will change as the input data changes. In order to be meaningful, it is required that although the location of the cluster centers may change, the general layout of the distributions should not. This can be tested by ensuring that the distances between distributions over several repetitions remain similar. Running the algorithm several times, we observe that the actual locations of the cluster center peaks do change, but the general distribution of peaks remains similar (see Figure 6). Comparing the final distances between the three curves over several runs, we see that these relative distances remain close (see Figure 7). This is the desired result.



(a) first run



(b) second run



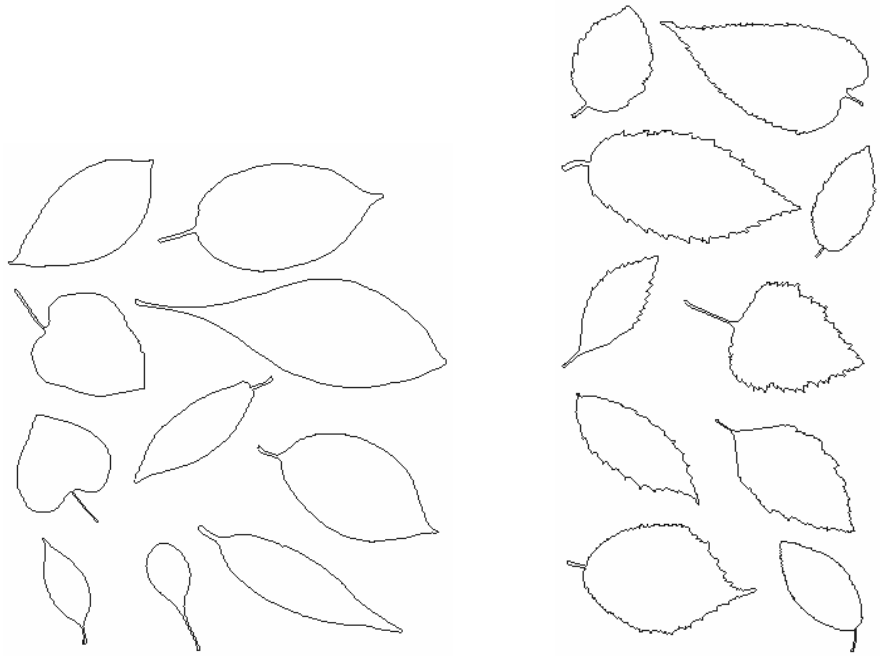
(c) third run

Figure 6: Distribution over three different sets of cluster centers generated from the validation curves. Left plots: circle; center plots: circle+sine curve; right plots: circular star.

D(1,2)	D(2,3)	D(1,3)
0.5443	0.5704	0.0944
0.5186	0.5398	0.0900
0.5232	0.5481	0.0947
0.5229	0.5357	0.0919
0.5352	0.5799	0.0856
0.5168	0.5536	0.0944
0.5087	0.5318	0.0864
0.5305	0.5623	0.0910
0.5196	0.5338	0.0923
0.5446	0.5661	0.0846

Figure 7: Distances over many trial runs between (1) circle, (2) circle+sine curve, (3) circular star..

The next step was to run the algorithm on a set of hand-picked leaves displaying the desired variation in serration. 10 species with smooth boundaries were chosen, along with 10 species with serrated boundaries, an example of each is seen in Figure 8.



(a) *(b)*

Figure 8: (a) 10 smooth species, (b) 10 serrated species.

From each species, 10 example leaves were chosen, and cluster centers were found from the boundary data of these 200 leaves. Five new examples of each leaf were used for testing. Each new leaf was predicted to be from the species of the leaf with the most similar distribution. As the algorithm only takes into account very local boundary information, it is not expected that full species classification can happen with this information alone. Instead, the wavelet distance on its own should be able to distinguish between smooth and serrated leaves, matching smooth leaves to smooth species, and serrated leaves to serrated species, even if the specific species match is incorrect. The results of this test are as follows:

Identified correct species	46%
Identified incorrect species with correct serration	100%

We see that the wavelet model is capturing local boundary information as desired.

5 Combining with IDSC

The goal is now to combine the wavelet distances with the original IDSC distances to train a better overall classifier. On the same smooth/serrated data set as above, the IDSC algorithm alone had the following results:

Identified correct species	62%
Identified incorrect species with correct serration	53%

This shows that while it is better at predicting the correct species, when IDSC fails to find the species, it predicts the correct serration no better than chance. From this comparison, we are convinced that the wavelets are capturing information that is independent from the IDSC measure, and that combining the two results should have positive results.

The distance from a new leaf to a species is the shortest distance from that leaf to any individual leaf in the training data of that species. Given the wavelet distances and the IDSC distances:

$$dW(\ell, S_k) = \text{closest wavelet distance from leaf } \ell \text{ to any leaf in species } S_k$$

$$dI(\ell, S_k) = \text{closest IDSC distance from leaf } \ell \text{ to any leaf in species } S_k$$

We want to set up a probability model to solve for the most likely species of the new leaf.

$$\text{Species}(\ell) = \underset{S_k}{argmax}\ P[\ell \in S_k \mid dW(\ell, S_k),\ dI(\ell, S_k)]$$

Using Bayes' Rule, a Naïve Bayes classification model can be constructed from probabilities than can be calculated from the training data:

$$\begin{aligned}\text{Species}(\ell) &= \underset{S_k}{argmax}\ P[\ell \in S_k \mid dW(\ell, S_k),\ dI(\ell, S_k)] \\ &= \underset{S_k}{argmax}\ \frac{P[dW(\ell,S_k),\ dI(\ell,S_k) \mid \ell \in S_k] \cdot P[\ell \in S_k]}{P[dW(\ell,S_k),\ dI(\ell,S_k)]}\end{aligned}$$

The first model is constructed to validate that combining the two sets of distances does indeed produce a better overall classification scheme. The training and test data are again the data set of 10 smooth and 10

serrated species. Here the wavelet distance is used to provide a binary serration value, based on the serration of the closest leaf in the training data:

$$Ser(\ell) = \begin{cases} 1 \; if \; \ell \; is \; serrated \\ 0 \; if \; \ell \; is \; unserrated \end{cases}$$

The IDSC distances are used to generate a rank vector for each leaf, IDSC(ℓ), ranking the possible species S_k in order from most likely to least likely, based on the closest distance from the leaf to any leaf of that species:

$$IDSC(\ell) = [S_1, S_2, ..., S_{20}]$$

The rank value R_I is then the index into this vector for any species S_k:

$$R_I(\ell, S_k) = \text{rank } \# \text{ of } S_k \text{ in } IDSC(\ell)$$

This adjusted setup with Naïve Bayes is now:

$$\begin{aligned} \text{Species Prediction of leaf } \ell \; &= \underset{S_k}{argmax} \; P[\ell \in S_k \mid R_I(\ell, S_k), \; Ser(\ell)] \\ &= \underset{S_k}{argmax} \; \frac{P[R_I(\ell,S_k), \; Ser(\ell) \mid \ell \in S_k] \; P[\ell \in S_k]}{P[R_I(\ell,S_k), \; Ser(\ell)]} \end{aligned}$$

For completeness, each of the necessary probabilities can be computed as follows:

$$P[R_I(\ell, S_k), \; Ser(\ell) \mid \ell \in S_k] = P[R_I(\ell, S_k) \mid \ell \in S_k] \; * \; P[Ser(\ell) \mid \ell \in S_k]$$
(assume independence)

$$P[R_I(\ell, S_k) \mid \ell \in S_k] = \frac{\#\; examples\; of\; S_k\; with\; S_k\; in\; rank\; R_I(\ell,S_k)}{\#\; examples\; of\; S_k}$$

$$P[Ser(\ell) \mid \ell \in S_k] = \frac{\#\; examples\; of\; S_k\; classified\; as\; serration\; Ser(\ell)}{\#\; examples\; of\; S_k}$$

$$P[\ell \in S_k] = \frac{\#\; examples\; of\; S_k}{\#\; leaves\; total} = \frac{1}{20}$$

$$P[R_I(\ell, S_k), \; Ser(\ell)] = \frac{\#\; times\; rank\; R_I\; is\; assigned\; with\; serration\; Ser(\ell)}{total\; \#\; leaves}$$

These probabilities are then smoothed to ensure that no probabilities are actually zero. The serration is smoothed with a small linear term:

$$P[Ser(\ell) \mid \ell \in S_k] = \frac{top \; + \; 0.1}{bottom \; + \; \#examples \; * \; 0.1}$$

The IDSC rankings are smoothing with a Gaussian smoothing term, where half a Gaussian is fit to the distribution over the ranks of each species separately:

$$P[R_I(\ell, S_k) \mid \ell \in S_k] = \frac{top \; + \; \phi_k(R_I)}{bottom \; + \; \sum_k \phi_k(R_I)}$$

$$\text{where } \phi_k(r) = \text{Gaussian for species } k$$

Training the classifier with the above described probability model, and testing on 5 new examples of each species, the following results were obtained:

	Wavelet alone	*IDSC alone*	*Wavelet + IDSC*
Identified correct species	46%	62%	71%
Identified incorrect species with correct serration	100%	53%	100%

It is seen that adding pure serration information has improved the overall classification results.

To construct an improved classification scheme on the entire 7481 leaf data set, several methods of combining the two sets of distance were considered. Each species of leaf in the system has been classified by a botanist as either serrated or not. But what defines serration in botany is inconsistent with the information retrieved by the wavelet serration detection. For example, a leaf with a few large scale spikes is formally classified as serrated, but locally the vast majority if its boundary is smooth (see Figure 9).

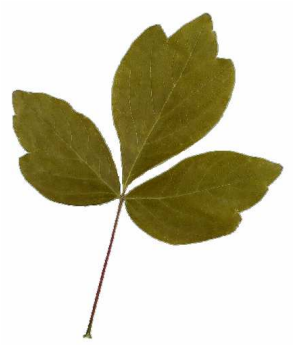


Figure 9: Species Acer negundo is classified as serrated in terms of botany, but its serrations are too large and smooth to be detected as serrated using the wavelet model.

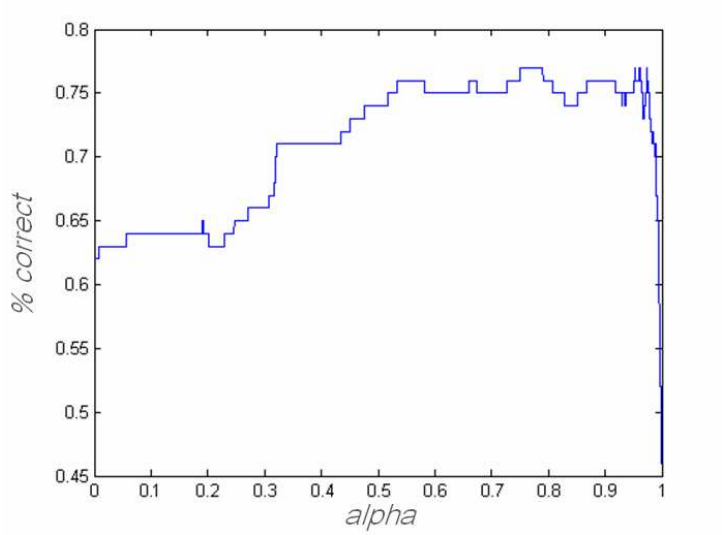
Using the Naïve Bayes system as described above on the entire data set, with the official botany serration classification determining the binary serration value of each species, the combined system is only able to predict species correctly about 20% of the time. A better way of combining the distances must be used.

Limiting the serration information a binary value loses a significant amount of information. There are many ways to include more of this information in the classifier. However, using a simple ranking system as described above for the IDSC distances is not appropriate, as this incorporates too much information from leaves whose serration type is very different from the leaf being considered. For example, in the example with 10 smooth species and 10 serrated species, if a new serrated leaf ranks a smooth species as the 12^{th} or 19^{th} most likely, this should probably not affect the classification decision very much, as long as it the smooth species more likely to be in the lower half. But forcing a rank decision for all species does not allow for this. Similarly, using the actual distances to each species type weights too heavily the insignificant distances. Neither of these models is able to produce more than 30% accuracy on the smooth/serrated data set.

Instead, a simple linear weighting between the two distances is used, to find *dC*, the combined distance:

$$dC(\ell_1, \ell_2) = \alpha \; dW(\ell_1, \ell_2) + (1 - \alpha) \; dI(\ell_1, \ell_2)$$

Trained over the smooth/serrated data set, it was found that the optimal weighting value should be 0.77 (see Figure 10).



$$\alpha_{optimal} = 0.77$$

Figure 10: The percentage of correctly classified species in the combined model as a function of the weighting constant α.

Using this model, matching each leaf to the species of its nearest neighbor using the new combined distance values, improved distances are found over the entire data set. The results are as follows:

Distance Model	***Correct Species Identified***
Wavelet Alone	20%
IDSC Alone	54%
Combined	64%

6 In Practice

In practice, the Electronic Field Guide returns images of the top 5, 10 or 20 best matches. It is therefore desirable to examine how the percentage of correctly identified species increases as the number of top matches is enlarged. A plot of the old IDSC results and the new combined results is shown in Figure 11. We see that the correct percentage increases quickly with the addition of a few more top matches, and that the combined results are indeed more accurate than the old IDSC prediction alone.

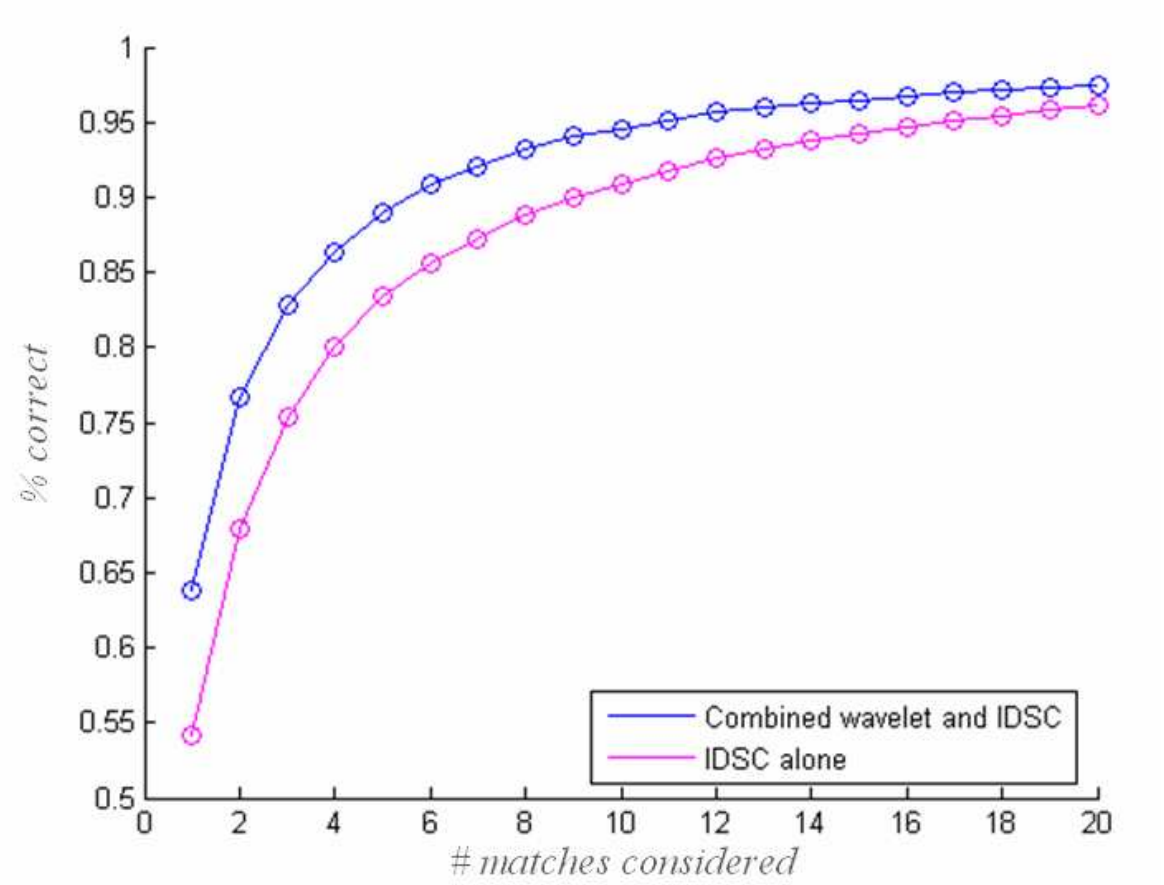


Figure 11: The percentage of correctly identified species in the top n matches, as n = 1,…,20.

In order for the electronic field guide to be practical to use over paper documents, the system must be able to return results in near real-time. Fortunately, the vast majority of calculations required for this algorithm are preprocessing. Reading in all the leaves, applying the wavelet transformations, performing the K-Means clustering, and calculating the distributions of each leaf takes several hours on a personal computer running Matlab, but this all only has to be done once. When an image of a new leaf is given to the system, only one set of wavelet transforms needs to be calculated, and the distribution of this set of points over the already known 36 clusters must be found. The one new distribution must then be compared to all distributions in the system. This whole process takes on average 0.92 seconds, easily satisfying the near real-time requirement.

7 Conclusions

It is seen that the wavelet transformation over several scales is able to capture local boundary information. This information is able to distinguish between different types of serration on the boundary of a leaf. Using this information in combination with the previously implemented IDSC leaf distance measure, a better overall classification scheme is produced. All necessary calculations can be done in real time to make this a realistic system to use in the field.

References

[1] Gaurav Agarwal, Haibin Ling, David Jacobs, Sameer Shirdhonkar, W. John Kress, Rusty Russell, Peter Belhumeur, Nandan Dixit, Steve Feiner, Dhruv Mahajan, Kalyan Sunkavalli, Ravi Ramamoorthi, Sean White. "First Steps Toward an Electronic Field Guide for Plants". Taxon, vol. 55, no. 3, Aug. 2006.

[2] Ali N. Akansu, Richard A. Haddad. Multiresolution Signal Decomposition. Academic Press, 2 edition, 2001.

[3] Cene C.-H. Chuang, C.-C. Jay Kuo. "Wavelet Descriptor of Planar Curves: Theory and Applications". IEEE Transactions of Image Processing, Vol. 5, No. 1, January 1996.

[4] Haibin Ling, David W. Jacobs. "Using the Inner-Distance for Classification of Articulated Shapes". IEEE Conference on Computer Vision and Pattern Recognition, vol. II, June 2005.

[5] Pedro F. Felzenszwalb, Jushua D. Schwartz. "Hierarchical Matching of Deformable Shapes". IEEE Conference on Computer Vision and Pattern Recognition, 2007.

[6] Jitendra Malik, Serge Belongie, Thomas Leung, Jainbo Shi. "Contour and Texture Analysis for Image Segmentation". International Journal of Computer Vision, vol. 34, no. 1, July 2001.

[7] Stephane Mallat. "A Wavelet Tour of Signal Processing". Academic Press, Chestnut Hill, Massachusetts, 1999.