A wavelet-based neuro-fuzzy system for data mining small image sets

Brendon J. Woodford **Da Deng** **George L Benwell**

Department of Information Science
University of Otago,
PO Box 56, Dunedin, New Zealand,
Email: bjwoodford@infoscience.otago.ac.nz

Abstract

Creating a robust image classification system depends on having enough data with which one can adequately train and validate the model. If there is not enough available data, this assumption may not hold and would result in a classifier that exhibits poor performance, thus lowering it's acceptability. This paper offers a solution to the problem of training and testing a neuro-fuzzy system for the purpose of image recognition when there are a limited number of images. Features of interest are segmented from each image and then used to train a neural-fuzzy system. This increases the number of data examples used to train the system. The neuro-fuzzy system is then tested on the entire data set set of full images. A high level of classification accuracy has been obtained using this method. This solution has two advantages; one, it overcomes the problem of limited data examples for training a classification model and two, rules can be extracted from the neuro-fuzzy model for further analysis. We apply this system to the problem to detection of pest damage on images of apples in New Zealand orchards.

Keywords: Multimedia data mining, spatial data mining, pattern recognition

1 Introduction

Up till now, active research in data-mining has been conducted with a number of assumptions in mind that relate to the characteristics of the data set. What happens if we can not satisfy these assumptions?

The first is that the data set can easily be obtained. If one is to apply data mining to an application area, especially one that is comparatively new, then the chance of sourcing an appropriate data set may be reduced.

The second deals with the size of the data set and follows on from the first assumption. In some disciplines, data had not been frequently collected as there was no call to do so until recently.

New Zealand's horticulture is one such area where data about certain processes, especially the production of export quality apples, has only recently begun to collect any data in a formal process that deals with apple quality. With such a limited data set, conventional data-mining algorithms may not be as applicable to elicit any knowledge about these processes.

For example when apples grow on trees in orchards, a visual inspection of their status normally governs the decision about whether the apple has been inflicted with pest damage and this is turn drives the decision about what insecticide to apply. Therefore this visual inspection process is an important part of the growers decision-making ability. They need to be able to identify different types of pest damage, some of which is very subtle and this damage may be inflicted on the apple itself, on the leaves of the apple tree, or both.

For potential apple growers, an intelligent system that could detect the type of pest, based on an image of the damaged fruit would be of potential use. More importantly, if a set of rules could be derived that aided the apple grower to aid them in detecting the type of pest damage then such knowledge could be of considerable value. However if their is little or no data from which a system could be built, then this becomes a problem, especially if the primary data type are images.

This paper proposes a method that addresses the problem of creating a suitable image recognition model when there is a limited data set from which the model can be generated. This solution has two advantages; one, it overcomes the problem of limited data examples for training a classification model and two, rules can be extracted from the neuro-fuzzy model for further analysis.

2 The problem of image recognition

Natural images like those shown in Figure 1 reflect the variability and type of the pest damage dealt to growing apples. In order to have a robust intelligent image recognition system that detects the presence of this damage normally requires many example images of these damaged apples to be presented to it for training and subsequent testing. If there are only a limited number of these images available then we need to approach the problem using a different tact.

Figure 1: Examples of pest damage to apples and leaves

Images like those in Figure 1 although they only have one apple or leaf in each shot, contain many examples of the pest damage on it. Because we are interested in the damage without too much regard to the apple or the leaf itself, it would be logical to extract the pest damage from the image and use these examples to train an appropriate classifier. This increases the number of training examples for the classifier. Testing of the classifier proceeds by comparing the trained classifier against complete images of damaged apples or leaves.

There are however three challenges to this approach. Firstly, the images of the pest damage are in full colour and of varying orientation, size, and shape. In order to successfully train a classification model, a suitable feature extraction method is required. And secondly, there needs to be a feasible way in which an entire image can be processed by the trained classifier. And finally, the selection of a classifier is difficult in the sense of finding one with satisfactory accuracy and knowledge extraction ability.

2.1 Extracting pest damage from images of damaged apples

Our approach to solving the first problem combines image segmentation and wavelet transformation to produce a set of feature vectors used to train our classifier. The colour image segmentation method devised by Ma & Manjunath (1997), Deng & Manjunath (2001) is used to segment regions which corresponds to the damage on the apple. Once the images have been segmented, the identified regions are then manually extracted from the image. Figure 2(a) displays an apple with stem damage whilst Figure 2(b) shows the output of the segmentation algorithm. When

(a) Original Image (b) Segmented Image

Figure 2: Original and segmented fruit damage

the same image segmentation algorithm is applied to leaf damage, the algorithm convienantly segments the damaged region from the rest of the leaf. Figure 3(a) is an example of leafroller damage and Figure 3(b) indicates the segmented regions of this image.

(a) Original Image (b) Segmented Image

Figure 3: Original and segmented leaf damage

Each colour image of the pest damage is resized to a 32x32 image and separated into three matrices corresponding to the original images' Red, Green, and Blue colour components. A 4-Layer 2D Daubechies wavelet transformation (DWT))(Daubechies 1990) is then applied to each colour component resulting in a 32x32 matrix. A sub-matrix of 16x16 is then extracted from the top left hand quartile of each colour component matrix resulting in 256 wavelet coefficients. The three 16x16 colour component submatrices are then combined to form a feature vector of 768 coefficients. Figure 4 outlines this process.

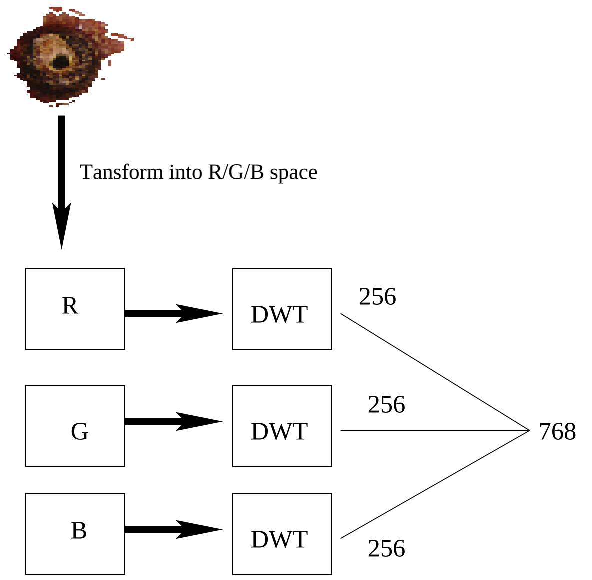

Figure 4: Process of applying wavelet transformation

For example Figure 5 is an example of the output of this process. The leftmost image is that of codling moth stem damage. This is the segmented damage taken from the image in Figure 2(b). After transforming this image using the method described in Figure 4 three sets of 32x32 wavelet coefficients, one for each of the Red, Green, and Blue colour components are generated. The middle image depicts the 32x32 set of wavelet coefficients for the Red colour component. The rightmost image is the 16x16 submatrix extracted from this 32x32 matrix and forms part of the feature vector the classifier is trained on.

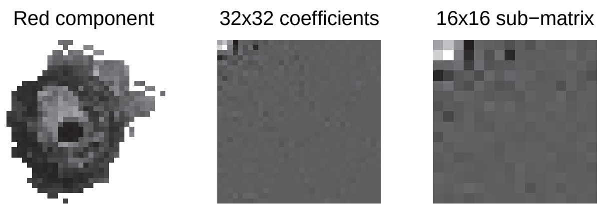

Figure 5: Output of the process in Figure 4

The justification for using the wavelet transformation is related to its multi-resolution analysis of an image, which in turn allows for the detection of the discontinuities resulted from the precense of pest damage on the apple or leaf. This has already been proven with using wavelets as the basis for image indexing (Wang et al. 1997, 1998) and image recognition (Woodford & Kasabov 2001).

2.2 The wavelet-based neural-fuzzy image recognition system

The solution to the second problem involves combining traditional image filtering techniques with a classifier creating an adaptive image filter for the purpose of image recognition.

Here we convolve a 32x32 window over the original image. The window is initially positioned at the top leftmost position of a 128x128 image and the output of each 32x32 window of the image is then converted to a set of wavelets using the scheme outlined in Subsection 2.1 that generates a set of 768 coefficients which are then passed to the classifier. The window is then shifted along by a predefined number of pixels and the another set of wavelet coefficients generated. This proces is repeated until the 32x32 window has completely covered the entire image.

The choice of size of the window was governed by finding an appropriate compromise between speed and time to process the entire image. Having a small window would incur large time delays before the system responded with an output as the window would need to be shifted many times over the image to completely cover it. If the window was too large, then subtle information related to pest damage would be lost. We choose to have a 32x32 window that was shifted 16 pixels along the image. This meant the system would have to process 64 separate segments for each image. By creating an overlapping window where we offset the shift by 24 pixels both in the horizontal and vertical directions we were able to decrease the number of segments to 49 resulting in a marked speedup. The coefficients generated from each window become the input to our classification model which can then identify the presence of a specific type of damage on the apple. Figure 6 outlines this process.

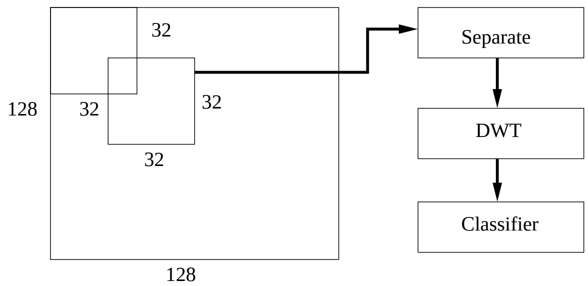

Figure 6: The image recognition process

However one question remains. What classifier do we use? We had already stated that we are working with a data set that has a limited number of examples. Traditional connectionist-based learning mechanisms such as the Multi-Layer Perceptron (MLP) (Rumelhart et al. 1986) normally exhibit good performance when they are trained and validated using a satisfactory number of examples. Indeed the problem is compounded by the fact it is a difficult task to elicit any knowledge in form of rules from these structures. Recent solutions to this problem of rule-extraction such as NEFCLASS (Nauck & Kruse 1996), FuNN (Kasabov et al. 1996), and specific approaches (Craven & Shavlik 1997) still rely on the assumption that they operate on a suitably sized data set.

The Evolving Fuzzy Neural Network (EFuNN) (Kasabov 1998*a*,*b*, 2001) addresses the problems offering an on-line, adaptive, learning mechanism, where rules can be extracted from the system. Because of its single-pass and adaptive learning capability the EFuNN can satisfactorily learn from small sized data sets as has been demonstrated in (Woodford et al. 1999, Kasabov et al. 2000). In this study the authors applied to the EFuNN to the challenging task of image recognition by training and testing it on an image data set of pest damage to apples and leaves. The same technique described in Figure 4 was used to produced a set of wavelet coefficients. Unlike this study, the entire image was transformed instead of transforming the segmented damage. A classification accuracy 48% was obtained using the EFuNN. This result was better than the other classifiers used to compare the performance of the EFuNN.

In a similar study Woodford (2001) compared the ability of the EFuNN against the Support Vector Machine (SVM) (Cortes & Vapnik 1995) using a similar data set. The results reported by this study indicated the EFuNN was a superior classifier achieving a classification accuracy of 54% compared to the SVM at 42%.

Although the results of the experiments reported in these works indicated that the EFuNN achieved a good level of performance, there was scope for improvement and in this paper we also use EFuNN as the image recognition engine for this work.

3 The recognition of pest damage on apples

3.1 Method

96 images of damaged apples were used as the data set for the experiment. The images were of three different types of pest damage. Codling moth which only inflicts damage to the apple itself, appleleaf curling midge that inflicts damage to both the leaves and fruit, and leafroller that also inflicts damage to the leaves and fruit. The distribution of the images are contained in Table 1.

Pest	Fruit	Leaf
Appleleaf-curling midge	10	33
Codling Moth	44	
Leafroller	29	44

Table 1: Distribution of pest damage examples

We then applied the method outlined in Subsection 2.1 to create a set of 160 images which were then separated into five distinct classes: appleleaf curling midge fruit damage, appleleaf curling midge leaf damage, codling moth damage, leafroller fruit damage, and leafroller leaf damage.

To train the EFuNN, 140 randomly selected instances from the pool of 160 were selected where same number of training examples were selected for each of the five classes. By adopting this method, we could then test the generalisation capability of the EFuNN when tested on an entire image. This training/testing strategy was repeated ten times. The parameters for the EFuNN were: Number of membership functions = 3, sensitivity threshold = 0.95, learning rate = 0.25, error threshold = 0.01. The single-pass learning mode was used to train the EFuNN.

Once the EFuNN was trained, it was tested on all 96 full sized images using the method outlined in Subsection 2.2. As the 32x32 subwindow was passed over the image, the contents of the subwindow were transformed into a set of 768 wavelet coefficients and passed to the EFuNN. If there was pest damage present within this sub-window, the EFuNN would output a high value close to 1.0. And if there was no pest damage the EFuNN would report a low value close to 0. For example if an image of codling moth damage was being scanned by this method, the output of the EFuNN would be [0 0 0.95 0 0] where there was a presence of this type of damage and [0 0 0.01 0.0] where there was not. The system would store the outputs of the EFuNN until the system had scanned the entire image. The vector with the highest output would then be compared to a known output vector for that entire image.

Table 2 shows the result of the testing of the EFuNN on all ten runs. With an average recognition rate of 75%, this method has dramatically increased the performance of the EFuNN compared to the previous studies.

Of the ten runs, run number 5 was the best in terms of a lower number of rules resulting in the highest classification. An analysis of the extracted rules can be found in Table 3.

From the results of the table we can see that leaf damage is harder to detect than fruit damage due

Run	RMS (Training)	No. Rules	Accuracy
1	0.0045	84	73.75%
2	0.0023	68	75.83%
3	0.0029	84	72.70%
4	0.0024	86	74.79%
5	0.0045	68	77.08%
6	0.0023	78	77.08%
7	0.0034	80	75.83%
8	0.0012	68	72.70%
9	0.0013	82	76.69%
10	0.0045	86	73.75%

Table 2: Results from ten runs of the EFuNN

Pest Damage	No. Rules
Appleleaf curling midge fruit damage	1
Appleleaf curling midge leaf damage	10
Codling moth damage	17
Leafroller fruit damage	15
Leafroller leaf damage	25

Table 3: Analysis of the rule set from the best performing EFuNN run

to the number of rules accommodating each class of damage. More interestingly was the fact that there was only 1 rule accommodating appleleaf curling midge fruit damage. This could be due to the fact that there is only a slight variation between instances of this type of damage. Leafroller leaf damage was the hardest to detect as there is a high variability between examples of this type of damage.

3.2 Discussion

The results obtained from this study indicated that the EFuNN had successfully learnt the templates and was able to generalise the templates to recognise damage that was similar or the same as what it had been trained on. Table 4 shows the confusion matrix from run 5 of the system. Each category is listed columnwise with the classification outcome listed in each row of the table.

Classified as	alm-f	alm-l	cm	lr-f	lr-l
alm-f	**7**	0	2	1	0
alm-l	0	**8**	0	2	0
cm	0	0	**28**	4	0
lr-f	2	0	7	**16**	0
lr-l	0	4	0	0	**15**
Total					**74/96**

Table 4: Confusion matrix from run 5

When a misclassification was made, the EFuNN reported a sensible result. For example when the EFuNN was tested on images of appleleaf curling midge leaf damage (alm-l), it misclassified it as leafroller leaf damage (lr-l). The same occurred with fruit damage. Codling moth fruit damage (cm) was sometimes misclassified as appleleaf curling midge fruit damage (alm-f) or leafroller fruit damage (lr-f). In most cases there were few misclassifications.

4 Comparison with other classifiers

To justify the results obtained from our study, we compared the results obtained from the EFuNN using three other classifiers. The k-means classifier as it is a well documented traditional statistical classifier, the MLP as it is also neural network, and the SVM since it is the current state-of-the-art in classifiers. The PRTools Pattern Recognition Toolbox for MATLAB (Duin 2000) was used as the implementation for these three classifiers.

The parameters used for each classifier were

- **K-means classifier:** 5 clusters
- **MLP:** 10 hidden nodes, error goal = 0.02, momentum = 0.95, learning rate = 0.01
- **SVM:** A polynomial kernel of degree 4.

Using the same training data sets created for the previous experiment, we applied the each classifier to this task. For testing the classifiers, the same method for extracting 32x32 windows from the original image was used and the feature vector presented to the classifier. The results of testing the classifiers can be found in Table 5.

Run	K-means	MLP	SVM
1	44.79%	36.45%	52.25%
2	43.75%	19.79%	53.20%
3	43.75%	34.37%	51.34%
4	43.75%	35.41%	45.36%
5	43.75%	37.50%	52.25%
6	39.58%	30.20%	56.69%
7	44.49%	38.54%	50.25%
8	44.79%	34.37%	53.34%
9	37.50%	36.45%	59.12%
10	42.70%	22.91%	57.23%

Table 5: Results from the other three classifiers

The results obtained indicated that the EFuNN compared favourably with the other three classifiers and on average outperformed them by a factor of 20%. This is an interesting result as one would expect the MLP to be good if not better than the EFuNN as it uses multi-pass training.

5 Conclusion

In this paper we have demonstrated a novel method for the purpose of constructing robust classifiers for image recognition where there are a limited number of examples with which to adequately train and test the classifier.

We have combined a number of techniques to produce a hybrid system that exploits the power of advanced neuro-fuzzy systems along with conventional image processing techniques to create a image recognition system to detect pest damage.

The added benefit of this system using the EFuNN is that knowledge in the form of rules can be extracted from this learning system. We are currently looking at ways of reducing this ruleset to make it more concise.

Future work will involve evaluating our approach against other rule extraction methods such as decision trees Mitchell (1997) and the XCS classifier (Wilson 1995).

6 Acknowledgements

We thank Professor Nikola Kasabov of the KEDRI Institute at the Auckland University of Technology for his involvement in the initial work that contributed to this research.

References

Cortes, C. & Vapnik, V. (1995), ‘Support-Vector Networks’, *Machine Learning* **20**(3), 273–297.

Craven, M. W. & Shavlik, J. W. (1997), 'Using Neural Networks for Data Mining', *Future Generation Computer Systems* **13**(2–3), 211–229.

Daubechies, I. (1990), 'The Wavelet Transform, Time-Frequency Localization and Signal Analysis', *IEEE Transactions on Information Theory* **36**(5), 961–1005.

Deng, Y. & Manjunath, B. S. (2001), 'Unsupervised Segmentation of Color-Texture Regions in Images and Video', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **23**(8), 800–810.

Duin, R. P. W. (2000), *PRTools Version 3.0: A Matlab Toolbox for Pattern Recognition*, Delft University of Technology.

Kasabov, N. (1998*a*), ECOS: A Framework For Evolving Connectionist Systems and the ECO Learning Paradigm, *in* 'Proceedings of ICONIP'98, Kitakyushu, Oct 1998', Ohmsha, Ltd: Tokyo, Japan, pp. 1232–1236.

Kasabov, N. (1998*b*), Evolving Fuzzy Neural Networks - Algorithms, Applications and Biological Motivation, *in* 'Proceedings of Iizuka'98, Iizuka, Japan', World Scientific, pp. 271–274.

Kasabov, N. (2001), 'Evolving Fuzzy Neural Networks for Supervised/Unsupervised On-Line, Knowledge-Based Learning', *IEEE Transactions on Systems, Man and Cybernetics, Part B: Cybernetics* **31**(6), 902–918.

Kasabov, N. K., Israel, S. A. & Woodford, B. J. (2000), 'The Application of Hybrid Evolving Connectionist Systems to Image Classification', *Journal of Advanced Computational Intelligence* **4**(1), 57–65.

Kasabov, N., Kim, J. S., Watts, M. & Gray, A. (1996), 'FuNN/2- A Fuzzy Neural Network Architecture for Adaptive Learning and Knowledge Acquisition', *Information Sciences - Applications* **101**(3-4), 155–175.

Ma, W. Y. & Manjunath, B. S. (1997), Edge Flow: A Framework of Boundary Detection and Image Segmentation, *in* 'Proceedings of the 1997 Conference of Computer Vision and Pattern Recognition (CVPR'97)', IEEE Computer Society, pp. 744–749.

Mitchell, M. T. (1997), *Machine Learning*, MacGraw-Hill.

Nauck, D. & Kruse, R. (1996), Neuro–Fuzzy Classification with NEFCLASS, *in* P. Kleinschmidt, A. Bachem, U. Derigs, D. Fischer, U. Leopold-Wildburger & R. Möhring", eds, 'Operations Research Proceedings 1995', Springer–Verlag: Berlin, Germany, pp. 294–299.

Rumelhart, D. E., Hinton, G. E. & Williams, R. J. (1986), *Parallel Distributed Processing, Vols 1 and 2*, The MIT Press: Cambridge, MA.

Wang, J. Z., Wiederhold, G., Firschein, O. & Wei, S. X. (1997), Wavelet-based image indexing techniques with partial sketch retrieval capability, *in* 'Proceedings of the Fourth Forum on Research and Technology Advances in Digital Libraries (ADL'97)', IEEE Press, pp. 1–9.

Wang, J. Z., Wiederhold, G., Firschein, O. & Wei, S. X. (1998), 'Content-based image indexing and searching using Daubechies' wavelets', *International Journal of Digital Libraries(IJODL)* **1**(4), 311–328.

Wilson, S. W. (1995), 'Classifier Fitness Based on Accuracy', *Evolutionary Computation* **3**(2), 149–175.

Woodford, B. J. (2001), Comparative analysis of the EFuNN and the Support Vector Machine models for the classification of horticulture data, *in* N. K. Kasabov & B. J. Woodford, eds, 'Proceedings of the Fifth Biannual Conference on Artificial Neural Networks and Expert Systems', University of Otago Press, pp. 70–75.

Woodford, B. J. & Kasabov, N. K. (2001), A wavelet-based neural network classifier for temporal data, *in* 'Proceedings of the 5th Australia-Japan Joint Workshop on Intelligent and Evolutionary Systems', pp. 79–85.

Woodford, B. J., Kasabov, N. K. & Wearing, C. H. (1999), Fruit Image Analysis Using Wavelets, *in* K. Ko & N. Kasabov, eds, 'Proceedings of the ICONIP/ANZIIS/ANNES'99 International Workshop', University of Otago Press, pp. 88–91.