Using the Bees Algorithm to Optimise a Support Vector Machine for Wood Defect Classification

D.T. Pham, Z. Muhamad, M. Mahmuddin,
A. Ghanbarzadeh, E. Koc, S. Otri

*Manufacturing Engineering Centre (MEC), Cardiff University,
The Parade, Cardiff, CF24 3AA, UK*

Abstract

This paper describes a new application of the Bees Algorithm to the optimisation of a Support Vector Machine (SVM) for the problem of classifying defects in plywood. The algorithm, which is a swarm-based algorithm inspired by the food foraging behaviour of honey bees, was also employed to select the components making up the feature vectors to be presented to the SVM. The objective of the work was to find the best combination of SVM parameters and data features to maximise defect classification accuracy. The paper presents the results obtained to demonstrate the strengths of the Bees Algorithm as an optimisation tool.

Keywords: Bees Algorithm, Support Vector Machine, optimisation.

1. Introduction

Many complex multi-variable optimisation problems cannot be solved easily. This has generated interest in “intelligent” search algorithms that find near-optimal solutions in reasonable running times. The Bees Algorithm developed in the authors’ laboratory is inspired by the food foraging behaviour of honey bees and could be regarded as belonging to the category of “intelligent” optimisation tools [1].

This paper presents an application of the Bees Algorithm to the problem of identifying defects in plywood veneer. Veneer sheets can contain defects, which could create quality problems when the sheets are bonded together. Researchers have developed systems for automatically detecting and identifying defects in plywood veneer. At the heart of such systems there is usually a classifier module that receives features of artefacts detected in wood veneer images and classifies those artefacts accordingly. Different types of classifiers have been constructed [2, 3, 4]. In this work, the Support Vector Machine (SVM), well known for its high classification accuracies, was adopted. The Bees Algorithm was employed both to optimise the parameters of the SVM and to select the features to be provided to the classifier.

The paper is organised as follows. Section 2 introduces the SVM. Section 3 explains how the Bees Algorithm was applied to the problem of SVM parameter optimisation and wood image feature selection. Section 4 presents the results obtained. Section 5 concludes the paper.

2. Support Vector Machines

2.1 The optimal hyperplane

The basic concepts of SVM are detailed in [5, 6, 7]. Given a training set of examples $(x_i,\ y_i)$, $i=1,\ 2,\ ...,\ m$ where input pattern $x_i \in R^n$ and class $y_i \in \{+1,\ -1\}$, the aim of the SVM is to find the optimal hyperplane that will classify each pattern x_i into the correct class y_i. If the patterns are linearly separable, the following expressions can be used to give the parameters w and b of the hyperplane:

$$\langle w \cdot x_i \rangle + b \geq +1 \ for\ y_i = +1 \quad (1)$$

$$\langle w \cdot x_i \rangle + b \leq -1 \ for\ y_i = -1 \quad (2)$$

Combining inequalities (1) and (2) gives [8]:

$$y_i(\langle w \cdot x_i \rangle + b) - 1 \geq 0 \quad \forall i = 1,\ ...,\ m \quad (3)$$

The SVM finds the optimal hyperplane by solving the minimisation problem represented by expression (4):

$$\underset{w,b}{Min}\ \frac{1}{2} w^T w \quad (4)$$

$$subject\ to:\ y_i(\langle w \cdot x_i \rangle + b) - 1 \geq 0$$

To solve this quadratic optimisation problem one must find the saddle point of the Lagrange function:

$$L_P(w,\ b,\ \alpha) = \frac{1}{2} w^T \cdot w - \sum_{i=1}^{m} \left(\alpha_i y_i (\langle w \cdot x_i \rangle + b) - 1\right) \quad (5)$$

where α_i denotes Lagrange multipliers; $\alpha_i \geq 0$. The saddle point can be located by minimising the Lagrange function L_P with respect to the primal variables (*w* and *b)* and maximising L_P with respect to the non-negative dual variables, α_i. The following equations are obtained after differentiating Eq. (5) with respect to w and b:

$$\frac{\partial}{\partial w} L_P = 0, \quad w = \sum_{i=1}^{m} \alpha_i x_i y_i \quad (6)$$

$$\frac{\partial}{\partial b} L_P = 0, \quad \sum_{i=1}^{m} \alpha_i y_i = 0 \quad (7)$$

Substituting Eqs. (6) and (7) into Eq. (5) yields the dual Lagrangian $L_D(\alpha)$ to be maximised:

$$\underset{\alpha_i}{Max}\ L_D(\alpha_i) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,\ j=1}^{m} \alpha_i \alpha_j y_i y_j \langle x_i \cdot x_j \rangle \quad (8)$$

$$subject\ to:\ \alpha_i \geq 0 \quad i = 1,\ ...,\ m \quad and \quad \sum_{i=1}^{m} \alpha_i y_i = 0$$

As mentioned previously, to find the optimal hyperplane, maximising the dual Lagrangian $L_D(\alpha_i)$ with respect to non-negative α_i is required. This quadratic optimisation problem can be solved by using a standard optimisation program. When the optimal values α_i^* of α_i have been determined, the optimal decision hyperplane is given by:

$$f\left(x, \alpha_i^*, b^*\right) = \sum_{i=1}^{m} y_i \alpha_i^* \langle x_i \cdot x \rangle + b^* \quad (9)$$

For non-zero α_i^*, b^* can be found from the Kuhn-Tucker condition [9]:

$$y_i\left(x_i \cdot w^* + b^*\right) - 1 = 0 \quad i = 1,\ ...,\ m \quad (10)$$

where, using Eq. (6),

$$w^* = \sum_{i=1}^{m} \alpha_i^* y_i x_i \quad (11)$$

Note that vectors x_i for which Eq. (10) holds are called support vectors.

2.2 The optimal hyperplane for non-separable data

In non-separable cases, the goal is to build a hyperplane that will produce the smallest number of classification errors. Slack variables $\xi_i \geq 0 \quad i = 1,\ ...,\ m$ are introduced in Inequalities (1) and (2) such that

$$\langle w \cdot x_i \rangle + b \geq +1 - \xi_i \quad for\ y_i = +1 \quad (12)$$

$$\langle w \cdot x_i \rangle + b \leq -1 + \xi_i \quad for\ y_i = -1 \quad (13)$$

ξ_i relax the constraints on the location of the data relative to the hyperplane. The optimisation problem becomes:

$$\underset{w,b,\xi}{Min}\ \frac{1}{2}w^T w + C\sum_{i=1}^{m}\xi_i \qquad (14)$$
$$subject\ to: y_i\left(\langle w\cdot x_i\rangle + b\right) + \xi_i - 1 \geq 0, \quad \xi_i \geq 0$$

In (14), C is a weight representing the trade-off between misclassifying certain points and correctly classifying others.

Again, the Lagrangian method can be used to solve the above optimisation problem.

The Lagrangian $L_D(\alpha_i)$ is to be maximised, i.e.

$$\underset{\alpha_i}{Max}\ L_D = \sum_{i=1}^{m}\alpha_i - \frac{1}{2}\sum_{i,\ j=1}^{m}\alpha_i\alpha_j y_i y_j\left\langle x_i\cdot x_j\right\rangle \quad (15)$$
$$subject\ to: 0 \leq \alpha_i \leq C,\ i = 1,\ ...,\ m\ and$$
$$\sum_{i=1}^{m}\alpha_i y_i = 0$$

Note that (15) is the same as (8) for the case of linearly separable data, except that α_i is now bounded by C.

The optimum hyperplane can be found as described previously once the values of α_i have been determined.

2.3 Non-linear SVM

A non-linear SVM maps the training samples from the original input space into a higher-dimensional space using a kernel function $k(.\ ,\ .)$ [9]. When applied to two points $x_i\ and\ x_j$, $k(x_i,\ x_j)$, is a generalised form of the inner product $\left\langle x_i\cdot x_j\right\rangle$ in Eq. (8).

The Lagrangian maximisation problem becomes:

$$Max\, L_D(\alpha) = \sum_{i=1}^{m}\alpha_i - \frac{1}{2}\sum_{i,\ j=1}^{m}\alpha_i\alpha_j y_i y_j k\left\langle x_i, x_j\right\rangle \quad (16)$$
$$subject\ to: 0 \leq \alpha_i \leq C,\ i = 1,\ ...,\ m\ and$$
$$\sum_{i=1}^{m}\alpha_i y_i = 0$$

Possible kernel functions include the polynomial kernel, radial basis function (RBF) kernel and sigmoid kernel [9], which are shown here as functions (17), (18) and (19).

Polynomial kernel:
$$k(x_i,\ x_j) = (1 + x_i\cdot x_j)^d \qquad (17)$$

Radial basis function kernel:
$$k(x_i,\ x_j) = \exp(-\gamma\left\|x_i - x_j\right\|^2) \qquad (18)$$

Sigmoid kernel:
$$k(x_i, x_j) = \tanh(kx_i\cdot x_j - \delta) \qquad (19)$$

3. Feature selection and parameter optimisation

In this work, the RBF was used as kernel. The classification accuracy depends on the γ value of the RBF [8, 10, 11] and the weighting factor C. These two parameters were optimised using the Bees Algorithm. The classification accuracy is also affected by the type of data to be classified, in particular, the features that make up such data. The Bees Algorithm was also employed to obtain the optimal set of features for the given wood defect classification problem.

3.1 Optimisation procedure

Figure 1 shows the main steps involved in applying the Bees Algorithm to feature selection (part I) and parameter optimisation (part II).

Part I:

(1) *Scaling*: All data were scaled between [0, 1] or [-1, 1] to avoid larger numerical ranges dominating smaller ones. Another reason for data scaling was to avoid the possibility of overflows [12].

(2) *Feature selection*: Random subsets of features were created from a set of 17 wood image attributes [14] to form “scout bees” in feature space, each subset representing a bee.

(3) *Parameter optimisation*: For each of the scout bees formed in step (2), the best combination (γ, C) was found.

(4) *Neighbourhood search*: Bees were recruited for the more promising sites discovered by the scout bees.
(5) *Parameter optimisation*: For each recruited bee, the best combination (γ, C) was found.
(6) *Random search*: Other scout bees were distributed randomly in search space.
(7) *Parameter optimisation*: The best combinations of (γ, C) parameters were found to the new scout bees. Steps (4) – (7) were repeated till some stopping criteria were met.

Part II
(1) *Parameter selection*: Tuples (γ, C) were randomly picked from the allowed parameter space, $\gamma \in [0.1, 1]$ and $C \in [900, 1200]$, to form "scout bees" in that space, each tuple representing a bee.
(2) *SVM construction*: SVM classifiers were constructed corresponding to the bees formed in step (1), the given set of features and the data to be classified.
(3) *Fitness evaluation*: The accuracies of the SVM classifiers constructed in step (2) were determined.
(4) *Neighbourhood search*: Bees were recruited for the more promising sites discovered by the scout bees.
(5) *Random search*: Other scout bees were distributed randomly in search space.
(6) *SVM construction*: SVM classifiers were constructed corresponding to the recruited bees formed in step (4), other scout bees formed in step (5), the given set of features and the data to be classified.
(7) *Fitness evaluation*: The accuracies of the SVM classifiers constructed in step (6) were determined. Steps (1) – (7) were repeated until some stopping criteria were met.

3.2 SVM network training procedure

The purpose of SVM training was to achieve the minimum classification error. The error associated with an input pattern was computed as the sum of the squared differences between the desired and actual outputs of the network corresponding to the presented pattern. Error calculation was repeated for all the patterns in the training set and the error components for all the patterns were summed to yield the value of the error function for the SVM network.

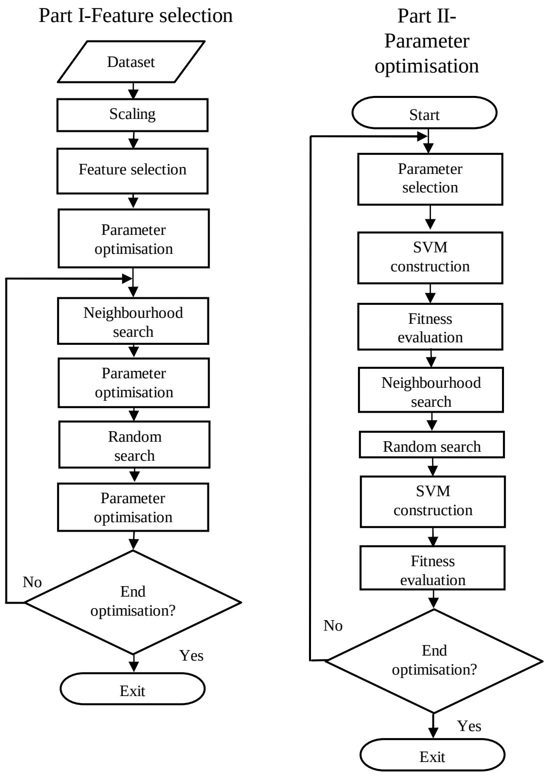

Fig. 1. Feature selection and parameter optimisation procedures.

As mentioned previously, for feature selection, each bee represented a selected feature subset. The Bees Algorithm was used to search the space of features to the feature subset producing the smallest value of the error function.

Table 1
Parameters used for feature selection and SVM parameter optimisation

Parameter	Parameter opt.	Feature selection and parameter opt.	
n	35	25	10
m	10	5	5
nsp	20	15	5
nep	30	20	10
e	4	2	2
ngh_γ	0.1	-	0.1
ngh_C	1	-	1

In parameter optimisation, each bee represented a gamma (γ) value and a penalty error (C) value. The Bees Algorithm was employed to search for the combination (γ, C) giving the smallest error.

Table 1 shows the parameters of the Bees Algorithm used. An explanation for these parameters can be found in [1]. Two different cases were studied: (i) only the parameters of the SVM were to be optimised and (ii) both the feature set and the SVM parameters were to be optimised.

The optimisation procedure consisted of the following steps.

1. Generate an initial population of bees.
2. Apply the training data set to determine the value of the error function associated with each bee.
3. Based on the error value obtained in step 2, create a new population of bees comprising the best bees in the selected neighbourhoods and randomly placed scout bees.
4. Stop if the value of the error function has fallen below the predetermined threshold or after repeating a certain number of iterations.
5. Else, return to step 2.

Table 2
Pattern classes and the number of examples used for training and testing

Pattern class	Total	Used for training	Used for testing
1	20	16	4
2	20	16	4
3	20	16	4
4	16	13	3
5	20	16	4
6	8	6	2
7	20	16	4
8	20	16	4
9	20	16	4
10	20	16	4
11	20	16	4
12	20	16	4
13	8	6	2
Total	232	185	47

3.3 Experiments

The wood veneer defect data used for the work reported in [2] was adopted in this study. The data had been obtained using an Automated Visual Inspection (AVI) system [13, 14]. The data consists of 232 examples with 17 features. Table 2 shows thirteen different classes of artefacts and the number of examples in each class. The initial classification of these examples had been performed by human inspection. For the support vector machine experiments, for each class, 80% (185 in total) of the examples were selected randomly for the training set and the remaining 20% (47 in total) for the test set. The LIBSVM software (version 2.83) [15] was used to implement SVM classifiers when (γ, C) and the training examples were provided.

4. Results

Figure 2 and Table 3 show the classification accuracies obtained for the two cases studied. The Bees Algorithm was applied 18 times to optimise the SVM parameters alone and another 4 times to optimise both SVM parameters and the set of input features.

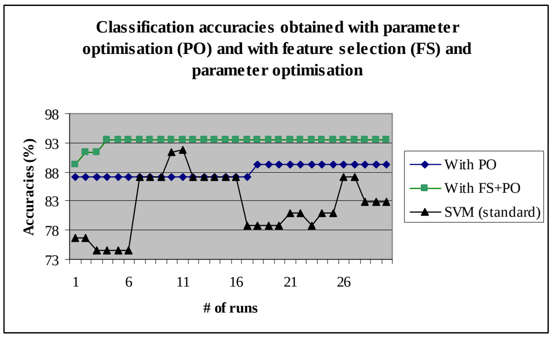

Fig. 2. Classification accuracies for 30 runs of the Bees Algorithm.

Table 3 also presents the results for a standard SVM. It can be seen that using the Bees Algorithm to optimise the SVM and feature set has increased the classification accuracy compared to the standard case.

Table 3
Defect identification accuracies

Method	Accuracy (%)(mean)
SVM with parameter optimisation	88.16
SVM with feature selection and parameter optimisation	93.33
SVM (standard)	82.42

5. Conclusion

This paper has presented the application of the Bees Algorithm to the problem of feature selection and parameter optimisation for a Support Vector Machine for the task of classifying defects in wood veneer sheets.

The accuracies obtained are higher than that obtained with a standard SVM. This work therefore reconfirms the usefulness of the Bees Algorithm as an optimisation tool.

6. Acknowledgements

This work was supported by the European Regional Development Fund (ERDF) Objective 1 SUPERMAN Project and by the EU FP6 I*PROMS Network of Excellence.

References

[1] Pham D. T., Ghanbarzadeh A., Koc E., Otri S., Rahim S. and Zaidi M. *The bees algorithm – a novel tool for complex optimisation problem*. In: Pham D.T., Eldukhri E., Soroka A. J. ed(s) 2nd Virtual International Conference on Intelligence Production Machines and Systems (IPROMS 2006). Elsevier, Oxford, 2006, pp 454-459.

[2] Pham D. T., Soroka A. J., Koç E., Ghanbarzadeh A., Otri S. and Packianather M. *Optimising neural networks for identification of wood defects using the Bees Algorithm*. 4th International IEEE Conference on Industrial Informatics. (Aug. 2006), Singapore, pp 1346-1351.

[3] Pham D. T. and Alcock R. J. *Smart inspection systems: techniques and applications of intelligent vision*. Oxford: Academic Press, 2003.

[4] Ruz G. A. and Estévez P. A. *Image segmentation using fuzzy min-max neural networks for wood defect detection*. In: Pham D.T., Eldukhri E., Soroka A. J. ed(s) 1st Virtual International Conference on Intelligence Production Machines and Systems (IPROMS 2005). Elsevier, Oxford, 2005, pp 183-188.

[5] Cristianini N. and Shawe-Taylor J. *An introduction to support vector machines*. Cambridge: Cambridge University Press, 2000.

[6] Vapnik V. *Universal learning technology: Support Vector Machine*, Special Issue on Information Utilizing Technologies for Value Creation, NEC Journal of Advanced Technology. Vol. 2(2), 2005, pp 137-144.

[7] Cortes C. and Vapnik V. S*upport-vector networks*, Machine Learning. Vol. 20(3), (Sep. 1995), pp 273-297.

[8] Huang C. L. and Wang C. J. *GA-based feature selection and parameters optimization for support vector machines*. C.-L. Huang, C.-J. Wang / Expert Systems with Applications. Vol. 31(2), 2006, pp 231–240.

[9] Burges C. *A tutorial on support vector machines for pattern recognition*. Data Mining and Knowledge Discovery. Vol. 2, 1998, pp 121–167.

[10] Cherkassky V. and Ma Y. *Practical selection of SVM parameters and noise estimation for SVM regression*. Neural Netw. Vol. 17(1), (Jan. 2004), pp 113-126.

[11] Zhang X. and Jiao L. *Simultaneous feature selection and parameters optimization for SVM by immune clonal algorithm*. Advances in Natural Computation. 2005**,** pp 905-912.

[12] Hsu C. W., Chang C. C. and Lin C. J. *A practical guide to support vector classification*. Available at http://www.csie.ntu.edu.tw/~cjlin/papers/guide/guide.pdf, (2003).

[13] Pham D. T. and Alcock R. J. *Automatic detection of defects on birch wood boards.* Proc. I Mech E, Part E, J. of Process Mechanical Engineering. 210, 1996, pp 45-52.

[14] Packiananther M. S. and Drake P. R. *Identifying defects on plywood using a minimum distance classifier and a neural network*. In: Pham D.T., Eldukhri E., Soroka A. J. ed(s) 1st Virtual International Conference on Intelligence Production Machines and Systems (IPROMS 2005). Elsevier, Oxford, 2005, pp 543-548.

[15] Chang C. C. and Lin C. J. *LIBSVM: a library for support vector machines*, 2001. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.