Available online at www.sciencedirect.com
ScienceDirect
Journal of Sound and Vibration 299 (2007) 672–682
JOURNAL OF SOUND AND VIBRATION
www.elsevier.com/locate/jsvi

Short Communication

Impact detection in an aircraft composite panel—A neural-network approach

J.R. LeClerc, K. Worden*, W.J. Staszewski, J. Haywood

University of Sheffield, Department of Mechanical Engineering, Mappin Street, Sheffield S1 3JD, UK

Received 19 January 2006; received in revised form 8 June 2006; accepted 24 July 2006
Available online 9 October 2006

Abstract

This paper presents the latest results from a programme of work aiming to design impact detectors for structures. The results are from an aircraft component which is substantially more complex than the structures previously investigated by the authors. The paper also illustrates three different approaches to the impact location problem, namely, regression, classification and a combination of both.

1. Introduction

This short paper presents the latest results in a programme of research aiming to design impact location and quantification systems for aircraft structures. In the previous stages, typified by Worden and Staszewski [1], the experimental structures were rather simple in material and geometrical terms and it was impossible to say with confidence whether the observed successes were due to the power of the analysis methods or to the simplicity of the structures. The current paper makes a significant advance in the programme of work by investigating a full-scale aircraft component of considerable complexity.

Another objective of this paper is to investigate an alternative analysis method to that proposed in Ref. [1]. In that study, a neural network was used to predict the x- and y coordinates of the impact on the basis of time-varying strain data recorded from a number of sensors. This is a *regression* approach to the problem. An alternative strategy is to divide the structure into a number of substructures and simply aim to determine which substructure an impact occurs in. Because the problem is simply to assign one of a small number of labels, it is ideally posed in terms of *classification*. Again, a neural network can be applied. Another possible strategy is to use a combination of the two approaches; in the first place one can use a classifier to locate the substructure containing the damage and then one can use a regression network trained specifically for that substructure to locate the damage more accurately.

The layout of the paper is as follows: Section 2 describes the experimental structure and how the primary measurement data were recorded and converted to features for analysis. Section 3 gives the results of applying the regression methodology to the location problem and the Section 4 gives the results from the classification

*Corresponding author.
E-mail address: k.worden@sheffield.ac.uk (K. Worden).

0022-460X/$ - see front matter © 2006 Published by Elsevier Ltd.
doi:10.1016/j.jsv.2006.07.019

approach. Section 5 describes the results of the combined approach. The paper is rounded-off with brief conclusions.

2. Test setup, data capture and feature extraction

The structure under investigation was a substantial composite aircraft component—a section of one of the flaps. A photograph is shown in Fig. 1. Due to the geometrical complexity of the structure it is difficult to give a concise summary of the dimensions, if a top view is taken as in the schematic in Fig. 2, the dimensions are approximately 1000 mm × 710 mm. In Fig. 2, the screwlines show the positions of the major ribs and spars, there are also numerous stringers attached to the underside of the centre region. The interior of the component below the leading and trailing edges is filled with aluminium honeycomb as indicated in Fig. 3.

The top surface of the component was instrumented with 9 piezoceramic patches. These were PIC-155 piezoceramics (manufactured by PI Ceramics), each was 10 mm in diameter and 1 mm thick. The positions of the sensors are indicated in Fig. 2. The previous study in Ref. [1] not only constructed the impact location neural networks, but also showed how to carry out sensor optimisation as a subset selection procedure using a genetic algorithm. To carry out such a program here would have required more sensors and rather more instrumentation, so a fairly minimal sensor set was placed here according to Engineering judgement. The impacts were applied using a PCB instrumented hammer. The levels of force applied were kept low in order to avoid damaging the structure. As the object of the exercise was damage location, the force signals were not recorded. The component was placed on a substantial metal table on two layers of bubble wrap. As the component is nominally symmetrical about the centreline and the sensors are symmetrically placed, impacts were restricted to half of the top surface, i.e. centre region U in Fig. 2 and the corresponding halves of the leading and trailing regions.

As discussed later, two sets of measurements were taken. The first comprised 251 impacts taken at random positions over the region of interest. The second set of 317 impacts were taken on a regular mesh of 6 cm × 6 cm cells over most of the region of interest—the cells were 4 cm × 6 cm on the leading edge. The exact position of the impact was marked by coating the hammer tip with chalk before the impact.

The primary data recorded were the time-varying strains at each sensor location. The data were recorded using an *NCode* system attached to a laptop running *NSoft* software (in fact only the *DataGate* options were

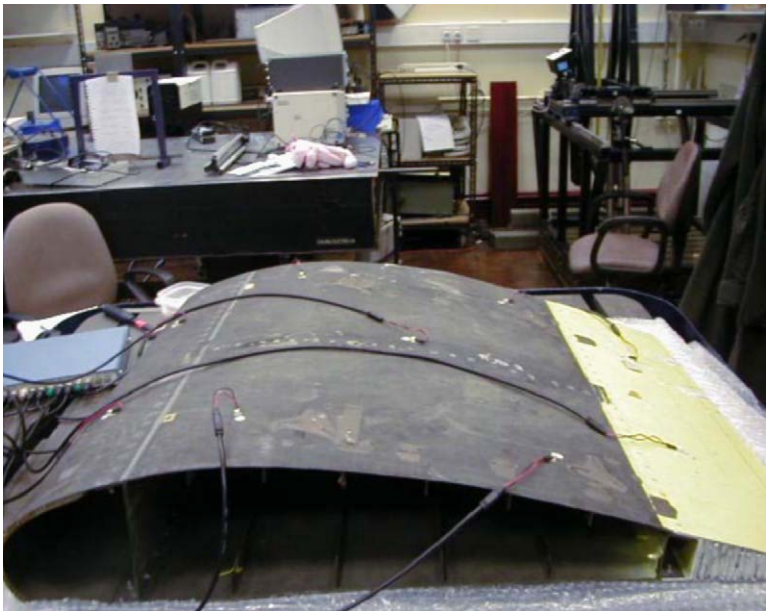

Fig. 1. Composite aircraft component.

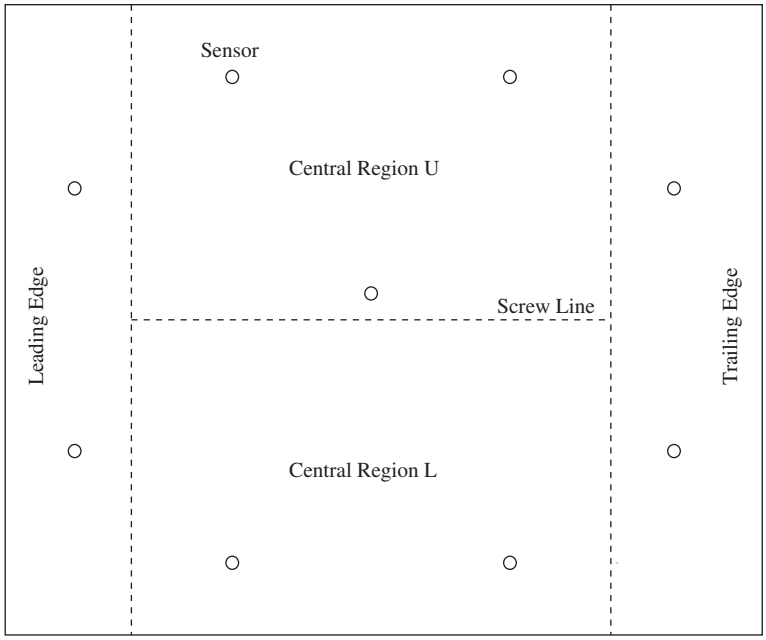

Fig. 2. Schematic of aircraft component showing sensor positions.

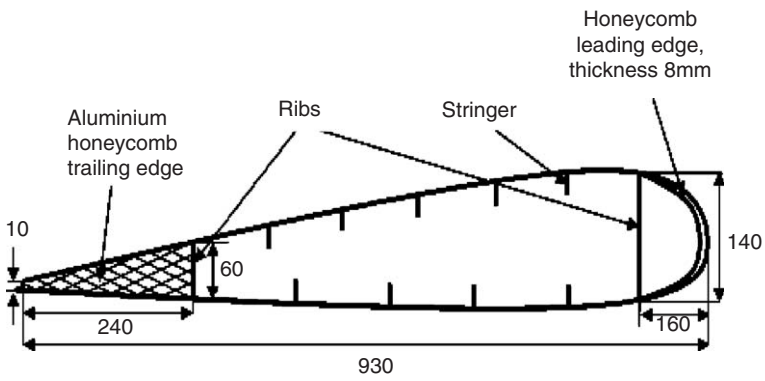

Fig. 3. Schematic representation of the wing flap section. Dimensions in mm (not to scale).

used). The sampling frequency was 6250 Hz and 2001 points were sampled for each channel per impact. A pre-trigger was set to ensure that all of the appropriate waveforms were recorded.

Once the primary data were recorded, the next task was to extract appropriate low-dimensional features for the pattern analysis to follow. A number of features were assessed: arrival time of the signal, risetime, maximum amplitude, arrival time of maximum and duration. In order to facilitate this stage, each of the signals was mean-removed and enveloped using a procedure based on the Hilbert transform as follows.

The signals from each sensor were polluted by spurious mean levels introduced by the instrumentation; the signals were therefore adjusted by removing the mean. This gave signals for the wave arrivals at the various sensors as illustrated in Fig. 4. In order to facilitate determining the start and end of the signal, the signals were enveloped. First the Hilbert transform of each signal $a(t)$ was obtained as follows

$$\tilde{a}(t) = \frac{-1}{\pi} P \int_{-\infty}^{\infty} \frac{a(t)}{\tau - t} \, \mathrm{d}\tau, \tag{1}$$

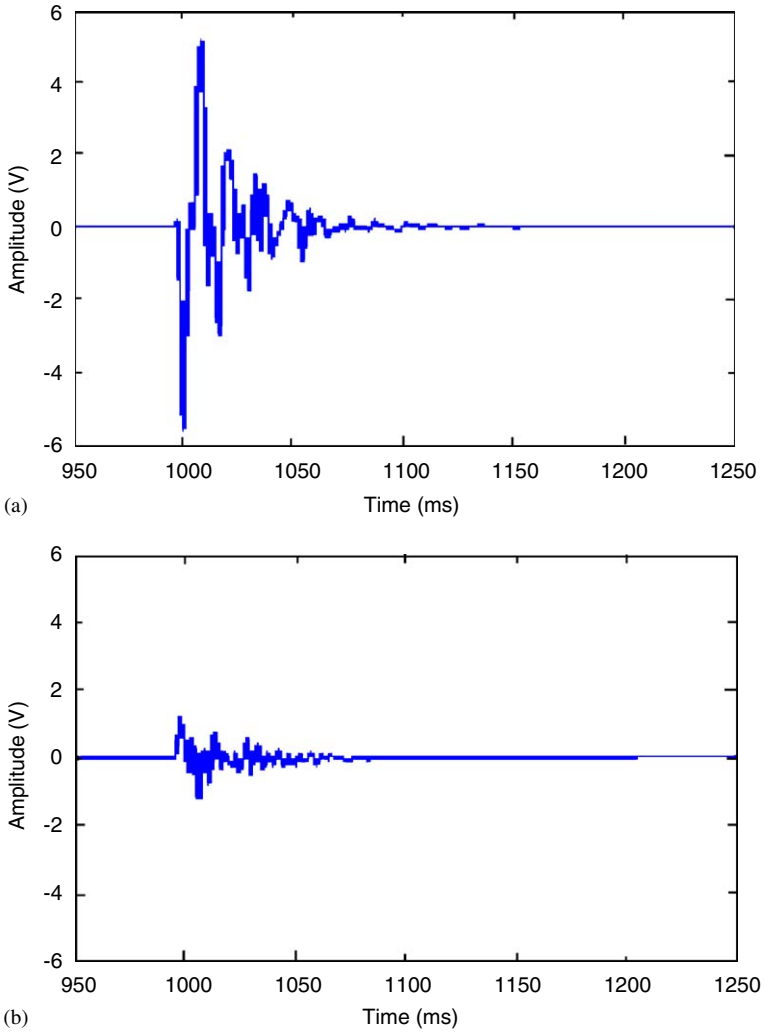

Fig. 4. Example of normalised sensor signals: (a) sensor 4 and (b) sensor 5. Impact location coordinates (417.5,175.5).

where P denotes the Cauchy principle value. Next, the so-called analytic signal $s(t)$ was formed from

$$s(t) = a(t) + i\tilde{a}(t) \tag{2}$$

and finally, the envelope $e(t)$ was computed from

$$e(t) = \sqrt{a(t)^2 + \tilde{a}(t)^2}. \tag{3}$$

The envelopes of the signals given in Fig. 4 are shown in Fig. 5.

On the basis of experience acquired during previous experiments [1], It was decided to use the maximum amplitude and arrival time. So for each impact, two features were extracted per channel, this gave an 18-dimensional feature vector. A more principled approach to feature selection could have been made which extended the GA optimisation described in Ref. [1] to an extended set of candidate features for each sensor. This strategy was not adopted here as the main concern was simply to see if an impact locator could be obtained for the complex structure using a similar feature set to that used for a simple structure.

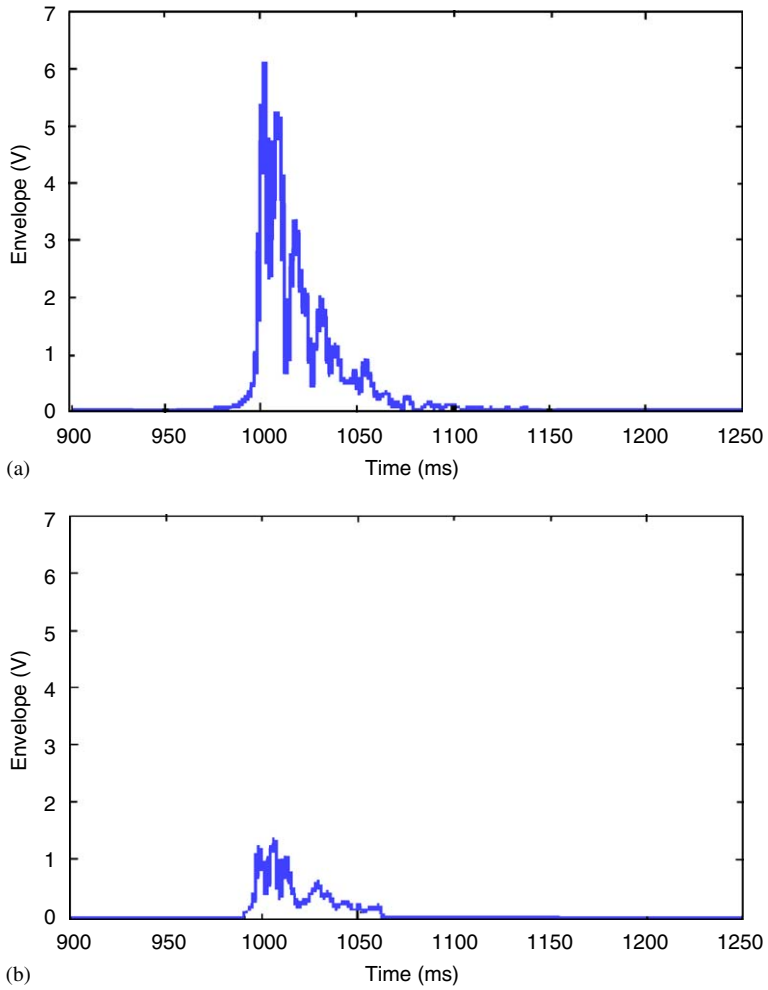

Fig. 5. Examples of signal envelopes corresponding to signals in Fig. 4.

3. Results from a regression approach

The first approach to impact location investigated was a regression approach. This means a neural network was trained to predict the x and y coordinates of the impact when presented with the features extracted from the time-varying strains. As in Ref. [1], a multi-layer perceptron (MLP) neural network was used [2]. As there were 18 input features, this was the required number of input neurons for the network. The outputs were required to be the x and y coordinates of the impact, so two output neurons were needed. This leaves the number of hidden neurons undecided. In order to choose this number in a principled manner, the procedure suggested in Ref. [2] was adopted. The experimental data were divided into three sets: a training set, a validation set and a testing set. The randomly distributed impacts were used for the training sets, while the impacts defined on a regular mesh were divided between the validation and testing sets. Many neural networks were trained with varying initial conditions, numbers of hidden nodes and stopping times. The network weights were fixed using the training data only. The optimum network structure was defined as that which minimised the error over the validation set. The error was defined as in Ref. [1]. The mean x error (in mm) was multiplied by the mean y error to give an error in terms of area; this was then normalised and presented as a percentage of the overall test area (approximately half the surface area of the component). Having minimised the validation error by adjusting the network parameters, the final performance of the network was assessed by generating the area error over the testing set.

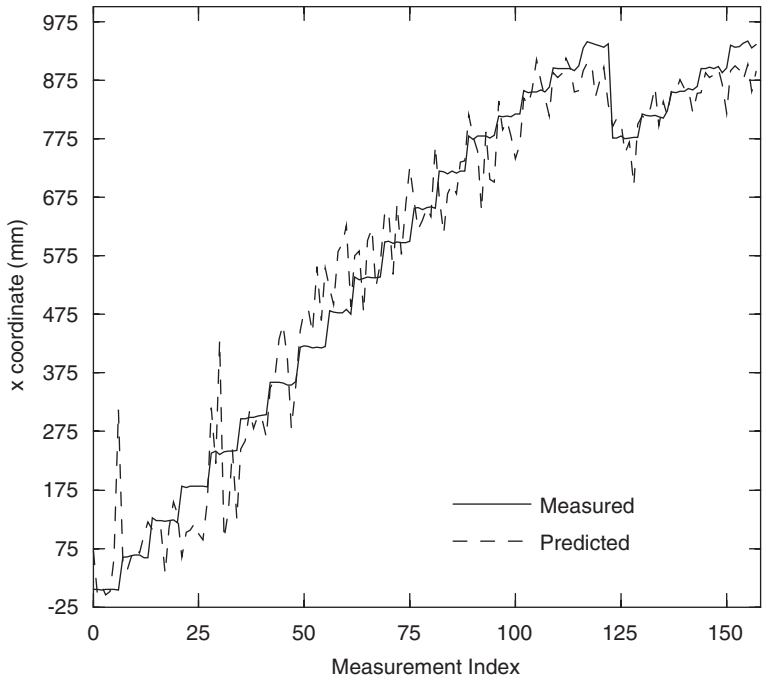

Fig. 6. Comparison of measured x coordinate of impact (solid line) and regression network prediction (dashed line).

After carrying out this procedure, the best validation error by a network was 1.76%. (This corresponded to a training error of 0.87%.) In absolute terms, the mean x coordinate error was 44.3 mm and the y coordinate error was 33.6 mm, giving an absolute error area of 1488.48 mm^2. The optimum network had 20 hidden neurons and was trained for 70 000 cycles. Note that according to usual practice, the amount of training data available (251 patterns) would not be adequate to train a network of this size. The usual rule of thumb is that 10 training patterns are required for each weight in the network and the 422 weights in this network suggest that 4220 patterns should be available for training. However, controlling the length of training time acts as a means of regularisation [3] and could alleviate overtraining. The early stopping approach can be shown to be equivalent to adding noise in training, which was the regularisation approach used in Ref. [1]. Further support for the idea that the network is not overtrained is provided by the fact that the best network was assessed using an independent testing set, and an error of 1.92% was obtained. Overall, this compares favourably with the result obtained in Ref. [1] of 1.5% for a considerably less complex structure instrumented with more sensors.

Figs. 6 and 7 show comparisons between the predicted and measured x and y coordinates, respectively. Agreement is fairly good with the majority of large errors in the region of the boundaries.

Returning briefly to the question of overtraining, Fig. 8 shows the best validation error of all the trained networks as a function of the number of hidden units. The area error is seen to stabilise quite quickly. Although the optimum of 1.76% is obtained at 20 hidden units, the next best validation error is at only 9 units, corresponding to 191 weights. This network has a validation error of 1.81% and a corresponding test error of 2.31%. If one were to set a maximum allowed error of 2.0%, one would find that the network with 6 hidden units is sufficient. This network gives a validation error of 1.95% and test error of 2.07%, i.e. a reduction of 14 hidden units causes a deterioration of only 0.15% in the test error. The network with 6 hidden units has 128 independent weights, so even this network would cause some concern from the point of view of overtraining if one had not used an appropriate regularisation scheme.

4. Results from a classification approach

As discussed above, an alternative approach to the location problem is to try and locate the damage to within a substructure. This converts the problem into a classification problem with each substructure

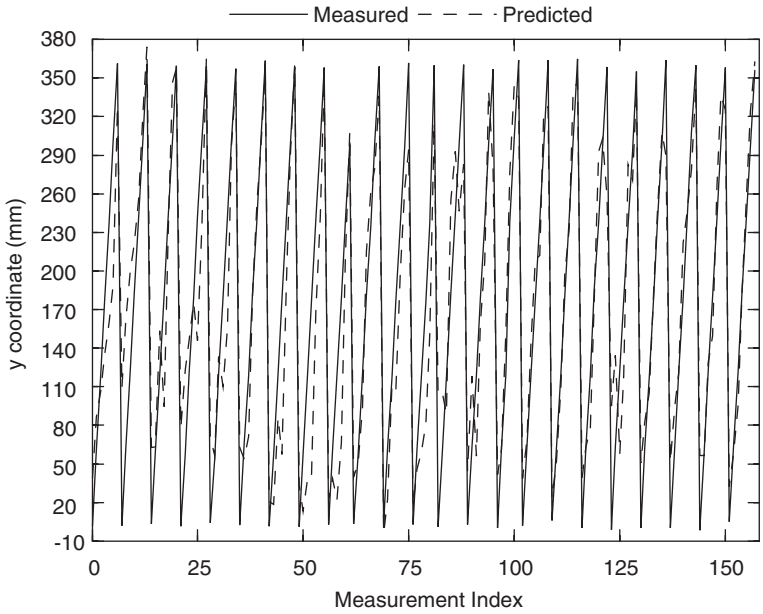

Fig. 7. Comparison of measured y coordinate of impact (solid line) and regression network prediction (dashed line).

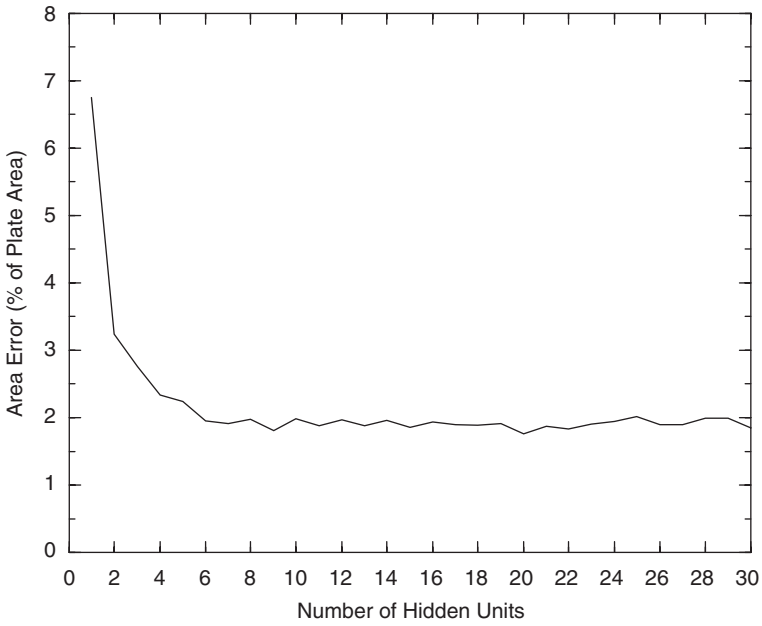

Fig. 8. Regression network validation error as a function of the number of hidden units.

represented by a unique class label. For the purposes of illustration, the structure here is divided into three regions (see Fig. 2): the upper-half of the leading edge, centre region U and the upper-half of the trailing edge.

The same input features are used as for the regressor. This time the neural network is trained as a classifier and has one output assigned to each class (i.e. three outputs). During training, if a pattern corresponding to class i is presented, the network is asked to respond with a unit value at output i and zeros at the other outputs. This approach is termed the *1 of M* strategy. It can be shown [3], that if the network is trained in this manner,

Table 1
Confusion matrix for best neural network-testing set

Prediction	1	2	3
True class 1	65	2	0
True class 2	2	60	1
True class 3	0	0	28

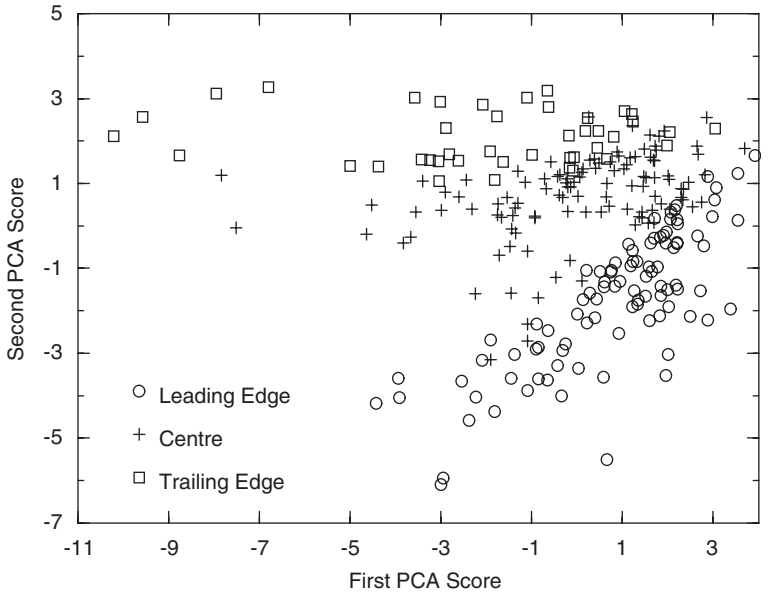

Fig. 9. PCA projection of classification training data: leading edge data (circles), central region (crosses), trailing edge (squares).

it will respond not with ones and zeros, but with the a posteriori class probabilities. The optimal decision rule is therefore to assign the class label corresponding to the highest output (most probable).

As with the regression network, the number of hidden nodes, stopping times, etc. are established by training many networks and optimisating the results on the validation set. In this case, the optimum corresponds to the minimum classification error. When the networks were trained on the features here, the best network gave a validation probability of error of 0.025 (corresponding to training probability of 0). The best network had only two hidden nodes and was trained for 80 000 cycles. When the testing data was presented, the neural network gave a probability of error 0.032, corresponding to 5 misclassifications over the testing set of 158 points. The confusion matrix for the classifier is given in Table 1. This classifier has only 42 independent weights, so having used a regularisation strategy as before, there were no real concerns about overtraining as there were approximately 5 patterns per weight.

The excellent results here with only two hidden units suggest that the data for the three classes are separable in a fairly simple manner. This was investigated using principal component analysis (PCA). The 18-dimensional feature data were projected onto the first two principal components in order to give a two-dimensional representation as an aid to visualisation [4]. The first two PCA scores for the training data are plotted in Fig. 9. It is clear from this picture that the data are simply separable as reflected in the zero misclassification rate on the training data. (Note that there appears to be some overlap, this is due to the fact that the first two principal components only retain 51% of the variance.)

When the testing data are projected onto the same principal components, the scores shown in Fig. 10 are obtained. The five misclassifications indicated in Table 1 are highlighted. As one would expect, they are in the regions of overlap between classes. The coordinates of the impacts which were misclassified are: (242, 358),

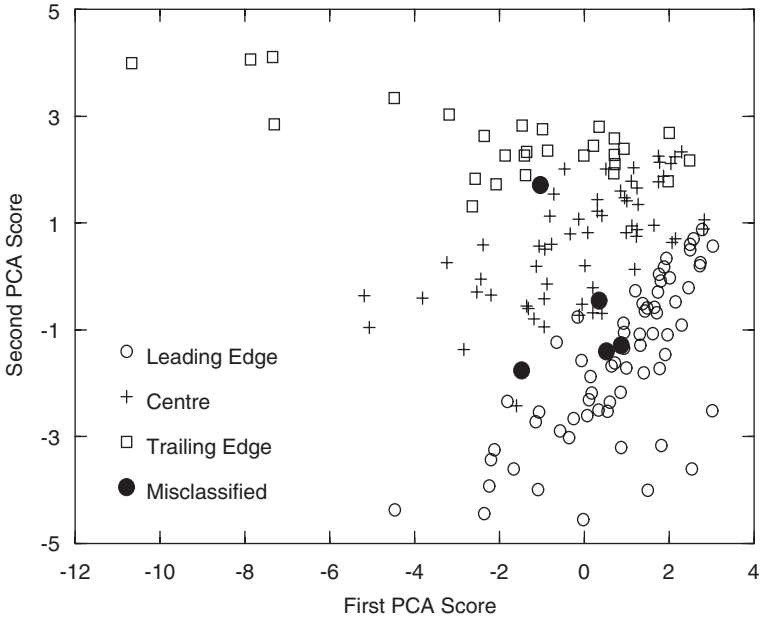

Fig. 10. PCA projection of classification testing data: leading edge data (circles), central region (crosses), trailing edge (squares), misclassified points (filled circles).

Table 2
Confusion matrix for the two-class classifier network-testing set

Prediction	1	2
True class 1	65	2
True class 2	2	89

(657, 360), (720, 360), (779, 119) and (814, 0). The first of these is in the geometrical region of the border between the trailing edge and the central region, the rest are in the area between the leading edge and the central region.

5. Results from a combined approach

The classifier has proved very successful. It provides a high classification rate and only makes mistakes on impacts which are near the borders of the class regions. This opens up a new line of enquiry. If the three geometrical regions are easy to classify, one might successfully train a regression network for each region. Data would first be passed to the classifier to assign the appropriate network. In order to investigate this, the leading edge data were separated out. (They are arguably the more important data, as the leading edge is most likely to suffer an impact during flight.)

As before, a number of neural networks were trained. The outputs were defined so that the first indicated the class membership of leading edge impacts, the second output simply indicated that the impact was elsewhere on the surface. The lowest probability of misclassification on the validation set occurred for a network with a single hidden node. This probability was 0.0126 corresponding to 2 out of 159 impacts misclassified. The probability on the testing set was 0.0253 corresponding to 4 out of 158 impacts. The confusion matrix for the classifier is given in Table 2. This network had only 23 independent weights, so in this particular case there were in fact more than 10 training patterns per weight.

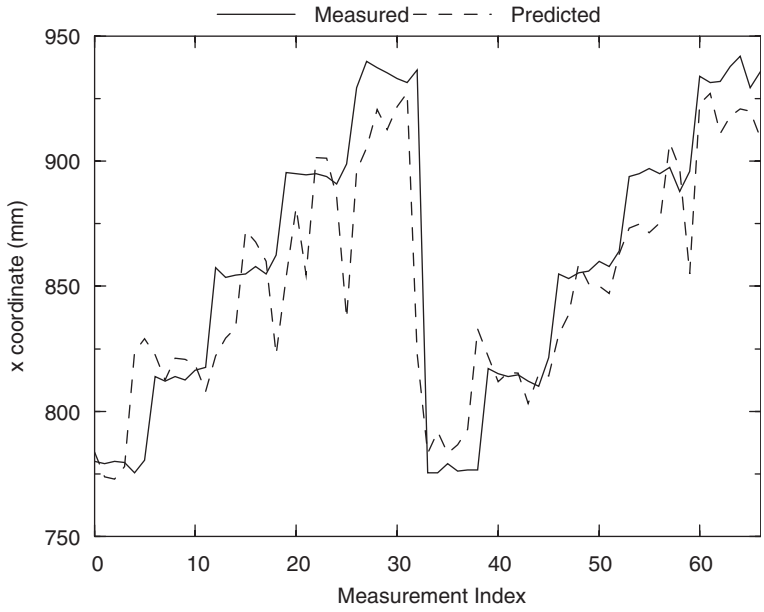

Fig. 11. Comparison of measured x coordinate of impact (solid line) and leading edge regression network prediction (dashed line).

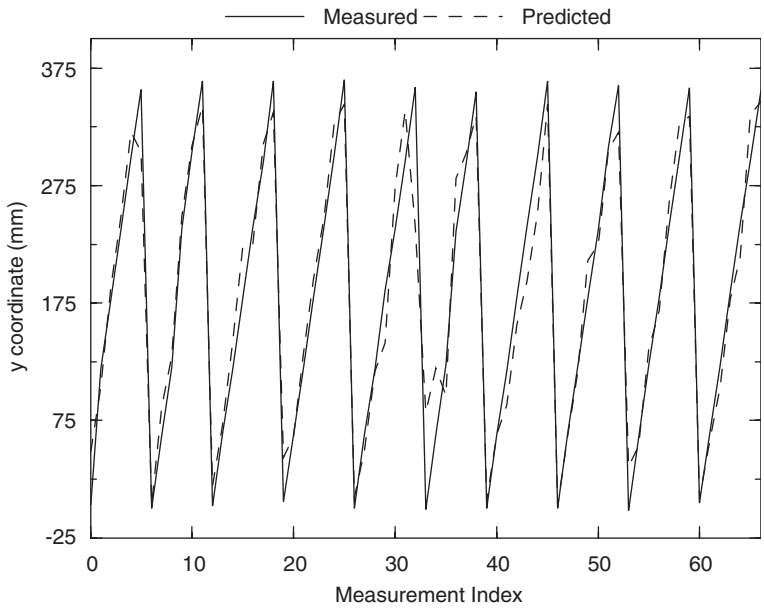

Fig. 12. Comparison of measured y coordinate of impact (solid line) and leading edge regression network prediction (dashed line).

Having established that a classification network could effectively identify impacts on the leading edge, the leading edge impacts only were used to train a regression network to refine the impact location.

The results were very good. The training run gave an area error of 0.04%, the validation error was 0.50% and the final test error was 0.48%. (Note that these values are still normalised with respect to half of the area of the top surface of the component.) The main improvements over the results shown in Figs. 9 and 10 were in the regions of the boundaries as shown in Figs. 11 and 12. As an aid to visualising the performance of the leading edge network, Fig. 13 shows the error area superimposed on the actual structure.

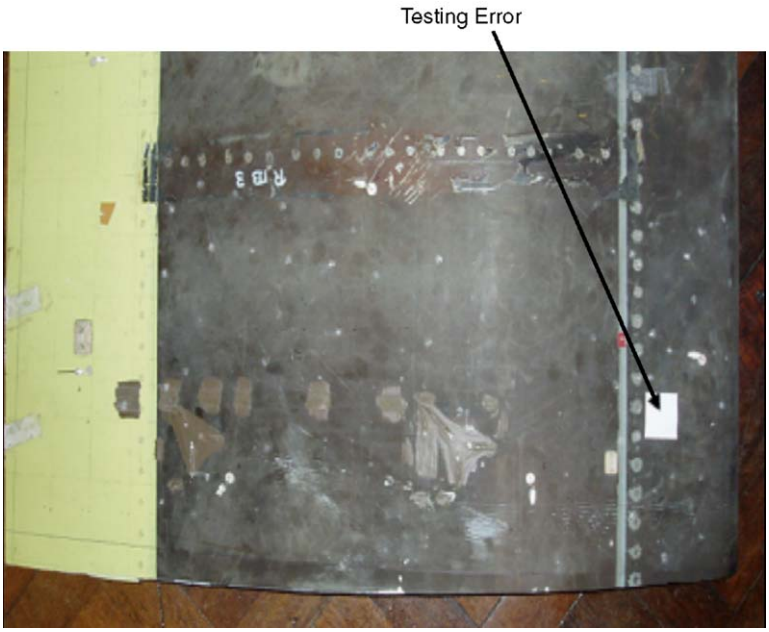

Fig. 13. Localisation error for leading edge regression network on testing set.

6. Conclusions

The conclusions from this work are fairly straightforward. In the first case, it is shown that the neural-network-based impact location strategy developed elsewhere for simple structures, is applicable to larger structures with a high degree of material and geometrical complexity. It is also shown that the impact location can also be solved as a classification problem, by localising the impact to within a small number of substructures. A hybrid approach is proposed whereby an initial classifier assigns a substructure to the impact and then a regression network specifically trained for that substructure refines the location estimate by providing coordinates. This approach is briefly illustrated here and shown to have great promise.

There is an important caveat relating to this work which must be mentioned. Due to the expense of the experimental structure and the intention to work with it further, it was considered undesirable to cause any damage at this stage. This meant that all impacts were conducted at levels unlikely to cause damage. This meant in turn that the various neural networks were trained to locate only *non-damaging* impact events. In reality, it is likely that a working system would be required to discard such events and only relate information about *damaging* impact events. The important outcome of the current work is that impacts can be located on complex structures using features from strain wave propagation in the structure; further work remains to show that these features contain information which distinguishes between non-damaging and damaging impact events and whether the features allow accurate identification of the location of the former.

References

[1] K. Worden, W.J. Staszewski, Impact location and quantification on a composite panel using neural networks and a genetic algorithm, *Strain* 36 (2000) 61–70.
[2] L. Tarassenko, *A Guide to Neural Computing Applications*, Arnold, Paris, 1998.
[3] C.M. Bishop, *Neural Networks for Pattern Recognition*, Cambridge University Press, Cambridge, 1998.
[4] K. Worden, G. Manson, Visualisation and dimension reduction of high-dimensional data for damage detection, *Proceedings of the 17th International Modal Analysis Conference*, Florida, USA, 1999.