

## Research Article

# Locating Impact on Structural Plate Using Principal Component Analysis and Support Vector Machines

Heming Fu<sup>1</sup> and Qingsong Xu<sup>2</sup>

<sup>1</sup> Department of Computer and Information Science, Faculty of Science and Technology, University of Macau, Macau

<sup>2</sup> Department of Electromechanical Engineering, Faculty of Science and Technology, University of Macau, Macau

Correspondence should be addressed to Qingsong Xu; [qsxu@umac.mo](mailto:qsxu@umac.mo)

Received 27 January 2013; Accepted 27 March 2013

Academic Editor: Chengjin Zhang

Copyright © 2013 H. Fu and Q. Xu. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

A new method which integrates principal component analysis (PCA) and support vector machines (SVM) is presented to predict the location of impact on a clamped aluminum plate structure. When the plate is knocked using an instrumented hammer, the induced time-varying strain signals are collected by four piezoelectric sensors which are mounted on the plate surface. The PCA algorithm is adopted for the dimension reduction of the large original data sets. Afterwards, a new two-layer SVM regression framework is proposed to improve the impact location accuracy. For a comparison study, the conventional backpropagation neural networks (BPNN) approach is implemented as well. Experimental results show that the proposed strategy achieves much better locating accuracy in comparison with the conventional approach.

## 1. Introduction

Damage detection is very meritorious in specifying and diagnosing the location and severity of damage for decreasing accidents and maintaining the stability of various structures [1, 2]. Structural health monitoring involves the process of realizing the damage detection and characterization for the concerned structures [3]. In order to implement structural health monitoring, a variety of approaches have been reported in previous works [4, 5]. Majority of the investigations have been focused on impacted metallic and composite plate structures which are widely used in different engineering scenarios such as aircrafts and automobiles. This research is concentrated on the study of impact location detection for a metallic structural plate subject to low-velocity impacts.

In the literature, various soft computing algorithms have been popularly employed for signal processing dedicated to impact detection [6]. For instance, Jones et al. implemented the location and magnitude estimation by applying backpropagation neural networks (BPNN) to experimental data, which were acquired by sensors attached on a fully clamped plate [7]. The authors adopted several methods to improve the accuracy of detection, such as removing DC shift and high

frequency noise and only using impacts within the range of training set. More recently, it has been demonstrated that the neural-networks-based impact location strategy is applicable to more complex structures such as real aircraft component along with a high degree of material and geometrical complexity [8]. Hence, it is evident that the impact detection investigations on simple structures have practical meanings.

As the damage severity is correlated with the kinetic energy of impact, it has been shown that the in-plane strain waves caused by the impact-induced damage in fiber-reinforced composite plates are associated with the type and extent of damage [9]. It is noticeable that, in order to ensure the consistence of experimental condition, the impacts are commonly produced at the levels which will not cause damage to the structures. This means that the detection methods concern nondamaging impact events.

The impact detection can be resolved as classification or regression problems. It has been well recognized that neural networks (NN) are appropriate to handle complicated classification or regression problems [10]. Neural networks are easy to implement because their realizations do not require the establishment of mathematical models. However, they exhibit some challenges in terms of overtraining and

generalization since their optimal parameters are difficult to determine. Comparing with neural networks, an alternative approach—support vector machines (SVM)—offers better generalization and more stable performance. In the literature, the SVM was employed in [11] to solve the classification problem. It was demonstrated that a result which is better than that of conventional NN can be achieved. Besides, the least squares support vector machines (LS-SVM) have been adopted in engineering applications [12, 13]. Considering that the conjugate gradient method cannot solve the KKT system of LS-SVM directly owing to the coefficient, an enhanced LS-SVM with the Hilbert transform was proposed to reinforce the convexity and computational speed [14]. However, majority of existing SVM-based works are dedicated to the classification problems [15]. It has been shown that the regression method is capable of providing better accuracy than the classification approach in solving the impact detection problem [8]. Hence, an SVM regression-based approach is devised in the present research.

Furthermore, most of regression-based approaches so far developed are based on some transforms of the original sensor signals. Then, a single-layer regression model is established to predict the output once new input occurs [7, 8]. Nonetheless, preliminary experimental investigations reveal that such an intuitive approach is not able to produce satisfactory impact detection accuracy in the current research. To this end, a new method of two-layer SVM regression is proposed for predicting the low-velocity impact location in this paper. The method consists of two general steps in terms of data preprocessing and model establishment for the prediction. In the data preprocessing part, the time-varying strain data are acquired using surface-mounted piezoelectric sensors on the plate. Before the training of SVM for subsequent prediction, the principal component analysis (PCA) is executed to extract principal components from the mass sensor output signals. As a result, less computational cost would be implemented as the dimensions of data are greatly reduced with minimal information loss. The two-layer SVM regression model is then implemented along with the model parameters adjusted using grid search and cross-validation method. The first layer contains four models which handle training data acquired from each of the four sensors, respectively. After that, the second layer of model is constructed by these four models outputs which involve underlying information of the sensor locations. Preliminary comparison shows that the two-layer SVM regression model leads to a better accuracy than the model which is built without knowing location information of sensor. To illustrate the effectiveness of the proposed detection strategy, the conventional BPNN method has been implemented for a comparative study. Results demonstrate the efficiency of the proposed strategy for the impact location detection explicitly.

The remainder of the paper is organized as follows. Section 2 presents a brief review of the PCA and SVM algorithms. The experimental setup is described in Section 3, which is then adopted for the data acquisition as conducted in Section 4 where the location detection strategy is outlined in detail. A series of experimental studies are carried out in Section 5 along with a comparative study with respect

to BPNN-based approach. Finally, Section 6 concludes this paper.

## 2. Algorithms

In this section, the basic PCA and SVM algorithms are briefly reviewed.

*2.1. PCA.* Principal component analysis (PCA) is a widely used statistical analysis technique which aims at selecting the most significant patterns or features from multivariate data set to simplify complicated problems. In the area of computer science, this technique is often responsible for “data preprocessing” or “dimensionality reduction” while extracting effective features to retain most of information of the original data. More precisely, the basic theory of PCA is an orthogonal transformation that adjusts the direction of a new coordinate system in which the variance with most of information can be projected onto the first coordinate.

PCA can also be explained by using formulas as follows. After PCA receives a data matrix  $\mathbf{X}$  (in this research,  $\mathbf{X}$  consists of all of the data in training testing sets), a set of principal components can be returned as the eigenvectors of the correlation matrix. The formula of correlation matrix  $\mathbf{R}$  can be expressed in the following form:

$$\mathbf{R} = E[\mathbf{X}\mathbf{X}^T] = \frac{1}{N}\mathbf{X}\mathbf{X}^T, \quad (1)$$

where  $N$  denotes the number of data in  $\mathbf{X}$ . Afterwards, the eigen problem

$$\mathbf{R}\mathbf{q} = \lambda\mathbf{q} \quad (2)$$

is solved, in which  $\mathbf{q}$  stands for the eigenvectors of  $\mathbf{R}$  and an associated  $\lambda$  indicates the eigenvalue of  $\mathbf{R}$ . There are many eigenvalues for  $\mathbf{R}$  which are denoted by  $\lambda_j$  for  $j = 1$  to  $m$ . Each eigenvalue  $\lambda_j$  is associated with an eigenvector  $\mathbf{q}_j$ .

As a result, the transformation function  $\mathbf{T} = [\mathbf{q}_1, \mathbf{q}_2, \dots, \mathbf{q}_m]$  can be used to calculate the projection  $\mathbf{a}$  of the new matrix  $\mathbf{Y}$  on that principal direction as shown below

$$\begin{aligned} \mathbf{a} &= [\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_m]^T \\ &= [\mathbf{Y}^T\mathbf{q}_1, \mathbf{Y}^T\mathbf{q}_2, \dots, \mathbf{Y}^T\mathbf{q}_m]^T \\ &= \mathbf{T}^T\mathbf{Y}, \end{aligned} \quad (3)$$

where the principal component  $\mathbf{a}_j$  represents the projection of  $\mathbf{Y}$  onto the principal direction.

In this research, the original data matrix  $\mathbf{X}$  is manipulated using PCA to obtain the transformation function  $\mathbf{T}$ . After that,  $\mathbf{T}$  is applied on  $\mathbf{X}$  again to calculate the projection  $\mathbf{a}$  in a new coordinate system. As the projection  $\mathbf{a}$  produced by PCA distributes in descending order in principal directions, a user-defined tolerance can be chosen to reduce the dimensions while minimizing the information loss. In this paper, the tolerance is selected as 98% by trial and error.

2.2. *SVM*. In view of the difficulties suffered by traditional neural network approaches, such as generalization and probability of overfitting, support vector machines (SVM) have been proposed which belong to supervised learning algorithms. SVM has become a popular active research area as it promotes the development of machine learning theories and techniques. SVM can be used to solve both classification and regression problems.

In classification problems, the goal of SVM is to produce a classifier that can well separate new data into two classes based on the training and testing data sets. In the case of nonlinear examples, it transforms the data in nonlinear input space to higher dimensional linear feature space through a nonlinear mapping and then finds the optimal hyperplane to classify the data.

As for nonlinear regression problems, SVM performs similar to the situation in nonlinear classifications. The nonlinear mapping can be used to map the data into higher dimensional linear feature space, and then it employs the kernel function to eliminate the effect of dimensionality. Besides, it adopts an  $\varepsilon$ -insensitive loss function given by

$$\begin{aligned} \max_{\alpha, \alpha^*} W(\alpha, \alpha^*) = \max_{\alpha, \alpha^*} & \left[ \sum_{i=1}^l \alpha_i^* (y_i - \varepsilon) - \sum_{i=1}^l \alpha_i (y_i + \varepsilon) \right. \\ & - \frac{1}{2} \sum_{i=1}^l \sum_{j=1}^l (\alpha_i^* - \alpha_i) \\ & \left. \times (\alpha_j^* - \alpha_j) K(x_i, x_j) \right] \end{aligned} \quad (4)$$

along with the following constraints:

$$0 \leq \alpha_i, \quad \alpha_i^* \leq C, \quad i = 1, 2, \dots, l, \quad (5)$$

$$\sum_{i=1}^l (\alpha_i - \alpha_i^*) = 0. \quad (6)$$

Based on the solution of  $(\alpha_i, \alpha_i^*)$ , the target nonlinear function is given by

$$f(x) = \sum_{i=1}^l (\alpha_i - \alpha_i^*) K(x_i, x) + b. \quad (7)$$

In (4) and (5), the parameters  $\varepsilon$  of the loss function and the constant  $C$  affect the calculation results, and they are usually defined by user. To generate an optimal setting of these parameters to maximize the performance of study, genetic algorithms or exhaustive search method are usually adopted. In this paper, the grid search algorithm coupled with cross-validation is chosen to find the best pair of  $\varepsilon$  and  $C$  owing to its convenience and exhaustive capability. Actually, both methods will be assigned a searching range with a small interval, and then each pair of  $\varepsilon$  and  $C$  is computed to take out the best values. As a result, an optimal pair of  $\varepsilon$  and  $C$  in a specific area could be obtained. The details of applying SVM to the present research by developing a novel two-layer SVM regression model are described in Section 4.1 later.

### 3. Experimental Setup

The experimental setup consists of a rectangular 491 mm  $\times$  392 mm clamped plate, four piezoelectric patch sensors pasted on the plate by conductive glue, a data acquisition board assembled in personal computer, and an instrumented hammer with rubber head for knocking. The experimental setup is depicted in Figure 1(a), and a photo of the plate structure is shown in Figure 1(b).

The plate is fastened onto a heavy metal table using screws located on the four corners. The four sensors are mounted on the plate surface by conductive glue. The distances between sensors and the nearest edges are 170 mm (horizontally) and 100 mm (vertically), respectively. The dimension of the plate and the location of sensors are shown in Figure 2.

For data acquisition purpose, a data acquisition board (NI PCI-4472) manufactured by National Instruments, Inc. is inserted in a PCI slot of personal computer. The time-varying data are gathered by a program developed with LabVIEW software. All the sensors are sensitive to the impacts applied on the plate. In this situation, when the hammer knocks on the plate, the voltages collected by sensors increase rapidly initially and then decay to zero quickly. During this transient response time, the sensors output voltage signals are recorded and saved in data files. The data acquisition rate is set as 100 kHz and the number of acquired samples is selected as 10000. The format of data storage in the file is presented in Figure 3.

In the data file, there is a 10000  $\times$  6 matrix corresponding to the impact. The first column is the time stamp of sampling, the second is the impact force magnitude, and the remaining four columns are the voltages acquired by the first, second, third, and fourth sensor, respectively. The last four columns of the voltage values will be used in the later training and prediction procedures.

### 4. Data Acquisition and Impact Detection Strategy

4.1. *Data Acquisition*. To detect the location of impact using SVM regression models, two sets of impact data have been acquired with the aforementioned experimental equipment. The first set includes a grid of 63 impacts (nine impacts equally separated horizontally by seven impacts equally separated vertically), which is shown in Figure 4. It is used for the SVM model training. Another set of 100 random impacts, which is differed from the training set, is presented in Figure 5.

In order to quantify the detection performance, a root mean square (RMS) error function, which was used in previous work [7], is employed here to provide a measure of detection accuracy. The RMS error function is expressed by

$$E_{\text{RMS}} = \frac{1}{n} \sqrt{\sum_{i=1}^n ((x - x_i)^2 + (y - y_i)^2)}, \quad (8)$$

where  $n$  is the amount of testing impacts,  $(x, y)$  are the real coordinates of the impacts within testing set in unit of

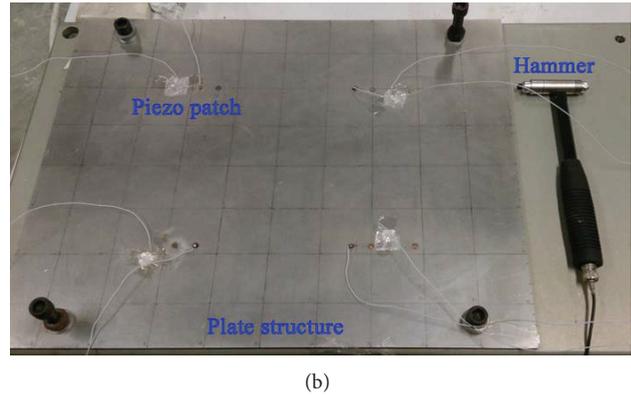
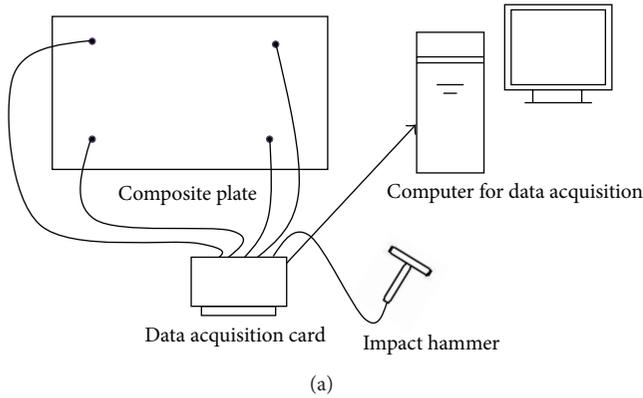


FIGURE 1: Experimental setup for impacts. (a) Schematic of hardware connection, (b) photo of the plate structure.

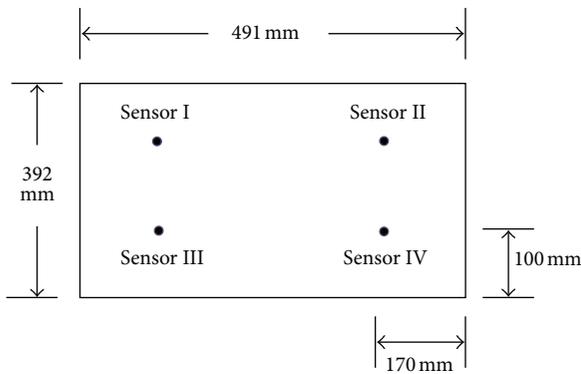
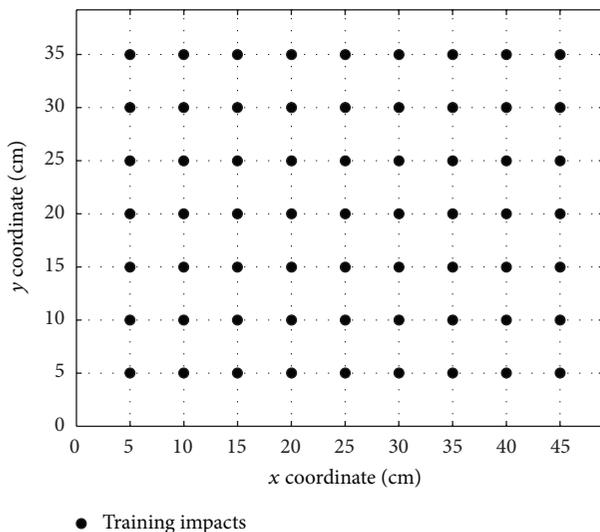


FIGURE 2: Dimensions of plate and location of sensors.

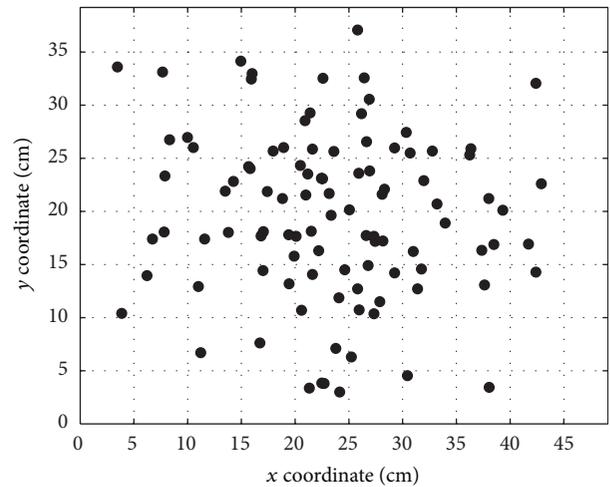
Time	Hammer	Sensor 1	Sensor 2	Sensor 3	Sensor 4
0.000000	0.001527	0.002631	0.002451	-0.002360	-0.000214
...	...	...	...	...	...

FIGURE 3: Format of data storage in data file.



● Training impacts

FIGURE 4: Distribution of training impacts.



● Testing impacts

FIGURE 5: Distribution of testing impacts.

centimeter, and  $(x_i, y_i)$  are the predicted results of SVM in unit of centimeter. The error function gives the radial distance between the real and computed impact location in centimeters.

Alternatively, another evaluation approach [6] is employed to better visualize the results. By this method, the averaged errors of  $x$  and  $y$  coordinates are calculated, respectively. Then, the product of them is considered as the evaluation factor. The smaller the ratio of the factor to the area of plate, the better the detection accuracy for the impact location.

$$P = \frac{(1/n^2) \sum_{i=1}^n (x - x_i) \sum_{i=1}^n (y - y_i)}{S}, \quad (9)$$

where  $S$  denotes the area of the plate structure and  $P$  is the ratio of error area to the whole plate area.

**4.2. Proposed Detection Strategy.** The whole procedure of the proposed impact detection strategy is depicted in Figure 6. The procedure is implemented in Matlab with toolboxes of

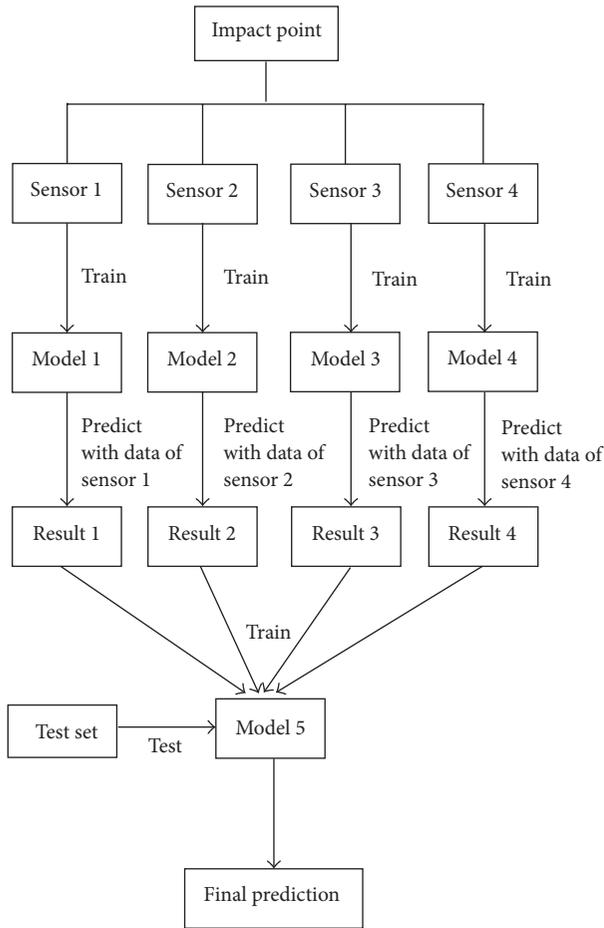


FIGURE 6: Flow chart of the proposed impact detection strategy.

PCA and SVM algorithms. Generally, the strategy contains the following two steps.

The first step is data preprocessing for data dimension reduction, which extracts the principal components from each data file using PCA. As impact happens, each of the four sensors collects 10000 voltage values. In each data file, except for the first column of time stamp and second column of impact force, one impact corresponds to the other four columns of piezoelectric sensor outputs (10000 × 4). Using PCA, all the impact data inputs into PCA are projected on another coordinate system, in which the projections of original data are distributed in descending order in the principal directions. Only a part of values represent the dominant feature of information. After defining a tolerance of information loss, 19 main components with nearly complete information are selected. Hence, the data dimension is significantly reduced from 10000 to 19 for each sensor output. With these reduced components, the result is still accurate for the purpose of impact location.

The second step concerns the regression model development. The four sensors represent four positions on the plate, and the generated main components (19 × 4 matrix) have direct relation to the impact. However, direct use of the data from those four independent sensors cannot offer a good

detection performance. To build a relationship among those four sensors, this paper proposes a new method which merges the four models corresponding to these sensors into a single model. This step is further divided into the following three substeps.

- (1) For the model training process,  $M$  data files are used. Each data file records the four sensors' outputs associated with each impact. From each data file, a  $19 \times 4$  matrix is generated using the PCA results of the four sensors' data. For the convenience of SVM regression, the matrix is transposed to generate a  $4 \times 19$  format, where each row represents one sensor output related to one impact. Thus, a total of  $M$  impacts allows the generation of an  $M \times 19$  matrix from each sensor output. Using this  $M \times 19$  matrix as the training input and the corresponding coordinates of the  $M$  impacts as desired output, one SVM regression model is obtained for each sensor output. Hence, four SVM models are developed for the four sensors.
- (2) Using the training sets in previous Step (1) as the testing sets for each SVM model, one set of the impact location is estimated by the SVM model. Hence, the four SVM models predict four sets of the impact location. It is worth mentioning that each of the four SVM models represents one sensor mounting site. Thus, the underlying reason of the different predicted impact locations lies in the different mounting site of the four sensors. Using the four locations predicted for the  $M$  impacts, that is, an  $M \times 4$  matrix, as the training input and the corresponding actual coordinates of the impacts as desired output, the fifth SVM regression model is established.
- (3) For the purpose of model testing, other  $N$  data files are used. In the testing process, the  $N$  data files are processed by the same way as interpreted in Step (1) and Step (2). That is, four matrices ( $N \times 19$ ) are generated to produce the first layer of four SVM models, and then the four SVM models' outputs ( $N \times 4$ ) are employed to establish the second layer of SVM model to predict the impact location, as depicted in Figure 6.

## 5. Results and Discussions

**5.1. SVM with PCA.** Using the approach described previously, the SVM model is trained with the training set as shown in Figure 4 and then tested using the original testing set as illustrated in Figure 5.

In order to enhance the accuracy, a grid search method coupled with cross-validation is applied to find the optimum parameters for SVM. Then, by analyzing the result, the mean error of  $x$  coordinates is found to be 2.6847 cm and the mean error of  $y$  coordinates is 2.023 cm, which give an area error corresponding to 0.28% of the plate area. Besides, the RMS errors for  $x$  and  $y$  coordinates are 4.0755 and 3.0809 cm, respectively. As a result, the overall RMS error of the results is 0.5109 cm.

The discrepancy between the predicted and actual  $x$  coordinates is illustrated in Figure 7. In addition, the error

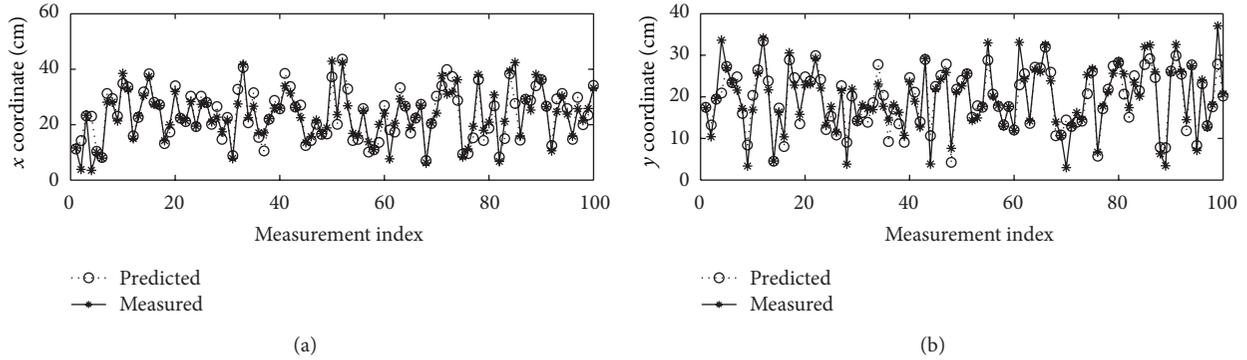


FIGURE 7: (a) Comparison of measured  $x$  coordinates and predicted results (100 impacts for test). (b) Comparison of measured  $y$  coordinates and predicted results (100 impacts for test).

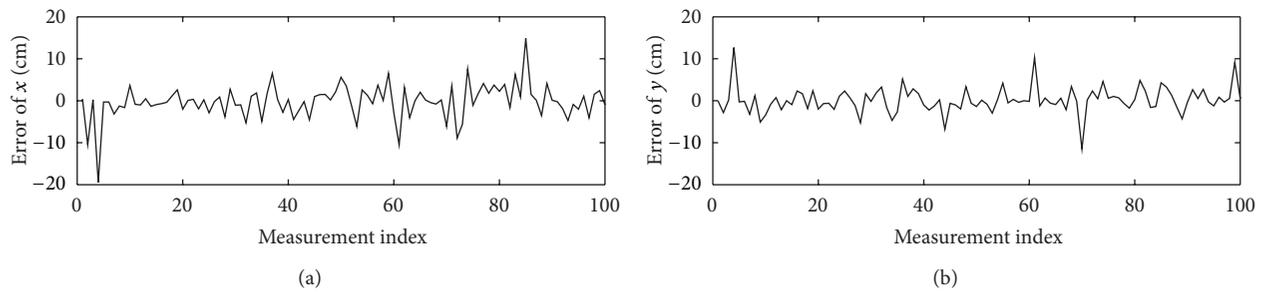


FIGURE 8: (a) Error of predicted and measured  $x$  coordinates (100 impacts for test). (b) Error of predicted and measured  $y$  coordinates (100 impacts for test).

between each set of the predicted and actual  $y$  coordinates is given in Figure 8. It is observed that some specific points exhibit large error. By inspecting the properties of those impact points, it is further found that the apparent errors are dependent on whether the location of testing impact lies in the training region or not.

Hence, it is desirable to enhance the detection accuracy by changing the constitution of the testing set. By comparing the predicted and original coordinates, it is observed that test impacts which locate outside the training set have higher errors than those inside. One reason lies in that the detection is more difficult when the impact is outside of the training region. Thus, it is reasonable to delete the impacts outside the training set so as to construct a new testing set. The sites of deletion are illustrated in Figure 9.

Consequently, the SVM models are trained with the original training set and tested with the modified testing set. The final results give an RMS error of 0.343 cm. The area of error occupies 0.129% of the plate size. Moreover, the RMS errors of the  $x$  and  $y$  coordinates are derived as 2.6954 and 1.8543 cm, respectively. The differences between the predicted and real values for the  $x$  and  $y$  coordinates are shown in Figures 10 and 11, respectively.

To illustrate the necessity of modifying the testing set, a comparison between the results obtained from the original testing set with 100 impacts and modified testing set with 91 impacts is presented in Table 1. The comparison covers the mean errors of  $x$  and  $y$  coordinates, the error area with

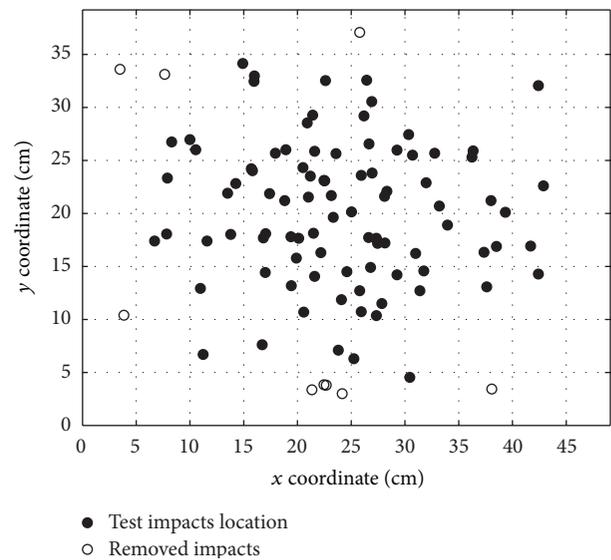


FIGURE 9: Site of testing set after deletion.

respect to plate size, RMS errors of  $x$  and  $y$  coordinates, and the integration of them, that is, the RMS error. It is found that the detection using the modified set accomplishes much better result than that using the original set. Evidently, the comparison result indicates the reason why the modified testing set is employed in the forgoing and subsequent testing experiments.

TABLE 1: Comparison of predicted results from different test sets with 100 and 91 impacts.

Data set	Mean error $x$ (cm)	Mean error $y$ (cm)	Error area	RMSE $x$ (cm)	RMSE $y$ (cm)	RMS error (cm)
Testing set (100 impacts)	2.6847	2.023	0.28%	4.0755	3.0809	0.5109
Modified testing set (91 impacts)	1.8346	1.3542	0.129%	2.6954	1.8543	0.3430

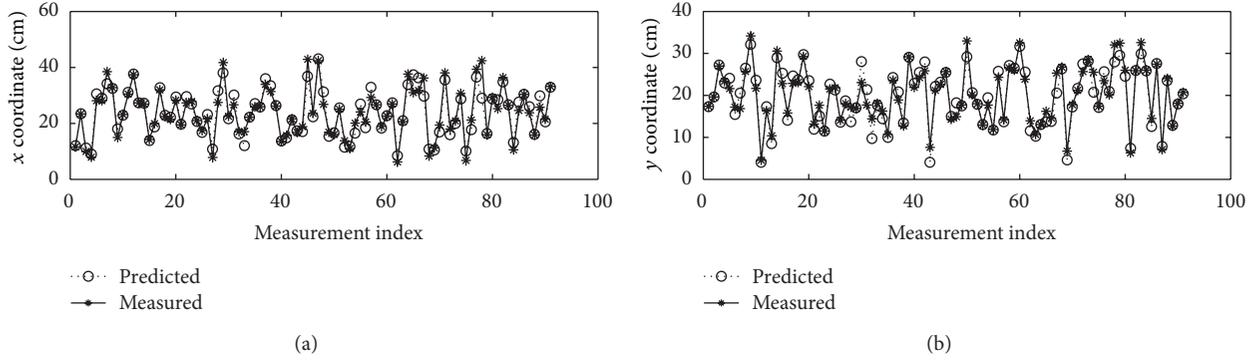


FIGURE 10: (a) Comparison of measured  $x$  coordinates and predicted results (91 impacts for test). (b) Comparison of measured  $y$  coordinates and predicted results (91 impacts for test).

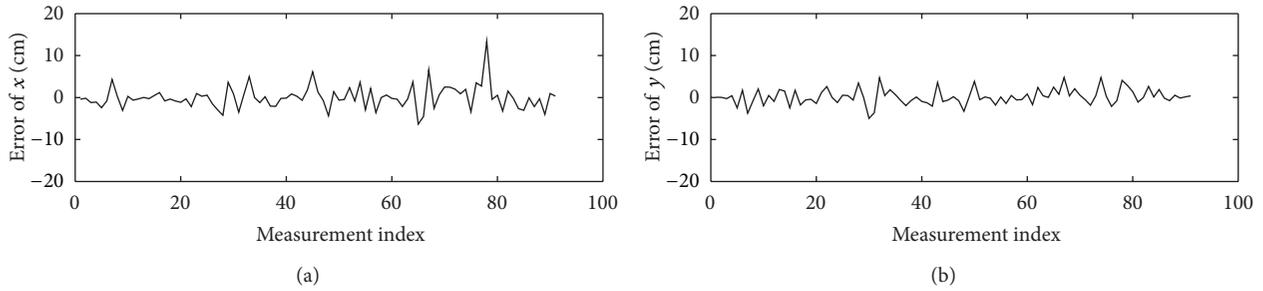


FIGURE 11: (a) Error of predicted and measured  $x$  coordinates (91 impacts for test). (b) Error of predicted and measured  $y$  coordinates (91 impacts for test).

TABLE 2: Comparison between SVM and BPNN results.

	RMSE $x$ (cm)	RMSE $y$ (cm)	RMSE area	RMS error (cm)
PCA + BPNN	5.8955	4.9034	1.5019%	0.7668
PCA + SVM	2.6954	1.8543	0.2597%	0.3430

5.2. *BPNN with PCA.* To illustrate the detection performance of the proposed approach in comparison with other method, a three-layer BPNN with 76:11:2 topology is adopted. The data preprocessing is also conducted by using the PCA. Because of the random initial conditions, the program uses an iterative loop to explore the optimal number of neurons (i.e., 11 neurons) in the hidden layer. The best result predicted by the BPNN is listed in Table 2 with comparison to the SVM result.

5.3. *Discussions on Detection Results.* It is evident from Table 2 that the SVM delivers better detection accuracy in comparison with BPNN result. As compared to SVM, the reason why the BPNN performance is worse may arise from

the insufficient number of the training data sets. In this sense, the SVM is capable of accomplishing better location detection accuracy than BPNN using a small set of training data.

The location detection results generated by the proposed strategy are also compared with typical previous works based on regression modeling. For example, [7] adopted the integrated real and imaginary parts of FFT results for the impact strain signals as inputs and the RMS error as shown in (8) to evaluate the performance. The RMS error of detection result was 1.56 cm, which is much larger than the RMS error (0.343 cm) achieved by the current work.

In addition, another work in [6] gathered the magnitude and time after impact of maximum response as input for neural networks training. That study resulted in an error area of 1.5% with respect to the plate size. Recalling that the current research obtains an RMS error of 0.343 cm and error area of 0.129% using SVM regression, a much better result has been achieved by the proposed approach in comparison with that in the previous work [6]. Therefore, the experimental results reveal that the proposed two-layer SVM in combination with PCA exhibits more powerful capacity of generalization and offers higher accuracy than conventional approaches.

## 6. Conclusions

This paper proposes a new approach by integrating PCA and two-layer SVM to predict the location of impacts on a plate structure. By extracting principal components, the PCA is capable of improving the computational efficiency. Using the PCA results, a two-layer SVM framework incorporating five SVM models has been built for predicting the  $x$  and  $y$  coordinates separately. The models have been trained using a set of 63 impacts and tested using another set of 91 impacts. The presented detection strategy delivers an RMS error of 0.343 cm, which indicates a better performance than that of the conventional backpropagation neural networks based method.

In the future, the effects of noise and variation of impact magnitudes on the detection accuracy will be investigated. Moreover, with an online real-time realization [13, 16], the whole prediction system is expected to meet the actual requirements and increase the prospect of applications. Besides, severe impacts generally lead to real damage. Hence, the issue of whether the damage affects the subsequent prediction will be investigated as well.

## Acknowledgments

This work was supported by the Macao Science and Technology Development Fund under Grants 024/2011/A and 070/2012/A3 and the Research Committee of the University of Macau under Grant MYRG083(Y1-L2)-FST12-XQS. The authors would like to express their gratitude to Mr. Kehn Wong for the technical discussions.

## References

- [1] J. Haywood, P. T. Coverley, W. J. Staszewski, and K. Worden, "An automatic impact monitor for a composite panel employing smart sensor technology," *Smart Materials and Structures*, vol. 14, no. 1, pp. 265–271, 2005.
- [2] H. Sohn, C. R. Farrar, N. F. Hunter, and K. Worden, "Structural health monitoring using statistical pattern recognition techniques," *Journal of Dynamic Systems, Measurement and Control*, vol. 123, no. 4, pp. 706–711, 2001.
- [3] L. E. Mujica, J. Vehí, W. Staszewski, and K. Worden, "Impact damage detection in aircraft composites using knowledge-based reasoning," *Structural Health Monitoring*, vol. 7, no. 3, pp. 215–230, 2008.
- [4] V. Giurgiutiu, "Tuned Lamb wave excitation and detection with piezoelectric wafer active sensors for structural health monitoring," *Journal of Intelligent Material Systems and Structures*, vol. 16, no. 4, pp. 291–305, 2005.
- [5] Y. Lu and J. E. Michaels, "A methodology for structural health monitoring with diffuse ultrasonic waves in the presence of temperature variations," *Ultrasonics*, vol. 43, no. 9, pp. 717–731, 2005.
- [6] K. Worden and W. J. Staszewski, "Impact location and quantification on a composite panel using neural networks and a genetic algorithm," *Strain*, vol. 36, no. 2, pp. 61–70, 2000.
- [7] R. T. Jones, J. S. Sirkis, and E. J. Friebele, "Detection of impact location and magnitude for isotropic plates using neural networks," *Journal of Intelligent Material Systems and Structures*, vol. 7, no. 1, pp. 90–99, 1997.
- [8] J. R. LeClerc, K. Worden, W. J. Staszewski, and J. Haywood, "Impact detection in an aircraft composite panel—a neural-network approach," *Journal of Sound and Vibration*, vol. 299, no. 3, pp. 672–682, 2007.
- [9] S. E. Watkins, F. Akhavan, R. Dua, K. Chandrashekhara, and D. C. Wunsch, "Impact-induced damage characterization of composite plates using neural networks," *Smart Materials and Structures*, vol. 16, no. 2, pp. 515–524, 2007.
- [10] S. F. Masri, A. W. Smyth, A. G. Chassiakos, T. K. Caughey, and N. F. Hunter, "Application of neural networks for detection of changes in nonlinear systems," *Journal of Engineering Mechanics*, vol. 126, no. 7, pp. 666–676, 2000.
- [11] H. Song, L. Zhong, and B. Han, "Structural damage detection by integrating independent component analysis and support vector machine," in *Advanced Data Mining and Applications*, vol. 3584 of *Lecture Notes in Computer Science*, pp. 670–677, Springer, Berlin, Germany, 2005.
- [12] P. K. Wong, Q. Xu, C. M. Vong, and H. C. Wong, "Rate-dependent hysteresis modeling and control of a piezostage using online support vector machine and relevance vector machine," *IEEE Transactions on Industrial Electronics*, vol. 59, no. 4, pp. 1988–2001, 2012.
- [13] Q. Xu and P. K. Wong, "Hysteresis modeling and compensation of a piezostage using least squares support vector machines," *Mechatronics*, vol. 21, no. 7, pp. 1239–1251, 2011.
- [14] J. Xie, "Improved least square support vector machine for structural damage detection," in *Proceedings of the 2nd International Conference on Computer Engineering and Technology (IC CET '10)*, pp. 237–240, April 2010.
- [15] P. M. Pawar and S. N. Jung, "Support vector machine based online composite helicopter rotor blade damage detection system," *Journal of Intelligent Material Systems and Structures*, vol. 19, no. 10, pp. 1217–1228, 2008.
- [16] J. Zhang, T. Sato, and S. Iai, "Support vector regression for online health monitoring of large-scale structures," *Structural Safety*, vol. 28, no. 4, pp. 392–406, 2006.



# Hindawi

Submit your manuscripts at  
<http://www.hindawi.com>

