

**Solving the Damage Localization Problem in  
Structural Health Monitoring Using Techniques  
in Pattern Classification**

**CS 229 Final Project  
Due Dec. 14, 2007**

**Hae Young Noh, Allen Cheung, Daxia Ge**

## Introduction

Structural health monitoring (SHM) for civil engineering applications involves the use of sensing networks in order to diagnose damage in large scale civil structures such as buildings or bridges. Health monitoring will result in large savings in maintenance costs as well as improving life safety by providing a real-time updating of a structure's integrity. Recent events such as the bridge collapse in Minneapolis as well as aging infrastructure illustrate the need to improve structural monitoring.

Recent research has shifted towards using statistical algorithms and time series modeling to detect damage [1]. Statistical methods are advantageous, because they do not require the user to create a structural, usually finite element, model for the whole structure. This results in significant savings in power consumption as well as data transmission, making them ideal for use in low power wireless networks. Interest in using wireless networks for SHM has increased due to the lower cost and ease of installation [2].

Damage diagnosis is divided into three categories: damage identification, measurement of damage extent, and damage localization. Extensive work has been done in the area of damage identification and localization. In particular, Nair et al. (2006) show that an algorithm based on autoregressive time series modeling of a structure's acceleration response to ambient vibrations is capable of detecting and measuring the extent of damage on a structure. However, the damage localization problem remains unsolved. Localization is important, because it will allow responders to streamline efforts in repair and maintenance.

This paper examines the use of pattern classification techniques in machine learning to solve the damage localization problem. Following, Nair et al. (2006), autoregressive coefficients from a structure's acceleration response are selected as the feature vector. Data is collected by simulation using the ASCE Phase I Benchmark Structure [3]. It is shown that using established pattern classification techniques, it is possible to localize damage in a structure. Further work is shown in examining the optimal number of parameters and number of training examples for this problem.

## Time Series Modeling by Autoregressive Coefficients

Nair et al. (2006) show that the autoregressive (AR) coefficients from a structure's accelerations under ambient vibrations are directly related to the structure's natural frequency and by extension its stiffness [1]. The autoregressive time series model can be represented as a linear difference equation with a transfer function, with the locations of the poles given by the AR coefficients; thus it follows that AR coefficient values will be a function of the structural stiffness. Thus changes to a structure's stiffness matrix as a result of permanent damage will change the structure's AR coefficients. For a more complete derivation of the AR algorithm, see Nair et al. (2006).

The AR algorithm is performed according to the following steps as was previously done by Sohn et al. (2001) and Nair et al. (2006):

The acceleration time history is normalized and standardized according to the following equation:

$$\tilde{x}(t) = \frac{(x(t) - \mu)}{\sigma} \quad (1)$$

where  $x(t)$  is the acceleration time history at a point  $t$  in time,  $\mu$  and  $\sigma$  are the mean and standard deviation of the time history, and  $\tilde{x}(t)$  is the normalized and standardized time history.

Let  $\tilde{x}_i(t)$  be the normalized and standardized acceleration signal from the  $i^{\text{th}}$  sensor. In order to increase the number of features,  $\tilde{x}_i(t)$  is divided into  $N$  chunks. Each chunk is an independent feature. More specifically, Nair et al. (2006) show that if the acceleration time history can be shown to be stationary, then the acceleration signal of each chunk will be independent and identically distributed. Although this has been done, it will not be shown here for brevity. The AR model is given by the following equation:

$$\tilde{x}_{ij}(t) = \sum_{k=1}^p \alpha_{ijk} \tilde{x}_{ij}(t-k) + \varepsilon_{ij}(t) \quad (2)$$

where  $\tilde{x}_{ij}(t)$  is the normalized acceleration signal of the  $j^{\text{th}}$  chunk of the  $i^{\text{th}}$  sensor,  $\alpha_{ijk}$  is the  $k^{\text{th}}$  AR coefficient from the  $j^{\text{th}}$  chunk of the  $i^{\text{th}}$  sensor,  $p$  is the AR model order, and  $\varepsilon_{ij}(t)$  is the residual term. The Burg algorithm (Brockwell and Davis, 2003) is used to estimate the AR coefficients. In this study a model order of 5 and a chunk size of 4000 are used. The optimal AR order  $p$  and optimal chunk size  $N$  can be determined according to methods proposed by Nair et al. (2006). Again for brevity, they will not be discussed.

## ASCE Phase I Benchmark Structure

The ASCE Phase I Benchmark Structure is a standardized simulation problem for the development of structural health monitoring algorithms for civil structures. The benchmark structure is a Matlab based simulation of a structure's response to ambient vibrations. Damage can be specified according to 6 predetermined damage patterns or user defined on any combination of beams, columns, and braces. Output is given in the form of 16 acceleration time histories, representing acceleration from 8 different sensors in the  $x$  and  $y$  directions. Extensive data can be gathered by specifying the random seed for force generation. The model is based off of an existing structure which was constructed and tested at the University of British Columbia. The structure is a two bay, four story, rectangular steel braced frame. It is modeled as a 120 degree of freedom system with various loading options. The Phase I Benchmark is chosen instead of a more recent benchmark, because the ability to assign user defined damage patterns is critical in order to build a suitable training and test data set required for the implementation of pattern classification algorithms.

## Pattern Classification

Once the AR algorithm is performed on the acceleration time histories, the AR coefficients are assembled into a feature vector. Previous work done by Nair et al. (2006) and confirmed by this study show that the first AR coefficient is sufficient to pick up changes in structural stiffness resulting from damage. For this reason the feature vector  $X$  is chosen to be a vector of the first AR coefficient from all the sensors available.

$$X_j = \begin{bmatrix} \alpha_{1j1} \\ \alpha_{2j1} \\ \dots \\ \alpha_{nj1} \end{bmatrix} \quad (3)$$

where  $n$  is the number of sensors, and  $j$  is the index of the current chunk. For the benchmark structure  $n = 16$ , representing 8 accelerometers each recording acceleration in the  $x$  and  $y$  directions. The training set size  $m$  is equal to the number of chunks per run  $N$  multiplied by the number of runs. A run is defined as one single output gathered from simulation by subjecting the structure to one ground motion and subjecting it to one damage case.

Damage location is discretized according to the floor on which the damage has occurred. This results in 5 possible classifications for damage location: undamaged, 1<sup>st</sup> floor, 2<sup>nd</sup> floor, 3<sup>rd</sup> floor, or 4<sup>th</sup> floor. Thus this becomes a multi-category classification problem. One way to solve this problem is to use binary classification methods and implement them in a “one-versus-all” (1va) or “one-versus-one” (1v1) approach (Duan et al. 2003).

In this study two binary classification methods are used: regularized logistic regression using Newton’s method to solve the gradient ascent problem and support vector machines (SVM) using sequential minimal optimization (SMO) with a linear kernel. The SMO algorithm is used as derived by Platt (1998).

Gradient ascent attempts to maximize the following log-likelihood function:

$$\theta_{MAP} = \arg \max_{\theta} \sum_{i=1}^m [y^{(i)} \log(h_{\theta}(x^{(i)})) + (1 - y^{(i)}) \log(1 - h_{\theta}(x^{(i)}))] - \lambda \|\theta\|^2 \quad (4)$$

where  $\theta_{MAP}$  is the estimate of the model parameters,  $y^{(i)}$  is the  $i$ th class label, either 0 or 1,  $x^{(i)}$  is the  $i$ th feature vector, and  $h_{\theta}(x)$  is the sigmoid function given by:

$$h_{\theta}(x) = \frac{1}{1 + e^{(-\theta^T x)}} \quad (5)$$

$\lambda \|\theta\|^2$  is the Bayesian regularization term.  $\theta$  is assumed to be  $\sim N(0, \tau^2 I)$ .

$$\lambda = \frac{1}{2\tau^2} \quad (6)$$

$\lambda$  is chosen using cross-validation. The log-likelihood function is maximized using the Newton-Raphson method.

$$\theta^{(t+1)} = \theta^{(t)} - H^{-1} \nabla_{\theta} l \quad (7)$$

The parameters  $\theta$  are updated according to the previous equation until convergence has occurred or until some specified maximum number of iterations has taken place.  $l$  is the log-likelihood function,  $H$  is the Hessian matrix of  $l$  with respect to  $\theta$  and  $\nabla_{\theta} l$  is the gradient of the log-likelihood function.

SVM attempts to solve the Dual Problem. The problem simplifies to a maximization problem given by the following expression:

$$\max_{\alpha} \sum_{i=1}^m \alpha_i - \frac{1}{2} \sum_{i=1}^m \sum_{j=1}^m y^{(i)} y^{(j)} \alpha_i \alpha_j \langle x^{(i)}, x^{(j)} \rangle \quad (8)$$

such that:

$$\alpha_i \geq 0 \quad (9)$$

$$\sum_{i=1}^m y^{(i)} \alpha_i = 0 \quad (10)$$

This maximization problem is solved using SMO as derived by Platt (1998). A description of that algorithm is beyond the scope of this paper. For further details see Platt (1998).

Both the 1v1 and 1v1 approach for multi-classification are used with a “winner-takes-all” (WTA) and a “max-wins voting” (MWV) strategy (Duan et al. 2003). In addition, each algorithm was performed with the dimension of the feature vector reduced using Principal Components Analysis (PCA). The feature dimension  $n = 16$  for the benchmark structure is manageable – 8 sensors measuring acceleration in the  $x$  and  $y$  directions. However, for a larger structure such as a suspension bridge or skyscraper,  $n$  may grow into the thousands, especially if the  $z$  direction of acceleration is also considered. PCA provides a way to make computations manageable. For this study, the dimension of the feature vector was reduced to the first two principal components.

The following damage patterns were introduced to the benchmark structure.

Table 1 – Damage Patterns and Class Labels

Damage Pattern	Class Label
Undamaged	1
All 1 <sup>st</sup> Floor Braces Damaged	2
All 1 <sup>st</sup> Floor Beams Damaged	2
All 2 <sup>nd</sup> Floor Braces Damaged	3
All 2 <sup>nd</sup> Floor Beams Damaged	3
All 3 <sup>rd</sup> Floor Braces Damaged	4
All 3 <sup>rd</sup> Floor Beams Damaged	4
All 4 <sup>th</sup> Floor Braces Damaged	5
All 4 <sup>th</sup> Floor Beams Damaged	5

10 runs are generated for each damage state, each with different random seeds. Each run is divided into 10 chunks according to the algorithm described above. Therefore the size of the total data set  $m$  is 900. Two damage states are considered – damage to beams and damage to braces – in order to determine the feasibility of localizing multiple types of damage even though the damage is labeled only according to location. It was found in the analysis that the damage localization algorithm trained on brace damage data alone was incapable of localizing beam damage. The data are labeled strictly according to which floor the damage took place on.

All algorithms were trained on 70% of the data, with the other 30% reserved for testing. The size of the training set is 630. The size of the test set is 270. Through cross validation, it was found that an optimal value of  $\lambda$  for gradient ascent is 0.01 and an optimal value of  $C$  for SVM is 50. The results of each algorithm are compared in Table 2. Figure 1 shows the distribution of features when plotted according to their first two principal components. A test case was determined to be correctly assessed if the algorithm predicted the floor on which the damage occurred (or no damage). An incorrect assessment was recorded as an error.

Table 2 – Test Error Results for Selected Algorithms

	Gradient Ascent	SVM
1v1 MWV	2.2%	0.37%
1va WTA	2.6%	0%
1v1 MWV PCA	64.8%	49.6%
1va WTA PCA	65.2%	62.6%

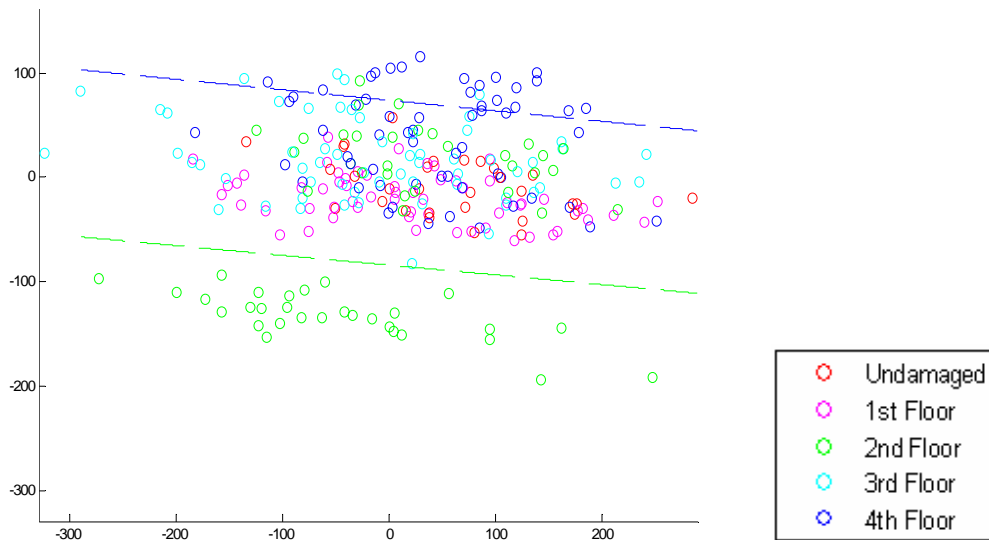


Figure 1 – Distribution of Features According to First Two Principal Components

The pattern classification algorithms used are able to predict with high levels of accuracy the location of damage in the benchmark structure. As expected, PCA increases the amount of test error, because information is lost. Nevertheless, even with the dimensionality of the feature vector reduced to 2, the 1v1 SVM algorithm is able to predict with 50% accuracy the location of damage in the structure.

## Conclusion

A pattern classification approach using supervised learning shows excellent results when tested on simulated data. SVM-based multi-classifier algorithms are able to achieve almost 100% success rate in identifying the location of damage. The advantage of this approach is that once a training data set has been built, testing is relatively simple and low in computational complexity.

The problem with implementing this approach in practice to monitor existing structures is the lack of and difficulty in acquiring a large enough training set. Because acquiring a training set experimentally would be prohibitively expensive, future research work should examine whether a training set created using simulated data can be used to predict the location of actual damage on an experimentally tested structure. In addition to this, future research may consider less severe damage states, other types of damage, other ways of discretizing damage location, and whether training data for one type of structure can be applied successfully to similar structures. However, despite the difficulties with applying this approach in practice, supervised learning approaches offer a great deal of potential in solving the damage localization problem.

## References

Nair, K.K., Kiremidjian and K.H. Law. (2006). “Time Series Based Damage Detection and Localization Algorithm with Application to the ASCE benchmark Structure,” *Journal of Sound and Vibration* 291, 349-368.

Sohn, H., Farrar, C.R., Hunter, H.F. and Worden, K., (2001), “Applying the LANL Statistical Pattern Recognition Paradigm for Structural Health Monitoring to Data from a Surface-Effect Fast Patrol Boat”, *Los Alamos National Laboratory Report LA-13761-MS*, Los Alamos National Laboratory, Los Alamos, NM 87545.

Brockwell, R. J. and Davis, R. A. (2003). *Introduction to Time Series and Forecasting*, Springer, 2<sup>nd</sup> Ed., 456p.

Platt, John. *Fast Training of Support Vector Machines using Sequential Minimal Optimization*, in *Advances in Kernel Methods – Support Vector Learning*, B. Scholkopf, C. Burges, A. Smola, eds., MIT Press (1998)

Duan, Kaibo, S. Sathiya Keerthi, Wei Chu, Shirish Krishnaj Shevade, and Aun Neow Poo. (2003). “Multi-Category Classification by Soft-Max Combination of Binary Classifiers”, *Multiple Classifier Systems*, pp. 125-134

