Civil Engineering Faculty Publications   Civil Engineering

9-2011

Application of Noise Cancelling and Damage Detection Algorithms in NDE of Concrete Bridge Decks Using Impact Signals

Gang Zhang
Professional Service Industries

Ronald S. Harichandran
University of New Haven, rharichandran@newhaven.edu

Pradeep Ramahulli
Pacific Northwest National Laboratory

Comments

This is the authors' accepted version of the article published in *Journal of Nondestructive Evaluation*. The final publication is available at Springer via http://dx.doi.org/10.1007/s10921-011-0114-8

Application of Noise Cancelling and Damage Detection Algorithms in NDE of Concrete Bridge Decks using Impact Signals

Gang Zhang[1],[*]; Ronald S Harichandran[2]; and Pradeep Ramuhalli[3]

Abstract: Delamination is a commonly observed distress in concrete bridge decks. Among all the delamination detection methods, acoustic methods have the advantages of being fast and inexpensive. In traditional acoustic inspection methods, the inspector drags a chain alone or hammers on the bridge deck and detects delamination from the "hollowness" of the sound. The signals are often contaminated by ambient traffic noise and the detection of delamination is highly subjective. This paper describes the performance of an impact-based acoustic NDE method where the traffic noise was filtered by employing a noise cancelling algorithm and where subjectivity was eliminated by introducing feature extraction and pattern recognition algorithms. Different algorithms were compared and the best one was selected in each category. The comparison showed that the modified independent component analysis (ICA) algorithm was most effective in cancelling the traffic noise and features consisting of mel-frequency cepstral coefficients (MFCCs) had the best performance in terms of repeatability and separability. The condition of the bridge deck was then detected by a radial basis function (RBF) neural network. The performance of the system was evaluated using both experimental and field data. The results show that the selected algorithms increase the noise robustness of acoustic methods and perform satisfactorily if the training data is representative.

Keywords: Delamination; Concrete bridge decks; Acoustic NDE; Noise cancellation; Mel-frequency cepstral coefficients; Neural Network.

---

[1] Research Associate, National Research Council, 6300 Georgetown Pike, Federal Highway Administration, McLean, VA 22101

[2] Formerly Professor and Chairperson, Department of Civil and Environmental Engineering, 3546 Engineering Building, Michigan State University, East Lansing, MI 48824-1226. Currently Dean, Tagliatela College of Engineering, University of New Haven, 300 Boston Post Road, West Haven, CT

[3] Senior Research Scientist, Pacific Northwest National Laboratory, 902 Battelle Boulevard, P.O. Box 999, MSIN K5-26, Richland, WA,99352

[*] Corresponding Author: 6300 Georgetown Pike, Federal Highway Administration, McLean, VA 22101, Tel.: +1-202-493-3488; Fax: +1-202-493-3161; Email:civilzhang@gmail.com

Introduction

Delamination refers to the separation of a layer of concrete from the main body and is a major form of damage in concrete decks. It is usually caused by traffic loads, corrosion of steel reinforcement and freeze-thaw of the water in concrete. The initial stage of delamination occurs under the surface and is difficult to detect through visual inspection. As the delamination propagates, it leads to spalling of the bridge deck, causes an uncomfortable driving experience and can be potential threat to public safety. Small delaminated areas can be repaired by patching. A very large area of delamination usually requires replacement of the entire deck, which is expensive and causes significant user delay. It is therefore advantageous to detect delamination at an early stage to reduce the cost of repair.

Many methods have been considered for the inspection of bridge deck systems. In the impact echo method [1, 2] delamination is detected by measuring the time interval for the P-wave to be reflected between the surface of the deck and that of the defect and the recorded signal is analyzed in the frequency domain. The delamination is characterized by peaks in the frequency spectrum. In the ultrasonic pulse velocity method [3] delamination is detected by measuring the arrival time of ultrasonic waves inside concrete. The existence of defects changes the path of wave propagation and hence changes the arrival time. In both methods, sensors must be coupled to the surface for reliable measurements. In addition, the signal obtained from the impact-echo test in real situations can be difficult to interpret when the surface of the defect is irregular. Although methods that do not require full surface contact do exist [4], these techniques require sophisticated signal processing techniques and advanced sensors and are therefore impractical for the inspection of a large area like a bridge deck. Ground penetrating radar [5] detects objects in the concrete by measuring contrast in the dielectric properties of the materials. This is a non-contact method that is non-destructive and fast, but it is not sensitive in detecting delaminations, voids and cracks filled with air because the contrast between the dielectric constants of air and concrete is small. Infrared thermography [6, 7] can also be used to detect delamination in concrete. Concrete and defects usually have different thermal properties that can be captured by the temperature distribution when there is a heat transfer. This makes the method environment dependent. In X-ray imaging [8], materials with different densities are represented by the darkness of the pixels. Defects and other objectives with different densities can then be identified by analyzing the X-ray images. But the X-ray source is bulky, needs considerable power, and is a safety concern for the inspector, making it impractical for

the inspection of bridge decks. Sounding methods [9-11] detect delamination by the "hollowness" of the sound when excited by hammer impacts or a chain-drag. Good concrete with no delamination produces a clear, ringing sound, while delaminated concrete is characterized by a dull, hollow sound. Next to visual inspection, mechanical sounding is the most prevalent method used for the inspection of concrete bridges [12]. However, there are two major problems associated with the traditional sounding: (1) the detection is subjective; and (2) the effectiveness of the method is affected by the level of ambient noise.

Although several attempts have been made to improve sounding methods, research on this topic is still limited. Researchers at the Michigan Department of Transportation (MDOT) designed a cart-like device for delamination detection [11]. The impulse was created by the chattering of two rigid wheels with the concrete and the vibration of the concrete was captured by a transducer coupled to the ground through soft tires and liquid in the wheels. The recorded signals were first truncated to retain only 5 ms following the impulse and filtered by a fixed band pass filter with cut-off frequencies at 300 and 1200 Hz. The processed signals were recorded on charts. The delamination was detected by listening to the filtered sound through earphone. The method showed signs of improvement but the signal processing algorithm was primitive and the detection is still subjective. Henderson et al. [10] used sound signals created by dragging a chain. The traffic noise was isolated by sound proofing around the chains. A computer algorithm based on linear prediction coefficients (LPC) was used to analyze the recorded signals and perform the detection. Although this technique showed promise, the method had two major drawbacks. First, the traffic noise was reduced only by physical isolation, which can be ineffective at high noise levels and for complex sound fields encountered on highway bridges. Second, traffic noise is usually non-stationary and simple filtering using LPC can be inadequate.

The goal of the research described in this paper was to develop an automated inspection system to accurately detect delamination in concrete bridge decks. This was achieved by accomplishing the following tasks:

1. Develop a noise cancelling algorithm that can cancel or separate ambient noise from field measurements. There are different noise cancelling algorithms and each has its own range of application. The performance of different algorithms was evaluated so that the optimal algorithm could be selected.

2. Develop algorithms that can differentiate between soundings on solid concrete and those on delaminated concrete. Two steps were needed for this objective. First, the "hollowness" of the acoustic signals needed to be quantified to eliminate subjectivity. Second, an automatic detection method was needed to classify the signals into two groups: those from solid and those from delaminated concrete. These algorithms needed be robust, fast and effective in the field.

Noise Cancelling Algorithms

Noise cancelling is a basic yet difficult problem. Fig. 1 shows the spectra of a typical impact signal and traffic noise and there is significant overlap between the two. This means that the noise cannot be filtered using simple band-pass filters. Fig. 2 shows that the properties of the traffic noise change due to changing traffic. Due to the non-stationarity and unpredictability of traffic noise, the noise cancelling algorithm should be adaptive and require no prior information about the noise. Extensive research has been performed on this topic and various types of algorithms have been proposed [13-16]. This section compares the performance of the following commonly used algorithms to help select an effective algorithm for traffic noise cancellation: spectrum subtraction, adaptive filters, independent component analysis (ICA) and modified ICA. The performance of the different algorithms was evaluated by visual inspection as well as a numerical criterion.

In spectrum subtraction [13], it was assumed that the spectrum of the noisy recording could be expressed by the summation of the target signal and the noise, which can be expressed in the frequency domain as:

$$M\left(e^{j\omega}\right) = S\left(e^{j\omega}\right) + N\left(e^{j\omega}\right) \tag{1}$$

where $M\left(e^{j\omega}\right)$, $S\left(e^{j\omega}\right)$ and $N\left(e^{j\omega}\right)$ refer to the spectrum of the noisy measurement, original signal and traffic noise, respectively.

If the noise is short-term stationary, the spectrum of the noise can be estimated from the "quiet" period where the target signal is absent, and the spectrum of the estimated signal can then be estimated by:

$$\hat{S}\left(e^{j\omega}\right) = \left[\left|M\left(e^{j\omega}\right)\right| - \left|\mu\left(e^{j\omega}\right)\right|\right] \tag{2}$$

where, $\hat{S}\left(e^{j\omega}\right)$ and $\mu\left(e^{j\omega}\right)$ are the estimated spectrum of the signal and the average spectrum of the noise in the "quiet" period, respectively.

This method requires that the properties of the noise in the "quiet" period are the same as those in the period when the signal is present. This requirement cannot be guaranteed in real situations as the noise in the adjacent lanes change due to different types of passing vehicles.

Adaptive filtering provides one solution to this problem. In this method, there are two microphones. The primary microphone records a mixture of the target signal and the noise and the reference microphone records a filtered version of the noise. The adaptive filter estimates the noise component in the primary microphone from the recording of the reference microphone. The coefficients of the filter adaptively change such that the filter output is a best estimate of the noise in the sense of mean-square-error (MSE). The noise in the primary recording is then cancelled by subtracting the estimated noise from the recording of the primary microphone. Fig. 3 shows the signal flow of the adaptive filter [16].

This method is adaptive and is able to cancel noise with changing properties. However, it requires a reference input (the noise) that contains no target signal component. This cannot be guaranteed in practice, because the target signal will always leak into the reference microphone. When the reference input contains some of the target signal, a part of the signal will inevitably be cancelled and the output of the system will be distorted. Also, the length of the filter must be predefined, although in reality, the length of the filter is unknown. In order to have a better estimate, a long filter length is preferred, but this will increase the computation time.

To release the requirement that the reference signal be pure noise, independent component analysis (ICA) was employed. Multiple microphones are used in this method, and the recorded signals are viewed as a mixture of the signal and noise. The signal and the noise are assumed to be independent, which is usually the case. A de-mixing matrix is used to separate the signal and the noise. The elements of the matrix adaptively change such that the statistical independence between output channels is maximized. The outputs of the algorithm will be scaled versions

of the original sources because the sources are mutually independent. The signal flow of traditional ICA is shown in Fig. 4 [14].

ICA does not require prior information about the sources, but it requires that the recordings be a linear mixture, meaning that different sources in the mixture should arrive at all microphones at the same time. This requirement cannot be easily satisfied in the field because the distance between the source and different microphones are usually different.

To overcome the requirement for a linear mixture, a modified ICA [15] was used in this study. The method is briefly described as follows. First, the number of samples representing the largest delay between the arrival times of different sources is estimated through:

$$L = \frac{F_s}{v / (d_1 - d_2)} \tag{3}$$

where F_s is the sampling frequency (Hz) of the measurement, v is the velocity of the sound in air, and $d_1 - d_2$ is the distance between the two microphones.

After the maximum delay is estimated, each measurement channel is shifted by an increment of one sample at a time until the maximum delay is reached. The shifted measurement is then rearranged and fed into the traditional ICA and the resulting independent components are scaled and shifted versions of each source. The outputs of the traditional ICA are grouped into different groups based on a similarity measurement as the independent components from the same source have greater similarity. The individual source is then estimated from the independent components identified from the same source. The steps of this modified ICA are shown in Fig. 5 [15].

To evaluate the performance of the different algorithms, the impact signal recorded in a quiet laboratory environment and traffic noise recorded near a highway were mixed in the computer to create various noisy measurements. The algorithms summarized above were used to estimate the target signal. The results are shown in Fig. 6 through Fig. 9. Based on visual inspection of the recovered signal, the modified ICA has the best performance.

A numerical performance criterion was used to provide an objective comparison of different algorithms. In this criterion, the estimated signal was decomposed into four parts: the part from the desired source (s_{target}), the part from unwanted sources (e_{interf}), measurement noise (e_{noise}) and artifacts (e_{artif}). In this study, because the recordings were obtained by mixing the impact signal obtained in a quiet environment with noise recorded near a highway, there was no contribution from unwanted sources and measurement noise (i.e., there was no e_{interf} or e_{noise}). Therefore,

$$\hat{s} = s_{target} + e_{interf} + e_{noise} + e_{artif}$$
$$= s_{target} + e_{artif}$$

(4)

The four noise components can be obtained through orthogonal projections. s_{target} can be calculated from a simple inner product [17]:

$$s_{target} = \frac{\langle \hat{s}_j, s_j \rangle s_j}{\|s_j\|^2}$$

(5)

where s_j is the original signal and $\hat{s}_j$ is the estimated signal. Once s_{target} is available, e_{artif} can be calculated as:

$$e_{artif} = \hat{s} - s_{target}$$

(6)

The performance of different algorithms was measured by the signal to distortion ratio (SDR) given by:

$$SDR = 10\log_{10} \frac{\|s_{target}\|^2}{\|e_{interf} + e_{noise} + e_{artif}\|^2}$$
$$= 10\log_{10} \frac{\|s_{target}\|^2}{\|e_{artif}\|^2}$$

(7)

The SDRs of the different noise cancelling algorithms are shown in Table 1. The SDR measure also indicates that the modified ICA had the best performance and therefore it was selected as the best algorithm for impact-based delamination detection.

Feature Extraction Algorithms

After the noise in the recordings was removed by implementing the noise canceling algorithm, the next step was to relate the characteristics of the acoustic signals with the existence of delamination. Fig. 10 shows typical impact signals from solid and delaminated concrete. It is very difficult to differentiate the two signals by inspecting the waveforms. Therefore, other characteristics of the signals need to be extracted for the purpose of detection. Even though delamination of the concrete bridge deck is characterized by a hollow sound, this criterion is highly subjective and difficult to implement in an automatic detection algorithm. One way to eliminate the subjectivity is to parameterize the acoustic signals using different models. The parameters in these models are called features of the signals and the process of parameterization is called feature extraction. Different models represent a signal in different ways and extract different features of the signal. In this section, five different features were extracted and compared to select the best one.

Frequency characteristics are probably the most widely used feature in the processing of acoustic signals since they have a clear physical meaning. The effectiveness of the sub-band energy as a candidate feature was evaluated. To compute sub-band energy, the frequency spectrum was filtered by 16 rectangular filters evenly spaced on the frequency axis. The energy of the filter output was used as features of the signal.

Similar to sub-band energy, the energy of a wavelet packet tree can also be used as features of signals. In this research, the signals were decomposed to level 4 using Haar wavelets. The energy of the 16 sub-bands at the lowest (the 4th) level was extracted as candidate features.

Even though detection of delamination by the "hollowness" of the impact signal may be subjective, it is undeniable that the human ear is effective at detecting differences in sounds. Psycho-acoustic based features that approximate human hearing may therefore also be effective. Mel-frequency cepstral coefficient (MFCCs) [18] are one set of pyscho-acoustic features and were selected as candidate features of the signal because they closely approximate the human hearing system and were widely used in speech recognition. To calculate MFCCs, the spectrum of the signal is filtered by a series of triangular filters whose center frequency is evenly distributed on the Mel-frequency scale. The Mel-frequency is proportional to the log of the regular frequency scale. The energy of the

output of each filter is then calculated and the MFCCs are calculated by taking the discrete cosine transform of the log of the energy. In this research, a total of 16 MFCCs were extracted for each signal.

During the feature extraction process, it is desirable to transform the signal into a lower dimension with minimum loss. Energy loss is a commonly used cost function during dimension reduction. A good representation will retain as much energy as possible using the least number of features. Principal component analysis (PCA) [19] can be used to achieve this goal. PCA finds an optimal linear transformation such that the energy contained in a certain number of features is maximized. To find the features of the signal using PCA, a certain number of signals were used as training data and the principal components of the training signals were found using the Hebbian learning algorithm [20]. The signals (both training and testing) were projected onto these principal components and the magnitudes of the projection were used as one of the candidate features.

Another cost function to minimize during feature extraction is the amount of information. ICA is able to find the optimal transformation by minimizing the loss of information. The steps to find ICA-based features are similar to those for extracting MFCCs. The triangular filter used for MFCCs is replaced by the frequency spectrum of the independent components of the training signals.

For the purpose of delamination detection, the features of the acoustic signals need to have two properties. First, the features of the signal from concrete in the same condition must be consistent; i.e., the results must be repeatable or have intra-class similarity. Second, the difference between features from the solid and delaminated concrete must be large enough so that they can be easily separated from one another; i.e., the results must be separable. Assuming that the features from solid and delaminated concrete are random variables, repeatability can be measured by the coefficient of variation. For multiple random variables, the repeatability of the extracted feature can then be calculated as:

$$REP = \sqrt{\frac{(N_S - 1)\left(\frac{\sigma_S}{\mu_S}\right)^2 + (N_D - 1)\left(\frac{\sigma_D}{\mu_D}\right)^2}{N_S + N_D - 2}} \tag{8}$$

where μ_S and σ_S are the mean and variance for features of signals from solid concrete, μ_D and σ_D are the mean and variance for features of signals from delaminated signals, and N_S and N_D are the number of signals from solid and delaminated concrete, respectively. The lower the value is, the better the repeatability of the features will be. The REP measure may become ill-conditioned if one or both means of the features is close to zero.

Similarly, the separability of the features can be calculated through:

$$SEP = \frac{|\mu_S - \mu_D|}{\sqrt{\dfrac{(N_S - 1)\sigma_S^2 + (N_D - 1)\sigma_D^2}{N_S + N_D - 2}}} \tag{9}$$

A high SEP value indicates better separability.

Another way to measure separability is through mutual information. Each feature contains a certain amount of information about the condition of the concrete and it is desirable to select those features that contain more information. Mutual information is one way of measuring the amount of information. It measures the reduction in uncertainty after observation of a feature. A feature with a high mutual information value contains more information about the class it belongs to. The value of mutual information can be calculated as [21]:

$$H(C \mid \underline{x}) = -\int_{\underline{x}} p(\underline{x}) \left(\sum_c p(C \mid \underline{x}) \log(p(C \mid \underline{x})) \right) d\underline{x} \tag{10}$$

where $p(\underline{x})$ is the probability density function (PDF) for feature vector $\underline{x}$ and $p(C \mid \underline{x})$ is the conditional PDF of the class label after feature vector $\underline{x}$ is observed.

The REP, SEP and mutual information performance measures of the feature extraction algorithms considered are shown in Fig. 11 through Fig. 13. PCA has a high REP because this measure was ill-conditioned due to the small mean of the features extracted by PCA. The SEP values for PCA and ICA are very high and are truncated in Fig. 12. The reason for the high SEP values is that the variance for features extracted by PCA and ICA was very small and the expression of SEP was ill-conditioned.

A weighted rank was used to take both repeatability and separability into account when selecting the best feature extraction algorithm. The weight assigned to REP was 0.5; SEP and mutual information shared the remaining 0.5, as they are both measures of separability. The weighted rank of the different algorithms is listed in Table 2. MFCC had the lowest weighted rank and the best overall performance.

The classification error of different features using a linear Bayesian classifier was also used to evaluate the performance of different feature extraction algorithms. A total of 105 impact signals from concrete with no delamination (ND) and 132 from concrete with shallow delamination (SD) were used. 16 features were extracted for each algorithm. Since not all features are useful for detection purposes and "unwanted" features will decrease the accuracy of the detection and increase the computational load for classification, four features with the highest mutual information values were selected as useful features and were used to train the classifier. This is because the optimal performance of the Bayesian classifier was reached when the number of the features was four. Detailed information about the effect of the number of features are described in the next section. The trained classifier was then used to classify the testing data and the error rate (the percentage of misclassification) of different algorithms was used as the criterion to evaluate the performance of the candidate feature extraction algorithms for the damage detection. The error rates using different candidate features are shown in Table 3. In this table, error 1 refers to the case where the concrete was solid but was classified as delaminated and, error 2 refers to the case where the concrete was delaminated but was classified as solid. The effectiveness of the algorithms based on the error rates agree well with the weighted rank.

Based on the results of weighted ranks as well as the error rates, the MFCC was selected as the feature extraction algorithm for delamination detection.

Delamination Detection Algorithms

The problem of detecting delamination from extracted features can be formulated as classifying the recorded signals into two groups: signals from solid concrete and those from delaminated concrete. The dividing line or decision surface cannot be drawn empirically "by eye" and should be determined optimally with respect to certain criteria. Commonly used classifiers include the Bayesian classifier, support vector machine (SVM), multilayer perceptron (MLP), and radial basis function (RBF) network. This section compares these classification algorithms to select the best for delamination detection.

The Bayesian classifier [21] finds the dividing surface by minimizing the probability of erroneous classification. The probability of erroneous classification can be expressed as:

$$
\begin{aligned}
P_e &= P\left(C_1 \middle| x \in C_2\right) + P\left(C_2 \middle| x \in C_1\right) \\
&= \int_{-\infty}^{x_0} P(x \mid C_2)\,dx + \int_{x_0}^{\infty} P(x \mid C_1)\,dx
\end{aligned}
\tag{11}
$$

where $P\left(x \middle| C_1\right)$ is the conditional PDF of the feature x if the signal belongs to class 1, and $P\left(x \middle| C_2\right)$ is the conditional PDF of the feature x if the signal belongs to class 2.

The probability of erroneous classification can be shown as the shaded area in Fig. 14 and can be minimized when the threshold is at the intersection of the two PDFs. Therefore, the optimal threshold can be expressed as:

$$
g(x) = P(x \mid C_1) - P(x \mid C_2) = 0
\tag{12}
$$

The probability density function that is used most often in practice is the normal (Gaussian) distribution for which the expression of the threshold can be further simplified. The conditional probability density function of a jointly normal vector x can be expressed as:

$$
P\left(x \middle| C_i\right) = \frac{1}{(2\pi)^{l/2} \left|\Sigma_i\right|^{1/2}} \exp\left(-\frac{1}{2}\left(x - \mu_i\right)^T \Sigma_i^{-1}\left(x - \mu_i\right)\right)
\tag{13}
$$

where Σ_i is the covariance matrix of each class and μ_i is the mean value.

The decision surface for class i can be expressed as:

$$g_i(x) = -\frac{1}{2}(x-\mu_i)^T \Sigma_i^{-1}(x-\mu_i) - \frac{l}{2}\ln(2\pi) - \frac{1}{2}\ln|\Sigma_i| \qquad (14)$$

If the elements of the feature vectors are mutually independent and have a normal distribution, the decision surface will have a quadratic form and the classifier is called a quadratic Bayesian classifier. In addition, if the variances of the elements are equal, the decision surface will reduce to a hyper-plane and the classifier is called a linear Bayesian classifier.

The performance of the Bayesian classifiers was evaluated using the same data mentioned in the previous section. Since the selection of training samples was random, the results for each simulation could be different. To accommodate the variance due to the random selection of the training samples and to have a fair comparison, the upper limit for the 95% confident interval (CL) of the error rates was used to measure the performance. Fig. 15 shows the performance of the Bayesian classifiers. For the linear Bayesian classifier, the error rate dropped with an increase in the number of features, but the increase in performance was not significant when the number of features exceeded four. This explains why the number of features used to evaluate different feature extraction algorithms is four. For the quadratic Bayesian classifier, the error rate first decreased and then increased. The reason for this was that information contained in the redundant features was not useful for or had a negative effect on the classification.

The Bayesian classifiers require prior information about the underlying distribution of features, which is not always available. Linear classifiers which are not dependent on the underlying distribution of the training data [22] provide one solution to this problem. Linear classifiers separate two classes by a hyper-plane. One commonly observed linear classifier is the support vector machine (SVM).

Assume that the decision surface is:

$$g(x) = w^T x + b \qquad (15)$$

13

where w and b are the weighting vector and bias vector of the decision surface, respectively.

The separation margin can then be expressed as:

$$r = \frac{g(x_{s1}) + g(x_{s2})}{\|w\|} = \frac{2}{\|w\|} \tag{16}$$

where $g(x_{s1})$ and $g(x_{s2})$ are supporting vectors in the two classes, as shown in Fig. 16.

The optimal decision surface can be found by maximizing the separation margin. In the case where the data cannot be separated by a hyper-plane, the data may be transformed using a non-linear transformation such that the transformed data is linearly separable. The transformation function is called the kernel function. The performance of the SVM is dependent on the kernel function used.

For a linear SVM, the kernel function can be written as:

$$K(x, x') = x \cdot x' \tag{17}$$

and for a quadratic SVM, the kernel function will have the form:

$$K(x, x') = (x \cdot x')^2 \tag{18}$$

where $\cdot$ refers to the dot product operation. The kernel function was determined in the training process by minimizing the training error (cost function). Fig. 17 shows the performance of SVMs with linear and quadratic kernels. The error rate decreased with an increase in the number of features. However, the improvement in performance was not significant when the number of features exceeded 5. The SVM with a quadratic kernel had a better performance than that with a linear kernel.

Another way to separate data that is not linearly separable is to use a non-linear classifier whose decision surface has a non-linear form. The multi-layer perceptron (MLP) neural network [20] belongs to this category. It consists of three parts: an input layer, one or more hidden layers, and an output layer. The input layer consists of a series of

neurons that receive the input data, in this case the extracted features of the impact signals. The hidden layer consists of several layers of neurons, which take the outputs of the previous layer as inputs, compute the outputs and feed them to neurons in the next layer. The neurons in the output layer receive the outputs from the last hidden layer and compute the final output. Each neuron in the system is a computation unit. The neuron computes the weighted sum of the inputs and the summation is fed into an activation function that produces the output. The weights that connect different neurons are updated based on the error signal, which is the difference between the actual output of the network and the desired output. The update process is called back-propagation because the process goes layer by layer from the output to the input. The structure of the MLP is shown in Fig. 18.

The performance of the MLP can be influenced by the number of hidden layers and the number of neurons in each layer. There is no systematic way to find the optimal structure. Due to the relatively low dimension of the input (maximum dimension is 16), a two-layer MLP was used and the number of perceptrons in each hidden layer was assumed to be the same. The activation function for all perceptrons was chosen to be the log-sigmoid function. The performance of the MLP is shown in Fig. 19 in which MLP44 means that the number of neuron in both hidden layers is four. Similar to the linear Bayesian classifier, the performance of the MLP was unsatisfactory when the number of features was too small or too large. This further confirmed that redundant features had a negative effect on performance. MLP44 was found to have the lowest error rate and the optimal performance was reached when the number of features was eight.

Another type of non-linear classifier is the radial basis function (RBF) neural network [20]. Fig. 20 shows the structure of the RBF network which consists of three layers: the input layer, the hidden layer and the output layer. The difference between the RBF and MLP is that the RBF has only one hidden layer. The input layer obtains the input from the environment. The hidden layer applies a non-linear transformation through a radial basis function. The output of the system is computed by sending the weighted sum of the outputs of the hidden neurons to an activation function. The difference between the outputs of the network and the desired output is called the error signal. The synaptic weights that connect different neurons are updated such that the error signal is minimized. After the synaptic weights are optimized, the network can be used to classify the new data.

Similar to the MLP, the structure of the RBF also affects the performance. Fig. 21 compares the effect of the number of neurons in the hidden layer. The best performance is reached when the number of neurons is 20. Fig. 22

15

depicts the effect of the spread of the activation function and shows that the performance improves when the spread of the RBF increases but the improvement is not significant when the spread is greater than 10. The best performance was reached when the number of neurons in the hidden layer was 20 and the spread of the radial basis function (in this case, Gaussian function) was 10.

Fig. 23 compares the performance of the best classifiers from each category. The RBF with 20 hidden neurons and a neuron spread of 10 had the best performance and the optimal performance was reached when the number of features was eight. These parameters were used in the delamination detection algorithms in this research.

Performance Verification

The performance of different algorithms in each category was evaluated in the previous sections. An automatic impact-based delamination detection (AIDD) system [23] was developed using the optimal algorithm in each category (i.e., MFCC features and RBF classifier). The impact was created by the free fall of the impactor from a constant height. The impactor was a stainless steel bar of 25 mm (1 inch) diameter with a ball-shaped head. The impact and ambient sound were recorded by a condenser microphone. The AT831b condenser microphone produced by Audio-Technica was used. The frequency response of the microphone is shown in Fig. 24 and indicates that the microphone had good sensitivity in the frequency range of interest. The microphone was directional and recorded the sound within a narrow cone, which helped limit extraneous noise. Two microphones were mounted on the cart. The primary microphone (Mic 1) was mounted under the base of the cart and pointed toward the impact point to record the impacting sound. A sound proofing curtain was mounted as a physical barrier to block traffic and wind noise. The secondary microphone (Mic 2) was mounted outside the sound shield pointing away from the impacting point to measure the ambient noise. The signals from both microphones were collected by a National Instruments data acquisition card (Model No. USB 6211), which has a USB interface and a maximum sampling frequency of 250 kHz. A sample frequency of 10 kHz was used and the length of the signal was 3 seconds. The use of multiple impacts increased the accuracy of the detection. The prototype of the system is shown in Fig. 25.

To test the performance of the proposed algorithms, signals with different signal to noise ratios (SNR) were obtained by mixing the impact signal obtained in a quiet laboratory environment with traffic noise recorded on a highway bridge. The impact sound was created by impacting the surface of a concrete slab with artificial

delamination as shown in Fig. 26. The thickness of the slab was 229 mm and the depth of the two "delaminated" parts were 76 mm (for the left block) and 152 mm (for the right block), respectively, to simulate different delamination depths. The depth of the delamination was controlled by the location of the separation. The acoustic properties of the materials were not tested because the results should not be dependent on these. Trial tests showed that the sound produced from the 152 mm delamination was very similar to that produced by the solid concrete. Methods such as impact-echo can have better performance in detecting deep delaminations, but require data in a broader frequency range. The method outlined here focuses on acoustic methods and the human ear is not very sensitive to high frequency signals. Therefore the signals from the 152 mm block were considered as "solid" in this study.

A total of 228 impacts were recorded on different days to account for variance between days. 120 recordings were obtained from solid concrete and 108 recordings were obtained from delaminated concrete. 40 randomly selected impacts were used for feature extraction and classifier training. The remaining signals were classified by the trained classifier. The average error rates under different conditions were calculated using the AIDD system are listed in Table 4. The proposed algorithms performed well in a quiet environment, yielding an error rate of only 2.3%. However, the accuracy of the algorithms dropped (error rate increased) as the noise level increased if the signals were not pre-processed with the modified ICA algorithm. When the signals were filtered with the modified ICA algorithm, the detection algorithm became less noise sensitive and the error rate remained constant (around 5%) for all noise levels considered.

To evaluate the detection algorithms under field conditions, tests were performed on two bridges using the AIDD system. Both bridges had concrete decks with delaminations. The concrete condition at several spots was first identified through traditional bar tapping (i.e., impacting the bridge deck using a steel bar and listening to the sound). The AIDD system shown in Fig. 25 was then used to test these spots and the sound signals were collected using the data acquisition card.

In real situations, the training signals can only be obtained from existing recordings. To simulate this, data obtained from the two bridges was divided into four groups and labeled as A, B C and D and different combination of the four groups were used as the training data. One group that was not in the training pool was used as testing pool. The training data and testing data were randomly selected from the recordings in the training pool and the

testing pool. The number of impacts in the training data and testing data were 150 and 100, respectively, and were fixed for all cases for fair comparison. Table 5 shows the error rates based on different training sets. If the number of groups in the training pool increased, the average error rate dropped for most cases (except for group D, possibly due to variance in the data). This indicates that the performance increases when more data is included in the training pool. Even with the limited amount of training data in this study, the performance of the system was still satisfactory with a maximum error rate of around 17%. As more bridges are inspected and more data becomes available, the performance will improve.

Conclusions

This paper describes algorithms to improve the performance of traditional impact-based delamination detection methods whose performance is affected by traffic noise in adjacent lanes and subjectivity of the inspection. To eliminate the influence of traffic noise, different noise cancelling algorithms were evaluated using both visual inspection as well as a numerical criterion. Modified independent component analysis (ICA) was selected as the noise cancelling algorithm in this work because of its good performance. Subjectivity in the delamination detection was removed by extracting features of the signal. Mel-frequency cepstral coefficients (MFCCs) of the signals were selected as features for delamination detection because they had the best overall performance in terms of repeatability and separability among all the feature extraction algorithms considered. Delamination detection was posed as a classification problem and different classifiers were tested to select the best classifier. An RBF network with 20 hidden neurons and a neuron spread of 10 had the lowest error rate, was selected as the classifier for delamination detection, and the best performance was reached when the number of features was eight. It is possible that a combination of different feature extraction algorithms and different classifiers could produce better results than the use of a single set of features with a single classifier. However, this was not investigated.

Tests under different noise levels showed that the proposed method increased the noise robustness of traditional sounding methods. Field tests demonstrated that the performance of the proposed algorithm was satisfactory even with the limited amount of training data. As more data becomes available for training, the performance will further improve.

Acknowledgements

References

1. Gibson, A. and J.S. Popovics, *Lamb wave basis for impact-echo method analysis*. Journal of Engineering Mechanics, 2005. **131**: p. 438.

2. Sansalone, M., *Impact-Echo: The Complete Story*. ACI Structural Journal, 1997. **94**(6): p. 777-786.

3. Martin, J., et al., *Ultrasonic Tomography of Grouted Duct Post-Tensioned Reinforced Concrete Bridge Beams*. NDT & E International, 2001. **34**(2): p. 107-113.

4. Shevaldykin, V., A. Samokrutov, and V. Kozlov, *Ultrasonic Low Frequency Short-Pulse Transducers with Dry Point Contact*. Proceedings of Non-Destructive Testing in Civil Engineering (NDT-CE), 2003. **16**(19.9).

5. Bungey, J.H., *Sub-surface Radar Testing of Concrete: A Review*. Construction and Building Materials, 2004. **18**(1): p. 1-8.

6. Brink, A., et al. *Application of Quantitative Impulse Thermography for Structural Evaluation in Civil Engineering - Comparison of Experimental Results and Numerical Simulations*. in *Quantitative Infrared Thermography*. 2002. Croatia.

7. Clemena, G.G. and W.T. McKeel, *Detection of Delamination in Bridge Decks with Infrared Thermography*. Transportation Research Record, 1978(664): p. 180-182.

8. Malhotra, V.M. and N.J. Carino, *Handbook on Nondestructive Testing of Concrete*. 2nd ed. 1991, Boca Raton, FL: CRC Press.

9. ASTM, *Standard Practice for Measuring Delaminations in Concrete Bridge Decks by Sounding*. ASTM-D4580. 2003, West Conshohocker, PA: ASTM Internatinal.

10. Henderson, M.E., G.N. Dion, and R.D. Costley. *Acoustic Inspection of Concrete Bridge Decks*. in *SPIE*. 1999. Newport Beach, CA.

11. MDOT, *Instruction Manual for Bridge Deck Delamination Detector*. 1977, Michigan Department of Transportation: Lansing, MI.

12.   Rolander, D.D., et al., *Highway Bridge Inspection: State-of-the-practice survey*. Journal of the Transportation Research Board, 2001. **1749**: p. 73-81.

13.   Boll, S., *Suppression of Acoustic Noise in Speech using Spectral Subtraction*. IEEE Transactions on Acoustics, Speech and Signal Processing, 1979. **27**(2): p. 113-120.

14.   Comon, P., *Independent Component Analysis: A New Concept?* Signal Processing, 1994. **36**: p. 287-314.

15.   Koldovsky, Z., P. Tichavsky, and E. Oja, *Efficient Variant of Algorithm FastICA for Independent Component Analysis Attaining the Cramér-Rao Lower Bound*. IEEE Transactions on Neural Networks, 2006. **17**(5): p. 1265-1277.

16.   Widrow, B. and M. Hoff, *Adaptive Switching Circuits*. IRE WESCON Convention Record, 1960: p. 96-104.

17.   Vincent, E., R. Gribonval, and C. Fevotte, *Performance Measurement in Blind Audio Source Separation*. IEEE Transactions on Audio, Speech, and Language Processing, 2006. **14**(4): p. 1462-1469.

18.   Zheng, F., G. Zhang, and Z. Song, *Comparison of Different Implementations of MFCC*. Journal of Computer Science and Technology, 2001. **16**(6): p. 582-589.

19.   Lee, S.M., et al., *Improved MFCC Feature Extraction by PCA-Optimized Filter Bank for Speech Recognition*. IEEE workshop on Automatic Speech Recognition and Understanding, 2001: p. 49-52.

20.   Haykin, S., *Neural Networks: A Comprehensive Foundation*. 2nd ed. 1999, Englewood Cliffs, NJ: Prentice Hall.

21.   Kay, S.M., *Fundamentals of Statistical Signal Processing, Volume II: Detection Theory*. 1998, Englewood Cliffs, NJ: Prentice-Hall.

22.   Theodoridis, S. and K. Koutroumbas, *Pattern Recognition*. 2nd ed. 2003, San Diego, CA: Academic Press.

23.   Zhang, G., *Automatic Delamination Detection of Concrete Bridge Decks using Acoustic Signatures*, in *Department of Civil and Environmental Engineering*. 2010, Michigan State University: East Lansing, MI.

Table 1. Performance of Different Noise Cancelling Algorithms

	Spectral Subtraction	Adaptive Filter	ICA	Modified ICA
SDR (dB)	-6.51	-6.21	-10.81	1.21

Table 2. Weighted Rank of Different Feature Extraction Algorithms

	Sub-band	WPT	MFCC	PCA	ICA
REP	2	3	1	5*	4
SEP	5	3	4	2*	1*
Mutual Info.	1	4	2	5	3
Weighted Rank	2.5	3.25	2	4.25*	3*

Note: * means the performance measure is ill-conditioned.

Table 3. Error Rates of Different Feature Extraction Algorithms

Algorithm	Sub-band	WPT	MFCC	PCA	ICA
Error 1 (%)	7.0	13.2	4.0	21.0	8.6
Error 2 (%)	9.1	17.0	6.3	28.5	11.2
Total Error (%)	15.8	30.4	10.2	49.6	19.8

Table 4. Performance under Different Noise Levels

SNR (α)	Measurements	Filtered Signals			Noisy Signals		
		Error 1	Error 2	Total	Error 1	Error 2	Total
∞	$m = s$	1.3	0.5	1.8	N/A	N/A	N/A
10	$m = 10s + n$	4.4	1.0	5.4	4.5	4.0	8.5
1	$m = s + n$	4.8	0.1	4.9	9.1	5.3	14.4
0.1	$m = 0.1s + n$	5.2	0.1	5.4	15.4	5.0	20.3

Table 5. Performance in the Field

Groups in Training	Groups in Testing	Error 1 (%)	Error 2 (%)	Total Error (%)	Average Error (%)
B	A	1.7	4.1	5.8	
C	A	13.3	0.3	13.6	8.7
D	A	6.7	0.0	6.7	
BC	A	1.9	1.6	3.5	
BD	A	2.3	0.1	2.4	5.3
CD	A	9.8	0.2	10.0	
BCD	A	2.4	0.3	2.6	2.63
A	B	0.0	12.5	12.6	
C	B	14.2	5.7	19.9	14.6
D	B	5.4	6.0	11.4	
AC	B	1.5	6.6	8.1	
AD	B	9.8	5.4	15.2	10.1
CD	B	0.4	6.8	7.2	
ACD	B	1.4	6.0	7.4	7.4
A	C	0.0	24.6	24.7	
B	C	2.3	16.7	19.0	21.0
D	C	4.8	14.5	19.3	
AB	C	0.9	17.5	18.4	
AD	C	0.3	15.9	16.2	17.4
BD	C	2.4	15.3	17.6	
ABD	C	1.2	14.5	15.6	15.6
A	D	0.3	6.4	6.7	
B	D	0.6	9.5	10.1	8.8
C	D	7.7	1.8	9.5	
AB	D	0.2	8.5	8.7	
AC	D	15.9	2.3	18.1	14.3
BC	D	11.0	5.1	16.1	
ABC	D	10.3	3.0	13.3	13.3
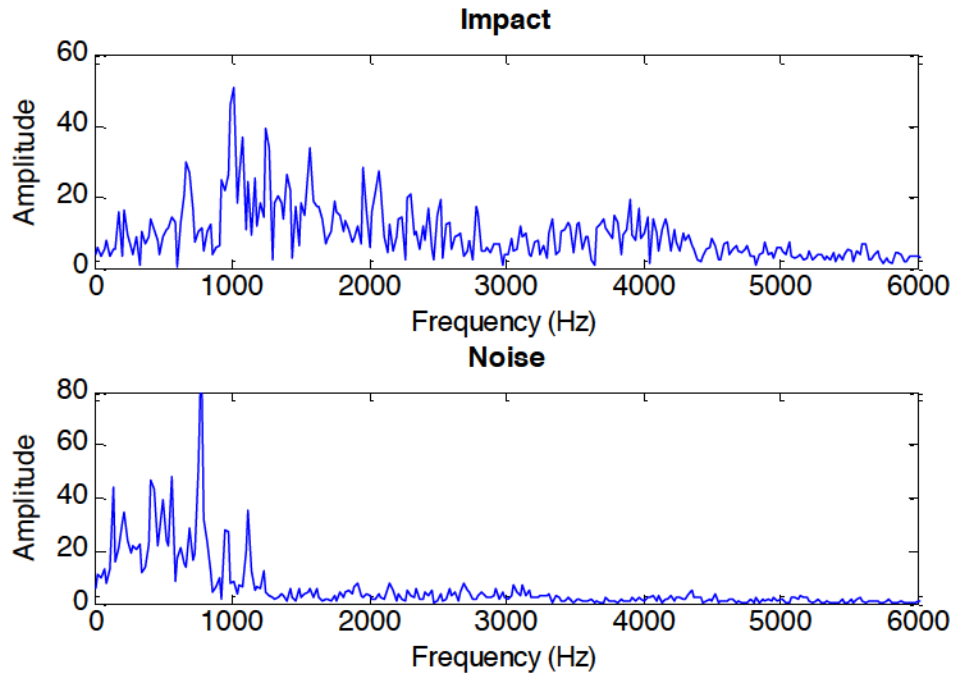
Fig. 1. Spectrum of Impact and Traffic Noise
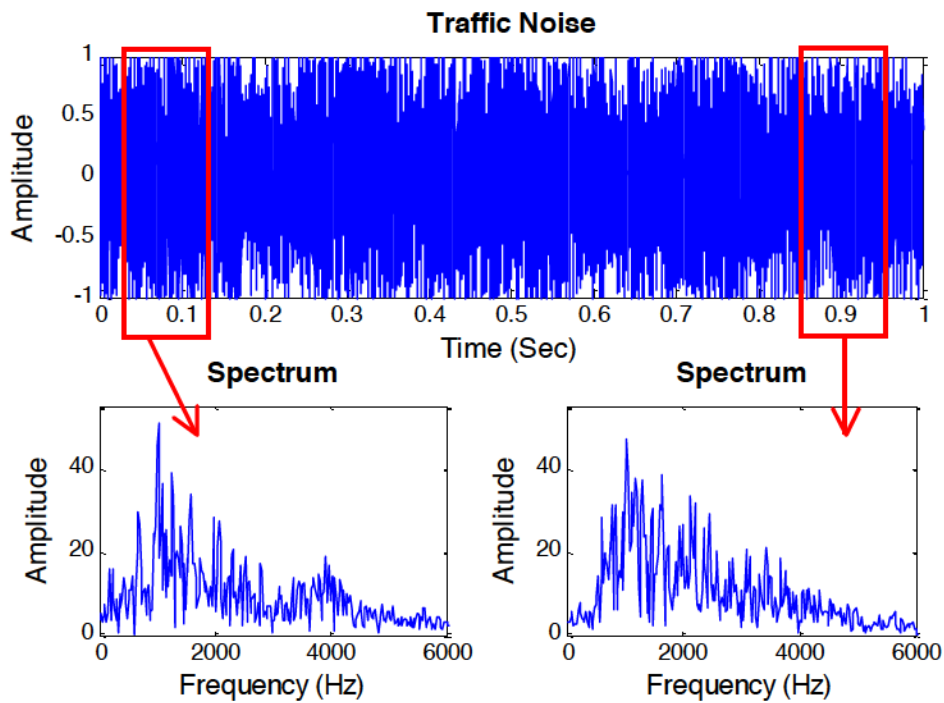
Fig. 2. Non-Stationarity of Traffic Noise
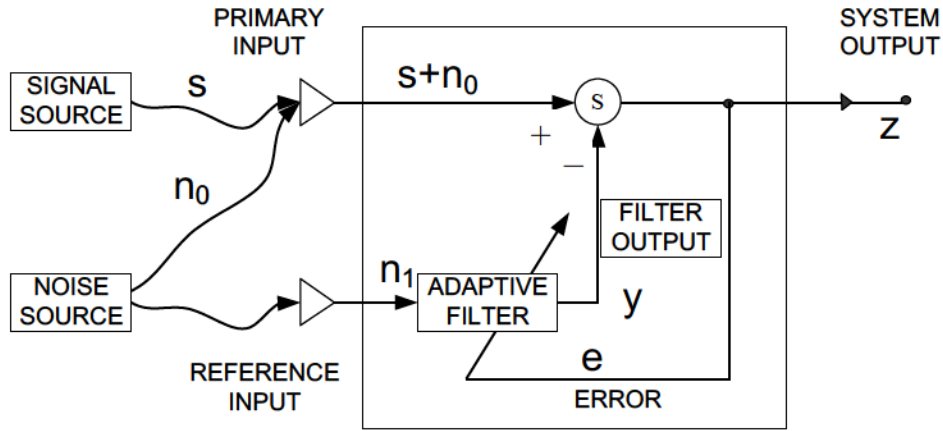
Fig. 3. Signal Flow of the Adaptive Filter
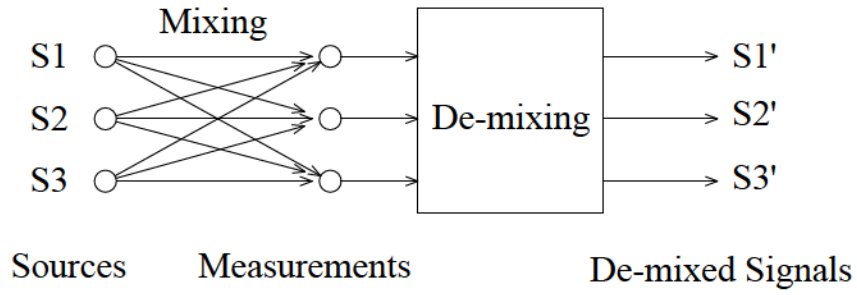
Fig. 4. Illustration of Mixing and Demixing

Fig. 5. Modified ICA

24
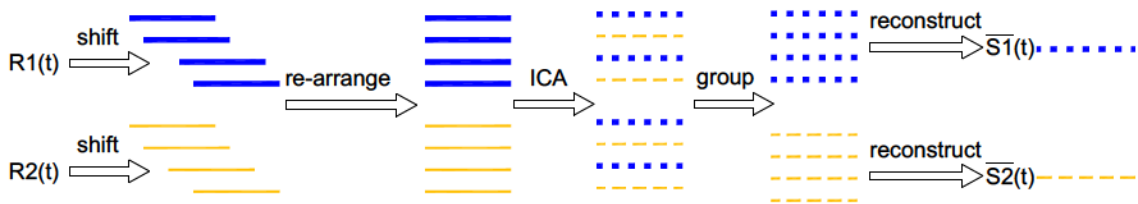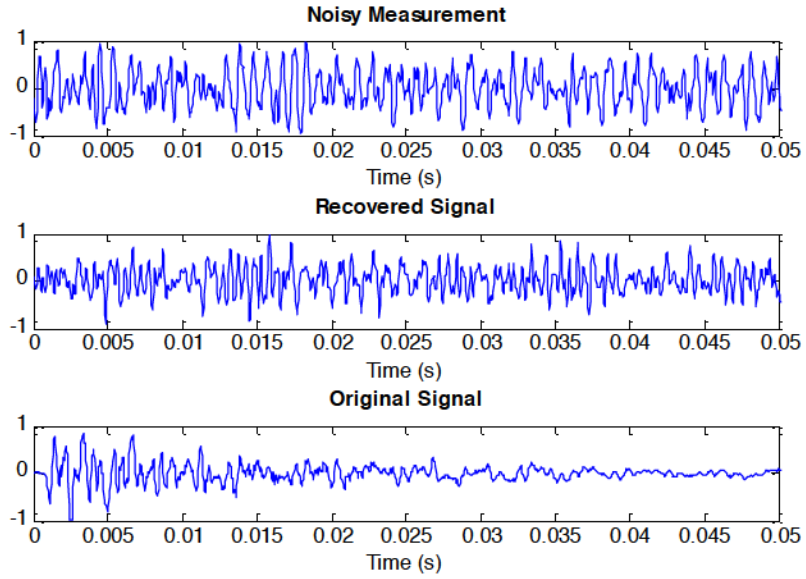
Fig. 6. Performance of Spectrum Subtraction
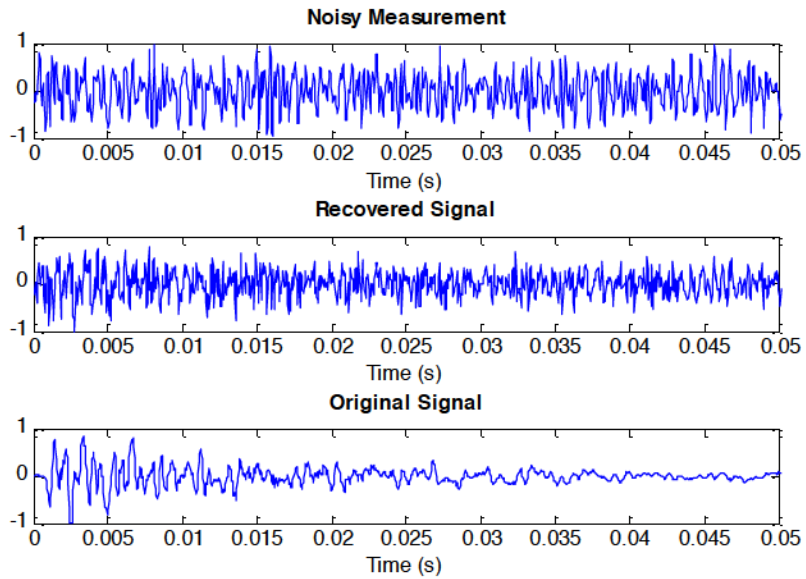
Fig. 7. Performance of Adaptive Filter
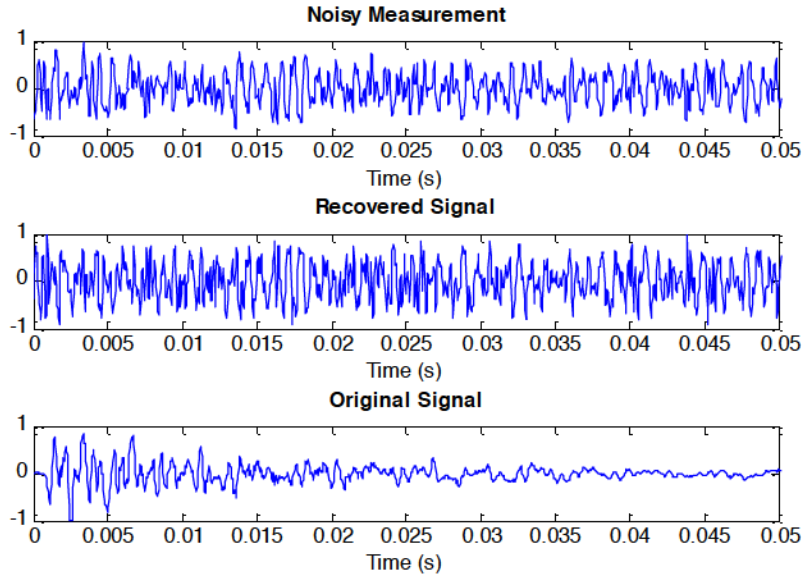
Fig. 8.   Performance of ICA
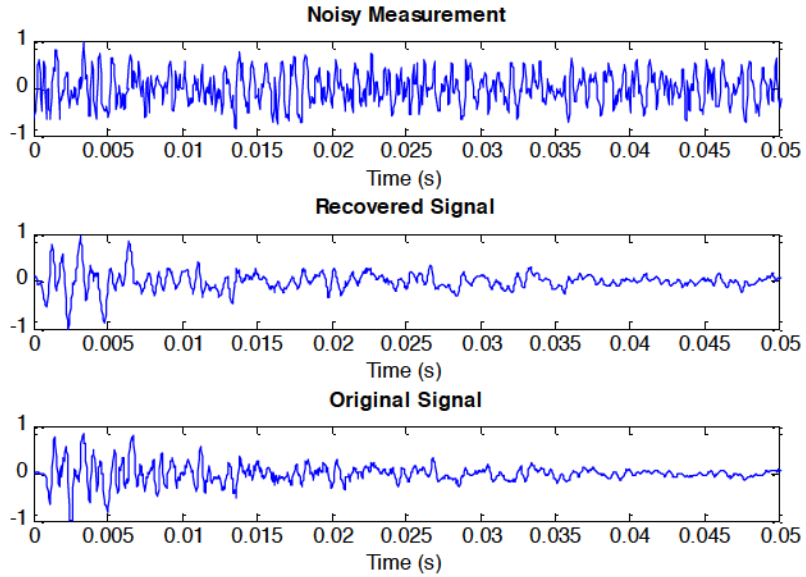
Fig. 9.   Performance of Modified ICA
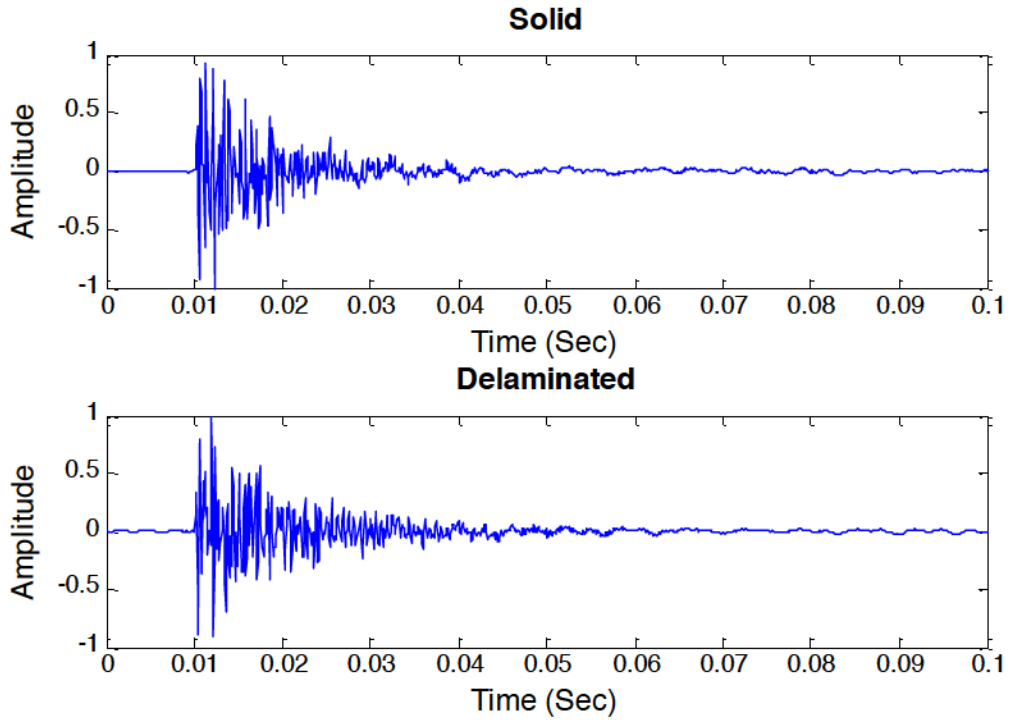
Fig. 10. Example Impact Signals
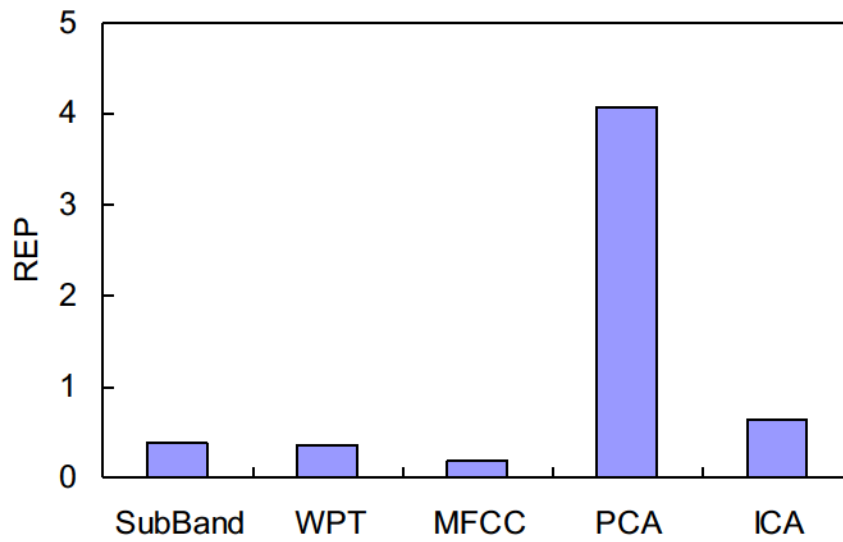
Fig. 11. REP of Different Feature Extraction Algorithms
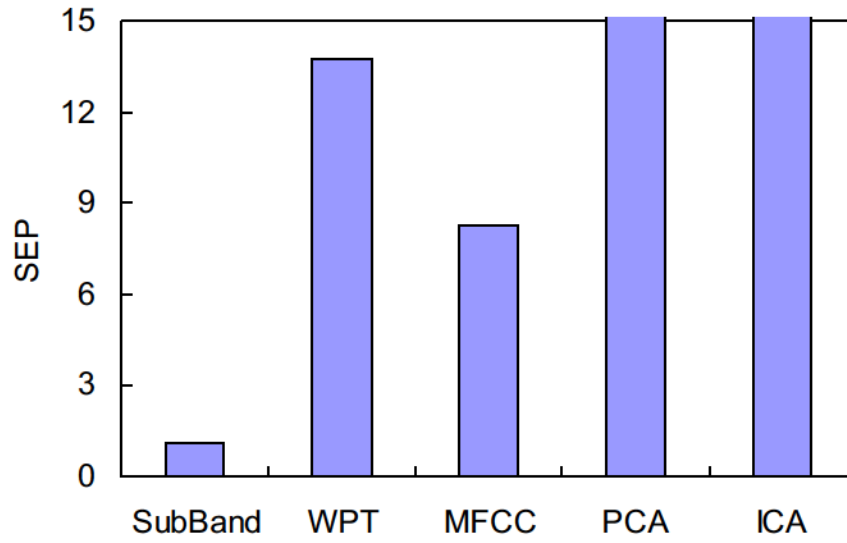
Fig. 12. SEP of Different Feature Extraction Algorithms
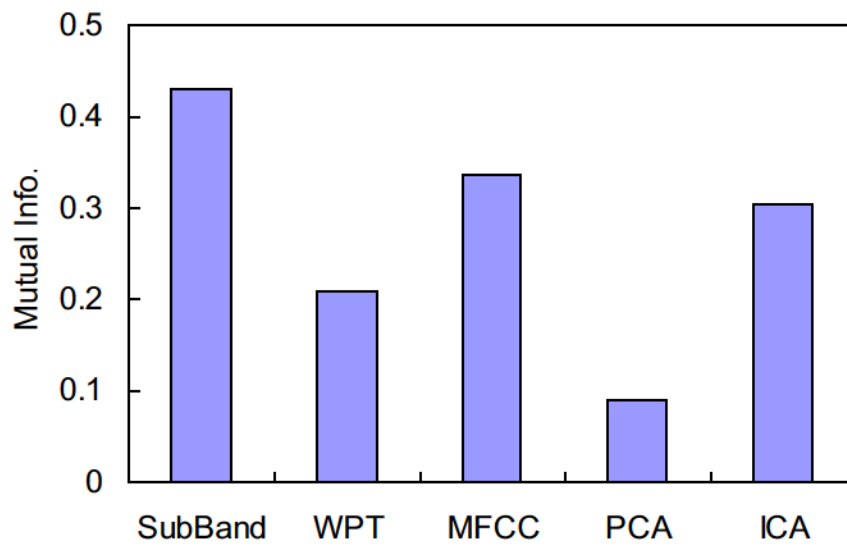
Fig. 13. Mutual Information of Different Feature Extraction Algorithms

$$P(x|C_1) \qquad P(x|C_2)$$

$$\int_{-\infty}^{x_0} P(x|C_2)\,dx \qquad \int_{x_0}^{\infty} P(x|C_1)\,dx$$

$$x_0$$
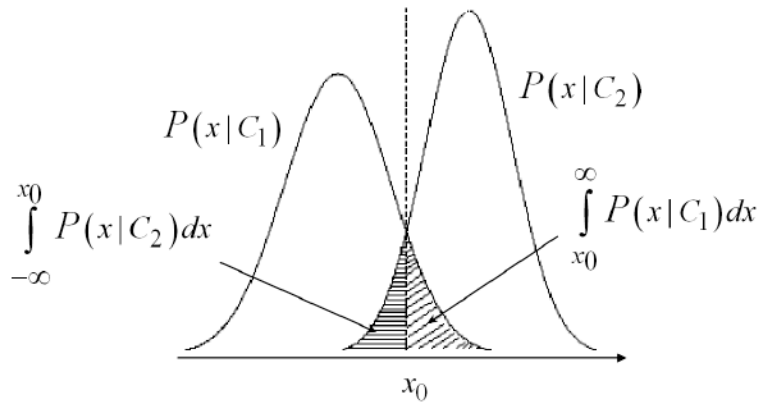
Fig. 14. Threshold of Bayesian Classifiers
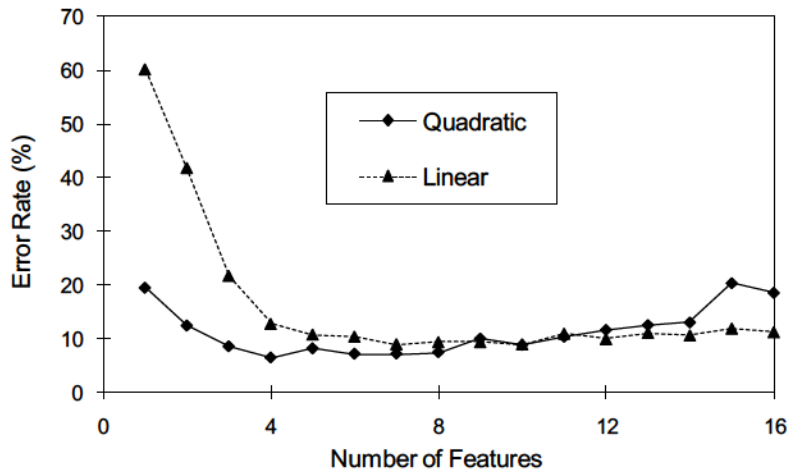
Fig. 15. Performance of Bayesian Classifiers
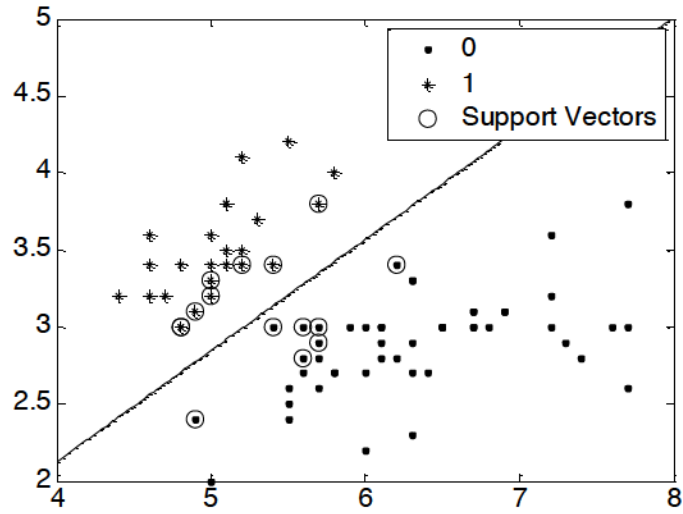
Fig. 16. Support Vector Machine
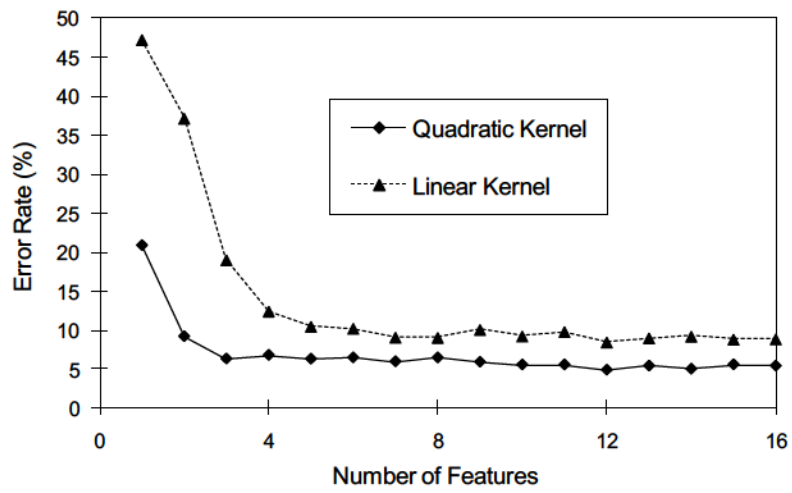
Fig. 17. Performance of SVM

30
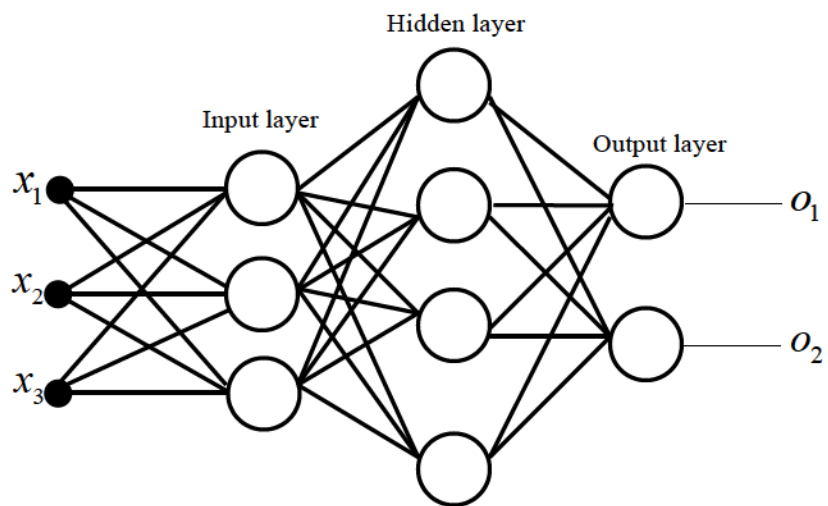
Fig. 18. Structure of MLP
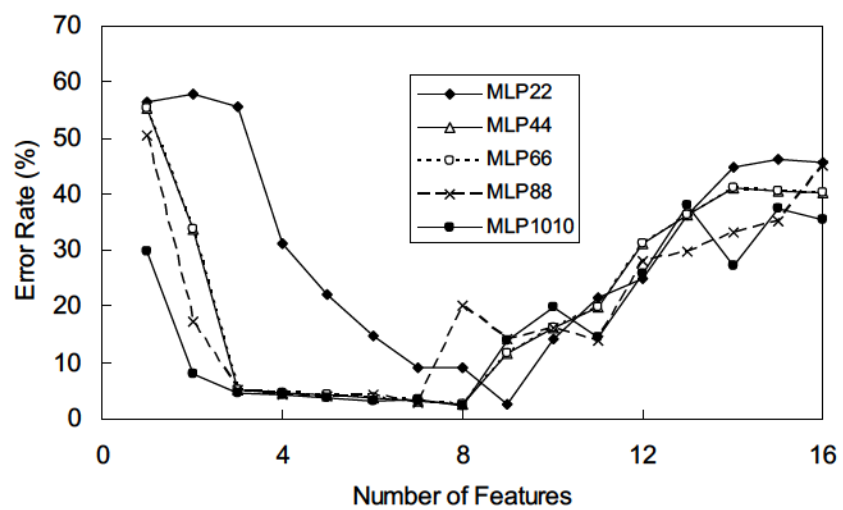
Fig. 19. Performance of MLP
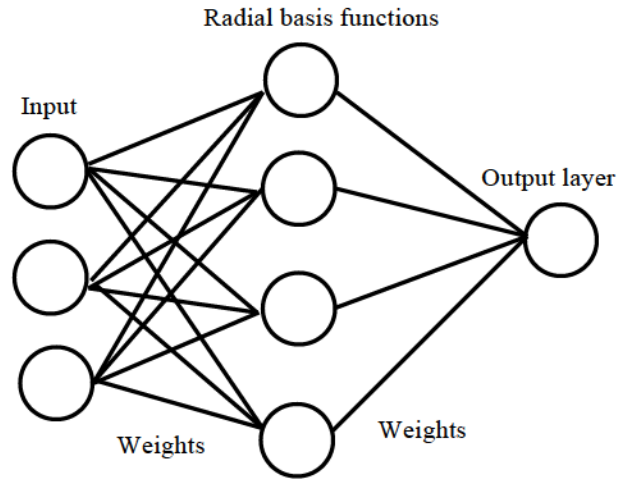
Fig. 20. Architecture of Radial Basis Function Network
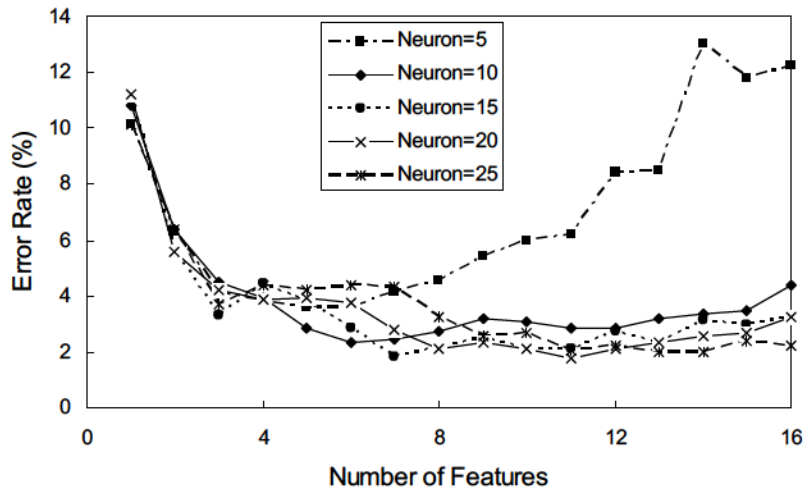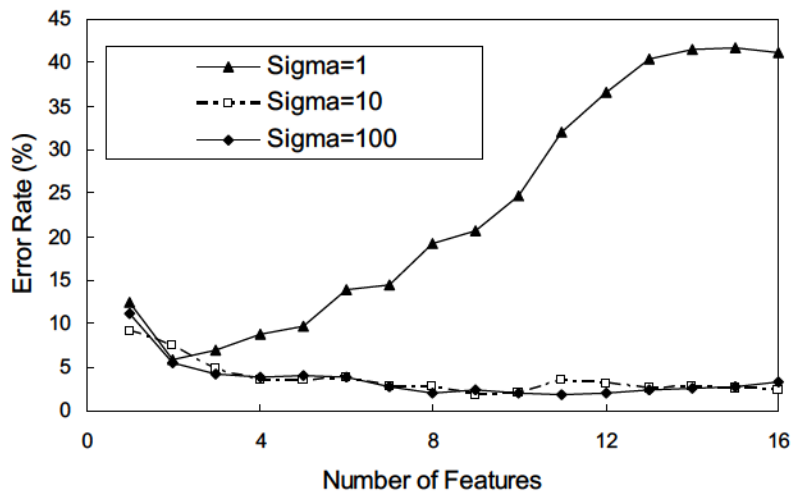

Fig. 21. Effect of the Number of Neurons

32
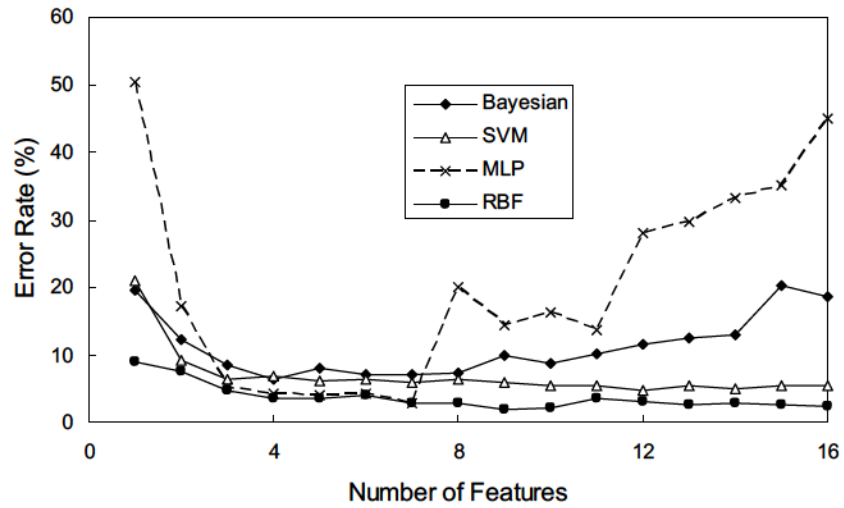
Fig. 22. Effect of Spread of Radial Basis Function

Fig. 23. Comparison of Different Classifiers

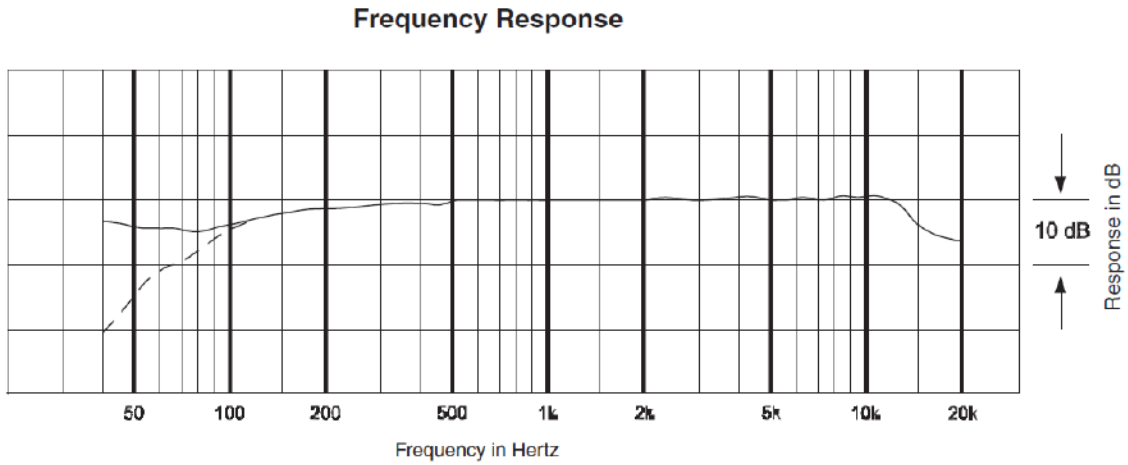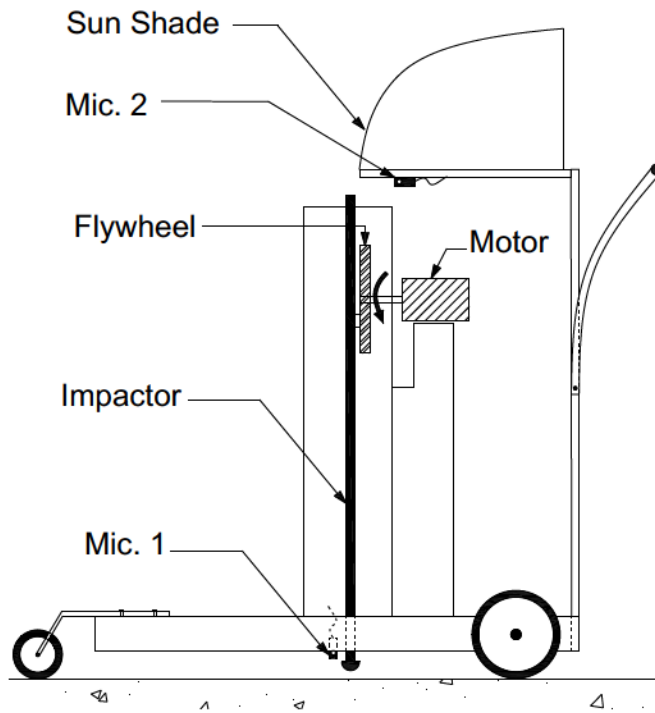**Frequency Response**

Fig. 24. Frequency Response of the Microphone

Fig. 25. Schematic of Impacting Cart

Fig. 26. Test Slab