

Comparison of Different Classification Algorithms for Underwater Target Discrimination

Donghui Li, Mahmood R. Azimi-Sadjadi, and Marc Robinson

Abstract—Classification of underwater targets from the acoustic backscattered signals is considered here. Several different classification algorithms are tested and benchmarked not only for their performance but also to gain insight to the properties of the feature space. Results on a wideband 80-kHz acoustic backscattered data set collected for six different objects are presented in terms of the receiver operating characteristic (ROC) and robustness of the classifiers wrt reverberation.

Index Terms—K-nearest neighbor (K-NN) classifier, neural networks, probabilistic neural networks (PNNs), support vector machines (SVMs), underwater target classification.

I. INTRODUCTION

THE problem of classifying underwater targets from the acoustic backscattered signals involves discrimination between targets and nontarget objects as well as the characterization of background clutter. Several factors that complicate this process include: nonrepeatability and variation of the target signature with aspect angles, range and grazing angle, competing natural and man-made clutter, highly variable and reverberant operating environment, and lack of any a priori knowledge about the shape and geometry of the nontargets.

A number of different classification schemes have been developed in recent years. A good review of the previous methods is provided in [1]. The method in this reference uses a wavelet packet-based classification scheme to discriminate mine-like and nonmine-like objects from the acoustic backscattered signals. The classifier was a back-propagation neural network (BPNN). A separate multispect fusion system was also implemented to improve the classification accuracy by observing the properties of the returns in several consecutive aspects. The test results on an 80-kHz data set were presented, which showed the promise of the system. With the exclusion of the work by Carpenter and Streilein [2], which used an ARTMAP-based classification system and provided reasonable single-aspect and multispect results, all other schemes almost exclusively used a BPNN classifier with different sets of features.

This brief paper studies the performance of other classifiers that use different decision rules on this problem. The purpose is to determine an optimum classification scheme for this problem, and also to infer clues about the properties of the data and more specifically the features. These clues could potentially provide

helpful insight into the feature space and distribution of the features, which in turn leads to a better understanding of this complex problem. Several different classification algorithms are considered and benchmarked. Among these are the multivariate Gaussian classifier, the evidential K-nearest neighbor (K-NN) classifier [4], probabilistic neural network (PNN) [5], [6] and support vector machines (SVM) [7]–[9]. The performance of these systems are then compared with that of the BPNN [1] on the wideband (80 kHz) data set provided by Coastal Systems Station (CSS), Panama City, FL.

II. BRIEF REVIEW OF DIFFERENT CLASSIFIERS

A brief review of different classification methods that are considered in this study is given here.

A. Multivariate Gaussian Classifier

This simple classifier [3] was used to give a benchmark for comparison with the other more complicated ones. This classifier works well when the data is assumed to be clustered in two distinct groups with normal distribution. In other words, the feature vectors are assumed to be simply randomly corrupted versions of two prototype vectors. Clearly, in case of the acoustic return features, this assumption is not a valid one as the change in the scattering and physical properties of the objects as a function of aspect and grazing angles, etc. cannot be modeled by a random process. In a two-class case (i.e. c_i , $i = 1, 2$), a very simple discriminant function $g(\mathbf{y}) = \ln[p(\mathbf{y}|c_1)/p(\mathbf{y}|c_2)] + \ln[P(c_1)/P(c_2)]$ is used, where $p(\mathbf{y}|c_i)$ is the a priori conditional probability density function (assumed to be normal) and $P(c_i)$ is the class prior probability.

B. Evidential K-Nearest Neighbor (K-NN) Classifier

In our two-class classification problem, $C = \{c_1, c_2\}$, where the possible hypotheses are either target or nontarget, an evidential K-NN classifier [4] finds the K nearest neighbors (training samples) of an unknown pattern and then prescribes a belief and plausibility measure to each neighbor or item of evidence. To each item of evidence, a basic belief assignment (BBA) is assigned. A common approach [4] is to find some type of a distance measure from the unknown pattern to each piece of evidence and then use $m^k(c_i) = \alpha \exp[-\gamma_i d_k^\beta]$ where $m^k(c_i)$ is the BBA for the k^{th} item of evidence in the neighborhood, i is the class label of this item of evidence, $0 < \alpha < 1$ is a constant, γ_i is a class dependant constant, β is an integer constant (typically 1 or 2) and $d_k = d(\mathbf{x}, \mathbf{x}_i^k)$ is a distance measure (e.g. Euclidean distance) between the evidence \mathbf{x}_i^k belongs to class c_i and the unknown pattern \mathbf{x} . The remaining belief not

Manuscript received May 9, 2001; revised June 5, 2002. This work was supported by the Office of Naval Research, Biosonar Program, under Grant N00014-99-1-0166 and Grant N00014-01-1-0307. Data and technical support were provided by the NSWC, Coastal Systems Station, Panama City, FL.

The authors are with the Department of Electrical and Computer Engineering, Colorado State University, Fort Collins, CO 80523 USA (e-mail: azimi@engr.colostate.edu).

Digital Object Identifier 10.1109/TNN.2003.820621

assigned to class i is assigned to the set of classes C , of which c_i is a subset. Thus, $m^k(C) = 1 - \alpha \exp[-\gamma_i d_k^\beta]$. Clearly, the farther away a piece of evidence is from the unknown sample, the less effect it will have on the belief in a class and it will be assigned completely to the set of classes C . For this two-class problem the belief caused by each evidence is placed in the same class, which is known from the training labels. Then, the BBA of all the K neighbors are combined using the Dempster's rule of combination [4]. Using this "orthogonal sum of the beliefs" rule, the BBAs, m_1 and m_2 , associated with two independent items of evidence can be combined to produce the BBA for the collective evidence, i.e., $m = m_1 \oplus m_2$ by

$$m(A) = \frac{\sum_{B_1 \cap B_2 = A} m_1(B_1)m_2(B_2)}{1 - \sum_{B_1 \cap B_2 = \emptyset} m_1(B_1)m_2(B_2)}. \quad (1)$$

Then, the class with the highest pignistic probability is declared as the winner. This probability is defined as

$$BetP(c_i) = \sum_{c_1 \in A} \frac{m(A)}{|A|} \quad \forall c_i \in C \quad (2)$$

where $|A|$ represents the cardinality of $A \subseteq C$.

C. PNN

PNN was introduced [5], [6] to implement the Parzen nonparametric probability density function (pdf) estimation method. To overcome the structural and computational (during the testing phase) of this network Streit *et al.* [12] introduced a modified version of the PNN that substantially reduces the number of neurons in the recognition layer by using Gaussian mixture models and the expectation maximization (EM) algorithm to estimate the network parameters. This algorithm not only reduces the computational costs during the training, but it also overcomes the issues with unbalanced training problems and the optimal kernel selection.

In this algorithm, for any class c_i , $i = 1, \dots, K$ the class conditional distribution is modeled approximately by a Gaussian mixture as

$$p(\mathbf{y}|c_i) = \sum_{j=1}^{M_i} \pi_{ji} p_{ji}(\mathbf{y}; \mu_{ji}, \Sigma_{ji}) \quad (3)$$

where M_i is the number of Gaussian components in class c_i that is generally decided experimentally, π_{ji} 's are the weights of the components (note, that $\sum_{j=1}^{M_i} \pi_{ji} = 1$), $p_{ji}(\mathbf{y}; \mu_{ji}, \Sigma_{ji})$ denotes the multivariate Gaussian density function of the j^{th} component in class c_i with μ_{ji} and Σ_{ji} being its mean vector and covariance matrix, respectively. The algorithm in [12] achieves the maximum likelihood (ML) estimates of these parameters via iterative computation when the observations can be viewed as incomplete data.

If the Gaussian mixture model is a good assumption, the modified PNN classifier will achieve high classification accuracy. Although the number of Gaussian components can always be increased so that the mixture model can better approximate the distribution, the computational cost may become unacceptable.

D. SVM

SVM [7]–[9] are powerful learning systems primarily for two-class problems. SVM maps the input patterns into a higher dimensional feature space through some nonlinear mapping chosen a priori. A linear decision surface is then constructed in this high dimensional feature space. Thus, SVM is a linear classifier in the parameter space, but it becomes a nonlinear classifier as a result of the nonlinear mapping of the space of the input patterns into the high dimensional feature space. The process involves solving a quadratic programming problem. SVM has been shown [7] to provide high generalization ability.

For a two-class problem, assuming the optimal hyperplane in the feature space is generated, the classification decision of an unknown pattern \mathbf{y} will be made based on

$$f(\mathbf{y}) = \sum_{i=1}^N \alpha_i y_i K(\mathbf{x}_i, \mathbf{y}) + b_0 \quad (4)$$

where $\alpha_i \geq 0$, $i = 1, 2, \dots, N$ are nonnegative Lagrange multipliers that satisfy $\sum_{i=1}^N \alpha_i y_i = 0$, $\{y_i | y_i \in \{-1, +1\}\}_{i=1}^N$ are class labels of training patterns $\{\mathbf{x}_i | \mathbf{x}_i \in R^N\}_{i=1}^N$, and $K(\mathbf{x}_i, \mathbf{y})$ for $i = 1, 2, \dots, N$ represents a symmetric positive definite kernel function that defines an inner product in the feature space [9]. This shows that $f(\mathbf{y})$ is a linear combination of the inner products or kernels.

Note that output $f(\mathbf{y})$ corresponds to the distance d of the unknown pattern \mathbf{y} to the hyperplane in the feature space where the distance of the support vectors to the hyperplane is presumed to be 1.

The kernel function enables the operations to be carried out in the input space rather than in the high-dimensional feature space. Some typical examples [7] of kernel functions are $K(\mathbf{u}, \mathbf{v}) = \mathbf{v}^T \mathbf{u}$ (linear SVM); $K(\mathbf{u}, \mathbf{v}) = (\mathbf{v}^T \mathbf{u} + 1)^n$ (polynomial SVM of degree n); $K(\mathbf{u}, \mathbf{v}) = \exp\{-\|\mathbf{u} - \mathbf{v}\|_2^2 / \sigma^2\}$ (RBF SVM); $K(\mathbf{u}, \mathbf{v}) = \tanh(\kappa \mathbf{v}^T \mathbf{y} + \theta)$ (two layer neural SVM) where σ , κ , and θ are constants. However, a proper kernel function for a certain problem is dependent on the specific data and till now there is no good method on how to choose a kernel function. In this paper, the choice of the kernel functions was studied empirically and optimal results were achieved using second-order polynomial kernel function $K(\mathbf{u}, \mathbf{v}) = (\mathbf{v}^T \mathbf{u} + 1)^2$.

The generalization ability of the SVM is controlled by two different factors: the training error rate and the capacity of the learning machine measured by its VC dimension [9]. The smaller the VC dimension of the function set of the learning machine, the larger the value of training error rate. We can control the tradeoff between the complexity of decision rule and training error rate by changing a parameter C [8] in the SVM.

III. RESULTS AND DISCUSSIONS

To examine the effectiveness of the above-mentioned classifiers, the algorithms were tested and benchmarked on the wideband 80 kHz data set provided by CSS. This data set contains backscattered signals corresponding to six different objects—two mine-like namely a bullet-shape metallic object and a truncated-cone-shape plastic object; and four nonmine-like, namely a water-filled drum, an irregular shape limestone rock, a smooth granite rock, and a water-saturated wooden

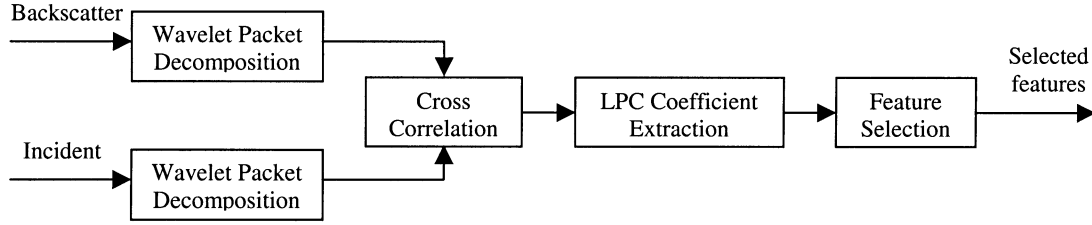


Fig. 1. Feature extraction and reduction processes.

log. The transmit signal was a linear frequency modulated (LFM) up-sweep with frequency range from 30 to 110 kHz. Each object was insonified at aspect angles from 0 to 355° with 5° separation. This resulted in 72 aspect angles out of which the even-angles were used in the training data set while the odd-angle samples were used as the testing samples. As a result, for each object there were 36 patterns (at different aspect angles) in the training or testing data sets. The training data set contained the feature vectors of backscattered data with synthesized reverberation effects with signal-to-reverberation (SRR)= 12 dB that corresponds to nominal operating conditions. The procedure for generating synthesized reverberation involves convolving the transmit signal with a random sequence and scaling the resultant signal according to the specified SRR [1]. The synthesized reverberation signal is then added to the backscattered signal to generate one “noisy realization.” The process is repeated for every aspect angle multiple times in order to generate a statistically rich data set for determining the generalization ability of the classifier. The testing data in our study contained two sets of ten and 50 noisy realizations with SRR = 12 dB for each aspect angle in the testing data in order to obtain statistically significant results.

The steps involved in the front-end feature extraction and selection system are shown in Fig. 1. A five-level wavelet packet (WP) decomposition [10] was applied to decompose the frequency spectra of the acoustic backscattered signals into several subbands. The subband features can provide sufficient discriminatory clues that aid target/nontarget discrimination. Only 12 subbands that reside within the frequency range of the transmit signal (i.e., 30–110 kHz) were selected. The transmit signal was also decomposed using the same WP tree structure. A third-order linear autoregressive (AR) model was employed to represent the spectral behavior of the signals. The AR coefficients were then used as features for classification. This led to a total of 48 features. To select an appropriate set of features, a function can be used to provide the discriminatory power of the individual features. In this study, the Batcharayan discriminant function [3] was used to evaluate the distance between the two classes for each feature and select 22 (out of 48) features with high discriminatory power. This measure was chosen based upon the observation of the histograms of the features exhibit a unimodal Gaussian distribution.

A. ROC Analysis and Classification Error of Different Classifiers

The classifiers are benchmarked on a data set containing ten noisy realizations with SRR = 12 dB in terms of their receiver operating characteristics (ROC) curves and the error loca-

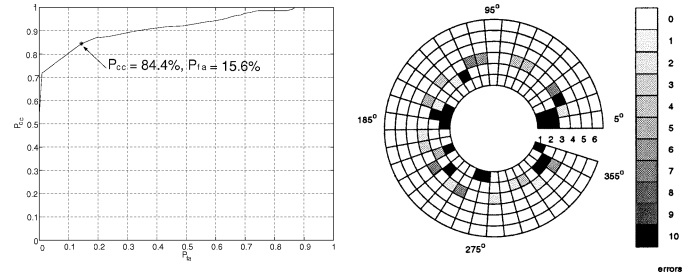


Fig. 2. Classification results of the multivariate Gaussian.

tion plots versus aspect angle. The ROC curve is the plot of the probability of correct classification rate P_{cc} versus probability of false-alarm P_{fa} . The “knee point” of this ROC is particularly important. This point corresponds to a decision threshold leading to $P_{cc} + P_{fa} = 1$. The error location plot, on the other hand, gives the corresponding classification error locations and their frequencies versus aspect angles for each object. In this plot, each ring corresponds to a particular object (i.e., objects 1–6) and each grid on the ring represents an aspect angle. In the counter-clockwise direction, the aspect angles start from 5° to 355° with 10° separation since there are 36 odd aspect angles for every object in the test set. The gray levels in each grid, which can vary from 0 to 10, represents the frequency of classification error at that particular aspect angle.

1) *Multivariate Gaussian*: As mentioned before, this classifier works well when the data is assumed to be clustered in two distinct groups with normal distribution. Fig. 2(a) shows the ROC curve for this classifier. At the knee point we have $P_{cc} = 84.4\%$ and $P_{fa} = 15.6\%$. Fig. 2(b) shows the corresponding error location plot. As evident from this plot the misclassification for this classifier occurs only for the first three objects. The less than adequate results of this classifier for this data set can be blamed on several factors including: impropriety of Gaussian assumption, inseparability of the clusters and the fact that the patterns are not randomly corrupted versions of some prototype vectors.

2) *Evidential K-NN*: When using the evidential K-NN, several parameters need to be preselected. We chose $\alpha = 0.95$, $\beta = 1$ and γ can be found heuristically for each class by estimating $\gamma_i = 1/\delta_i$, where δ_i is the mean of the distances between all combinations of two training vectors belonging to class i . A gradient descent algorithm [4] can also be used to find the optimum value of γ over the training set. This method is based upon minimizing classification error by finding the optimum γ . These methods were tested for various K ranging from 7 to 25 and the results indicated that the gradient-based approach performed slightly better than the heuristic-based method. This

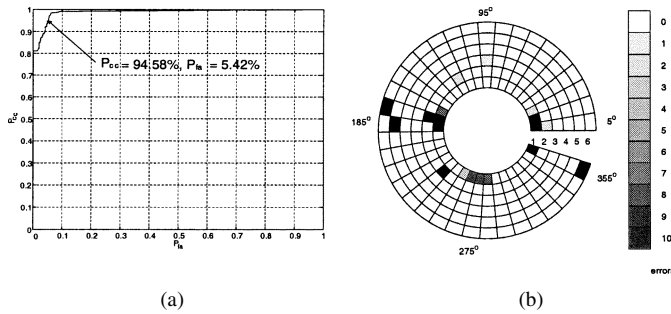


Fig. 3. Classification results of the evidential K-NN: $K = 7$ with optimized γ .

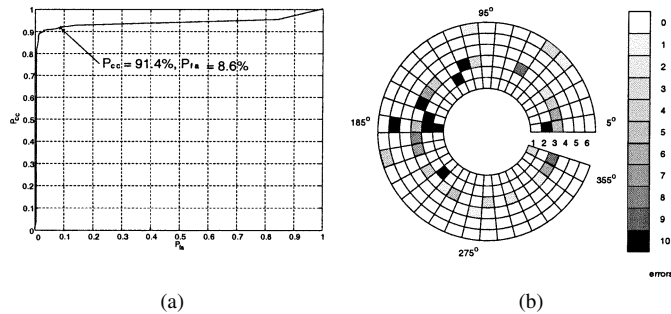


Fig. 4. Classification results of the original PNN ($\sigma = 2$).

study also revealed that the best results are obtained for $K = 7$. The performance slightly degrades as K increases. Fig. 3(a), (b) shows the ROC and error location plots for this classifier. At the knee point, we have $P_{cc} = 94.58\%$ and $P_{fa} = 5.42\%$. From the corresponding error location plot, it is interesting to see that the errors are concentrated only on a few aspects of objects 1, 2, 5, and 6 with high frequency of occurrence. This is in contrast to the multivariate Gaussian classifier where the errors were more widely spread among aspects of objects 1–3.

3) *PNN*: First, the original PNN [2] was implemented. The effects of the smoothing factor, σ on the performance of the PNN was studied first. It was empirically determined that the minimum classification error rate (6.16%) was achieved at $\sigma = 2$. Unlike BPNN, whose output units directly provide the a posteriori class conditional probabilities, the output units of the PNN are only proportional to these probabilities. Thus, the output units of PNN had to be scaled so that the sum of the outputs equals 1. In this case, the outputs could be compared with those of the BPNN and SVM. Fig. 4(a) shows the ROC curve of this PNN ($\sigma = 2$) with $P_{cc} = 91.4\%$ and $P_{fa} = 8.6\%$ at the knee point. Fig. 4(b) shows the corresponding error location plot based upon the decision threshold at the knee point of the ROC. Comparing this figure with Fig. 3(b) for evidential K-NN, it is clear that PNN has a slightly inferior performance. In particular, object 3 (water-filled drum) has a large number of classification errors, while the evidential K-NN had none. Note that although the error location plot in Fig. 3(b) was generated using the hard-limiter decision, this comparison is fair since we consider the optimal performance of each classifier irrespective of the choice of the decision threshold.

Since a PNN creates a pool of neurons for each class, the training data should be balanced to avoid disparity in approximating different density functions. In this application, the

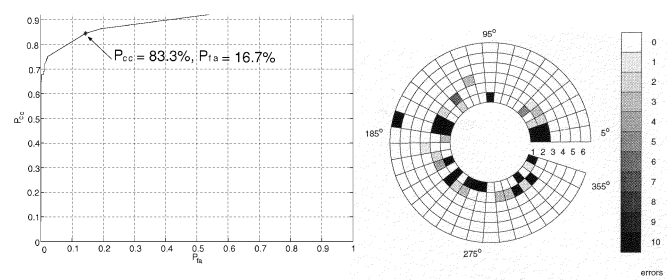


Fig. 5. Performance plots of the EM-PNN.

training was not balanced. To study the effect of the balanced data on the PNN, one half of the training samples for the nontargets were removed. Thus, the training set was left with 72 aspects of targets and 72 aspects of nontargets. For this balanced training set, a $\sigma = 4$ provided the best results. The minimum classification error for this choice of σ was 6.25%, only 0.09% less than the unbalanced data. On the ROC curve, the knee point was located at $P_{cc} = 94.03\%$ and $P_{fa} = 5.97\%$. This is an improvement of just above 2% over the unbalanced training set. These changes are very slight and from this observation it can be concluded that the unbalanced training set has very little impact on these results and that the poor performance of the PNN must be attributed to some other reasons.

The modified PNN using Gaussian mixture models and EM training was examined next. The purpose of the study was to determine if the relatively poor performance of the original PNN can indeed be attributed to the optimum kernel selection and/or the unbalanced training data problems. During the training phase, EM-PNN algorithm automatically selects the number of Gaussian mixtures to model the distribution of the training data for each class. For this particular application only 2 mixtures were selected for each class. Again the outputs of the EM-PNN were scaled and normalized so that they could be compared with the other classifiers. Fig. 5(a) shows the ROC curve of this EM-PNN. At the knee point of this curve, we have $P_{cc} = 83.3\%$ and $P_{fa} = 16.7\%$, which provides substantially worse performance comparing to the original PNN. Fig. 5(b) shows the corresponding error location plot based upon a hard-limiting threshold. As can be seen from Fig. 5(b), a large amount of errors are made for the targets. Since the EM-PNN does not require even training sets, one can conclude that this is not the reason for poor performance on the targets in the testing. To stay consistent, however, an EM-PNN was trained on the balanced data set of 72 aspects of both targets and nontargets. As expected, the results were very similar. The unbalanced data set had a hard-limiter threshold correct classification rate of 88.5%, while the performance on the balanced data improved by half a percentage point. From these results one can conclude that the poor performance of the PNN-type networks is not caused by the unbalanced data set or the choice of kernel.

4) *SVM*: The choice of C parameter [4] in SVM is very critical in order to have a properly trained SVM. Our studies on this problem indicated that when the value of C is in the range of ∞ to 0.001, the classification error rate of SVM remains relatively constant. When the value of C further decreases, the performance of SVM degrades rapidly. Moreover, when $C \leq 0.01$,

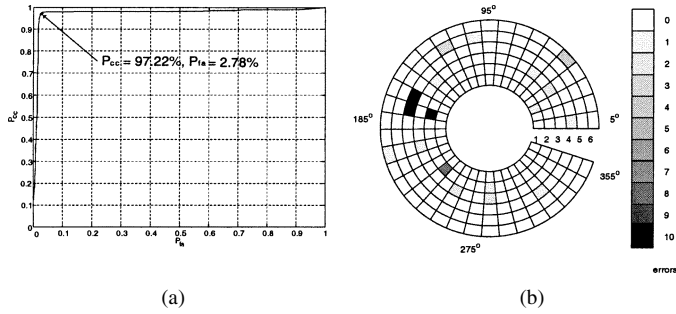
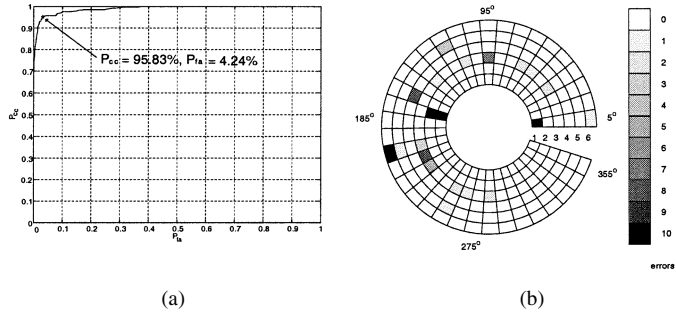
Fig. 6. Classification results of the SVM ($C = 0.001$).

Fig. 7. Performance plots of the BPNN.

the number of support vectors in SVM training was found to be between 25 and 27. For $C = 0.001$, the number of support vectors increased to 39 and the performance of SVM was slightly improved. Also, the training error rate for $C = 0.001$ was 0, i.e., SVM was perfectly trained. For larger values of C , the number of support vectors increased but the performance of SVM deteriorated drastically. Thus, the SVM for $C = 0.001$ was chosen for the rest of this study.

Fig. 6(a) shows the ROC curve of the SVM for $C = 0.001$. At the knee point of this curve, we have $P_{cc} = 97.22\%$ and $P_{fa} = 2.78\%$. As mentioned in Section III-C, the output of the original SVM relates to the distance, d , of the test pattern to the decision boundary where the distance of the support vectors to the decision boundary is assumed to be 1. The transformation $O = d/\sqrt{d^2 + 1}$, where O is the new output of the SVM, was used in order to generate output values $1 \leq O \leq -1$ that can be compared with those of the other classifiers, i.e. PNN and BPNN and to generate the ROC. Fig. 6(b) is the error location plot for the decision threshold corresponding to the ROC knee point. These results indicate excellent performance of SVM on this data set. Most of the classification errors for this classifier occur for objects 2 (truncated-cone-shape target) and 4 (limestone rock).

5) *BPNN*: As a benchmark, a two-layer BPNN with structure 22-42-2 was also used in this study. This network was trained using the standard BPNN training algorithm using an adaptive learning rate and momentum factor of 0.9. Ten different weight initialization trials were implemented and the network with the best performance on the validation data set was chosen. Fig. 7(a) shows the ROC curve of the BPNN for ten noisy realizations. At the knee point, we have $P_{cc} = 95.8\%$ and $P_{fa} = 4.2\%$. Fig. 7(b) is the corresponding error location plot at the ROC knee point. As can be observed for this classifier most of the errors occurred for the mine-like object 1 and nonmine-like objects 3 and 6. This is an interesting observation

TABLE I
COMPARISON OF ROBUSTNESS OF DIFFERENT CLASSIFICATION ALGORITHMS
FOR 50 NOISY REALIZATIONS (%)

Classification algorithms	μ	σ
Multi-Variate Gaussian	0.114	0.011
Evidential K-NN	0.068	0.0052
PNN	0.063	0.0051
EM-PNN	0.1142	0.008
SVM	0.026	0.0061
BPNN	0.039	0.0062

since the errors of all these classifiers occur for different objects and at different aspect angles. This suggests the possibility that further improvements could potentially be gained by fusing the results of these classifiers.

B. Robustness Analysis of Different Classifiers

To study the robustness and generalization of different classifiers to reverberation, we have studied the error rate statistics for a larger number of trials. For a given set of input pattern vectors, the number of misclassifications can be viewed as a random variable resulting in a randomly varying empirical error rate. In [15], irrespective of the pattern source and the type of the classifier, it is shown that this random variable follows a binomial distribution. For a very large number of patterns, this distribution approaches normal distribution. To estimate the error rate statistics of the classifiers, fifty different Monte Carlo trials were performed. For each trial, a different testing set with different reverberation realizations was used. As before, each set contained 216 backscattered signals corresponding to odd aspect angles and different synthesized reverberation sequences with $SRR = 12$ dB. The histograms of the classification errors for the multivariate Gaussian, evidential K-NN, PNN, EM-PNN, SVM, and BPNN classifiers were then formed for 50 error measurements. These were obtained under the assumption that false positive and negative errors have the same weighting, i.e., the false alarm and misclassification errors are summed up to a total classification error. The classification decision was made based on hard-limit threshold operation. The threshold associated with the “knee” point of the ROC curve was not used since each classifier has a different knee point and hence different threshold. Table I summarizes the classification error statistics, i.e., the mean, μ , and the standard deviation σ , extracted from the histogram associated with each classifier. The results in this table reinforce the fact that SVM gave the best overall results on this wideband data set. Although, PNN gave inferior results, comparing to all other classifiers except the multivariate Gaussian, the online updating feature of this classifier makes it attractive for *in situ* underwater target classification applications.

C. Discussion and Analysis

Each of these classification schemes and their respective results give insights into the 22-dimensional feature space of this complex problem. Even though the features are fairly unimodal individually, the distribution of these features in the 22-dimensional space will not necessarily follow a multivariate Gaussian

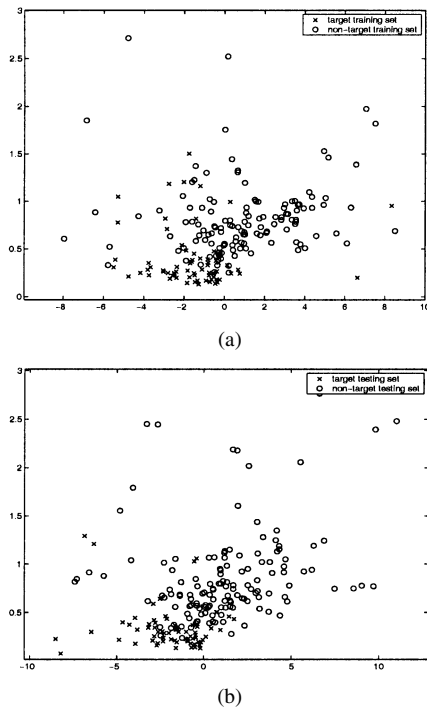


Fig. 8. Scatter plots of the first two features.

distribution for each class. Based upon the poor results of the multivariate Gaussian and the PNN classifiers, namely those of the EM-PNN, it can be argued that the distribution in each class cannot be represented by Gaussian mixtures. The fact that the original PNN provided somewhat better results is an indication that the data is not clustered, rather it is scattered in the feature space. To promote this belief, the scatter plots of the first two features with high discriminatory ability are presented in Fig. 8(a) and (b) for the training and testing data sets, respectively. It must be mentioned that there were several widely scattered points that could not have been included in these plots. Although, these plots present only the distributions of the first two features, they clearly show the scattered nature of this data set.

The evidential K-NN relies on the closeness of the applied pattern to the training samples in order to provide good performance. Although in certain regions of the feature space most of the nearby training samples to an unknown pattern are of the representative class, in the others proximity measure does not necessarily dictate to a stronger evidence of class membership. This implies that the features are mixed together in the feature space, and that the closest neighbors may not always give the best clues pertaining to class membership. As the size of the neighborhood (i.e., K) increased, performance decreased, inferring that the mixing of features is throughout the space. The fact that the two-layer BPNN performs better suggests that the features are fairly separable with a multidimensional convex-type decision regions. The SVM maps the features to a higher dimensional space and then uses an optimal hyperplane in the mapped space. This implies that though the original features carry adequate information for good classification, mapping to a higher dimensional feature space could potentially provide better discriminatory clues that are not present in the original feature space. Thus, each of these classifiers provides hints about the properties of the data in the 22-dimensional space and from these hints

it may be possible to devise a classification paradigm that will give perfect classification results on this data.

IV. CONCLUSION

In this brief paper, four different classification algorithms, namely multivariate Gaussian, evidential K-NN, PNN, and SVM are examined on a wide-band 80-kHz acoustic backscattered data set. The performance of these classifiers was then compared together and with those of the BPNN. The robustness and statistical confidence of these results were studied on a large number of trials. This study indicated that the wideband insonification provides much better robustness to reverberation when compared to the results on the 40-kHz data in [1]. Additionally, the results of these classifiers point to this interesting hypothesis that the data in this 22-dimensional feature space has very complex scattered as well as clustered natures. This, to some extent, explains the great performance of the SVM classifier, which maps the features to a higher dimensional space. Clearly, the choice of the kernel function has an important impact on the overall performance. The performance of the PNN and multivariate Gaussian were not as good as the other classifiers. This may be attributed to several factors including the scattered and mixed nature of the features, invalidity of the Gaussian assumption, and the unbalanced number of training samples for the target and nontarget classes (for the original PNN). Certainly, the behavior of each classifier gave valuable insights to the properties of the feature space.

REFERENCES

- [1] M. R. Azimi-Sadjadi, D. Yao, Q. Huang, and G. J. Dobeck, "Underwater target classification using wavelet packets and neural networks," *IEEE Trans. Neural Networks*, vol. 11, pp. 784–794, May 2000.
- [2] G. A. Carpenter and W. W. Streilein, "ARTMAP-FTR: a neural network for fusion target recognition with application to sonar classification," in *Proc. SPIE Int. Symp. Aerospace/Defense Sensing Control*, vol. 3392, Orlando, FL, Apr. 1998, pp. 342–356.
- [3] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*. New York: Wiley, 2001.
- [4] T. Denoeux, "A k -nearest neighbor classification rule based on dempster-shafer theory," *IEEE Trans. Syst., Man, Cybern.*, vol. 25, pp. 804–813, May 1995.
- [5] D. F. Specht, "Probabilistic neural networks," *Neural Networks*, vol. 3, no. 1, pp. 109–118, 1990.
- [6] D. F. Specht and P. D. Shapiro, "Generalization accuracy of probabilistic neural networks compared with backpropagation networks," in *Proc. IJCNN'91*, July 1991, pp. 458–461.
- [7] S. Gunn, "Support vector machines for classification and regression," Univ. Southampton, Southampton, U.K., ISIS Tech. Rep., May 1998.
- [8] C. Cortes and V. Vapnik, "Support vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, 1995.
- [9] V. Vapnik, *The Nature of Statistical Learning Theory*. New York: Springer-Verlag, 1995.
- [10] M. Vetterli and J. Kovacevic, *Wavelets and Subband Coding*. Upper Saddle River, NJ: Prentice-Hall, Apr. 1995.
- [11] L. Rabiner *et al.*, *Fundamentals of Speech Recognition*. Upper Saddle River, NJ: Prentice-Hall, 1993.
- [12] R. L. Streit and T. E. Luginbuhl, "Maximum likelihood training of probabilistic neural networks," *IEEE Trans. Neural Networks*, vol. 5, pp. 764–783, Sept. 1994.
- [13] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. Royal Stat. Soc.*, ser. B, vol. 39, pp. 1–38, 1977.
- [14] C. J. C. Burges, "A tutorial on support vector machines for pattern recognition," *Data Mining Knowledge Discovery*, vol. 2, no. 2, pp. 121–167, 1998.
- [15] J. Schurmann, "Pattern classification," in *A Unified View of Statistical and Neural Approaches*. New York: Wiley, 1996.