Classification of Underwater Objects Based on Probabilistic Neural Network

Jie Tian, Shanhua Xue, Haining Huang
Institute of Acoustics, Chinese Academy of Science, Beijing, 100190 China

Abstract

Classification of underwater objects remains a challenging and significant problem because of the complexity of underwater environments. In this paper, a probabilistic neural network (PNN) is used as a classifier to the automatic classification of underwater objects. Firstly, a process of multi-field feature extraction is employed to construct a feature vector. The multi-field feature extraction involves time-domain analysis, time-frequency distribution, spectra and bi-spectra analysis. Underwater target classification can be considered as a problem of small sample recognition, because samples acquired under different conditions often exhibit different clustering characteristics. Probabilistic neural network is chosen to discriminate underwater objects because of its simplicity, robustness to noise, and nonlinear decision boundaries. The PNN classifier is contrasted with a Gaussian classifier and a support vector machine (SVM) using lake or sea trial data. Experimental results indicated the PNN classifier is appropriate to this problem.

1. Introduction

Active sonar is a main method to detect and classify underwater objects such as artificial objects and stones. Commonly, underwater objects can be divided to suspended, bottomed or buried ones according to their positions. Suspended targets are relatively easy to detect. However, the detection of bottomed targets is complicated because the disturbing background mainly includes the sea floor reverberations. Moreover, the classification of buried targets is much more difficult, because this problem involves not only sea floor reverberation, but also sound absorption by the covering layer, as described in [1].

In this paper, an algorithm based on multi-field features and PNN classifier is employed to classify the underwater echo signals. Firstly, a feature extraction process is done to the received acoustic signals. That means the 1-D time series are transformed to a few characteristic spaces. The inherent information of signals can be extracted on the basis of these transformations, and then differences between the objects can be protruded in different projection spaces. Secondly, these features are combined and optimized through PCA. Finally, the signals are classified by a neural network classifier.

Generally, the training samples are only a small part of the set of objects to be recognized. The underwater acoustic environments are always so complicated, confusing the acoustic signals in different way, and the mechanism of these confusions is not linear, so the received signals may be time-varying, space-varying and nonlinear. That is to say, the classification of underwater objects should be considered as a small sample, nonlinear recognition problem. PNN is appropriate to the nonlinear problem because of its advantages. The PNN classifier is proved to reach a Bayesian optimal solution only if the training sample set is adequate, no matter how complicated is the classification problem. It is usually much faster to train a PNN than a traditional Back-propagation (BP) network. Although SVM is regarded as an appropriate classifier to the small sample recognition, PNN also has the advantage that it allows adding training samples without a long term retraining when classifying new cases. Accordingly, a PNN is used in this paper to classify real target signals and non-target signals, and compared with a SVM and a Gaussian classifier. Experimental results show the effectivity of the proposed classification approach.

2. Feature extraction

Underwater objects are usually immobile, and they do not radiate energy actively, so active sonars are used to detect them. Fig.1 shows the detection process, in which the target is embedded in the seafloor. Sonar array emits detecting signals, and this signal will be reflected by targets and the seafloor, then this reflected signal can be received by receiving array of the sonar. Characteristics of the object are carried by the echo signals which are commonly 1-D time series, and the

978-0-7695-3736-8/09 $25.00 © 2009 IEEE
DOI 10.1109/ICNC.2009.362

information of the target can be extracted by analyzing the echo signal.

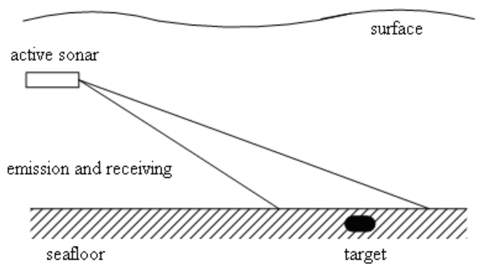

Fig.1 **Active sonar working process**

The shapes of man-made targets are commonly column, or sphere with a relatively small size. The material of its shell is mostly metal such as steel or aluminum, and there usually exists a cavum in it.

The objects can be classified by analyzing the echo signal structure, because the characteristics of targets, such as shapes, materials, structures, are implied in the echo signals. In this paper, a procedure is proposed, in which analyzing the echo signal is the first step of the classification, and features are extracted through the analyzing, then the features are arranged to a vector. Previous researches [2] and [3] advised that some integrated features may be efficient. These features involve four kinds of transformation, Auto-regressive (AR) modeling, power spectrum analysis, Wigner-Ville Distribution (WVD) and bi-spectrum analysis. The discrimination between the real targets and false targets are protruded into different projection spaces through these transformations. The feature vector is so-called multi-field features. As shown in Fig.2, the classification procedure includes preprocess, feature extraction, and classification.

Preprocess mainly means normalization of the time domain signals which include echoes from targets. Purpose of this process is to unify the signal level.

The features are extracted just as follows:

(1) Coefficients of time-domain AR model.

Time domain waveform can be differentiated by the coefficients of their AR models, and the waveform implies the target response. The coefficients of 2-order AR model can be extracted as 2 features.

(2) Power spectrum features

Power spectrum of a certain segment of signal reflects its frequency distribution, so it can be deduced that the spectrum also implies target features. 8 features are formed from the amplitude statistics and shape of the spectrum curve. These features include: mean of the amplitude, deviation of the amplitude, skewness of the amplitude, kurtosis of the amplitude, mean of the curve shape, deviation of the curve shape, skewness of the curve shape and kurtosis of the curve shape.

(3) WVD features

WVD is a kind of time-frequency analysis. WVD of a signal can be regarded as an image, whose two axes represent time and frequency respectively. This image shows the time-frequency distribution of the signal segment, and the features are also described along the two axes. The 16 features extracted from WVD are partially similar to the features in [3], just as follows:

- Standard deviation of the WVD projection on the time axis
- Standard deviation of the WVD projection on the frequency axis
- Standard deviation of the whole WVD
- Number of WVD values that exceed $0.4Max_{wvd}$, where Max_{wvd} means the absolute WVD maximum
- Standard deviation of the time bins of the values exceeding $0.4Max_{wvd}$
- Standard deviation of the frequency bins of the values exceeding $0.4Max_{wvd}$
- Mean of the frequency bins of the values

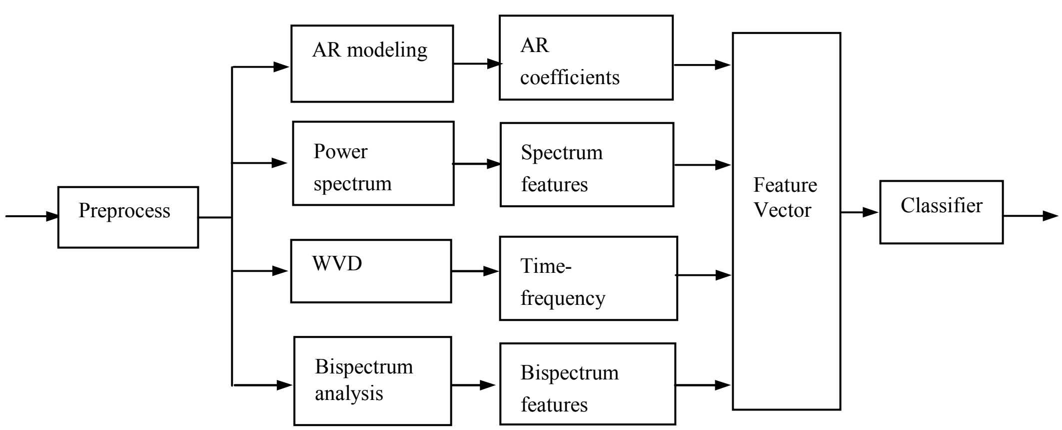

Fig.2 **Structure of underwater objects classification**

exceeding $0.4Max_{wvd}$

- Maximum frequency bin interval of the values exceeding $0.4Max_{wvd}$
- Maximum time bin interval of the values exceeding $0.4Max_{wvd}$
- Standard deviation of the WVD projection at the frequency axis, considering only the pixels whose absolute value exceeds $0.4Max_{wvd}$
- Standard deviation of the WVD projection at the time axis, considering only the pixels whose absolute value exceeds $0.4Max_{wvd}$
- Sum of the distance (on the time-frequency plane) between the positions of the values exceeding $0.4Max_{wvd}$ and the position of the absolute maximum itself
- Mean of the Viterbi line, representing the mean of the strongest frequencies
- Deviation of the Viterbi line, representing the dispersivity of the strongest frequencies
- Maximum value of the Viterbi line
- Length of the Viterbi line

(4) Bi-spectrum features

Underwater signal has been proved mostly nonlinear and non-Gaussian. Considering the mechanism of echo creation, the time domain features and frequency domain features may be both non-Gaussian, while high order statistical analysis is the main tool to non-Gaussian process, so bi-spectrum analysis seems a useful feature extraction method. The bi-spectra transform give birth to a complex matrix, which consists of both amplitude and phase information, but in this paper, only the amplitude information is used. The amplitude distribution of the bi-spectrum can be considered as an image as well as the WVD time-frequency distribution image, so the features are extracted in the similar way. Some of these features are defined similar to those in [3]:

- Standard deviation of the bi-spectrum
- Standard deviation of the bi-spectrum
- Mean of the values inside a given circle centered in the origin
- Absolute maximum of the bi-spectrum
- Numbers of bi-spectrum samples exceeding $0.2Max_{bis}$, where Max_{bis} represents the absolute maximum of the bi-spectrum (this feature can be symboled as N_{em})
- Numbers of samples that are contained inside the given circular region and exceed $0.2Max_{bis}$ (this feature can be symboled as N'_{em})
- Ratio of N'_{em} to N_{em}
- Standard deviation of the bi-spectrum along the diagonal
- Index of the bi-spectrum maximum along the diagonal
- Mean of the indexes of the samples along the diagonal that exceed $0.2Max_{bis}$
- Standard deviation of the indexes of the samples along the diagonal that exceed $0.2Max_{bis}$
- Standard deviation of the indexes of the samples along the diagonal that exceed $0.9Max_{bis}$

The features mentioned above can be combined to a vector, which contains 37 elements. That is to say, the input layer of the classifier should have 37 elements. But some of these features may contribute less to the classification than the others, so they are less effective, and it seems reducing the quantity of the features may leads to fast training of the network. So the feature vector is minified through Principal Component Analysis (PCA), known as a feature optimization tool. PCA is a kind of true eigenvector-based multivariate analyses. Often, its operation can be thought of as revealing the internal structure of the data in a way which best explains the variance in the data. If a signal is described in a high-dimensional data space, PCA supplies the user with a lower-dimensional description. PCA can be used for dimensionality reduction in a data set by retaining those characteristics of the data set that contribute most to its variance, by keeping lower-order principal components and ignoring higher-order ones. Such low-order components are supposed to contain the "most important" aspects of the data. The details of PCA tool can be found in [4]. Therefore, through PCA, the 37-D vector is reduced to 20 elements which preserve over 95% information of the 37 features.

3. Classifier

Bayes classifier has been regarded as one of the optimal classifier because that it reaches lowest error rate. Although it can be affirmed that absolute Bayes classifier is unattainable in practice, the Bayes rule provides the logical basis for all statistical algorithms. Bayes classifiers are unattainable because they assume complete information is known about the statistical distributions in each class, so the training set is supposed to be adequate and representative. It has been proven in [5] and [6] that PNN yield an approximate Bayes optimal decision surface under this supposition, and this kind of classifier also exhibits simplicity in training. Although there are several tools such as the BP network, support vector machines [7], and other similar methods for pattern classification could have been used, the PNN classifier was often chosen because of its simplicity, robustness to noise, and nonlinear decision boundaries [8], and the practical benefits that it allows adding or reducing training samples without a long term to retrain.

As shown in Fig.3, a PNN networks has four layers:

(1) Input layer

There is one neuron in the input layer corresponding to each predictor variable. The input neurons feed the feature vector values to each of the neurons in the hidden layer.

(2) Hidden layer

This layer is also called pattern layer, because it is composed of some typical sample sets. The hidden layer has one neuron for each pattern in the training data set. When presented with the **x** vector of input values from the input layer, a hidden neuron computes the Euclidean distance of the test case from the neuron's center point and then applies the radial basis function (RBF) kernel. The resulting sigma value is passed to the neurons in the next layer.

(3) Summation layer

The summation layer computes the probability of the given input **x** belonging to class i represented by the patterns in the pattern layer. There is one pattern neuron for each category of the target variable. The actual target category of each training case is stored with each hidden neuron. The weighted value coming out of a hidden neuron is fed only to the summation neuron that corresponds to the hidden neuron's category. The summation neurons simply summarize the values for the class they represent, so it gives a weighted vote for that category. Suppose that there are M categories in the training set $\{\omega_i, i = 1,2,\cdots,M\}$, and the typical sample set of each category has N_1, N_2, $\cdots$, N_M samples. Then for input vector **x**, the output of the k-th node in summation layer can be denoted as:

$$y_k(\mathbf{x}) = \frac{1}{N_k}\sum_{j=1}^{N_k} K\left(\left\|\mathbf{x} - \mathbf{c}_j^k\right\|, \sigma\right), \quad (1)$$

where $\mathbf{c}_j^k$ is the j-th typical sample of the j-th category, $\|\cdot\|$ means Euclidean distance. $K(\cdot)$ is a kernel function, and a Gaussian function is generally chosen.

If the input sample is a normalized vector, which means $\|\mathbf{x}\| = \|\mathbf{c}\| = 1$, equation (1) can be re-writen as:

$$\begin{aligned} y_k(\mathbf{x}) &= \frac{1}{(2\pi)^{\frac{N}{2}}\sigma^N} \cdot \frac{1}{N_k}\sum_{j=1}^{N_k}\exp\left(-\frac{\left\|\mathbf{x} - \mathbf{c}_j^k\right\|^2}{2\sigma^2}\right) \\ &= \frac{1}{(2\pi)^{\frac{N}{2}}\sigma^N} \cdot \frac{1}{N_k}\sum_{j=1}^{N_k}\exp\left(\frac{\mathbf{x}^T\mathbf{c}_j^k - 1}{\sigma^2}\right) \end{aligned} \quad (2)$$

(4) Decision layer

The decision layer picks the class for which the highest probability was obtained in the summation layer. That means a comparison of the weighted votes for each target category accumulated in the summation layer and uses the largest vote to predict.

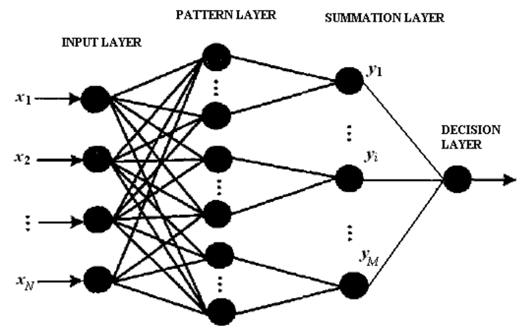

Fig.3 **Architecture of PNN**

Assuming that the prior probability and the misclassifying loss are equal for all classes, the Bayes rules use posterior probability $P(\omega_i \mid \mathbf{x})$ to make decision, which can be written as:

$$P(\omega_i \mid \mathbf{x}) = \frac{P(\mathbf{x} \mid \omega_i)P(\omega_i)}{P(\mathbf{x})}. \quad (3)$$

Then a universal expression of the Bayes optimal decision can be described as:

$$\mathbf{x} \in \omega_i,\ if\ \omega_i = \arg\max_{j=1}^{M}\left[P(\omega_j \mid \mathbf{x})\right] \quad i = 1,2,\cdots,M \quad (4)$$

Because $P(\mathbf{x})$ is a communal, (4) can be transformed to:

$$\mathbf{x} \in \omega_i,\ if\ \omega_i = \arg\max_{j=1}^{M}\left[P(\mathbf{x} \mid \omega_j)P(\omega_j)\right]$$
$$i = 1,2,\cdots,M \quad (5)$$

where $P(\mathbf{x} \mid \omega_j)$ is the conditional probability density of **x** given class i, and $P(\omega_j)$ is the occurrence probability of samples from class i.

The conditional probability $P(\mathbf{x} \mid \omega_j)$ is estimated using the Parzen window method [9], so the estimation is represented as:

$$\widehat{P}(\mathbf{x} \mid \omega_i) = \frac{1}{(2\pi)^{\frac{N}{2}}\sigma^N} \cdot \frac{1}{N_i}\sum_{j=1}^{N_i}\exp\left(-\frac{\left\|\mathbf{x} - \mathbf{c}_j^i\right\|^2}{2\sigma^2}\right) \quad (6)$$

where $\mathbf{c}_j^i$ is the member of class ω_i, N is the dimension of input vectors, and N_i the amount of class ω_i. It has been proved that, if the training samples are enough, (6) will approach the real $P(\mathbf{x} \mid \omega_j)$, and $P(\omega_i)$ can be estimated by:

$$\widehat{P}(\omega_i) = N_i \Big/ \sum_{i=1}^{M} N_i \quad (7)$$

(6) and (2) shows that PNN approximately realizes the Bayes decision rules.

4. Experiments

To evaluate the performance of the PNN classifier, two classifiers are used as comparison, a SVM with a 4-degree polynomial kernel and a Gaussian classifier which is similar to the one used in [10].

Some of the data used in the experiments are partially chosen from a lake trial made in 1999 and the others from a sea trial in 2000. The experimental samples were segments intercepted from the 1-D echo signal, consisting of 1024 points each.

The samples were divided into two classes, class A is man-made gas tank and class B stones with equivalent size.

The training set consists of 31 A class files and 34 B class files (one file is just one sample). 14 of the A class files are echoes from the lake trial, and 17 are from the sea trial. While half of B class files are echoes from the lake trial, and the others are from the sea trial. Hence, the input layer of PNN has 20 neurons, the pattern layer has 65 neurons, and the summation layer 2 neurons.

The testing set consists of 60 A class files and 60 B class files, and they are selected equally from the lake trial and the sea trial.

The classification experiments were based on the sets above mentioned, and the results are showed in Table.1, in which PNN shows best performance, and SVM gives similar results. Although Gaussian classifier performs well with the training set, its classifying results are almost unacceptable with the testing set.

Table.1 **Experimental results of classification**

classifiers	Training set		Testing set	
	Class A	Class B	Class A	Class B
Gaussian	100%	95%	100%	43.3%
SVM	90%	100%	88.3%	70%
PNN	90%	100%	88.3%	71.7%

These experiments may be not so sufficient to compare the classifiers, but it shows the proposed approach is effective.

5. Conclusion

An automatic classification approach is introduced in this paper to solve the underwater object classification problem. This approach is on the basis of multi-field feature extraction and employs a PNN classifier. The multi-field features include coefficients of the time-domain AR model, power spectrum features, time-frequency distribution features and bi-spectrum features, and they are combined to a 37 dimension feature vector. Then the dimension of this feature vector is reduced to 20 by use of PCA. The reduced feature vector is fed to a PNN classifier, which was chosen because of its simplicity, robustness to noise, nonlinear decision boundaries and conveniency of retraining. The PNN classifier seems appropriate to this kind of problems in the experiments. Data from a lake trail and a sea trail were used in the classification experiments, and the experimental results proved that the approach can effectively distinguish the two classes. As a comparison, a Gaussian classifier and a SVM classifier were also used in the experiments, and the results shows that the PNN is more appropriate than the Gaussian classifier, while SVM outputs almost similar results as PNN.

6. References

[1] Li Xiukun, Yang Shie, "Extraction of Features of Underwater Target", *Journal of Harbin Engineering University*, vol. 22, no.1, 2001, pp.26-29.

[2] Tian Jie, Xue Shanhua, Huang Haining, and Zhang Chuanhua, "Classification of underwater still objects based on multifield features and SVM", *Journal of Marine Science and Application*, vol. 6, no.1, 2007, pp.36-40.

[3] Andrea Trucco, "Detection of Objects Buried in the Seafloor by a Pattern-Recognition Approach", *IEEE Journal of Oceanic Engineering*, vol. 26, no.4, 2001, pp.769-782.

[4] Jolliffe I.T., *Principal Component Analysis*, Springer, NY, 2002.

[5] D. F. Specht, "Probabilistic neural networks for classification, mapping, or associative memory", *IEEE International Conference on Neural Networks*, vol.1, 24-27 July 1988, pp.525 – 532.

[6] D. F. Specht, "Probabilistic Neural Networks and the Polynomial Adaline as Complementary Techniques for Classification", *IEEE Trans. on Neural Networks*, vol. 1, no.1, 1990, pp.111-121.

[7] Vladimir N. Vapnik, *Statistical Learning Theory*, John Wiley & Sons Inc., 1998

[8] Edmond M. DuPont, Rodney G. Roberts, and Carl A. Moore, "Speed Independent Terrain Classifcation", *Proceedings of the 38th Southeastern Symposium on System Theory*, Tennessee Technological University, Cookeville, TN, USA, March 5-7, 2006, pp.241-244.

[9] E. Parzen, "On Estimation of a Probability Density Function and Mode," *Annals of Mathematical Statistics*, vol. 33, 1962, pp. 1065-1076.

[10] Frances B. Shin, David H. Kil, and Richard F. Wayland, "Active Impulsive Echo Discrimination in Shallow Water by Mapping Target Physics-Derived Features to Classifiers", *IEEE Journal of Oceanic Engineering*, vol. 22, no.1, 2001, pp.66-79.