

# A Boosting Approach for Object Classification in Biosonar Based Robot Navigation

Majid M. Beigi and Andreas Zell

**Abstract**—This paper addresses the problem of object classification in a biosonar based mobile robot in a natural environment using a boosting method. We present an algorithm based on gradient boosting for biosonar-based robots that recognize different objects such as different trees via reflected sonar echoes. Gradient boosting is a machine learning approach, that builds one strong classifier from many base learners. We present two kinds of base learners for the gradient boosting: Ordinary Least Squares (OLS) and kernel-based base learners. Compared with our previous works, in which we presented a time resolved spectrum kernel to extract the similarities between echoes, we get more efficient and accurate results with the newly proposed boosting method. We compare the methods in terms of sensitivity, specificity, accuracy and Matthew’s correlation coefficient and also the runtime of training and testing.

## I. INTRODUCTION

Bats can distinguish objects and their prey by emitting a series of ultrasound signals (chirps) that generally sweep covering frequencies from 22 to 100 kHz. They can separately perceive the delays of two concurrent echoes as little as 2 ms apart and resolve reflecting points as close together as 0.3 mm in range [10]. The acoustic image of a sonar target is apparently derived from time-domain or periodicity information processing by the nervous system. Inspired by the bat biosonar system, researchers have utilized ultrasonic sensing techniques for mobile robots (biomimetic robots) and tried to classify different textures and landmarks using received echo signals. Biosonar sensing involves the production of chirps, the reception of echoes from targets, signal analysis and target matching. By comparing the returning echoes (which are individually the superposition results of the reflected echoes) we aim at recognizing the objects. Gao et. al [11] presented a deformable template matching algorithm for classification of several types of brick walls, picket fences and hedges using sonar echoes. M. Wang et al. [13], [12] used different structural features in the frequency domain and also cross correlation as template matching algorithm for that task. In our previous works [18], [19] we suggested a kernel named *Time-resolved spectrum kernel* for matching the subsequences of time series (sonar echoes) and extracting the local similarities of echoes. The results outperformed other matching techniques [13], [12]. The time-resolved spectrum kernel simply measures the whole similarities of all subsequences of the time series in consideration. The more

two time series share similar subsequences, the more similar they are. A linear combination of kernels with different subsequence size ( $p$ -spectrum kernels) was a measure of similarity between two time series. Despite the accurate rate of classification, the training and testing speed were slow and the method is not applicable for real applications.

In this paper, we implement a simple yet powerful method for the problem at hand using boosting. Originally, boosting has been proposed in the 90’s (Freund and Schapire, 1996 [1]) as a method for classification and regression in which a fitting method or estimator, called the base learner, is fitted multiple times on re-weighted data and the final boosting estimator is then constructed via a linear combination of those base learners. In different works, it has been shown that boosting method outperformed other machine learning methods for high-dimensional data. It is empirically illustrated in Bühlmann and Yu [3] that boosting has mainly an advantage for data with high-dimensional predictors. Hoffmann et al. [2] used gradient boosting to classify high dimensional EEG signals in brain-computer interfaces and Jiao et al. [5] used this method for high dimensional protein classification and obtained satisfying results.

Similar to the above researches, we are also facing high-dimensional data in classification of sonar signals reflected by different kinds of trees (Fig. 3). In this paper, we study the efficiency of boosting methods for our classification task. We use the gradient boosting method with two kinds of base learners. The first one uses Ordinary Least Squares (OLS) regression and the other one uses the kernel function as base learner.

Compared with our previous works [18], [19], in which we presented a time resolved spectrum kernel to extract the similarities between echoes, we got more efficient and accurate results with the newly proposed boosting method.

The content of the rest of the paper is as follows: In Sec. II we describe the hardware and the experimental setup used for this work and the preprocessing of the data. In Sec. III the gradient boosting is described in detail. In Sec. IV we present two base learners for gradient boosting. The experimental results are presented in Sec. V, and Sec. VI draws the conclusion of this work.

## II. BIOSONAR BASED ROBOT

### A. Hardware

The implementation of the whole system consists of a mobile robot (Robin) with two PCs, a digital signal processing package, and a biosonar system (Fig. 1). The biosonar system includes a National Instruments NI6110 analog I/O

M. M. Beigi is Ph.D. student at the Department of Computer Science, University of Tuebingen, D72076, Germany. majid.beigi@uni-tuebingen.de

A. Zell is Professor at the Department of Computer Science, University of Tuebingen, D72076, Germany. Andreas.Zell@uni-tuebingen.de

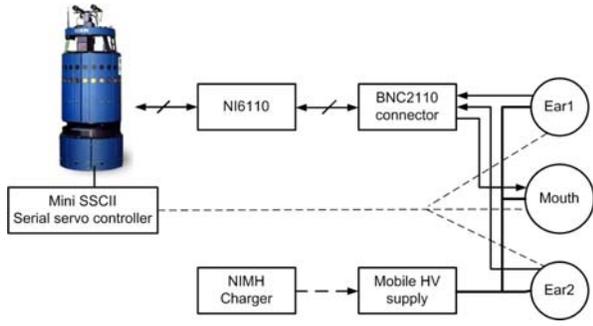


Fig. 1. Biosonar system configuration

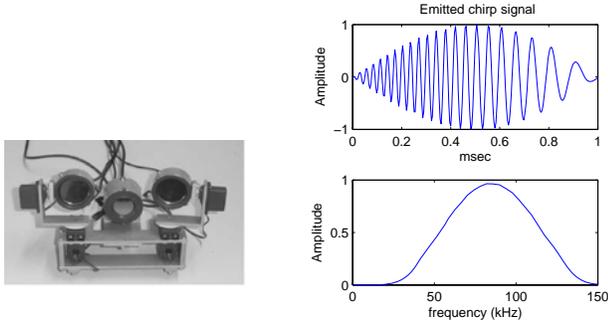


Fig. 2. Biosonar head (left). Emitted chirp signal and its frequency content (right).

card, a mini servo controller (module SSCII), a BNC2110 connector, and the biosonar head. The NI6110 card and the BNC2110 connector transfer chirp signals and receive the reflected echoes. The biosonar head (Fig. 2) consists of 3 Polaroid sensors in a triangular layout, similar to the layout of a bat’s mouth and ears: two Polaroid 600 sensors spaced 12.5 cm apart as *ears*, a Polaroid 7000 sensor as *mouth* in the middle between two ears. Each of the two ears has two degrees of angular freedom provided by two servo motors. These can be finely rotated to acquire local support. The Polaroid ultrasonic ranging system is most commonly used by the robotics research community. The emitted pulse was a linearly frequency modulated chirp sweeping from 20kHz to 120kHz in 1 ms (Fig.2). The maximum sampling speed of the NI6110 card is 5 MHz. We utilized 1 MHz in our research. The NiMH charger box provides the sensors with a 150V power supply. The mobile robot Robin is an autonomous mobile service robot that has two PCs inside, one is in charge of navigation control, the other one is responsible for signal data processing, feature extraction and decision making.

### B. Landmarks and Sensing Strategy

Through echolocation in darkness, a bat can perceive not only the position of an object, but also its 3D structure [6]. The recognizable target in nature works as a landmark for its navigation. For our sensory task – biosonar based mobile robot navigation in natural environments – these landmarks should be rich and easy to be found there. The criteria for selecting natural landmarks include observability, frequent



Fig. 3. Three different trees as biosonar landmarks. From left to right: Ficus, Bamboo, Schefflera.

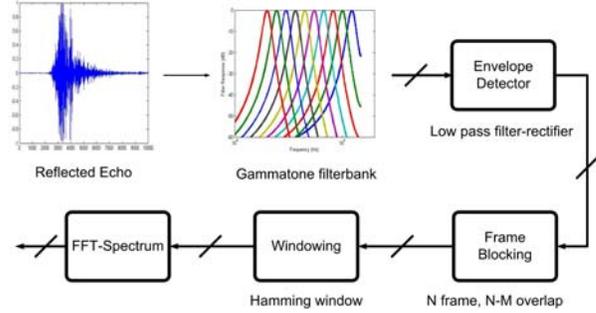


Fig. 4. Block diagram of the preprocessing steps for reflected echoes.

occurrence, uniqueness, temporal stability, fast and robust classification, and lateral compactness [7]. Considering those aspects, we selected three artificial trees with similar height of 1.7 m as shown in Fig. 3.

Compared with other researchers [8], [9], we used a different method for sensing the objects. We used a 0.5 degree angular stepsize for our scans, each tree was scanned 360 degrees in a circular movement of the robot and we collected echoes from all orientations of leaves and tree. The reflected echo contains the information about the geometry of the tree and is the superposition of all reflections.

### C. Data Processing

Fig. 4 shows the block diagram of the data acquisition and preprocessing procedure of reflected echoes. We passed the reflected echoes through a bank of 10 gammatone filters between 20 kHz and 120 kHz. In order to extract the envelope of the filtered signals, they were delivered to half-wave rectifiers.

The next step is *frame blocking*. In this step the signal blocked to frames of  $N$  samples, is separated from adjacent frames by  $M$  ( $M < N$ ) samples and has  $N - M$  overlaps. Considering the sampling frequency of the data acquisition part (1 MHz) and the minimum width of leaves of trees and axial resolution of transducers, we selected  $N = 32$  and 50% overlap for frames. The next step in the data preprocessing is to window each individual frame so as to minimize the signal discontinuities at the beginning and end of each frame. We used a *Hamming window* for this purpose. The last step is to calculate the average energy of each band of gammatone filter bank in each frame. The result is a feature matrix, where each column is a vector showing the average energy of each channel in one time frame. Fig. 5 shows the examples

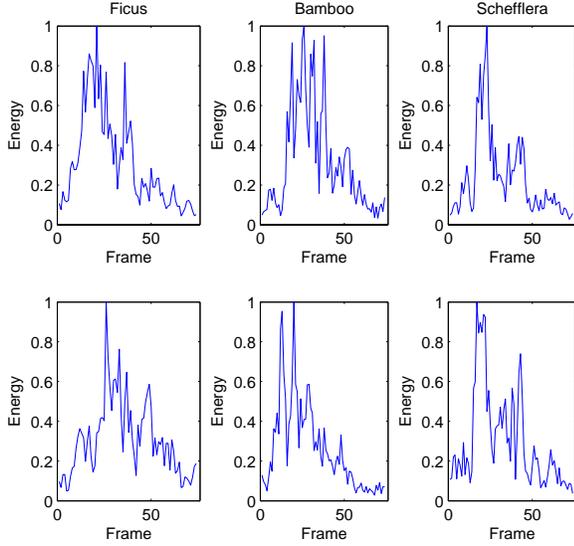


Fig. 5. Features: examples of the energy spectrum (output of gammatone filter centered around 50 kHz) for Ficus, Bamboo and Schefflera trees.

of the preprocessed reflected echoes from Ficus, Bamboo and Schefflera trees. We use this feature matrix for our classification task.

After the preprocessing steps for each echo (Fig. 4), we have a Matrix  $K$  of time series in which each cell is a time frame and its value is the average energy of each channel of gammatone filter. Furthermore we have

$$K = C \times S \quad (1)$$

where  $K$  is the number of features, with  $C$  the number of channels and  $S$  the number of samples in each channel. Now, the problem is very similar to a multichannel EEG classification. We use the gradient boosting algorithm for the sonar classification task as Hofmann et al. [2] did for the event classification from EEG signals.

### III. GRADIENT BOOSTING ALGORITHM

We here give a summary of the gradient boosting algorithm from [4]. Given a set of random input variables  $x = \{x_1, \dots, x_n\}$  and a random output variable  $y$  and some samples  $\{y_i, x_i\}_{i=1}^N$ , we want to find an approximation function  $F^*$  that can predict  $y$  from  $x$  such that over the joint distribution of  $y, x$  values, the expected value of a specific loss function  $L(y, F(x))$  is minimized:

$$\begin{aligned} F^*(x) &= \arg \min_{F(x)} E_{y,x} L(y, F(x)) \\ &= \arg \min_{F(x)} E_x [E_y L(y, F(x)) | x] \end{aligned} \quad (2)$$

Examples of different loss functions include squared error  $(y - F)^2$  and absolute error  $|y - F|$  for regression, and negative binomial log-likelihood,  $\log(1 + e^{-2yF})$ , when  $y \in \{-1, 1\}$  for classification.

$F(x)$  is a member of parameterized class of functions  $F(x; \mathbf{p})$ , where  $\mathbf{p} = \{p_1, p_2, \dots\}$  is a finite set of parameters whose joint values identify individual class members. In the gradient boosting method, we have the additive expansions of the form

$$F(x; \{\beta_m, a_m\}_{m=1}^M) = \sum_{m=1}^M \beta_m h(x; a_m) \quad (3)$$

in which  $\mathbf{p} = \{\beta_m, a_m\}$ . The generic function  $h(x; a)$  is called a base or weak learner and is a simple function of  $x$  with parameters  $a = \{a_1, a_2, \dots, a_M\}$ . The task is to find the parameters of weak learners through solving Eq. 2. For that, a typical parameter optimization method “greedy-stagewise” is used in which we optimize  $\{\beta_m, a_m\}$  after all of the  $\{\beta_i, a_i\} (i = 1, \dots, m-1)$  are optimized. Then, the recursive representation of the optimization method is as follows:

$$\{\beta_m, a_m\} = \arg \min_{\beta, a} \sum_{i=1}^N l(y_i, F_{m-1}(x_i) + \beta h(x_i; a)) \quad (4)$$

where the joint distribution of  $(x, y)$  is estimated by a finite data sample  $\{y_i, x_i\}_1^N$ , and we have

$$F_m = F_{m-1} + \beta_m h(x; a_m) \quad (5)$$

$\beta_m h(x; a_m)$  is an incremental *boost* and the best greedy direction step towards the data-based estimate of  $F^*(x)$ . Friedman [4] suggested a steepest-descent method to find that direction:

$$-g_m(x_i) = - \left[ \frac{\partial L(y_i, F(x_i))}{\partial F(x_i)} \right]_{F(x)=F_{m-1}(x)} \quad (6)$$

It gives the best steepest-descent step direction at  $F_{m-1}$ . We find the parameters  $a_m$  that produces  $h_m = \{h(x_i; a_m)\}_{i=1}^N$  most parallel to  $-g_m \in \mathbb{R}^N$ . So we have:

$$a_m = \arg \min_{a, \rho} \sum_{i=1}^N [-g_m(x_i) - \rho h(x_i; a)]^2 \quad (7)$$

and then Eq. 4 is converted to:

$$\beta_m = \arg \min_{\beta} \sum_{i=1}^N L(y_i, F_{m-1}(x_i) + \beta h(x_i; a_m)) \quad (8)$$

We consider a regularization term to avoid the resulting overfit problem by a large number of weak learners. This can be done by adding a *shrinkage factor*  $0 < \nu \leq 1$  to the Eq. 5:

$$F_m = F_{m-1} + \nu \beta_m h(x; a_m) \quad (9)$$

This can greatly improve the generalization performance of the algorithm. The general framework of the gradient boosting is as follows:

---

**Algorithm A** A general gradient boosting framework

---

```
1  $F_0(x_i) = 0, \forall i$  ;
2 for  $m \leftarrow 1$  to  $M$  do
3    $g_m(x_i) = \left[ \frac{\partial L(y_i, F(x_i))}{\partial F(x_i)} \right]_{F(x)=F_{m-1}(x)}$  ;
4    $a_m = \arg \min_{a, \rho} \sum_{i=1}^N [-g_m(x_i) - \rho h(x_i; a)]^2$  ;
5    $\beta_m = \arg \min_{\beta} \sum_{i=1}^N L(y_i, F_{m-1}(x_i) + \beta h(x_i; a_m))$ ;
6    $F_m = F_{m-1} + \nu \beta_m h(x; a_m)$ ;
7 end
```

---

In our classification task, we convert  $F(x_i)$  into a randomized predictor by using the soft-max function :

$$p(y_i = 1|x_i) = \frac{e^{F(x_i)}}{e^{F(x_i)} + e^{-F(x_i)}}$$

and use the Bernoulli log-likelihood for the loss function:

$$\begin{aligned} L(y, F) &= \log\left(\prod_{i=1}^N p(y_i = 1|x_i)^{y_i} p(y_i = 0|x_i)^{1-y_i}\right) \\ &= \sum_{i=1}^N \left[ 2y_i F(x_i) - \log(1 + e^{2F(x_i)}) \right] \\ &= \sum_{i=1}^N L(y_i, F(x_i)) \end{aligned}$$

which results in:

$$L(y_i, F(x_i)) = 2y_i F(x_i) - \log(1 + e^{2F(x_i)}) \quad (10)$$

and  $g_m$  in Eq. 6 is obtained with:

$$g_m(x_i) = 2(y_i - p_m(y_i = 1|x_i)) \quad (11)$$

and Eq. 8 is converted to:

$$\begin{aligned} \beta_m &= \arg \min_{\beta} \left\{ \sum_{i=1}^N 2y_i (F_{m-1}(x_i) + \beta h(x_i; a_m)) \right. \\ &\quad \left. - \sum_{i=1}^N \log\left(1 + e^{2(F_{m-1}(x_i) + \beta h(x_i; a_m))}\right) \right\} \end{aligned}$$

The pseudocode for the gradient boosting algorithm is given in the following.

After initialization, we calculate  $h$  and  $\beta$  to update the new  $F$ . This procedure is continued until a certain number of iterations  $M$  is reached. To prevent the overfitting or underfitting problems, we select optimum values of  $M$  and  $\nu$  in a cross-validation test.

#### IV. SELECTION OF THE BASE LEARNER

The function  $h$  can have any form that can be optimized over the parameter  $a$  to fit the training data. In this paper we consider two kinds of base learner as follows:

---

**Algorithm** Gradient boosting with Bernoulli log-likelihood loss function

---

```
1  $p_0(y_i = 1|x_i) = 0.5, F_0(x_i) = 0, \forall i$  ;
2 for  $m \leftarrow 1$  to  $M$  do
3    $g_m(x_i) = 2(y_i - p_{m-1}(y_i = 1|x_i)), \forall i$  ;
4    $h = \arg \min_{\hat{h}, a} \sum_{i=1}^N (-g_m(x_i) - \hat{h}(x_i; a_m))^2$  ;
5    $\beta_m = \arg \min \left\{ \sum_{i=1}^N 2y_i (F_{m-1}(x_i) + \beta h(x_i; a_m)) \right.$ 
6      $\left. - \sum_{i=1}^N \log\left(1 + e^{2(F_{m-1}(x_i) + \beta h(x_i; a_m))}\right) \right\}$ 
7    $F_m = F_{m-1} + \nu \beta_m h(x; a_m)$ ;
8    $p_m(y_i = 1|x_i) = \frac{e^{F_m(x_i)}}{e^{F_m(x_i)} + e^{-F_m(x_i)}}, \forall i$ ;
9 end
```

---

#### A. Ordinary Least Squares (OLS) Base Learner

The simplest function to use here for  $h$  is the OLS regressor as:

$$h(x) = \alpha_1 x + \alpha_2 = aX$$

where

$$a = \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix}, X = \begin{bmatrix} x \\ 1 \end{bmatrix}$$

By solving the Ordinary Least Squares (OLS) regression, we find the parameter  $a = [\alpha_1; \alpha_2]$  in Eq. 7:

$$a = (XX^T)^{-1}X^Tg$$

#### B. Kernel-based Base Learner

Kernel based regression methods are considered as the problem of finding the function  $f$  that minimizes the objective function

$$\min_{f \in \mathcal{H}} \frac{1}{m} \sum_{i=1}^m L(f(x_i), y_i) + \mu \|f\|^2 \quad (12)$$

where  $\mathcal{H}$  is the Reproducing Kernel Hilbert Space (RKHS) generated by the kernel  $k(\cdot, \cdot)$  and  $\mu$  is a parameter that trades off the quality of the regression function and the regularization term.

Recently different research has been done to use the idea of kernel in the boosting procedure [15], [16], [17] based on the representer theorem. According to the representer theorem (Kimeldorf et al.[14]) the optimal  $f(x)$  has the form:

$$f(x) = \sum_{i=1}^N \alpha_i k(x_i, x) \quad (13)$$

Similar to Eq. 5 in which  $F(x)$  is a summation of base learners,  $f(x)$  is also a summation of kernel functions. Compared with the other works [15], [16], we use the

TABLE I  
THE PERFORMANCE OF CLASSIFICATION IN SONAR SIGNALS USING GRADIENT BOOSTING METHOD

Method	TREE	Spec. (%)	Sen. (%)	Acc. (%)	Mcc.
OLS regression	Ficus	93.0	87.5	86.33	0.77
OLS regression	Bamboo	97.1	95.0	<b>96.1</b>	0.91
OLS regression	Schefflera	84.0	94.0	91.0	0.8
Kernel regression ( $\sigma = 0.1$ )	Ficus	88.1	93.0	91.1	<b>0.81</b>
Kernel regression ( $\sigma = 0.1$ )	Bamboo	99.1	80.0	86.6	0.80
Kernel regression ( $\sigma = 0.1$ )	Schefflera	75.0	95.0	88.6	0.75
Kernel regression ( $\sigma = 10$ )	Ficus	92.1	96.0	<b>95.1</b>	0.79
Kernel regression ( $\sigma = 10$ )	Bamboo	99.1	93.0	95.3	<b>0.93</b>
Kernel regression ( $\sigma = 10$ )	Schefflera	84.0	96.0	<b>92.9</b>	<b>0.82</b>

following simple base learner to bring the concept of the kernel function in the boosting procedure. We consider:

$$h(x, \gamma) = \alpha_1 k(x, x_\gamma) + \alpha_2 = \mathbf{a}K$$

where

$$\mathbf{a} = \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix}, K = \begin{bmatrix} k \\ 1 \end{bmatrix}$$

and again we use OLS regression to find the parameters  $\mathbf{a}$  and  $\gamma$ .

$$(h, \gamma) = \arg \min_{\hat{h}, \mathbf{a}, \gamma} \sum_{i=1}^N \left( -g(x_i) - \hat{h}(x, \gamma, \mathbf{a}) \right)$$

Optimizing the parameter  $\gamma$  means that each kernel function is selected once at most. This guarantees that the effect of some kernel functions is not excessively magnified and so prevents over-fitting in the boosting procedure.

## V. EXPERIMENT AND RESULTS

We gathered the sonar data, 720 echoes, each one 10000 data points, for each tree shown in Fig. 3. After preprocessing, those echoes were converted to 720 matrices of features, where in Eq. 1,  $K = 1980$  (number of features),  $C = 20$  (number of channels) and  $S = 99$  (number of data points or frames in each channel). So, in our boosting procedure,  $N$ , the number of examples is 720 and the number of features is 1980. But in training and in each boosting step, instead of using all features at once, we use only the features in each channel ( $S = 99$ ) and find the regressor  $h$  for that and repeat the boost step for all channels. Then, in testing we use the corresponding regressor (base learner) of all channels to estimate the final  $F(x)$ . In all experiments the maximum number of iterations of the boosting algorithm,  $M$ , and the shrinkage factor,  $\nu$ , were set to 100 and 1 respectively. For the kernel regression we chose the Radial Basis Function (RBF) kernel with different values of  $\sigma$ . After using 5-fold cross validation, the prediction quality was then evaluated by specificity (Spec.), sensitivity (Sen.), accuracy (Acc.) and Matthew's correlation coefficient (Mcc.) as follows:

$$Sen. = \frac{TP}{(TP + FN)}$$

$$Spec. = \frac{TN}{(TN + FP)}$$

$$Acc. = \frac{TP + TN}{(TN + FN + TP + FP)}$$

$$Mcc. = \frac{TP \times TN - FN \times FP}{\sqrt{(TN + FN)(TP + FN)(TN + FP)(TP + FP)}}$$

where  $TP$  = true positive,  $TN$  = true negative,  $FP$  = false positive and  $FN$  = false negative. Table I shows the results of the classifier.

The classification results are shown in table I. As we see from that table, the gradient boosting with kernel-based base learner shows a slight improvement in accuracy and Mcc. compared with the one that uses OLS base learner, and also the value of  $\sigma$  in RBF kernels affects the performance of the classifier.

In our previous works [18], [19], we presented a Time-resolved spectrum kernel to extract the similarities between the echoes. In those papers, it was shown an improvement compared with the previous work of our group (Wang et al. [12]), in which the classification was made through template matching in 2D sonar acoustic image using a 2D Discrete Cosine Transform.

Table II shows the results of the Time-resolved spectrum kernel [19]. Comparing the Tables I and II one sees that the boosting method could improve the performance of classification.

The other point is the running time of the boosting method. Table III shows the running time of the boosting and spectrum kernel methods on a PC with an Intel Core Duo processor (1.83GHz and 1GB RAM) while coded in Matlab 7.0 and in a 5-fold cross validation test. From this table, we see that the boosting approach is more efficient than the spectrum kernel for the classification of sonar data and needs much less time for training.

## VI. CONCLUSION

In this paper, we proposed a new machine learning approach for object classification in biosonar based robots. We used a regression approach in the gradient boosting which proved to be both more accurate and efficient than other

TABLE II  
THE PERFORMANCE OF CLASSIFICATION IN SONAR SIGNALS  
USING TIME-RESOLVED SPECTRUM KERNEL [19].

TREE	Spec. (%)	Sen. (%)	Acc. (%)	Mcc.
Ficus	89.0	87.5	88.0	0.72
Bamboo	91.3	95.0	92.1	0.91
Schefflera	80.0	90.0	93.0	0.80

TABLE III  
RUNNING TIME OF THE PROPOSED BOOSTING METHOD AND  
TIME-RESOLVED SPECTRUM KERNEL IN A 5-FOLD CROSS VALIDATION  
TEST.

	Training time	Testing time
Spectrum kernel	6 hours	2 mins
OLS regression Boosting	6 mins	10 seconds
Kernel regression Boosting	10 mins	20 seconds

previously proposed methods such as spectrum kernels and template matching using acoustic images. We suggested a simple base learner in the boosting method using the kernel matrix and showed that it outperformed the simple OLS regression. The main point of the signal preprocessing was using a filter bank like that of the hearing system of bats. With this filter bank, the one-dimensional sonar echoes were converted into shorter length but more informative multi-dimensional signals in which each dimension has the frequency content of a similar filter in the hearing system. After this conversion, the features were more distinguishable and the boosting method was able to classify them efficiently and we could get satisfying results. Without this filter bank, the boosting method could not classify the raw echoes accurately.

## VII. ACKNOWLEDGMENT

We would like to thank Maosen Wang for the preparation of the biosonar based robot. The biosonar head we used was built according to a design of Dr. Rolf Müller in the neurobiology lab of Prof. Dr. H.-U. Schnitzler, University of Tübingen, in a former joint research project.

## REFERENCES

- [1] Y. Freund and R. E. Schapire, "Experiments with a New Boosting Algorithm", *International Conference on Machine Learning*, pp. 148-156, 1996.
- [2] U. Hoffmann, G. Garcia, J.-M. Vesin, K. Diserens and T. Ebrahimi. "Boosting Approach to P300 Detection with Application to Brain-Computer Interfaces", in *Proceedings of the 2nd International IEEE EMBS Conference on Neural Engineering*, pp. 97-100, Arlington, Va, USA, March 2005.
- [3] P. Bühlmann and B. Yu, "Boosting with the  $L_2$  loss: regression and classification", *J. American Statistical Association*, vol 98, pp. 324-339, 2003.
- [4] J. H. Friedman, "Greedy Function Approximation: a Gradient Boosting Machine", *Annals of Statistics*, vol 29(5), pp. 1189-1232, 2001.
- [5] F. Jiao, J. Xu, L. Yu and D. Schuurmans, "Protein fold recognition using the gradient boost algorithm", in *Computational Systems Bioinformatics Conference (CSB2006)*, 2006.
- [6] J.-E. Grunwald, S. Schörnich and L. Wiegrebe, "Classification of natural textures in echolocation", in *Proc. Natl. Acad. Sci. USA*, vol. 101, pp. 5670-5674, 2004.
- [7] D. Ratner and P. McKerrow, "Landmark recognition with CTFM ultrasonic sensing", in *Proc. ACRA 2001*, pp. 104-110, 2001.

- [8] R. Muller, "A computational theory for the classification of natural biosonar targets based on a spike code", *Network: Computat. Neural Syst. J.*, vol. 14, pp. 595-612, 2003.
- [9] R. Kuc, "Transforming echoes into pseudo-action potentials for classifying plants", *J. Acoust. Soc. AM.*, vol. 110, pp. 2198-2206, 2001.
- [10] J. A. Simmons, N. Neretti, N. Intrator, R. A. Altes and M. J. Ferragamo, "Delay accuracy in bat sonar is related to the reciprocal of normalized echo bandwidth, or Q", in *Proc. Natl. Acad. Sci. USA*, vol. 101, pp. 3638-3643, March 2004.
- [11] W. Gao and Mark Hinders, "Mobile Robot Sonar Backscatter Algorithm for Automatically Distinguishing Walls, Fences, and Hedges", in *The International Journal of Robotics Research*, vol. 25(2), pp. 135-145, 2006.
- [12] M. Wang and A. Zell, Classification of natural landmarks with Biosonar, *Journal of the Acoustic Society of America (JASA)*, vol. 116, 2004.
- [13] M. Wang, *Natural Landmark Classification with a Biosonar based Mobile Robot*, PhD thesis, University of Tübingen, 2006.
- [14] G. Kimeldorf and G. Wahba, "Some results on tchebycheffian spline functions", *J. Math. Anal. Appl.*, vol. 33, pp. 82-95, 1971.
- [15] P. Sun and X. Yao, "Boosting Kernel Models for Regression", in *Sixth IEEE International Conference on Data Mining (ICDM'06)*, pp. 538-591, 2006.
- [16] L. Bo, L. Wang, and L. Jiao, "Training support vector machines using greedy stagewise algorithm", *Lecture Notes in Computer Science (PAKDD'05)*, vol. 3518, pp. 632-638, 2005.
- [17] G. Dai and D.-Y. Yeung, "Boosting Kernel Discriminant Analysis and Its Application to Tissue Classification of Gene Expression Data", *International Conference on Adaptive and Natural Computing Algorithms (ICANNGA 2005)*, vol 32, pp 53-58, 2005.
- [18] M. M. Beigi and A. Zell, "Synthetic Protein Sequence Oversampling method for Classification and remote homology detection in imbalanced protein data", *1st International Conference on Bioinformatics Research and Development (BIRD 2007)*, Berlin, Germany, pp. 263-277, 2007.
- [19] M. M. Beigi and A. Zell, "A novel kernel-based method for local pattern extraction in random process signals", *15th European Symposium on Artificial Neural Networks, Bruges, Belgium (ESANN 2007)*, pp. 265-270, April 2007.