Matched-field source localization using sparsely-coded neural network and data-model mixed training

Shougui Cai
College of Information Science and Electronic Eng.
Zhejiang University
Hangzhou, China
shouguicai@zju.edu.cn

Wen Xu
College of Information Science and Electronic Eng.
Zhejiang University
Hangzhou, China
wxu@zju.edu.cn

ABSTRACT

Source localization is a basic problem in underwater acoustics. Many solving approaches have been developed, and the matched-field processing (MFP) is one of the mostly-studied. However, MFP is sensitive to the mismatch problem and performs well only when the knowledge of ocean environment is accurate. Machine learning learns directly from the observation and can be designed to learn a generic model suitable for different scenarios. In this paper, source localization is viewed as a machine learning problem and a matched-field source localization model is learned by training a sparsely-coded feed-forward neural network with mixed environment models and data. Sparsely-coded network can prevent the model from over-learning. Results on SWellEx-96 experiment show that the learned model achieves good positioning performance in source range estimation for varying sound-speed profiles (SSP). Compared with Bartlett matched-field processing, machine learning model is more robust and thus has potential advantages in underwater source localization.

CCS CONCEPTS

• **Applied computing** → **Physical sciences and engineering**; • **Computing methodologies** → **Machine learning**;

KEYWORDS

Machine learning, source localization, underwater acoustics

1 INTRODUCTION

Matched-field processing (MFP) is a common technique for source localization in a underwater wave-guide[1–3], which matches measured acoustic pressure field data on an array of sensors with a replica field computed by a numerical propagation model for an assumed source range and depth. The processor output is maximum at the true source range and depth. However, MFP requires a pretty good knowledge of the environment, thus significant errors in the environment model can be introduced into the depth and range localization predictions[4].

Unlike MFP, machine learning methods do not require a good *a prior* information and can implement a required processing through learning from examples. Furthermore, a well designed structure can learn a generic model that works in different kinds of scenarios. This is meaningful to improving the robustness of matched-field source localization. Machine learning has obtained success in many areas, such as speech recognition, natural language processing and image processing. There are also applications of machine learning in underwater acoustics. For example, previous works have used artificial neural networks to classify whale sounds[5], locate targets[6] and discriminate depth[7]. A notable recent example using machine learning methods in underwater acoustic is the Niu's application of nonlinear classification to source localization[8]. As far as we can tell, there is no discussion on how to use a machine learning method to solve the mismatch problem in underwater source localization.

In this paper, the source localization problem is viewed within a machine learning framework, and a method that can tolerate the mismatch problem is proposed. As the sound-speed profile (SSP) in the water layer is the most important parameter needed to be known accurately[9], we primary focus on the SSP mismatch problem. In our simulations, two different degrees of error (a large one and a light one) in the knowledge of the sound-speed profile are chosen to train and test the model. Effects of such errors on positioning performance for various methods, including Bartlett matched-field processing, matched-covariance estimation (MCE)[10] and sparsely-coded feed-forward neural network (SCFNN), are compared. Treating different SSPs as different application scenarios, a generic model is learned by data-model mixed training, and the trained model is tested on varying SSPs. In Niu's work[8], he used a dense neural network to train the model, and the model performed well on the data of Noise09 experiment, which verified that feed-forward neural network (FNN) can achieve a good prediction performance when source localization is solved as a classification problem. However, as the author mentioned, the FNN classifier will be over-fitting when the SNR of training data is low, i.e. the model accuracy on the training set is much lower than the test set. In order to overcome this problem, an SCFNN is used in this paper. Besides, our models are trained and tested using SWellEx-96 experimental or simulated data.

WUWNet'17, November 2017, Halifax,NS,Canada

ACM ISBN 123-4567-24-567/08/06. . . $15.00
https://doi.org/10.475/123_4

2 SPARSE NEURAL NETWORK BASED SOURCE LOCALIZATION

In this section, we discuss how to establish an SCFNN for source localization prediction and how to train it with mixed data-model.

2.1 Neural networks and function approximation

As well known, neural networks models can be viewed as a mathematical function f. Taking feed-forward neural network (FNN) as an example, it defines a mapping $y = f(x; \theta)$ between the input x and the output y by parameter θ, where the parameters are to be learned by a rule. FNN is typically represented by composing together many different functions. There might have two function f^1, f^2 connected in a chain[11], to form $f(x) = f^2(f^1(x))$.

2.2 Source localization prediction model

In his paper[8], Niu assumed that there is a deterministic relationship between source range and sample-covariance matrix (SCM) and approximated this relationship by the FNN. Same as Niu did, we also use FNN to approximat the relationship between source range and SCM in this paper.

FNN extends linear models to represent nonlinear transformed format $\phi(x)$ of the input x. The transform function ϕ defines a hidden layer $h = \phi(x)$ and can be regarded as providing a set of features describing x, or as providing a new representation for x. The crucial problem here is how to choose the transform function ϕ. As the people engaged in machine learning usually do, we use a linear combination with nonlinear function to fit the basis functions,

$$h = g(W^{(1)}x + b^{(1)}) \tag{1}$$

Neurons between the hidden layer and the output layer are simply mapped by a linear function,

$$z = W^{(2)T}h + b^{(2)} \tag{2}$$

Then the output of the model is normalized by $softmax$ function, which is a common choice for multi-class classification task[12],

$$p(y_k|x) = softmax(z)_k = \frac{\exp(z_k)}{\sum_j \exp(z_j)} \tag{3}$$

where x is the input data, $p(y_k|x)$ is the probability that the measured signal x is transmitted from position k. W and b in Eq. (1) and Eq. (2) are the parameters to be learned.

Obviously, a learning criterion is needed. In most cases, the parametric model defines a distribution $p(y|x; \theta)$ and we can simply use the principle of maximum likelihood to determine the parameters in this model,

$$J(\theta) = -E_{x,y \sim p_{data}} \log p_{model}(y|x) \tag{4}$$

The objective function is $J(\theta)$ is described as the cross-entropy, equivalently the negative log-likelihood, between the training data p_{data} and the model distribution $p_{model}(y|x)$. As the maximum likelihood criterion is consistent, the model is capable of representing the training data distribution.

2.3 Training the model with sparsity constraint and mixed data-model

A. Sparsity constraint on neural networks

An SCFNN can be formed by adding sparsity constraint on networks and there are two main kinds of sparsity methods, including weight-level regularization and neuron-level regularization[11],

$$\tilde{J}(\theta) = J(\theta) + \alpha\Omega(\theta) + \beta\Omega(h) \tag{5}$$

where $\Omega(\theta)$ is the weight decay term and $\Omega(h)$ is the penalty on the activations of the units, α, β are hyper parameters that control the relative contribution of the two penalty terms. The weight decay term penalizes the size of the model parameters, while the activation penalty term encourages their activations to be sparse and makes the neural network to be sparsely coded. A sparsely-coded neural network encodes each input data as a sparse code firstly, and then accomplish the specific task with further processing.

In practical applications, we not only want the network to be sparse, but also want the model parameters to be sparse, the latter making the model more interpretable. In this paper, we use l_1-norm to promote sparse neurons activations, and constrain the l_2-norm of each column of the weight matrix $W_i^{(1)}$ to prevent any one hidden unit from having very large weights. The objective function is modified as below,

$$\begin{aligned} \tilde{J}(\theta) = & -E_{x,y \sim p_{data}} \log p_{model}(y|x) + \lambda \|h\|_1 \\ s.t. \quad & \left\| W_i^{(1)} \right\|_2 \leq C \quad \forall i = 1, \cdots, M \end{aligned} \tag{6}$$

where, M is the number of the neurons in hidden layer and C is the constraint coefficient. We choose $C = 1$ in our simulations.

B. Data-model mixed training

Another strategy used in this paper is training the neural network with mixed data-model, in order to increase the processing robustness. As neural networks are strong enough to learn regular pattern over a set of changing scenarios, when training the network, we can use the examples gathered form different mismatch scenarios to make the network be robust to mismatch. In our case, the receive acoustic pressure computed by various environment models, are combined with the actual data as the training set.

Neural network are easily to be over-learning, because of their powerful fitting ability and the noisy, discrepant data samples. Thus, sparsely-coded neural network is more preferred to be applied in data-model mixed training cases.

3 SIMULATION AND EXPERIMENTAL RESULTS

In this section, the SCFNN introduced above is used to learn the source range directly from the data of the SWellEx-96 experiment, and the performance of the classifier is compared with some other matched-field processing methods using simulation or experimental data, respectively. In addition, the influence of SSP mismatch on the performance of SCFNN classifier is investigated by simulations. In our application,

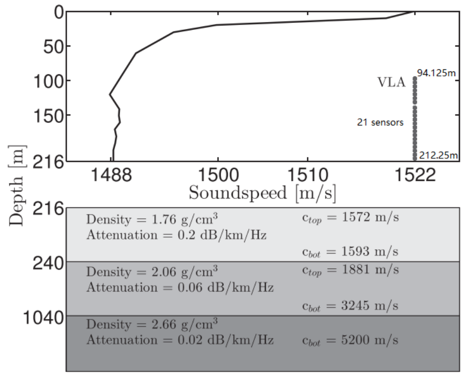

Figure 1: Environment model for SWellEx-96.

the robustness of the classifier is improved by training the model using data sampled under different SSP.

Simulation environment is the widely studied SWell96-Ex test, conducted in a shallow water waveguide environment with depth of 216 m. During the experiment, two moving sound sources are deployed in field, including a deep source (J-15) and a shallow source (J-13). In all of the following simulations, the shallow sound source is used, which was towed about 9 m in depth and emitted with 9 frequencies between 109 Hz and 385 Hz. The number of vertical array elements L is 21, and other specific deployment parameters are shown in Fig. 1.

3.1 Parameter settings

In the simulation part, acoustic data used to train and test the neural network is simulated using Kraken[13] with the environment model. The normalized SCMs of measured pressure at each frequency are used as network input data. In the input layer, the number of neurons D is $L^2 \times N_{fre}$(number of frequency used) and the number of neurons in the output layer (number of classes) is $K = 300$. Simply, the number of neurons in the hidden layer is set to be equal to the input layer, i.e. $M = D$. During the input data preprocessing, fast fourier transform duration is 1-seconds and the snapshot number N_s for constructing SCMs is 10. For the sake of learning speed and sparsity of hidden neurons, $ReLU$ activation [11] is applied. The training set contains 3000 samples sampled uniformly within 1.1475-8.6475 km in range, and the test set is another 300 data sampled within the same range. All of the noise in the simulations is set to be complex gaussian white.

Experimental data is from SWell96-Ex Event S5 vertical line array (VLA). The array recorded a total of 75 minutes data. In order to facilitate processing, 0-50 min data is taken as the training set. Consistent with the simulation part, the experimental trajectory was divided into 300 grids, 25 m each in range. The snapshot is also set as 1-second and finally 3000 SCMs would be. Each SCM is averaged at every two snapshots, 2700 of the samples are taken as the training set and the remainder as the test set.

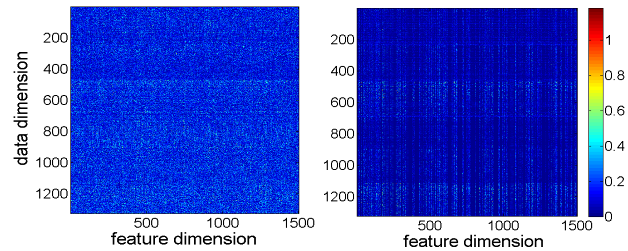

Figure 2: Weights in hidden layer (left: no constraint, right: with constraint, λ = 2.1e-5). Sparse constraint training makes the weight coefficients show group structure.

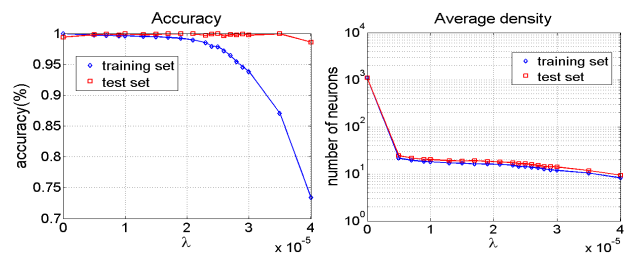

Figure 3: Model accuracy and average activation density for different λ. Regularization on neuron-level significantly reduces the average activation density and achieves good model accuracy. The model is trained on SWell96Ex-S5 experimental data.

3.2 The effect of sparsity constraint training

As shown in Fig. 3, the SCFNN efficiently prevents the model from over-learning, i.e. no over-fitting occurs. Besides, we can also find that the regularization degree on model affects the model accuracy and the average activation density of FNN's hidden layer. As the coefficient grows, the model accuracy on the test set also grows, which is good for improving the model performance. When the coefficient is too big, the model accuracy on the training set becomes much lower than the test set, which indicates under-fitting. The optimal coefficient λ in this paper is manually chosen by testing. On the other hand, as the coefficient increases, the order of magnitude on average activation drops from 10^3 to 10, and keeps stable. This is good for us to train the SCFNN.

Compared to the case of training without sparse constraints, SCFNN makes the weight coefficient in the hidden layer show group structure, i.e. the element of weight vector is either all zero, or basic zero. In our model, we choose $\lambda = 2.1 \times 10^{-5}$, and the number of feature vectors in hidden layer reduces from 1500 to 740. At the same time, the average activation neuron number in hidden layer is just 16 and the activation rate is only 1.1%.

To sum up, by using regularization strategy on neural networks, a sparse and low rank model is built, where the transferred representation is sparse and the rank of the learned

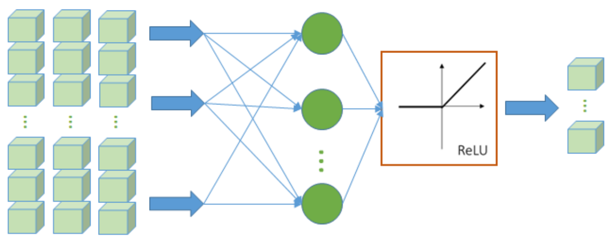

Figure 4: The learned sparse representation model. The learned feature space spans data (SCM) space likelihood that few basis functions ϕ explains a given data.

Table 1: Localization accuracy of SCFNN and MFP on SWell96Ex-S5 experimental data

Methods	SCFNN	MCE	Bartlett 1	Bartlett 2
109Hz	89.3%	72.3%	37.7%	3.7%
232Hz	97%	91%	17.7%	4.3%
385Hz	99.7%	97.7%	14%	0.67%
109,232,385Hz	99%	99.7%	40.7%	7.7%

Table 2: Absolute mean error of SCFNN and MFP on SWell96Ex-S5 experimental data(m)

Methods	SCFNN	MCE	Bartlett 1	Bartlett 2
109Hz	28.1	290.3	852.8	1219.5
232Hz	7.4	2.5	832.3	832.3
385Hz	0.08	0.58	1266.7	1756.3
109,232,385Hz	0.25	0.083	477.2	722.9

weight matrix is low. In our example, the input 1323-elements SCM data space can be spanned by the 740 feature vectors, and averagely, each data sample can be represented by only 16 features.

3.3 Comparison with conventional matched-field processing method

As a comparison, Bartlett processor is used here position the source. There are two main kinds of replica-field used in the Bartlett processor, one is simulated by Kraken (noted as Bartlett 2), the other is the measurement data (noted as Bartlett 1), same as the training data used in SCFNN. The accuracy and absolute mean error for different methods under different frequency are summed in Table 1 and Table 2. The model accuracy is defined as,

$$acc = \frac{1}{N}\sum_{n=1}^{N} I(y_n, t_n) \times 100\% \tag{7}$$

where t_n and y_n are the real and the predictive class of the n-th sample data separately, N is number of test samples,

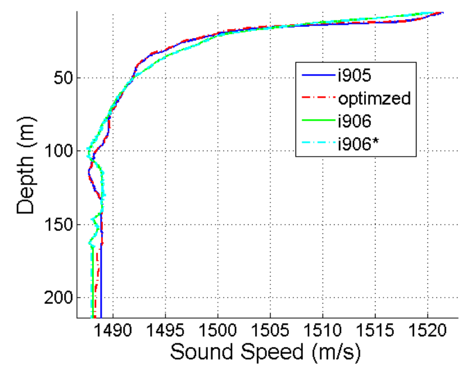

Figure 5: Plots of sound speed profiles. The i906 has significant change in shape from the optimized, while the change in the i905 is slight. The i906* is slightly changed from i906, for the sake of testing.

and $I(y_n, t_n)$ is defined as,

$$I(y_n, t_n) = \begin{cases} 1 & \text{if } y_n = t_n, \\ 0 & \text{otherwise.} \end{cases} \tag{8}$$

As we can see, whether using single frequency or multi-frequencies, the accuracy of SCFNN is always better than the Bartlett, and not worse than direct data match (noted as MCE), this is more obvious when it comes to the comparison of absolute mean error. It can be said that, the learned SCFNN works well on source localization and performs better than the Bartlett processor.

3.4 The influences of SSP mismatch on FNN classifier

In the MFP method, the model accuracy is heavily affected by the mismatch problem[4, 9]. The performance curves for SCFNN, Bartlett and MCE vs SNR are plotted by 1000 times Monte Carlo simulations. In Fig. 6, the legend 'FNN,i905' means that, the corresponding method is FNN and the test data is from i905 environment, rests are similar. The snapshot number here is 10. Simulation results show that when the change in SSP is relatively slight (see Fig. 5), SCFNN positions best, followed by MCE and Bartlett worst; when the change is relatively large (with shape varying), the accuracy order is unchanged, but the absolute mean error of SCFNN becomes larger than MCE. This is maybe caused by the noisy training data. In Fig. 5, the optimized one is the best SSP model for real environment of SWellEx-96 experiment, while, i906, i905 are the measured SSPs from different devices. In conclusion, SCFNN is also sensitive to SSP mismatch, but still performs better than Bartlett and the performance of SCFNN is close to the MCE method.

3.5 Increase model robustness by data-model mixed training

As mentioned in section 3.4, the SCFNN is also sensitive to SSP mismatch. When the environment SSP has a big change

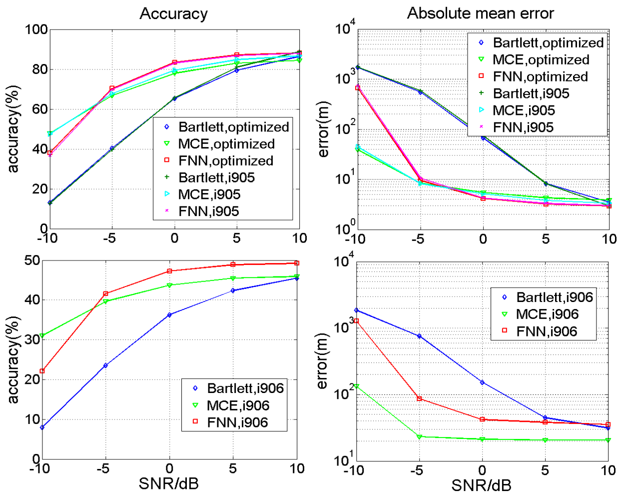

Figure 6: FNN positioning performance curve on simulation data (frequency: 109, 232, 385 Hz). FNN is also sensitive to SSP mismatch, but still performs better than Bartlett.

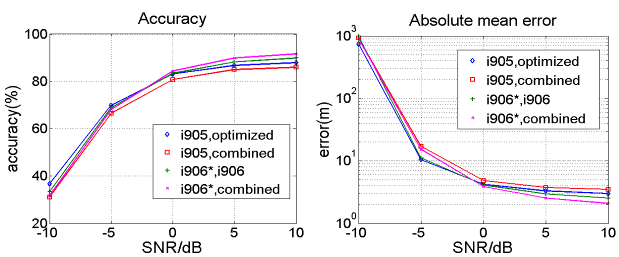

Figure 7: FNN positioning performance curve on simulation data. FNN model robustness can be significantly improved by data-model mixed training.

in shape, the classifier trained by single data set performs poorly. For example, the model, trained by data set corresponding to ssp-optimized (see Fig.5), performs poorly on ssp-i906*, and the accuracy drops about 40%, compared with the performance on ssp-i905. In this section, by combining the data collected from ssp-i906 and ssp-optimized as training set, the robustness of the classifier increases significantly; as Fig. 7 shows, the re-trained classifier predicts accurately on ssp-i906*, just as well as on ssp-i905. Although the accuracy for i905 has a little glissade compared with data training only case, the performance for i906 is improved. By mixed data-model training, the SCFNN classifier works well on two entirely different SSPs. Note that, in Fig. 7, the legend 'i905,combined' means the model is trained by mixed data, and then tested on ssp-i905. The rest legends are similar.

4 SUMMARY

In this paper, we propose a method that can help reduce the mismatch problem in matched-field source localization, by using a sparsely-coded feed-forward neural network(SCFNN), combined with data-model mixed training. The proposal is examined on SWellEx-96 experiment. To be specific, we firstly train and test a prediction model on the experimental data, and confirm that the SCFNN works well on source localization. Then, the influence of SSP mismatch on the SCFNN classifier is investigated by simulations. Finally, we train the SCFNN with mixed environment model data. It can be seen that the model robustness is significantly improved and the trained classifier performs well on varying SSP environments.

Comparing with dense neural networks, the sparsely-code neural network needs fewer basis functions to span the data space and is beneficial to capture characteristic of data distributions, which will make the model more descriptive.

The discussions on applying machine learning methods to overcome mismatch problem in underwater source localization are preliminary and only a fine-tuned FNN is used in this paper. Machine learning has potential advantages in unstable underwater acoustic environment and thus deserves more further efforts.

ACKNOWLEDGMENTS

This work is supported by the National Natural Science Foundation of China (No. 41376104) and the Fundamental Research Funds for the Central Universities.

REFERENCES

[1] Bucker, H. P. Use of calculated sound fields and matched-field detection to locate sound sources in shallow water. *The Journal of the Acoustical Society of America 59*, 2 (1976), 368–373.

[2] Baggeroer, A. B., Kuperman, W., and Schmidt, H. Matched field processing: Source localization in correlated noise as an optimum parameter estimation problem. *The Journal of the Acoustical Society of America 83*, 2 (1988), 571–587.

[3] Baggeroer, A. B., Kuperman, W. A., and Mikhalevsky, P. N. An overview of matched field methods in ocean acoustics. *IEEE Journal of Oceanic Engineering 18*, 4 (1993), 401–424.

[4] Tolstoy, A. Sensitivity of matched field processing to sound-speed profile mismatch for vertical arrays in a deep water pacific environment. *The Journal of the Acoustical Society of America 85*, 6 (1989), 2394–2404.

[5] Thode, A. M., Kim, K. H., Blackwell, S. B., Greene Jr, C. R., Nations, C. S., McDonald, T. L., and Macrander, A. M. Automated detection and localization of bowhead whale sounds in the presence of seismic airgun surveys. *The Journal of the Acoustical Society of America 131*, 5 (2012), 3726–3747.

[6] Steinberg, B. Z., Beran, M. J., Chin, S. H., and Howard Jr, J. H. A neural network approach to source localization. *The Journal of the Acoustical Society of America 90*, 4 (1991), 2081–2090.

[7] Ozard, J. M., Zakarauskas, P., and Ko, P. An artificial neural network for range and depth discrimination in matched field processing. *The Journal of the Acoustical Society of America 90*, 5 (1991), 2658–2663.

[8] Niu, H., Reeves, E., and Gerstoft, P. Source localization in an ocean waveguide using supervised machine learning. *The Journal of the Acoustical Society of America 142*, 3 (2017), 1176–1188.

[9] Feuillade, C., Del Balzo, D., and Rowe, M. M. Environmental mismatch in shallow-water matched-field processing: Geoacoustic parameter variability. *The Journal of the Acoustical Society of America 85*, 6 (1989), 2354–2364.

[10] Zhou, Y., and Xu, W. Matched-field source localization via statistical covariance matching. In *Oceans-San Diego, 2013* (2013), IEEE, pp. 1–5.

[11] Goodfellow, I., Bengio, Y., and Courville, A. *Deep learning*. MIT press, 2016.

[12] Bishop, C. M. *Pattern recognition and machine learning*. Springer, 2006.

[13] Porter, M. B. The Kraken normal mode program. Tech. rep., NAVAL RESEARCH LAB WASHINGTON DC, 1992.