Proceedings of the 28th IEEE
EMBS Annual International Conference
New York City, USA, Aug 30-Sept 3, 2006

Combining Nonlinear Dimensionality Reduction with Wavelet Network to Solve EEG Inverse Problem

Qing Wu, Lukui Shi, Youxi Wu, Guizhi Xu, Ying Li, and Weili Yan

Abstract—**An integrated multi-method system to analyze the neuroelectric source parameters of electroencephalography (EEG) signal is presented. In order to handle the large-scale high dimension data efficiently and provide a real-time localizer in EEG inverse problem, an improved isometric mapping algorithm is used to find the low dimensional manifolds from high dimensional recorded EEG. Then, based on reduced dimension data, a single-scaling radial-basis wavelet network module is employed to determine the parameters of different type of EEG source models. In our simulation experiments, satisfactory results are obtained.**

I. Introduction

COMPARED with other functional neuroimaging techniques, e.g. fMRI and PET, one of the main advantages of electroencephalography (EEG) is its high temporal resolution, which can follow the changes in neural activity on a millisecond time scale and enable us to observe the dynamic behavior of the human brain. So, an important aspect in EEG inverse problem should be calculating the parameters of the internal neuroelectric sources rapidly. Unfortunately, traditional iterative methods are often too time consuming to take advantages of the high temporal resolution of EEG. In this paper, the practice of using wavelet network (WN) combined with isometric mapping (ISOMAP) method for the EEG inverse problem is carried out. The disadvantage of the high computational cost of iterative approaches and the difficulty in the description of inverse model are avoided. A self-associative-memory is established in the training of WN, which can determine the inner relation between the EEG on the scalp and the electric current sources in the brain correctly. The parameters of the dipole sources can be worked out rapidly for new measured data, providing a novel approach to the dynamic analysis of EEG. Although taking the original multichannel EEG directly as the input of WN is objective, the amount of data is too large. In order to further reduce the computational cost, a nonlinear dimensionality reduction method ISOMAP is used to pretreat the original data and choose suitable characteristic patterns as input variables.

Manuscript received March 30, 2006. This work was supported by the Natural Science Foundation of Hebei Province, China under Grant E2005000024, and by the Ph. D. Foundation of Hebei Education Department under Grant B2004107.

Qing Wu, Lukui Shi and Youxi Wu are with the College of Computer Science and Software, Hebei University of Technology, Tianjin 300130, China (phone: 86-22-26582488-817; fax: 86-22-26564111; e-mail: qingwu@hebut.edu.cn).

Guizhi Xu, Ying Li and Weili Yan are with the Province-Ministry Joint Key Laboratory of Electromagnetic Field and Electrical Apparatus Reliability, Hebei University of Technology, Tianjin 300130, China (e-mail: gzxu@hebut.edu.cn).

II. An Improved ISOMAP Algorithm

Principal component analysis (PCA) and multidimensional scaling (MDS) are two classical dimensionality reduction methods. However, these kinds of linear algorithms in essence have no way to discover complex nonlinear manifold structure. Recently, several different algorithms, such as ISOMAP [1] and LLE [2], have been developed to perform dimensionality reduction of low-dimensional nonlinear manifolds embedded in a high dimensional space.

The main idea of ISOMAP is that classical MDS is applied to the geodesic distances to find a set of low-dimensional points with similar pairwise distances. The crux of the method is to estimate the geodesic distances between points, which represent the shortest paths along the curved surface of the manifold. For a neighboring point, the input space distance provides a good approximation to the geodesic distance. For faraway points, the actual geodesic distances are approximated by a sequence of short hops between neighboring points. For a data set X with N points in the input space, the main idea of the algorithm is: 1) Determine which points are neighbors on the manifold M based on the distances $d_X(i, j)$ between pairs of points (i, j) in X by a suitable way. 2) Estimate the geodesic distances $d_M(i, j)$ between all pairs of points. 3) Apply MDS on the pairwise distances to construct low dimensional coordinates. For details about ISOMAP please refer to [1].

A. Select the Landmark Set and Determine the Neighborhood Parameter

ISOMAP has two computational bottlenecks. The first is calculating the $N \times N$ shortest path distance matrix. The complexity is $O(kN^2 \log N)$ using Dijkstra's algorithm with Fibonacci heaps (k is the neighborhood size), and $O(N^3)$ using Floyd's algorithm. The second bottleneck is the MDS eigenvalue calculation, which involves a full $N \times N$ matrix and has complexity $O(N^3)$. This also leads to huge memory requirements.

In contrast to ISOMAP, the LLE algorithm has higher efficiency and lower memory requirements since the eigenvalue computation in LLE is sparse. If a sparse eigenvalue can be obtained in ISOMAP, its execution will speed up. LMDS [3] solves the problem by only preserves the geodesic distances between each point and some landmark points. Since landmark points are randomly chosen, they often cannot represent the true topology of the manifold in the

input space and even lead to worse results. Therefore, the stress function (SF) is used to evaluate quantitatively on landmark sets

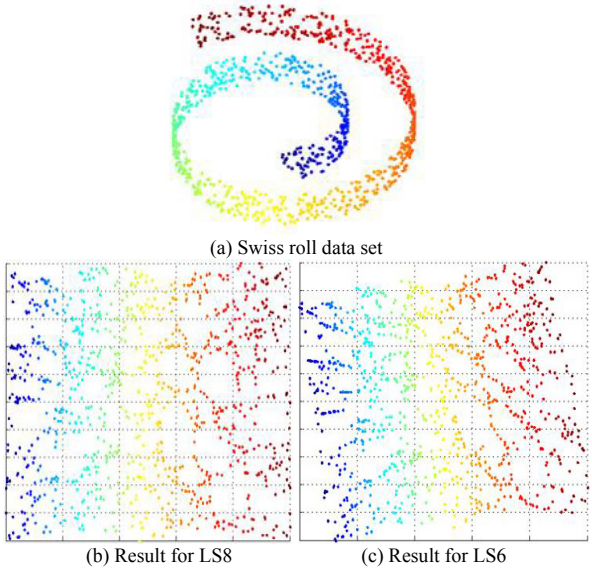
$$SF = \sqrt{\sum_{1\le i<j\le n}(d_M(i,j)-d_m(i,j))^2 \Big/ \sum_{1\le i<j\le n} d_M^2(i,j)} \quad (1)$$

where $d_M(i,j)$ is the geodesic distance between pairs of points in the input space, $d_m(i, j)$ the Euclidean distance in the embedded space and n the number of landmark points.

Here, we apply ISOMAP on the Swiss roll dataset. We randomly choose 10 landmark sets (LS1-10) with 20 data points for each set from Swiss roll data set with 1000 samples. The values of SF are given in Table I. The results prove that the smaller the value of SF is, the better the selected landmark set is. From Fig. 1, we can see (b) is much better than that in (c).

TABLE I
VALUES OF SF FOR DIFFERENT LANDMARK SETS

Landmark sets	Values of SF	Landmark sets	Values of SF
LS1	0.0181	LS6	0.0213
LS2	0.0180	LS7	0.0185
LS3	0.0192	LS8	0.0161
LS4	0.0186	LS9	0.0297
LS5	0.0173	LS10	0.0184



(a) Swiss roll data set

(b) Result for LS8 (c) Result for LS6

Fig. 1. Low-dimensional embeddings from ISOMAP algorithm for various landmark sets

The other parameter that affects the result of ISOMAP is the neighborhood size k. Experimental results on Swiss roll data set demonstrate that an improper neighborhood parameter k will lead to poor low-dimensional embeddings. Equation (1) can also be used to determine a proper neighborhood size. Let $k \in \{4, 5, 6, 7, 8, 9\}$ and randomly take 20 data points from Swiss roll data set as the landmark set. The values of SF are shown in Table II after running ISOMAP on the landmark set LS8 with different k. Experimental results also point out that the smaller the value of SF is, the better the low-dimensional embedding is; otherwise, the worse the result is.

TABLE II
VALUES OF SF FOR DIFFERENT NEIGHBORHOOD PARAMETER k

k	Values of SF	k	Values of SF
4	0.0359	7	0.0161
5	0.0228	8	0.1327
6	0.0196	9	0.1332

B. Main Steps of the Improved ISOMAP

The improved ISOMAP (IISOMAP) method that can automatically select a proper neighborhood size and an appropriate landmark set according to the stress function is as follows.

Step 1. Execute the ISOMAP algorithm on a randomly selected landmark set that contains n data points for a range of the neighborhood parameter k.

Step 2. Compute the values of SF for each k and chose the k with the smallest SF as the final obtained neighborhood parameter K.

Step 3. Calculate the geodesic distances between landmark points and other points for m random landmark sets using K.

Step 4. Map each set into the embedded space with ISOMAP.

Step 5. Figure out the values of SF on each landmark set, and take the set with the smallest SF as the last chosen landmark set M.

Step 6. Obtain the low-dimensional coordinates of the whole data set with the selected neighborhood size K and the landmark set M.

In the above algorithm, it is obviously $n << N$. The time complexity of computing the geodesic distance matrix is $O(knN\log N)$ using Dijkstra while MDS runs in $O(n^2N)$ time. Both are much smaller than those of ISOMAP.

III. CONSTRUCTIVE ALGORITHM OF A SINGLE-SCALING RADIAL-BASIS WAVELET NETWORK

In general, the implementation of wavelet network often suffers from the curse of dimensionality, and that it cannot determine the structure automatically. In order to avoid these, based on a single-scaling wavelet frame [4] and RBF network model, we propose a constructive method of a multi-dimensional WN, which can be used as a universal approximator. Here, single scaling means that a single dilation parameter is used in all of the dimensions of each wavelet. Due to the symmetry feature, it is natural for radial functions to use single-scaling manner when being expanded to high dimension. A single-scaling radial wavelet frame F of $L^2(R^d)$ can be built by scaling a mother function "Mexican hat" (2).

$$\psi(\boldsymbol{x}) = (d - \|\boldsymbol{x}\|^2)e^{-\|\boldsymbol{x}\|^2/2}, \quad \boldsymbol{x} \in R^d \quad (2)$$

$$F=\{\psi_{n,\boldsymbol{m}}(\boldsymbol{x})=2^{dn/2}\psi(2^{-n}\boldsymbol{x}-\boldsymbol{m}):n\in Z,\boldsymbol{m}\in Z^d\}\quad(3)$$

Let $\theta=\{(\boldsymbol{x}_i,\boldsymbol{y}_i):\boldsymbol{x}_i\in R^d,\boldsymbol{y}_i\in R^s,i=1,\cdots,N\}$ be a sample set provided under $\boldsymbol{y}=\boldsymbol{f}(\boldsymbol{x})$. If the radial basis functions in RBFN are replaced by F, while centers and radius parameters are replaced by the shift and scale parameters of the wavelets respectively, a good approximation of $\boldsymbol{f}$ can be obtain [5]:

$$\widetilde{\boldsymbol{f}}(\boldsymbol{x})=\sum_{n,\boldsymbol{m}}W_{n,\boldsymbol{m}}\psi_{n,\boldsymbol{m}}$$

where $W_{n,\boldsymbol{m}}$ is the weight value. Thus, the relationship between input and output for an unknown system can be deduced based on θ.

Fortunately, in the practical neuroelectric source localization, the learning samples are often not uniformly distributed in the entire domain of the input space. Rather, they are commonly sparse and distributed in some subspace in the form of clusters. So we start with the analysis of the sparseness of the learning samples and focus on reducing the redundancy of F based on localization of wavelets and time-frequency information of the samples' input vector. Then, we use an adaptive orthogonal projective algorithm that can automatically determine the network size and calculate the network weights in the light of the information given by the samples' output vector. Our main steps of the algorithm are summarized in the following.

First, "blank" wavelets are taken out by selecting a self-adaptable discrete wavelet frame based on the input sample data and the time-frequency localization property of wavelets. Let the effective time and frequency supports of ψ be defined respectively as

$$\mathrm{supp}(\psi,\varepsilon)=[x_0,x_1]$$
$$\mathrm{supp}(\hat{\psi},\hat{\varepsilon})=[\omega_0,\omega_1]$$

where ε and $\hat{\varepsilon}$ are two arbitrarily small positive number, and $\hat{\psi}$ is the Fourier transformation of ψ. Then, the time-frequency supports of $\psi_{n,m}$ can be deduced [6]

$$\mathrm{supp}(\psi_{n,m},\varepsilon)=[2^n(x_0+m),2^n(x_1+m)]$$
$$\mathrm{supp}(\hat{\psi}_{n,m},\hat{\varepsilon})=[2^{-n}\omega_0,2^{-n}\omega_1]$$

Step 1. Determine the range of dilation parameter *n*, and for each dilation *n*, calculate the range of translation parameter *m* according to input data.

$$n_{\min}=\mathrm{int}(-\lg(4\pi/\Delta x\omega)/\lg 2)$$
$$n_{\max}=\mathrm{int}(\lg(x_{\max}-x_{\min})/\lg 2)$$
$$m_{\min}=\mathrm{int}(x_{\min}/2^n)$$
$$m_{\max}=\mathrm{int}(x_{\max}/2^n)$$

From time and frequency boundary, the infinite wavelet frame F is shortened to

$$F'=\{\psi_{n,m}:n_{\min}\le n\le n_{\max};m_{\min}\le m\le m_{\max}\}\quad(4)$$

Step 2. Calculate the supports of each wavelet $\varepsilon-\mathrm{supp}(\psi_{n,\boldsymbol{m}})$ for a chosen ε, and search for such wavelets

$$\{\psi_{n,m}:\boldsymbol{x}_i\in\mathrm{supp}(\psi_{n,m}(\boldsymbol{x}),\varepsilon)\}$$

in F' whose support includes at least one $\boldsymbol{x}_i(i=1,\cdots,N)$, that is, eliminate the wavelets whose supports do not contain any data point $\boldsymbol{x}$.

Suppose $F''=\{\psi_1,\cdots,\psi_L\}$ be the resulting set of selected wavelets after the above two steps.

Second, to further remove redundant elements in F'', wavelets with small "contribution" are got rid of based on the output sample data and self-adapting orthogonal projection algorithm. Considering the wavelets in F'' as regressors, techniques of matching pursuits can be applied. And Gram-Schmidt algorithm is combined to orthonormalize and further selects optimal wavelets. Moreover, WN training is implemented during this executing process. The elements $\psi_j(j=1,2,\cdots,L)$ in F'' and output data rewrite as

$$\boldsymbol{\psi}_j=[\psi_j(\boldsymbol{x}_1),\cdots,\psi_j(\boldsymbol{x}_N)]^{\mathrm{T}}/\|[\psi_j(\boldsymbol{x}_1),\cdots,\psi_j(\boldsymbol{x}_N)]^{\mathrm{T}}\|$$
$$\boldsymbol{Y}=[\boldsymbol{y}_1,\cdots,\boldsymbol{y}_N]^{\mathrm{T}};\ \boldsymbol{y}_i=[y_{i1},y_{i2},\cdots,y_{is}];\ \boldsymbol{x}_i,\boldsymbol{y}_i\in\theta;\ i=1,\cdots,N$$

For the estimation of unknown $\boldsymbol{f}$, the "best" subset of $M=\{\boldsymbol{\psi}_1,\cdots,\boldsymbol{\psi}_L\}$ is selected by an iterative procedure which can be described as follows:

The first best wavelet $\boldsymbol{\psi}_{l_1}$ meets the requirement of $|<\boldsymbol{y},\boldsymbol{\psi}_{l_1}>|=\sup_{1\le l_j\le L}|<\boldsymbol{y},\boldsymbol{\psi}_{l_j}>|$ according to the maximum matching pursuit algorithm. The rest of the wavelet vectors $\boldsymbol{\psi}_j,j\in\{\{1,\cdots,L\}-l_1\}$ are orthogonalized with $\boldsymbol{\psi}_{l_1}$ and the result is expressed as

$$\boldsymbol{u}_j=\boldsymbol{\psi}_j-<\boldsymbol{\psi}_j,\boldsymbol{\psi}_{l_1}>\boldsymbol{\psi}_{l_1}.$$

Since $\boldsymbol{u}_j$ and $\boldsymbol{\psi}_{l_1}$ are orthogonal, the next selected best wavelet $\boldsymbol{\psi}_{l_2}$ corresponds to the $\boldsymbol{u}_j$ which is most adjacent to $\boldsymbol{y}$, that is,

$$l_2=\arg\max_j(|<\boldsymbol{u}_j,\boldsymbol{y}>|/\|\boldsymbol{u}_j\|)$$

Let $\boldsymbol{v}_{l_1}=\boldsymbol{\psi}_{l_1}$ and $\boldsymbol{v}_{l_2}=\boldsymbol{u}_{l_2}/\|\boldsymbol{u}_{l_2}\|$, and so $\boldsymbol{v}_{l_1}$ and $\boldsymbol{v}_{l_2}$ are standard orthogonal vector sets.

Repeat the above procedure, and suppose *i* is the current iterative label, $\boldsymbol{\psi}_{l_1},\boldsymbol{\psi}_{l_2},\cdots,\boldsymbol{\psi}_{l_{i-1}}$ are the wavelets selected before, then orthogonalize the $\boldsymbol{\psi}_j$ ($j\in\{\{1,2,\cdots,L\}-\{l_1,\cdots,l_{i-1}\}\}$) with $\boldsymbol{v}_{l_k}(k=1,2,\cdots,i-1)$, the result can be expressed as

$$\boldsymbol{u}_j^{(i)}=\boldsymbol{u}_j^{(i-1)}-<\boldsymbol{\psi}_j,\boldsymbol{v}_{l_{i-1}}>\boldsymbol{v}_{l_{i-1}}$$
$$\boldsymbol{u}_j^{(0)}=\boldsymbol{\psi}_j$$

Correspondingly

$$l_i=\arg\max_j(|<\boldsymbol{u}_j^{(i)},\boldsymbol{y}>|/\|\boldsymbol{u}_j^{(i)}\|)$$
$$\boldsymbol{v}_{l_i}=\boldsymbol{u}_{l_i}^{(i)}/\|\boldsymbol{u}_{l_i}^{(i)}\|$$

If the error after *h* iterations is smaller than a given value, the algorithm stopped. WN can be constructed as

$$\boldsymbol{y} = \boldsymbol{\Psi w}$$

where $\boldsymbol{\Psi} = [\boldsymbol{\psi}_{l_1}, \cdots, \boldsymbol{\psi}_{l_h}]$ and $\boldsymbol{w} = [w_{l_1}, \cdots, w_{l_h}]^{\mathrm{T}}$. In fact, the orthogonalization procedure is a division process, that is, dividing $\boldsymbol{\Psi}$ to

$$\boldsymbol{\Psi} = \boldsymbol{VA}$$

where $\boldsymbol{V}$ is composed by standard orthogonal vector, $\boldsymbol{V} = [\boldsymbol{v}_{l_1}, \cdots, \boldsymbol{v}_{l_h}]$, and the elements in the upper triangular matrix $\boldsymbol{A}$ are $\alpha_{ii} = \| \boldsymbol{u}_{l_i} \|$, $\alpha_{ij} = < \boldsymbol{\psi}_{l_j}, \boldsymbol{u}_{l_i} >$, $i = 1, \cdots, h$, $j = i+1, \cdots, h$.

So during the above procedure, the connection weight w_{ij} between the hidden node j and the output node i is calculated. The resulting WN for single-output is

$$\widetilde{f}_i(\boldsymbol{x}) = \sum_{j=1}^{h} w_{ij} \psi(2^{-n_j} \boldsymbol{x} - \boldsymbol{m}_j) \quad i = 1,2,\cdots,s \qquad (5)$$

As for a WN with s-output, the construction method is almost the same, the only difference being that the evaluation standard for the best wavelet vector in the i iteration is the average best basis, that is

$$l_i = \arg\max_j \left(\sum_{r=1}^{s} | < \boldsymbol{u}_j^{(i)}, \boldsymbol{y}^r > | / \| \boldsymbol{u}_j^{(i)} \| \right)$$

After h iteration steps

$$\boldsymbol{Y} = \boldsymbol{\Psi W}$$

The solution of the $h \times s$ weight matrix $\boldsymbol{W} = [\boldsymbol{w}^1, \cdots, \boldsymbol{w}^s]$, $\boldsymbol{w}^r = [w_{l_1}^r, \cdots, w_{l_h}^r]^{\mathrm{T}}$, $(r = 1, \cdots, s)$ is $\boldsymbol{W} = \boldsymbol{A}^{-1} \boldsymbol{V}^{\mathrm{T}} \boldsymbol{Y}$.

IV. Experimental Results

Using three source models – single dipole, disk, and line source – in [7], the capability of the proposed system on the EEG source localization problem is tested. The data sample set that relates the given parameters vector $\boldsymbol{y}$ of a certain source model to the scalp potential vector $\boldsymbol{x}$ can be produced by solving EEG forward problem.

In our simulation experiments, $\boldsymbol{x}$ is 137-dimensions, and $\boldsymbol{y}$ is 6, 7 and 9-dimensions for the single dipole, the disk, and the line source model respectively [7]. After 2000 patterns from each model are generated, we apply IISOMAP on this obtained dataset. Fig. 2 shows the relationship between ISOMAP dimensionality and residual variance. Fig. 3 shows the three-dimensional projection result, where the red, blue and green dots respectively denote the single dipole, the disk, and the line source model.

After the 137-dimensional data $\boldsymbol{x}$ is mapped into a 5-dimensional embedding with IISOMAP, three wavelet networks are independently trained based on the dimensionality reduction data set. In order to evaluate the performance of the trained WN, we generate a test set according to the method that generate the training set, and then test the WN with the test data. An average relative mean-square error is defined by

$$\text{Error} = \sum_{i=1}^{N} \sum_{j=1}^{s} (y_j(i) - \widetilde{y}_j(i))^2 \Big/ \sum_{i=1}^{N} \sum_{j=1}^{s} y_j^{\,2}(i) \qquad (6)$$

where y and $\widetilde{y}$ are the theoretical and wavelet network computed parameters. The total training and test error is 3.81% and 5.96%, respectively.

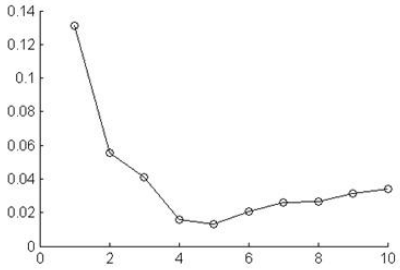


Fig. 2. Residual variance curve

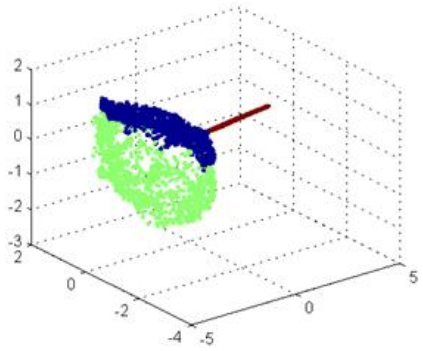


Fig. 3. Projection result by IISOMAP

V. Conclusion

The emulation results show that the wavelet network combined with nonlinear dimensionality reduction ISOMAP seems to be a reliable and fast way for EEG inverse problem. Although a relatively long time is needed for the training process, it can be done off-line. Moreover, the method is also flexible since it is not based on any particular assumption from prior knowledge.

References

[1] J. B. Tenenbaum, V. de Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, pp. 2319–2323, Dec. 2000.

[2] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, pp. 2323–2326, Dec. 2000.

[3] V. De Silva and J. B. Tenenbaum, "Global versus local methods in nonlinear dimensionality reduction," in *Advances in Neural Information Processing Systems*, vol. 15, pp. 705–712, 2002.

[4] T. Kugarajah and Q. Zhang, "Multidimensional wavelet frames," *IEEE Trans. Neural Networks*, vol. 6, pp. 1552–1556, Nov. 1995.

[5] Q. Zhang, "Using wavelet network in non-parametric estimation," *IEEE Trans. Neural Networks*, vol. 8, pp. 227–236, Mar. 1997.

[6] Q. Wu, X. Shen, and W. Yan, "Neuroelectric Source Localization with Wavelet Network," in *Proc. 4th Int. Conf. on Electromagnetic Field Problems and Applications*, Tianjin, China, Sept. 2000, pp. 454–457.

[7] M. Sonmez, M. Sun, C. C. Li, and R. J. Sclabassi, "A hierarchical decision module based on multiple neural networks," in *Proc. of the IEEE Int. Conf. on Neural Networks*, Houston, TX, June 1997, pp. 238–241.