Proceedings of the IASTED International Conference
ARTIFICIAL INTELLIGENCE AND SOFT COMPUTING
July 17-19, 2002, Banff, Canada

PATTERN DISCRIMINATION OF TIME SERIES EEG SIGNALS USING A RECURRENT NEURAL NETWORK

Osamu FUKUDA
Institute for Human Science and Biomedical Engineering
National Institute of Advanced Industrial Science and Technology
1-2-1, Namiki, Tsukuba, 305-8564, JAPAN.
E-mail: fukuda.o@aist.go.jp
Toshio TSUJI, Nan BU, and Makoto KANEKO
Department of the Artificial Complex Systems Engineering
Hiroshima University, Higashi-Hiroshima, Hiroshima, JAPAN
E-mail: tsuji@bsys.hiroshima-u.ac.jp

ABSTRACT
This paper proposes a new discrimination method of time series EEG signals using the Recurrent Log-Linearized Gaussian Mixture Network (R-LLGMN). The structure of R-LLGMN is based on a hidden Markov model, which has been well developed in the area of speech recognition. The weight coefficients in the network can be learned using the back-propagation through time algorithm. In order to examine the EEG discrimination ability of the proposed method, comparison experiments were conducted using the several discrimination methods, such as the statistical neural networks, recurrent neural filters, and hidden Markov models. It can be seen from the experimental results that R-LLGMN can achieve high discrimination performance.

KEY WORDS
Neural network, Pattern discrimination, EEG, Hidden Markov model.

1 Introduction

The electroencephalogram (EEG) includes very important information for a clinical diagnosis of neurological disorders or a mental disability, so that the EEG analysis is often utilized for medical examination in a clinic. On the other hand, this signal can be expected to be used as a control signal for a new type of a man-machine interface because the signal pattern changes depending on the internal or external factors such as intentions of movements, photic and auditory stimulation. This paper proposes a new discrimination method of EEG signals using a recurrent neural network.

Up to the present, several methods for the EEG pattern discrimination have been proposed. For example, Peltoranta *et al.* [1] used the self-organizing feature map, the learning vector quantizer (LVQ), the K-mean and neural networks (NNs) [1], and conducted the comparison experiments. The discrimination method using a NN was especially reported by Pritchard *et al.* [2], Pfurtscheller *et al.* [3][4] and Selvan and Srinivasan [5]. Pritchard *et al.* [2] attempted to classify Alzheimer's disease from the EEG signals using the Back Propagation Neural network (BPN). Pfurtscheller *et al.* [3][4] utilized the EEG as the brain computer interface (BCI). They classified the motion of the left and the right arms using the BPN with the LVQ. Selvan and Srinivasan [5] proposed a recurrent neural network including an adaptive filter, where they trained the network using real time recurrent learning. They also proposed a method to remove the ocular artifacts from the EEG signals.

While the BPN was utilized in most of the previous studies, Tsuji *et al.* proposed the log-linearized Gaussian mixture network (LLGMN) [6] based on a log-linear model and a Gaussian mixture model. LLGMN can acquire the log-linearized Gaussian mixture model through learning and calculate the *a posteriori* probability. Moreover, in order to cope with time-varying characteristics of the EEG signals, they combined this network with a Neural Filter (NF) [7]. Although this method attained relatively high classification rates, it was necessary to train two different types of NNs, that is, LLGMN and NF, therefore the learning procedure became quite complicated and general optimization was almost impossible.

In this paper, a novel NN, a Recurrent Log-Linearized Gaussian Mixture Network (R-LLGMN) is proposed by introducing the recurrent connection into LLGMN to classify a time sequence of EEG signals. Since this network is composed of a feedforward NN including a Gaussian mixture model and feedback connections from output to input, the pattern discrimination and the filtering process are unified together and realized in a single network. R-LLGMN includes a hidden Markov model (HMM) [8] in its structure and can regulate the weight coefficients by the back-propagation through time (BPTT) algorithm [9]. R-LLGMN ensures the pattern discrimination and the filtering process to be achieved at the same time and can attain high discrimination ability.

2 Recurrent Log-Linearized Gaussian Mixture Network

The structure of a proposed network is shown in Fig. 1. This network is a five-layer recurrent NN with a feedback connection between the fourth layer and the third layer. First of all, the input vector $\mathbf{x}(t) = [x_1(t), x_2(t), \cdots, x_d(t)]^{\mathrm{T}} \in \Re^d (t = 1, \ldots, T)$ is pre-processed with a non-linear computation and converted into the modified vector $\mathbf{X} \in \Re^H$:

$$\mathbf{X}(t) = [1, \mathbf{x}(t)^{\mathrm{T}}, x_1(t)^2, x_1(t)x_2(t), \cdots, x_1(t)x_d(t), x_2(t)^2, x_2(t)x_3(t), \cdots, x_2(t)x_d(t), \cdots, x_d(t)^2]^{\mathrm{T}}. \quad (1)$$

The first layer consists of H units corresponding to the dimension of $\mathbf{X}$(the dimension H is determined as $H = 1 + d(d + 3)/2$) and the identity function is used for activation of each unit. The input ${}^{(1)}I_h$ and the output ${}^{(1)}O_h$ of the hth unit in the first layer are defined as

$${}^{(1)}I_h(t) = X_h(t), \quad (2)$$

$${}^{(1)}O_h(t) = {}^{(1)}I_h(t). \quad (3)$$

Unit$\{c, k, k', m\}$ $(c = 1, \cdots, C\ ;\ k, k' = 1, \ldots, K_c\ ; m = 1, \ldots, M_{c,k})$ in the second layer receives the output of the first layer weighted by the coefficient $w^c_{k',k,m,h}$. The relationship between the input and the output in the second layer is defined as

$${}^{(2)}I^c_{k',k,m}(t) = \sum_{h=1}^{H} {}^{(1)}O_h(t) w^c_{k',k,m,h}, \quad (4)$$

$${}^{(2)}O^c_{k',k,m}(t) = \exp\left({}^{(2)}I^c_{k',k,m}(t)\right), \quad (5)$$

where C is the number of classes, K_c is the number of states, $M_{c,k}$ is the number of the components of the Gaussian mixture distribution corresponding to the class c and the state k [6].

The input into a unit $\{c, k, k'\}$ in the third layer integrates the outputs of units $\{c, k, k', m\}$ $(m = 1, \ldots, M_{c,k})$ in the second layer. The output in the third layer is that input weighted by the previous output in the fourth layer. The relationship in the third layer is defined as

$${}^{(3)}I^c_{k',k}(t) = \sum_{m=1}^{M_{c,k}} {}^{(2)}O^c_{k',k,m}(t), \quad (6)$$

$${}^{(3)}O^c_{k',k}(t) = {}^{(4)}O^c_{k'}(t-1) {}^{(3)}I^c_{k',k}(t), \quad (7)$$

where ${}^{(4)}O^c_{k'}(0) = 1.0$ for the initial state.

The fourth layer receives the integrated outputs of units $\{c, k, k'\}$ in the third layer. The relationship in the fourth layer is defined as

$${}^{(4)}I^c_k(t) = \sum_{k'=1}^{K_c} {}^{(3)}O^c_{k',k}(t), \quad (8)$$

$${}^{(4)}O^c_k(t) = \frac{{}^{(4)}I^c_k(t)}{\sum_{c'=1}^{C} \sum_{k'=1}^{K_{c'}} {}^{(4)}I^{c'}_{k'}(t)}. \quad (9)$$

At last, a unit c in the fifth layer integrates the outputs of K_c units $\{c, k\}$ $(k = 1, \ldots, K_c)$ in the fourth layer. The relationship in the fifth layer is defined as

$${}^{(5)}I^c(t) = \sum_{k=1}^{K_c} {}^{(4)}O^c_k(t), \quad (10)$$

$${}^{(5)}O^c(t) = {}^{(5)}I^c(t). \quad (11)$$

The output of the network ${}^{(5)}O^c(t)$ corresponds to the *a posteriori* probability of the input vector $\mathbf{x}(t)$ for the class c, while only the weight coefficients $w^c_{k',k,m,h}$ between the first layer and the second layer are adjusted by learning.

The network is learned with the teacher vector $\mathbf{T}^{(n)} = (T_1^{(n)}, \cdots, T_c^{(n)}, \cdots, T_C^{(n)})^{\mathrm{T}} (n = 1, \ldots, N)$ for the nth input stream $\mathbf{x}(t)^{(n)}$ at a time T. If the input stream $\mathbf{x}(t)^{(n)}$ is set for the class $\hat{c}$, then $T_{\hat{c}}^{(n)} = 1$, and $T_c^{(n)} = 0$ for all the other classes. R-LLGMN is trained with the given N vector streams which divided into L subsets, while each set consists of C stream classes $(N = L \times C)$. In this paper an energy function J for the network is defined as

$$J = \sum_{n=1}^{N} J_n = -\sum_{n=1}^{N} \sum_{c=1}^{C} T_c^{(n)} \log {}^{(5)}O^c(T)^{(n)}, \quad (12)$$

where ${}^{(5)}O^c(T)^{(n)}$ means the last output $(t = T)$ for stream $\mathbf{x}(t)^{(n)}$. The learning process is to minimize J, that is, to maximize the likelihood that each teacher vector $\mathbf{T}^{(n)}$ is obtained for the input vector $\mathbf{x}(t)^{(n)}$.

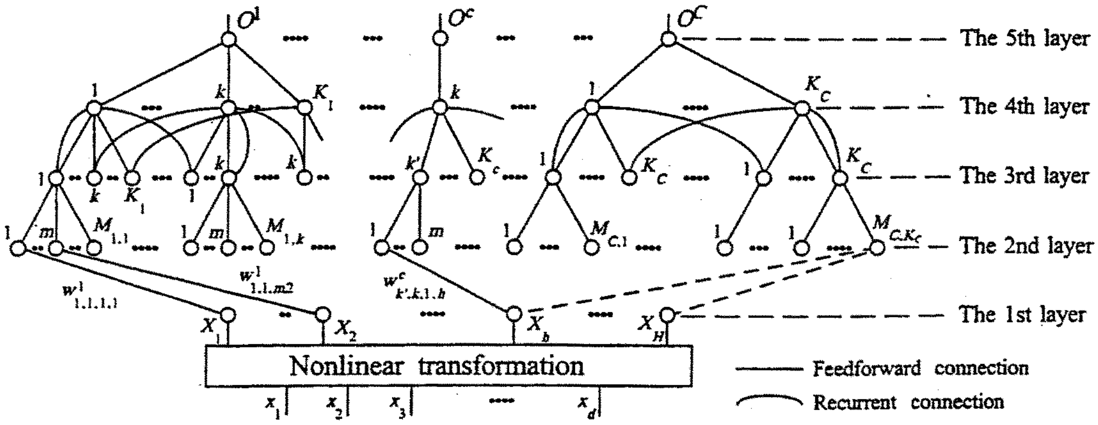

Figure 1. The structure of R-LLGMN

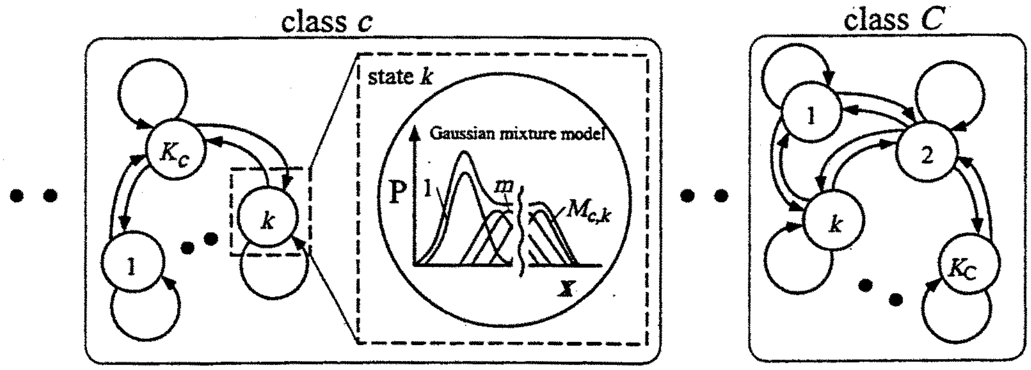

Figure 2. A dynamic probabilistic model with C classes and K_c states

3 Relation between R-LLGMN and Hidden Markov Model

This section proves that R-LLGMN can be interpreted as a hidden Markov Neural Network based on a continuous density hidden Markov model [10][11].

Let us consider a dynamic model [8], as shown in Fig. 2, where there are C classes in this model and each class c ($c \in \{1,\ldots,C\}$) is composed of K_c states. Suppose that, for the given time series $\tilde{X} = \mathbf{x}(1), \mathbf{x}(2), \cdots, \mathbf{x}(T)$ ($\mathbf{x}(t) \in \Re^d$), at any time $\mathbf{x}(t)$ must occur from one state k of class c in the model, *a posteriori* probability for class c, $P(c|\tilde{X})$, is derived as

$$P(c|\tilde{X}) = \sum_{k=1}^{K_c} P(c,k|\tilde{X}) \tag{13}$$

$$= \sum_{k=1}^{K_c} \frac{a_k^c(T)}{\sum_{c'=1}^{C}\sum_{k'=1}^{K_{c'}} a_{k'}^{c'}(T)}. \tag{14}$$

$$a_k^c(t) = \sum_{k'=1}^{K_c} a_{k'}^c(t-1)\gamma_{k',k}^c b_k^c(\mathbf{x}(t)) \; (t>1), \tag{15}$$

$$a_k^c(1) = \pi_k^c b_k^c(\mathbf{x}(1)), \tag{16}$$

where $\gamma_{k',k}^c$ is the probability of the state changing from k' to k in class c, and $b_k^c(\mathbf{x}(t))$ is defined as *a posteriori* probability for state k in class c corresponding to $\mathbf{x}(t)$. Also the *a priori* probability π_k^c equals to $P(c,k)|_{t=0}$.

When the *a posteriori* probability of state k in class c corresponding to $x(t)$, $b_k^c(\mathbf{x}(t))$, is approximated by summing up $M_{c,k}$ components of Gaussian mixture distribution [6][12], $\gamma_{k',k}^c b_k^c(\mathbf{x}(t))$ in the right side of (15) can be derived with the form

$$\gamma_{k',k}^c b_k^c(\mathbf{x}(t)) = \sum_{m=1}^{M_{c,k}} \gamma_{k',k}^c r_{c,k,m} g(\mathbf{x}(t); \mu^{(c,k,m)}, \Sigma^{(c,k,m)}) \quad (t>1), \tag{17}$$

where $r_{c,k,m}$, $\mu^{(c,k,m)} \in \Re^d$ and $\Sigma^{(c,k,m)} \in \Re^{d\times d}$ stands for the mixing proportion, the mean vector and the covariance matrix of each component $\{c,k,m\}$, respectively. Using the mean vector $\mu^{(c,k,m)} = (\mu_1^{(c,k,m)}, \cdots, \mu_d^{(c,k,m)})^{\mathrm{T}}$ and the inverse of the covariance matrix $\Sigma^{(c,k,m)-1} = [s_{ij}^{(c,k,m)}]$, the right side of (17) can be rewritten as

$$\begin{aligned}
&\gamma_{k',k}^{c} r_{c,k,m} g(\mathbf{x}(t); \mu^{(c,k,m)}, \Sigma^{(c,k,m)}) \\
&= \gamma_{k',k}^{c} r_{c,k,m} (2\pi)^{-\frac{d}{2}} |\Sigma^{(c,k,m)}|^{-\frac{1}{2}} \\
&\quad \times \exp\left[-\frac{1}{2}\sum_{j=1}^{d}\sum_{l=1}^{j}(2-\delta_{jl}) s_{jl}^{(c,k,m)} x_j x_l \right. \\
&\quad + \sum_{j=1}^{d}\sum_{l=1}^{d} s_{jl}^{(c,k,m)} \mu_j^{(c,k,m)} x_l \\
&\quad \left. - \frac{1}{2}\sum_{j=1}^{d}\sum_{l=1}^{d} s_{jl}^{(c,k,m)} \mu_j^{(c,k,m)} \mu_l^{(c,k,m)} \right], \qquad (18)
\end{aligned}$$

where δ_{ij} is the Kronecker delta: $\delta_{ij} = 1$ when $i = j$ and $\delta_{ij} = 0$ when $i \neq j$, and $|\cdot|$ stands for the matrix determinant. Then, taking logarithm of (18), we get

$$\gamma_{k',k}^{c} b_k^{c}(\mathbf{x}(t)) = \sum_{m=1}^{M_{c,k}} \xi_{k',k,m}^{c}(t), \qquad (19)$$

$$\begin{aligned}
&\xi_{k',k,m}^{c}(t) \\
&\triangleq \log \gamma_{k',k}^{c} r_{c,k,m} g(\mathbf{x}(t); \mu^{(c,k,m)}, \Sigma^{(c,k,m)}) \\
&= \beta_{k',k,m}^{c}{}^{\mathrm{T}} \mathbf{X}(t), \qquad (20)
\end{aligned}$$

where $\mathbf{X}(t) \in \Re^H$ and $\beta_{k',k,m}^{c} \in \Re^H$ are defined as

$$\begin{aligned}
\mathbf{X}(t) &= (1, \mathbf{x}(t)^{\mathrm{T}}, x(t)_1{}^2, x(t)_1 x(t)_2, \cdots, x(t)_1 x(t)_d, \\
&\quad x(t)_2{}^2, x(t)_2 x(t)_3, \cdots, x(t)_2 x(t)_d, \\
&\quad \cdots, x(t)_d{}^2)^{\mathrm{T}}, \qquad (21)
\end{aligned}$$

$$\begin{aligned}
\beta_{k',k,m}^{c} &= \left(\beta_{k',k,m}^{c,0}, \sum_{j=1}^{d} s_{j1}^{(c,k,m)} \mu_j^{(c,k,m)}, \cdots, \right. \\
&\quad \sum_{j=1}^{d} s_{jd}^{(c,k,m)} \mu_j^{(c,k,m)}, -\frac{1}{2} s_{11}^{(c,k,m)}, \\
&\quad -s_{12}^{(c,k,m)}, \cdots, s_{1d}^{(c,k,m)}, \cdots, \\
&\quad -\frac{1}{2}(2-\delta_{jl}) s_{jl}^{(c,k,m)}, \cdots, \\
&\quad \left. -\frac{1}{2} s_{dd}^{(c,k,m)} \right)^{\mathrm{T}}, \qquad (22)
\end{aligned}$$

$$\begin{aligned}
\beta_{k',k,m}^{c,0} &= -\frac{1}{2}\sum_{j=1}^{d}\sum_{l=1}^{d} s_{jl}^{(c,k,m)} \mu_j^{(c,k,m)} \mu_l^{(c,k,m)} \\
&\quad -\frac{d}{2}\log 2\pi - \frac{1}{2}\log|\Sigma^{(c,k,m)}| \\
&\quad + \log r_{c,k,m} + \log \gamma_{k',k}^{c}. \qquad (23)
\end{aligned}$$

Equation (21) describes the nonlinear pre-process for the first layer in R-LLGMN (see (1)), and $\xi_{k',k,m}^{c}$ can be expressed as the product of the coefficient vector $\beta_{k',k,m}^{c}$ and the modified input vector $\mathbf{X} \in \Re^H$. Hence, the model can be expressed as the neural network structure by using $\beta_{k',k,m}^{c}$ as the weight coefficients.

However, the definition of $\beta_{k',k,m}^{c}$ (22) indicates that most elements of $\beta_{k',k,m}^{c}$ are constrained by the statistical properties of the parameter $s_{i,j}^{(c,k,m)}$, and these constraints may cause a difficult problem in the learning procedure: how to satisfy the constraints during the learning of the weight coefficients. Therefore the new variable $Y_{k',k,m}^{c}$ and the new coefficient vector $\mathbf{w}_{k',k,m}^{c}$ are introduced to get rid of the constraints:

$$\begin{aligned}
Y_{k',k,m}^{c}(t) &\equiv \xi_{k',k,m}^{c}(t) - \xi_{K_C,K_C,M_{C,K}}^{C}(t) \\
&= \left(\beta_{k',k,m}^{c} - \beta_{K_C,K_C,M_{C,K}}^{C}\right)^{\mathrm{T}} \mathbf{X}(t) \\
&= \mathbf{w}_{k',k,m}^{c}{}^{\mathrm{T}} \mathbf{X}(t), \qquad (24)
\end{aligned}$$

where $\mathbf{w}_{K_C,K_C,M_{C,K}}^{C} = 0$ by definition.

This new parameter $\mathbf{w}_{k',k,m}^{c}$ has no constraints and is used as the weight coefficient in this paper. Subsequently, equation (15) can be rewritten in the form as

$$\begin{aligned}
a_k^c(t) &= \sum_{k'=1}^{K_c} a_{k'}^{c}(t-1)\gamma_{k',k}^{c} b_k^c(\mathbf{x}(t)) \\
&= \sum_{k'=1}^{K_c} a_{k'}^{c}(t-1)\exp\left[Y_{k',k,m}^{c}(t)\right] \\
&\qquad (t > 1). \qquad (25)
\end{aligned}$$

Comparing (13)-(16),(21),(24),(25) in HMM with (1)-(11) in R-LLGMN, we can see both of them are equivalent. It means that R-LLGMN regards the coefficient vector $\mathbf{w}_{k',k,m}^{c}$ as a weight coefficient vector and modifies them so as to optimize an energy function via learning with the BPTT.

4 EEG Pattern Discrimination

4.1 Experimental Conditions

The EEG signals measured from the electrodes were digitized by an A/D converter after they were amplified and filtered out through low-cut (3 Hz) and high-cut (40 Hz) analogue filters. The noise in the EEG signals can be removed significantly by the bipolar derivation between the two electrodes located at Fp1 and Fp2.

Subjects were seated on the chair, and a flash light (xenon, illuminating power: 0.176 [J], frequency of flashing: 4 [Hz]) was set at the distance of 50 cm apart from their eyes. First, the EEG signals were measured during photic stimulation by opening/closing eyes and an artificial light (60 seconds for each). The measured signals were used as learning data. Then, photic stimulation was given alternatively according to the pseudo-random series for 420 seconds, and the EEG signals were measured for the discrimination data.

The electroencephalograph used in the experiments has one pair of the electrodes, so that the spatial information of the EEG signals on the location of the electrodes cannot be utilized. The frequency characteristics of the EEG signals, however, significantly changes depending on the eye states. Therefore, the spectral information of the measured EEG signals were used as follows. The power spectral density function of the measured EEG signal was estimated using the FFT for every 128 sampled data. The function was divided into several ranges (from 0~35 Hz). The frequency bands of this range were determined based on the clinical use of the brain wave (delta, theta, alpha, beta). Time series of the mean values of the power spectral density function within each frequency ranges were calculated and normalized between [0, 1] in each range. Thus, the d-dimensional data were obtained and used as the input vector to the networks $x_1, x_2, \ldots, x_d$. However, there is almost no influence of the dimension number d [6], the experimental result with the 2-dimensional data ($d = 2$) is shown (corresponding to frequency range $0 \sim 8, 9 \sim 35$ [Hz]).

In the experiments, four discrimination methods were used for comparison, such as LLGMN, LLGMN with NF, HMM, and R-LLGMN. LLGMN is a feedforward type probabilistic neural network which is based on the log-linearized Gaussian mixture model [6]. This network corresponds to the special case of R-LLGMN when the length of time series, T, is 1, and $K_c = 1$ for each unit in the 5th layer. LLGMN used $M_1 = M_2 = 1$ components, and $L = 56$ training data for each class. As for LLGMN with NF, there were 8 units in NF and 168 data were used for the training. In NF, fully interconnected units in the second layer keep the internal representation, so that the time history of the input data can be considered [7]. The history of the input data was considered back to 5 steps. NF works like a filter and makes the *a posteriori* probability as the output from LLGMN to be smoother. On the other hand, in R-LLGMN, the parameters of the network were set as: $K_1 = K_2 = 1, M_{1,1} = M_{2,1} = 1, T = 5, L = 5$. In HMM, the number of states $K_1 = K_2 = 1$, the number of training data $L = 5$, the length of the training data $T = 5$ were used, and the number of quantization level QL varied from 2 to 4.

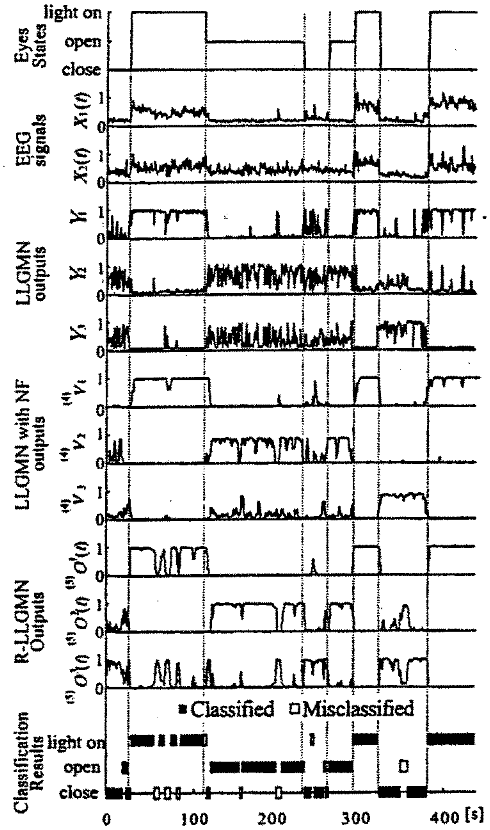

Figure 3. An example of the discrimination results for three types of photic stimulation (subject A)

4.2 Discrimination Result

Figure 3 shows an example of the discrimination result of subject A. In the figure, the timing of switching photic stimulation, the input EEG signals correspond to two frequency bands, the output of LLGMN, LLGMN with NF and R-LLGMN, and the discrimination results of the R-LLGMN are shown. The output of R-LLGMN is smooth and stable. R-LLGMN performs relatively high discrimination rate of 87.4% in this experiments.

Table 1 shows discrimination results for five subjects. The mean values and the standard deviations of the discrimination rate were computed for 10 kinds of initial

Table 1. discrimination results for three types of photic stimulation

Type of the methods		LLGMN	LLGMN with NF	HMM			R-LLGMN
				$QL=2$	$QL=3$	$QL=4$	
Subject A (male)	*CR*	75.8	84.4	74.3	88.6	82.9	84.3
	SD	0.5	0.1	0.0	0.0	0.0	0.1
Subject B (male)	*CR*	83.8	91.6	79.5	88.1	94.3	95.7
	SD	0.3	0.3	0.0	0.0	0.0	0.0
Subject C (female)	*CR*	69.1	78.3	64.3	90.5	92.9	82.8
	SD	4.5	2.3	0.0	0.0	0.0	4.1
Subject D (male)	*CR*	65.8	74.6	72.4	82.9	76.7	86.2
	SD	1.7	1.8	0.0	0.0	0.0	0.0
Subject E (male)	*CR*	77.2	87.1	46.2	69.0	84.8	87.7
	SD	1.9	0.9	0.0	0.0	0.0	0.0
Total	*CR*	74.3	83.2	67.3	83.8	86.3	87.7
	SD	1.8	1.1	0.0	0.0	0.0	0.8

CR : Classification rate [%], *SD* : Standard deviation [%], *QL* : Number of quantization level

weights, which were randomly chosen. According to the experimental results, except for LLGMN, all the other methods attained high discrimination rates. Speaking in general, LLGMN based on a static Gaussian mixture model does not fit for discrimination of the dynamic signal like the EEG, while the other methods contains the dynamic statistical model.

5 Conclusions

In this paper, the new recurrent neural network, R-LLGMN, has been proposed to perform a pattern discrimination for a time series of EEG signals. R-LLGMN is a recurrent NN including a Gaussian mixture model and a feedback connection from output to input. Therefore, this network ensures the pattern discrimination and the filtering process to be achieved at the same time.

The discrimination experiments for EEG signals under the photic stimulation have been carried out to examine the discrimination capability of the proposed network. The results of discrimination experiments showed that R-LLGMN performs the filtering process as well as the pattern discrimination together in the same network architecture and can realize a relatively high discrimination rate.

Future research will be directed toward improving the learning algorithm for the application of the discrimination of various bioelectric signals.

Acknowledgments

A part of this work was supported by Industrial Technology Development Organization (NEDO) of Japan.

References

[1] M. Peltoranta and G. Pfurtscheller, "Neural network based classification of non-averaged event-related EEG responses," Med and Biol. Eng. and Comput., 32, pp. 189-96, 1994.

[2] W. S. Pritchard, D. W. Duke, K. L. Coburn, N. C. Moore, K. A. Tucker, M. W. Jann and R. M. Hostetler, "EEG-based, neural-net predictive classification of Alzheimer's disease versus control subjects is augmented by non-linear EEG measures," *Elctroencephalography and clinical Neurophysiology*, Vol. 91, No. 2, pp. 118-130, 1994.

[3] D. Flotzinger, G. Pfurtscheller, C. Neuper, J. Berger and W. Mohl, "Classification of non-averaged EEG data by learning vector quantisation and the influence of signal pre-processing," *Medical Biological Engineering Computing*, Vol. 32, No. 5, pp. 571-576, 1994.

[4] G. Pfurtscheller, J. Kalcher, C. Neuper, D. Flotzinger and M. Pregenzer, "On-line EEG classification during externally-paced hand movements using a neural network-based classifier," *Elctroencephalography and clinical Neurophysiology*, Vol. 99, No. 5, pp. 416-425, 1996.

[5] S. Selvan and R. Srinivasan, "Removal of Ocular Artifacts from EEG Using an Efficient Neural Network based Adaptive Filtering Technique," *IEEE Signal Processing Letters*, Vol. 6, No. 12, pp. 330-332, 1999.

[6] T. Tsuji, O. Fukuda, H. Ichinobe and M. Kaneko, "A log-linearized Gaussian mixture network and its application to EEG pattern classification," *IEEE Trans. Systems, Man, and Cybernetics-Part C: Applications and Reviews*, Vol. 29, No. 1, pp. 60-72, 1999.

[7] O. Fukuda, T. Tsuji and M. Kaneko, "Pattern Classification of a Time-Series EEG Signal Using a Neural Network," *Transactions Institute of Electronics, Information and Communication Engineers*, Vol. J80-D-II, No. 7, pp. 1896-1903, 1997. (in Japanese)

[8] L.E. Baum and T. Petrie, "Statistical inference for probabilistic function of finite state Markov chains," *Ann. Math. Stat.*, Vol. 37, No. 6, pp. 1554-1563, 1966.

[9] P.J. Werbos, "Backpropagation through time: what it does and how to do it," *Proceedings of the IEEE*, Vol. 78, No. 10, pp. 1550-1560, 1990.

[10] L.E. Baum and G.R. Sell, "Growth functions for transformations on manifold," *Ann. Math. Stat.*, Vol. 27, No. 2, pp. 211-227, 1968.

[11] B.H. Juang, S.E. Levinson, and M.M. Sondhi, "Maximum Likelihood estimation for multivariate mixture observations of Markov chains," *IEEE Trans. Informat. Theory*, Vol. IT-32, No. 2, pp. 307-309, 1986.

[12] D.M. Titterington, A.F.M. Simth and U.E. Markov, *Statistical analysis of finite mixture distributions*, John Wiley & Sons, New York, 1985.