A Neural Network Architecture for Speech Segmentation Using Mean Field Annealing

C. G. Jeong and H. Jeong
Department of Electrical Engineering, POSTECH
P.O.Box 125, Pohang 790-600, South Korea

Abstract

As a dual algorithm to the Geiger–Girosi restoration scheme, a new segmentation method is introduced and used to demonstrate an approach to phoneme–boundary detection. Also we introduce a neural network suitable for this algorithm, which consists of sigmoid neurons and Sigma-Pi neurons. Experimental results show that the new algorithm is superior to the *forward-backward* algorithm and the Geiger–Girosi algorithm in terms of position accuracy and recognition accuracy as well as computational speed for phoneme–boundary detection.

I. Introduction

Many segmentation algorithms are based upon the *divergence test* [2], [8], log-likelihood [3] or any other test-statistics [6], [9]. Among them, the *divergence test* shows prominent results. This algorithm is, however, very sensitive to threshold values, and critical misses may occur at vowel to consonant(VC) transitions. In order to detect better the VC transition, Regine [2] proposed the *forward-backward* algorithm by introducing a backward search scheme to the *divergence test.*

On the other hand, Geman and Geman [10] introduced a generic stochastic framework for signal reconstruction, especially for optical images. Since then, some investigators [11] applied this paradigm into the speech signal processing. This approach, however, requires much computation time due to simulated annealing.

Recently Geiger and Girosi [5] proposed a fast method for the Geman algorithm using mean field approximation and *saddle point approximation.* This mean field solution was especially designed for signal restoration, and therefore shows poor estimates of *line processes*; in general, it tends to generate many false-alarms at object boundaries. To overcome this problem, we propose an approach opposite to this scheme, that is tailored to obtain accurate boundaries with the help of less accurate restoration, and that is suitable for neural network implementation.

II. Segmentation Framework

A Markov process together with its boundary process is defined in the *parameter space* (X, L) of size N: $X \triangleq \{X(k); k \in \{1, \cdots, N\}\}$ and $L \triangleq \{L(k); k \in \{1, \cdots, N\}\}$, where $X(k)$ is a signal process in the form of time sequence, cepstrum, or short time Fourier transform defined at time k. Also $L(k) \in \{0, 1\}$ is the so-called *line process* which determines the existence of discontinuity between $X(k-1)$ and $X(k)$. We assume that from the hidden process (X, L), the *observation*, $Y \triangleq \{Y(k);\ k \in \{1, \cdots, N\}\}$, is generated with additive Gaussian noise.

Based upon this stochastic model, we can find the MAP estimate of (X, L) from Y:

$$(\mathbf{x}^*, \mathbf{l}^*) = \arg\max_{(\hat{\mathbf{x}}, \hat{\mathbf{l}})} P(\hat{\mathbf{x}}(\mathbf{y}), \hat{\mathbf{l}}(\mathbf{y})|\mathbf{y}), \tag{1}$$

where $\hat{\mathbf{x}}(\mathbf{y})$ and $\hat{\mathbf{l}}(\mathbf{y})$ are the estimates of the realizations $\mathbf{x}$ and $\mathbf{l}$, respectively. Since X and L

0-7803-1901-X/94 $4.00 ©1994 IEEE

are Markov random fields, $P(\cdot)$ can be represented by the Gibbs distribution [7]:

$$P(\mathbf{x}, \mathbf{l}|\mathbf{y}) = \frac{1}{Z_0} \exp[-\beta U_0(\mathbf{x}, \mathbf{l}|\mathbf{y})], \tag{2}$$

where $U_0(\cdot)$ is an *energy function*, $1/\beta$ is a *temperature* constant, and Z_0 is a *partition function*:

$$Z_0 \triangleq \sum_{\mathbf{x}} \sum_{\mathbf{l}} \exp[-\beta U_0(\mathbf{x}, \mathbf{l}|\mathbf{y})]. \tag{3}$$

Following Geman's strategy [10], we can define $U_0(\mathbf{x}, \mathbf{l}|\mathbf{y})$ as

$$U_0(\mathbf{x}, \mathbf{l}|\mathbf{y}) \triangleq \sum_{k=1}^{N} \Big\{ ||x(k) - y(k)||^2 + \lambda(1 - l(k))||x(k) - x(k-1)||^2 + \gamma l(k) + \nu \sum_{i \in N_k} \frac{1}{(k-i)^2} l(k) l(i) \Big\}, \tag{4}$$

where λ, γ and ν are weighting parameters, $N_k \triangleq \{k-1, k-2, \cdots, k-d\}$ is a dth neighborhood of $L(k)$, and $||\cdot||$ is the L_p norm. Here p is the dimension of the signal process. Notice that this energy function consists of the three terms: *matching*, *smoothness*, and *a priori* knowledge about discontinuity. Considering (1)–(4) together, we note that the MAP estimate $(\mathbf{x}^*, \mathbf{l}^*)$ is a minimizer of the functional $U_0(\mathbf{x}, \mathbf{l}|\mathbf{y})$, as was first noticed by Geman [10].

As a special case of (4), a *weak membrane model* is

$$U_0(x, l|y) = \sum_{k=1}^{N} [(x(k) - y(k))^2 + \lambda(1 - l(k))(x(k) - x(k-1))^2 + \gamma l(k)]. \tag{5}$$

To solve this equation, Geman [10] proposed the *Gibbs Sampler*, and Blake and Zisserman [1] the *graduated nonconvexity* (GNC) algorithm, respectively. Recently, Geiger [5] derived a deterministic relaxation solution.

$$\left\{ \begin{array}{rcl} \bar{x}^{\tau+1}(k) & = & \bar{x}^{\tau}(k) - \mu\Big[(\bar{x}^{\tau}(k) - y(k)) + \lambda\{(\bar{x}^{\tau}(k) - \bar{x}^{\tau}(k-1))(1 - \bar{l}^{\tau}(k)) \\ & & -(\bar{x}^{\tau}(k+1) - \bar{x}^{\tau}(k))(1 - \bar{l}^{\tau}(k+1))\}\Big], \\ \bar{l}^{\tau}(k) & = & \frac{1}{1+\exp[-\beta(\lambda||\bar{x}^{\tau}(k) - \bar{x}^{\tau}(k-1)||^2 - \gamma(k))]}, \end{array} \right. \tag{6}$$

where τ is an iteration index, and μ a convergence rate. In particular, note that $\bar{x}^{\tau}(k)$ is recursively improved but $\bar{l}^{\tau}(k)$ is indirectly estimated from $\bar{x}^{\tau}(k)$. As a result, the resolution of $\bar{l}^{\tau}(k)$ is poor and an alternative approach is required.

III. Mean Field Approximation

To begin with, let us introduce an auxiliary process $s(k)$ having continuous value on real line:

$$l(k) \triangleq \frac{1}{1 + \exp(-\beta s(k))}, \quad s(k) \in (-\infty, \infty). \tag{7}$$

Next, substituting (4) into (3) yields

$$\begin{aligned} Z_0 &= \int \int \exp\Big[-\beta \sum_{k=1}^{N} \Big\{ ||x(k) - y(k)||^2 + \lambda(1 - l(s(k)))||x(k) - x(k-1)||^2 \\ &+ \gamma l(s(k)) + \nu \sum_{i \in N_k} \frac{1}{(k-i)^2} l(s(k)) l(s(i)) \Big\}\Big] ds(1) \cdots ds(N) dx(1) \cdots dx(N). \end{aligned} \tag{8}$$

Approximating Z_0 by replacing $x(k-1)$ by the mean field value $\bar{x}(k-1)$ and applying *saddle point approximation* to $s(k)$ will give

$$Z_0 \approx C \prod_{k=1}^{N} \int_{-\infty}^{\infty} \exp\Big\{-\beta(1 + \lambda(1 - l(\bar{s}(k))))||x(k) - m(k)||^2\Big\} dx(k)$$

$$\times \exp\Big\{-\beta\{\frac{\lambda(1-l(\bar{s}(k)))}{1+\lambda(1-l(\bar{s}(k)))}\|y(k)-\bar{x}(k-1)\|^2+\gamma l(\bar{s}(k))+\nu\sum_{i\in N_k}\frac{1}{(k-i)^2}l(\bar{s}(k))l(\bar{s}(i))\}\Big\}, \tag{9}$$

where C is a constant, and $m(k)$ is defined as

$$m(k) \triangleq \frac{y(k)+\lambda(1-l(\bar{s}(k)))\bar{x}(k-1)}{1+\lambda(1-l(\bar{s}(k)))}. \tag{10}$$

Shortly it will be shown that $m(k)$ is indeed the mean of $X(k)$. Note that (10) is a linear combination of the observation and the estimate $\bar{x}(k-1)$, i.e., $m(k) = \alpha y(k) + (1-\alpha)\bar{x}(k-1)$, $0 \leq \alpha \leq 1$, where $\alpha \triangleq 1/\{1+\lambda(1-l(\bar{s}(k)))\}$. Finally noting in (9) that $X(k)$ is a Gaussian process with the mean $m(k)$ and the variance, $1/\{2\beta(1+\lambda(1-l(\bar{s}(k))))\}$, we obtain

$$\begin{aligned} Z_0 &\approx C\exp\Big[-\beta\sum_{k=1}^{N}\Big\{\frac{\lambda(1-l(\bar{s}(k)))}{1+\lambda(1-l(\bar{s}(k)))}\|y(k)-\bar{x}(k-1)\|^2+\gamma l(\bar{s}(k)) \\ &+ \nu\sum_{i\in N_k}\frac{1}{(k-i)^2}l(\bar{s}(k))l(\bar{s}(i))-\frac{1}{\beta}\log\sqrt{\frac{\pi}{\beta(1+\lambda(1-l(\bar{s}(k))))}}\Big\}\Big]. \end{aligned} \tag{11}$$

We can now derive $\bar{s}(k)$ and $\bar{x}(k)$ from Z_0. First, the mean field $\bar{x}(k)$ can be represented by the partition function:

$$\begin{aligned} \bar{x}(k) &= \int_{-\infty}^{\infty} x(k)\frac{1}{Z_0}\exp\{-\beta U_0(x,l|y)\}\,dx(k), \\ &= y(k)-\frac{1}{2}\nabla_{y(k)}F(\bar{x},\bar{s},y), \end{aligned} \tag{12}$$

where F is the *free energy*: $F \triangleq -\log Z_0/\beta$. As a result, substituting (11) into (12), we have

$$\bar{x}(k) = y(k)+\frac{\lambda(1-l(\bar{s}(k)))}{1+\lambda(1-l(\bar{s}(k)))}(\bar{x}(k-1)-y(k)). \tag{13}$$

Note that (??) is same as (10), and that (13) represents an estimate $\bar{x}(k)$ from $y(k)$ with an innovation $(\bar{x}(k-1)-y(k))$ and a variance $(1+\lambda(1-l(\bar{s}(k))))$ as normalization factor.

Finally, to obtain an equilibrium solution of $\bar{s}(k)$, we set

$$\frac{\partial\bar{s}(k)}{\partial\tau} = -\frac{\partial F(\bar{x},\bar{s},y)}{\partial\bar{s}(k)}. \tag{14}$$

Next putting (11) into (14) results in (15).

Therefore, we have the final result:

$$\left\{\begin{aligned} \bar{s}^{\tau+1}(k) &= \bar{s}^{\tau}(k)+\mu\beta l(\bar{s}^{\tau}(k))(1-l(\bar{s}^{\tau}(k)))\big\{\tfrac{\lambda}{(1+\lambda(1-l(\bar{s}^{\tau}(k))))^2}\|y(k)-\bar{x}^{\tau}(k-1)\|^2 \\ &\quad -\gamma-\nu\textstyle\sum_{i\in N_k}\tfrac{1}{(k-i)^2}\{l(\bar{s}^{\tau}(i))+\bar{s}^{\tau}(2k-i)\}+\tfrac{\lambda}{2\beta(1+\lambda(1-l(\bar{s}^{\tau}(k))))}\big\}, \\ \bar{x}^{\tau}(k) &= y(k)+\tfrac{\lambda(1-l(\bar{s}^{\tau}(k)))}{1+\lambda(1-l(\bar{s}^{\tau}(k)))}(\bar{x}^{\tau}(k-1)-y(k)). \end{aligned}\right. \tag{15}$$

Notice that fixed points of these equations are the mean field approximation of the MAP estimates of (1). The importance of this equation is the fact that it is actually dual to the Geiger–Girosi equation (6). The major difference is that which of $X(k)$ and $L(k)$ is pursued at the cost of the other.

As a neural network architecture to implement (15), we show a basic part for a time section k in Fig. 1. In this figure, the disks and boxes denote neurons and the solid disk represents a switch controlled by the line process $L(k)$.

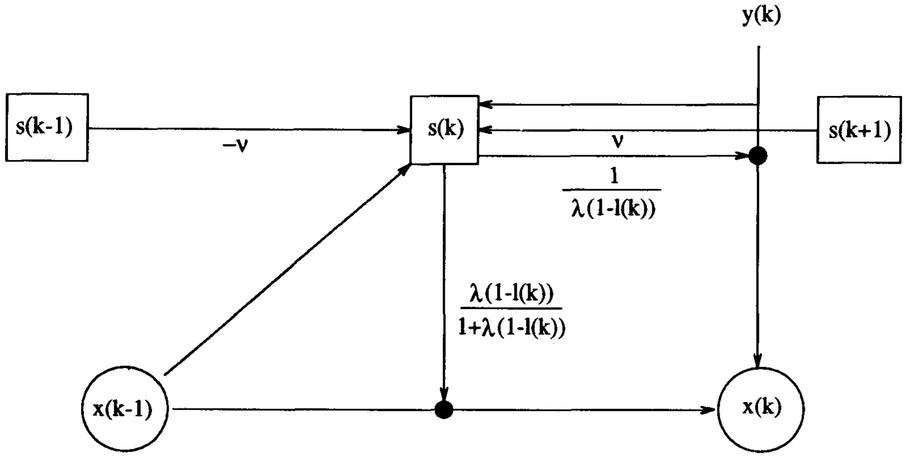

Fig. 1. A neural network at the section k; the neighborhood size $d = 1$ is considered.

IV. Experimental Results

As an application to speech segmentation, we will find phoneme–boundaries. To compare its performance, we chose the *forward-backward* algorithm [2] and the Geiger–Girosi algorithm as typical references among many other segmentation algorithms [6], [8], [9].

The training database consists of 602 phoneme segments and is used to determine the parameters for the three segmentation algorithms. After then, the second database, consisting of 2588 phoneme segments, is used to test the segmentation algorithms.

To begin with, we divide a large speech sample into possibly overlapping blocks of N frames and process each block in a batch. The last detected position of the previous segmentation result will become the starting position of the current N frames. For the purpose of simplicity, we consider only the case where $x(k)$ and $y(k)$ are represented in pth order LPC cepstrum. In this experiment all spectrum analysis are done using 256 point rectangular window spaced with half window size. Cepstrum analysis with $p = 16$ is carried out by the LPC Burg algorithm along with the cepstrum *liftering* [4].

One of the most hardest problems in the MRF framework is to track the Markov parameters. In our case, we estimated the parameters by trial and error. Typical parameters are $\gamma = 0.1$, $\lambda = 0.1$, and $\nu = 1.8$. Also the temperature $1/\beta$ is reduced by the schedule, $1/\beta(i) = T_0/(1+\log(i))$, $(i = 1, \cdots, 20)$. Here i is an annealing index and T_0 is an initial temperature. In addition, at each fixed temperature, (15) is updated up to 40 times($\tau = 1, 2, \cdots, 40$).

To compare our algorithm with the previous works, we implemented the *forward-backward* algorithm and the Geiger algorithm. Each algorithm is optimized so that it may give about the same number of segments as that of manual segmentation. In the *forward-backward* algorithm, we used $(\delta_v, \lambda_v) = (0.2, 400)$ and $(\delta_u, \lambda_u) = (0.8, 800)$ [2], so as to get phonemic units.

Some snapshot views of our algorithm are illustrated in Fig. 2. Note that the vertical bars in Fig. 2(a) shows the desired segmentation locations. As can be seen from Fig. 2(b) to 2(d), phoneme boundaries of abruptly changing points are detected first, and then less abrupt points appear. In this figures, the length of the vertical bars indicates the likelihood of boundaries.

The comparison with the *forward-backward* algorithm and the Geiger-Girosi algorithm is shown in Fig. 3. As can be seen from this figure, the new algorithm performs very well.

To estimate the performance quantitatively, we computed the miss and false-alarm rates and the position tolerance as shown in Fig. 4. We notice that the new algorithm performs well for any transitions such as CV, VV, VC, and CC on the average.

Table I shows the recognition rate at a typical position tolerance of ±42 msec. Here the

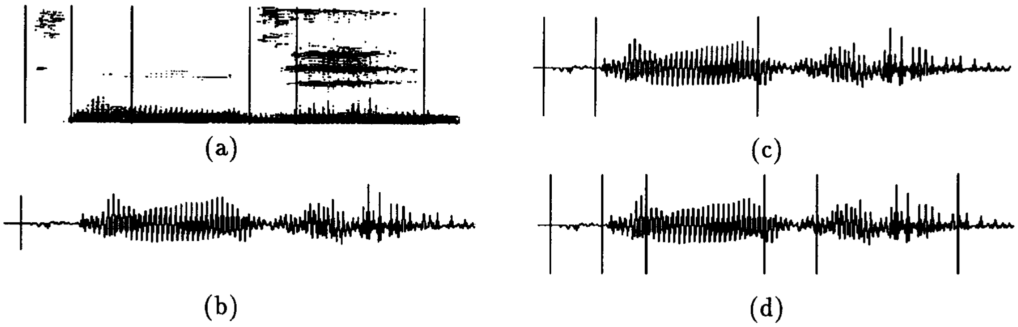

Fig. 2. Segmentation results (b)–(d) with the mannually segmented boundary (a). Here (b) results after 40 iteration($i = 1$), (c) after 80($i = 2$), (d) after 400($i = 10$), respectively.

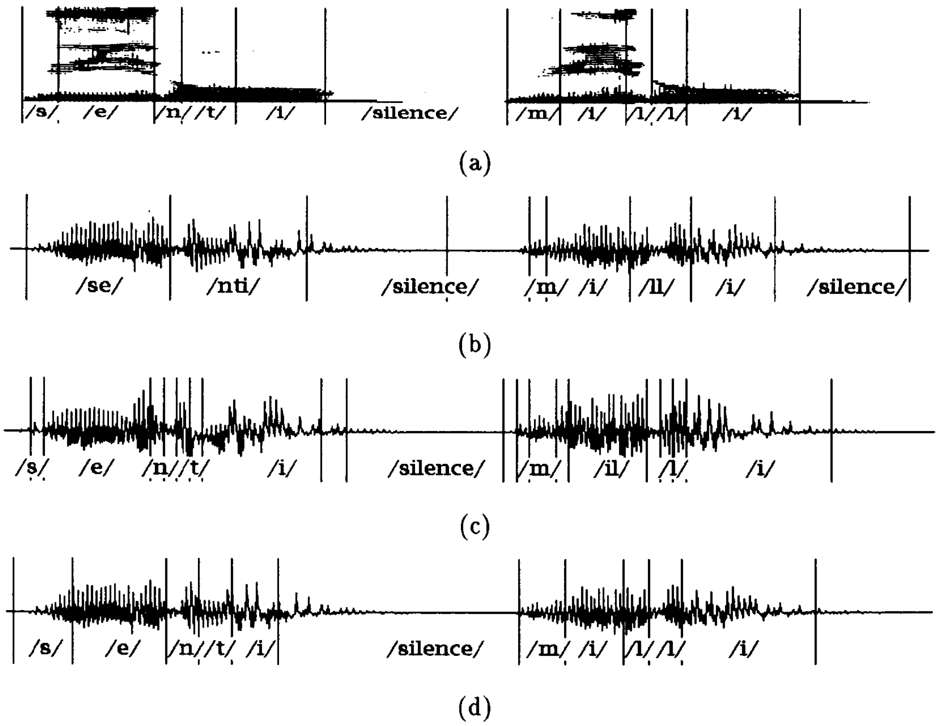

Fig. 3. Segmentation results: (a) a test speech spectrum, (b) the forward-backward algorithm, (c) Geiger-Girosi algorithm, and (d) the new algorithm.

performance index is the average of the *miss* and *false-alarm* rates, and the position tolerance of ±42 msec corresponds to $d + 1$ window rates(i.e., 512 samples). As can be seen from this table, the new algorithm shows about 5.3% and 33.8% better recognition rate than the *forward-backward* algorithm and the Geiger–Girosi mean field solution, respectively.

Finally the new algorithm is $\mathcal{O}(N^2)$ times faster for N speech frames than the forward-backward method achieving 0.75 phonemes boundaries per second by a 2.3 MFLOPS computer.

V. Conclusion

In conclusion, we derived a robust segmentation algorithm and its neural network architecture starting from the Geman algorithm and showed that it is actually dual to the Geiger–Girosi algorithm which is especially designed to signal restoration.

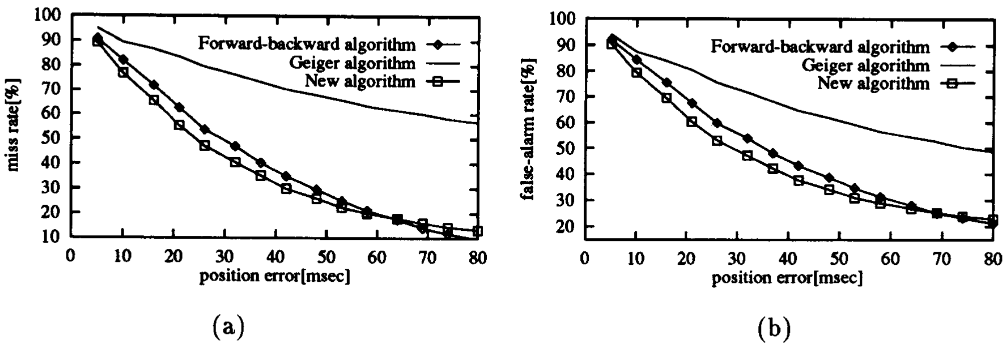

Fig. 4. The miss (a) and false-alarm rates (b) with respect to the position tolerance.

TABLE I

THE RATES OF MISSES AND FALSE-ALARMS WITHIN A ±42 MSEC POSITION TOLERANCE. THE PERFORMANCE INDEX IS THE AVERAGE OF THE MISS AND FALSE-ALARM RATES.

Recognition rate Algorithm	Miss	False-alarm	Performance index
Forward-backward algorithm	34.9%(903/2588)	43.3%(940/2171)	39.1%
Geiger algorithm	70.3%(1819/2588)	64.5%(1918/2973)	67.4%
New algorithm	29.9%(774/2588)	37.7%(1128/2991)	33.8%

Also the experimental results show that the new algorithm is better, about 5.3% – 33.8% in recognition rate, than the *forward-backward* algorithm and the Geiger–Girosi mean field solution.

REFERENCES

[1] A. Blake and A. Zisserman, *Visual Reconstruction*, Cambridge. MA: MIT Press, 1987.

[2] A. O. Regine, "A New Statistical Approach for the Automatic Segmentation of Continuous Speech Signals," *IEEE Trans. on ASSP*, Vol. 36, No. 1, Jan. 1988, pp. 29–40.

[3] A. Van Brandt, "Detecting and Estimating Parameter's Jumps Using Ladder Algorithms and Likelihood Ratio Test," *Proc. of ICASSP*, 1983, pp. 1017–1020.

[4] B. H. Juang, L. R. Labiner, and J. G. Wilpon, "On the Use of Bandpass Liftering in Speech Recognition", *IEEE Trans. on ASSP*, Vol. ASSP-35, No. 7, July 1987, pp. 947–1987.

[5] D. Geiger and F. Girosi, "Parallel and Deterministic Algorithms from MRF's: Surface Reconstruction," *IEEE Trans. on PAMI*, Vol. 13, No. 5, May 1991, pp. 401–412.

[6] Jan P. van Hemert, "Automatic Segmentation of Speech," *IEEE Trans. on SP*, Vol. 39, No. 4, April 1991, pp. 1008–1012.

[7] J. E. Besag, "Spatial Interaction and the Statistical Analysis of Lattice Systems," *J. Royal Statis. Soc., Ser. B 36*, 1974, pp. 192–236.

[8] M. Basseville and A. Basseville, "Sequential Detection of Abrupt Changes in Spectral Characteristics of Digital Signals," *IEEE Trans. on Information Theory*, Vol. IT-29, No. 5, Sept. 1983, pp. 709–724.

[9] R. J. Di Francesco, "Real-Time Speech Segmentation Using Pitch and Convexity Jump Models: Application to Variable Rate Speech Coding," *IEEE Trans. on ASSP*, Vol. 38, No. 5, May 1990, pp. 741–748.

[10] S. Geman and D. Geman, "Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images," *IEEE Trans. on PAMI*, Vol. 6, No. 6, Nov. 1984, pp. 721–741.

[11] Y. Zhao, et al., "Applying of the Gibbs Distribution to Hidden Markov Modeling in Speaker-Independent Isolated Word Recognition," *IEEE Trans. on SP*, Vol. 39, No. 6, June 1991, pp. 1291–1299.