Journal of Electronic Imaging 6(3), 286–293 (July 1997).

Modular expert network approach to histogram thresholding

Chun Hung Li
Peter K. S. Tam
Hong Kong Polytechnic University
Department of Electronic Engineering
Hung Hom, Hong Kong
E-mail: enptam@hkpucc.polyu.edu.hk

Abstract. *The problem of histogram thresholding is tackled using a modular expert network. The modular expert network is a network of expert modules modulated by a gating network. The expert modules incorporate individual experts' opinions on the thresholding problem. The difficult task of integration of conflicting experts' opinions is achieved through a training of the gating network using backpropagation. The resulting network achieves accurate modeling of the solution mapping through the efficient combination of existing experts. Experimental results show the superior performance of the modular network over classical algorithms. In particular, a near-optimal solution was shown to be achievable using a small training set. Application to a real-world biomedical cell segmentation problem is also given.* © *1997 SPIE and IS&T.* [S1017-9909(97)00603-X]

1 Introduction

In various applications of image processing such as template matching and morphological operations, the number of gray levels of the image often needs to be reduced. Such operations can be achieved efficiently through the use of the histogram thresholding operation. Thresholding is the segmentation of an image into different classes by comparing the gray level of a pixel with that of a set of thresholds. Bi-level thresholding is the simplest case where only one threshold is needed for segmenting an image into two classes. Due to its wide applications, many algorithms have been proposed for solving this problem. An in-depth analysis of the thresholding problem and a discussion of many thresholding algorithms can be found in the work of Haralick and Shapiro,[1] Sahoo *et al.*,[2] and Glasbey.[3] Lee *et al.*[4] gave a comparative performance study on several histogram thresholding algorithms along with contextual algorithms and give evaluations based on several criteria. They have come to the conclusion that different algorithms perform better under different criteria and more sophisticated algorithms need to be developed. In developing a better algorithm, it is observed that most of the thresholding algorithms make different inherent assumptions on the criteria for selecting the threshold. However, the relationship between these criteria and the segmentation result on a particular image is often unknown. Furthermore, the large variations shown in histograms of different images pose significant difficulties to the design of a good thresholding algorithm. It is often observed that a particular algorithm can work for some images while failing completely on others.[3,4]

In this paper, the histogram thresholding problem is tackled through the efficient use of existing algorithms and the learning capability of feed-forward networks. In the proposed modular expert network approach, each module is a classical expert algorithm and its output is modulated by a gating network. The network architecture is shown in Fig. 1. The employment of classical expert algorithms allows the integration of expert knowledge on problems that cannot be handled easily by simple network models. Since each expert's output may be close to some other expert's or may be different from others depending on the particular histogram, a modulation of the experts' outputs is needed to obtain the network output. The gating network achieves the integration of the experts' output by learning from teaching samples.

The modular expert network approach solves complex problems using the principle of ''divide and conquer,'' which often leads to simple and efficient algorithms. The idea of using a kind of modular network for learning was discussed in Nowlan *et al.*[5] Subsequently, an expectation maximization algorithm for training of a mixture of experts was investigated by Jordan and Jacobs.[6] Applications of the mixture of experts can also be found in Ref. 7. However, the expert modules in these approaches refer to generalized linear models that are more restrictive in many aspects, when compared with expert modules proposed in this paper.

In the subsequent sections, the modular network structure will be described in detail. The classical algorithms that form the modules will be selected. The training of the gating network that integrates the various modules will be demonstrated. Experimental results using simulated Gaussian mixtures will be given. Application to a real-world biomedical cell segmentation problem will also be investigated.

Paper NNT-06 received Jan. 11, 1997; revised manuscript received Feb. 25, 1997; accepted for publication Feb. 28, 1997.
1017-9909/97/$10.00 © 1997 SPIE and IS&T.

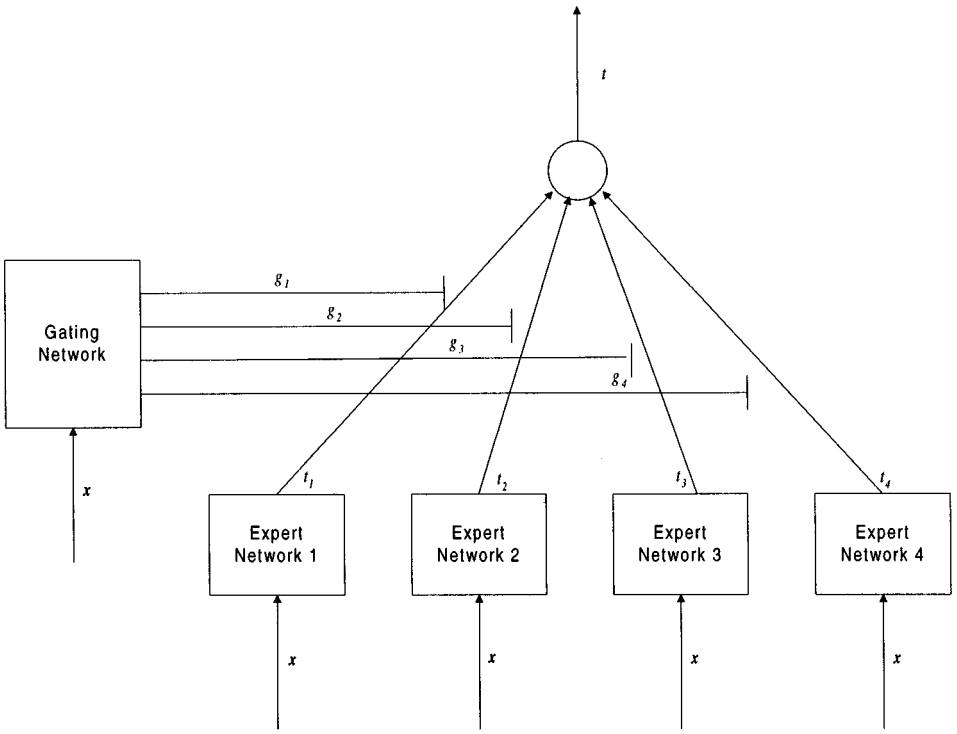

Fig. 1 Modular network.

2 Modular Expert Network

The modular expert network architecture proposed in this paper has a modified architecture when compared to the mixture-of-experts model of Jordan and Jacobs.[6] The expert network in Jordan's approach is linear with a single output nonlinearity with output given by

$$t_i = f(U_i \mathbf{x}), \tag{1}$$

where U_i is a weight vector and f is a fixed continuous nonlinearity. The use of generalized linear models does not allow easy integration of prior knowledge to the problems of thresholding, segmentation, etc. For example, the invariance in size and location properties are essential to various pattern recognition problems. Such an invariance can be efficiently captured using specific measures such as the invariant moments. However, neither the generalized linear model nor the simple feed-forward network can efficiently represent such information. Therefore, we propose to use modules with invariance properties by simply using existing classical algorithms, thus allowing the learning of much more complex mappings.

In Jordan's approach, training has to be applied to each component expert as well as to the gating network. In our proposed modular network approach, the expert network modules consist of nonlinear mappings that are predefined. Training only applies to the gating network, which controls the output of the individual expert network modules. The output of the network is given by

$$t = \sum_{i=1}^{N} g_i t_i, \tag{2}$$

where N is the number of expert modules. Since the expert modules are fixed, only the gating network need to be trained. The gating network can function as a softmax network corresponding to soft-split of the input space, or the gating network can be employed in a ''winner-take-all'' fashion, resulting in a hard partitioning of the input space. Hard partitioning of the input space is similar to the approach taken in classification and regression trees (CART).[8] The main differences between the proposed approaches and CART are the fixed network topology and the use of predefined experts.

Consider each expert module, having output t_i. Define an error criterion $E_x(t_i)$ of the output of each expert module. The gating network is trained with binary target values

$$g_i = \begin{cases} 1 & \text{if } E_x(t_i) = \min_i E_x(t_i) \\ 0 & \text{if } E_x(t_i) \geqslant \min_i E_x(t_i) \end{cases}. \tag{3}$$

The use of a ''winner-take-all'' scheme as the network output is based on the properties of the modular network. If the problem is close to regression problems with continuous output and low nonlinearity, a softmax gating network is more appropriate. If the problem is close to a classification problem with discrete output and high nonlinearity, it is in general more suitable to choose a ''winner-take-all'' scheme. The training of the gating network depends on teaching samples. The teaching samples can be obtained from histograms where optimal thresholds are known, either from manual inputs of human experts or from analytical derivations. However, in problems where the data possess high dimensionality, the choice of a finite set of teaching samples to adequately represent the input data is

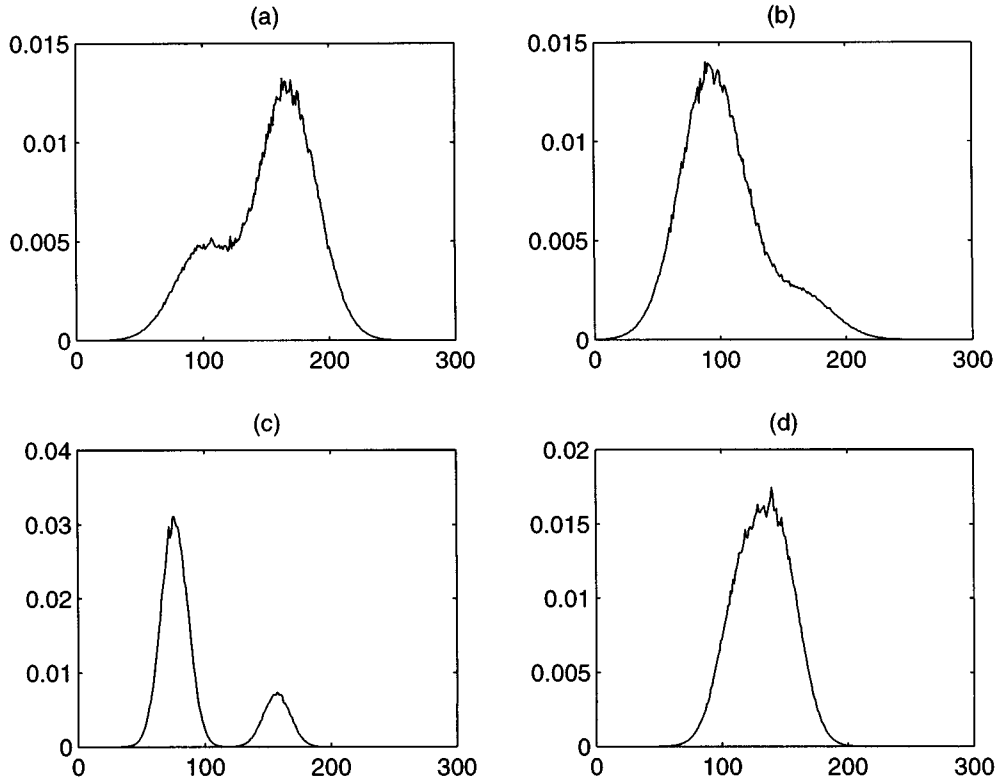

Fig. 2 Samples of synthetic histograms.

not a trivial task. In the next section, a histogram distribution model is introduced to allow the development of a systematic scheme for generating teaching samples, which gives good generalization to untrained testing samples. Furthermore, the distribution model enables quantitative evaluation of the network's performance through the use of statistical criteria of errors in thresholding.

3 Histogram Distribution Model

In this section, the histogram distribution model will be introduced as the framework for solving the histogram thresholding problem. The gray-level histogram can be approximated as a set of random realizations of a probability density function $h(x)$. The probability density function $h(x)$ can be modeled as a mixture of two probability distributions

$$h(x)=\rho_1 p_1(x)+\rho_2 p_2(x), \tag{4}$$

where ρ_1 and ρ_2 are the proportions of the two classes of objects, and $p_1(x)$ and $p_2(x)$ are the probability distributions of the two classes, respectively. The probability distributions can be modeled by standard distributions such as the Gaussian distribution or the Poisson distribution.

In this paper, the Gaussian distribution is chosen to illustrate the approaches. Similar derivations can also be obtained using other distributions. The validity of this choice is also justified by the central limit theorem, which states that the distribution of the sum of a large number of independent random variables will approach a normal distribution as the number of random variables increases.[9] The probability density functions $p_i(x)$, where $i = 1,2$, are thus

$$p_i(x)=\frac{1}{\sqrt{2\pi}\sigma_i}\exp\left(-\frac{(x-\mu_i)^2}{2\sigma_i^2}\right), \tag{5}$$

where μ_i and σ_i are the means and the standard deviations of the Gaussian distributions, respectively.

The observed histogram is only a realization of the density function, and such random realizations can be obtained by sampling N times the random variable with density function $h(x)$ for generating a histogram with N pixels. The methods for generating histograms by sampling various distributions can be found in Ref. 10. Alternatively, the observed histogram can be approximated by a signal dependent noise model.[11] Figure 2 shows some of the histograms generated with the observation model.

In order to compare the performances of different thresholding algorithms on the mixtures of Gaussians, the misclassification error is employed. In the case for a mixture of two Gaussians, the misclassification error when t is selected as the threshold is given by:

$$E(t)=\rho_2\int_{-\infty}^{t} p_2(x)dx+\rho_1\int_{t}^{\infty} p_1(x)dx. \tag{6}$$

In simulations, the discrete version of the above formula is implemented,

$$E(t)=\rho_2\sum_{x=0}^{x<t} p_2(x)+\rho_1\sum_{x=t}^{x=L} p_1(x). \tag{7}$$

With a total of N sample histograms, the average misclassification error for any chosen algorithm is defined as,

$$\overline{E}=\frac{\Sigma_j E_j}{N}, \tag{8}$$

where E_j is the misclassification error $E(t)$ for the j'th sample histogram.

The use of a single criterion is prone to bias of various kinds. An additional criterion, the accuracy, will be employed in order to make a fair comparison. The accuracy $A(t)$ is defined as the ratio between the minimum misclassification error and the misclassification error of the algorithm on the histogram,

$$A(t)=E(t_o)/E(t), \tag{9}$$

where $E(t_o)$ is the minimum misclassification error and t_o is the optimal threshold, defined by the threshold that minimizes the misclassification errors. This accuracy will take its maximum value of one when the threshold t selected by the thresholding algorithm gives the minimal possible classification error. With a total of N sample histograms, the average accuracy for any chosen algorithm is defined as

$$\overline{A}=\frac{\Sigma_j A_j}{N}, \tag{10}$$

where A_j is the accuracy for the j'th sample histogram. The optimal threshold t_o is the threshold that minimizes the error above criterion,

$$t_o=\arg\min_t E(t). \tag{11}$$

In designing the gating network for the thresholding application, the output layer of the network corresponds to the four gating outputs that modulate the four expert algorithms. The number of nodes in the input layer should be assigned according to the length of the histogram. However, a more efficient representation of the histogram data can be achieved through the use of invariant shape descriptors, i.e., variance, skewness, and kurtosis, which are the second-, third-, and fourth-order shape descriptors. One or two hidden layers can be employed in the application depending on the variations shown in the teaching and testing histograms.

In a supervised approach to thresholding, given h and t_o, we can design a network that is capable of regression such that the network would generate t that is close to t_o. However, the size of the histogram would imply the computationally infeasible training of a huge network.

4 Thresholding Algorithms as Experts

In the modular network approach to thresholding, classical algorithms will be incorporated to form expert modules. Such modules can be described by the following statistics of the histograms. The zeroth, first, and second moments of the foreground and background portions of the thresholded histogram are, respectively,

$$\begin{aligned} &m_{0a}=\Sigma_{i=0}^{t-1} h_1,\quad m_{0b}=\Sigma_{i=t}^{L-1} h_i,\\ &m_{1a}=\Sigma_{i=0}^{t-1} i h_i,\quad m_{1b}=\Sigma_{i=t}^{L-1} i h_i,\\ &m_{2a}=\Sigma_{i=0}^{t-1} i^2 h_i,\quad m_{2b}=\Sigma_{i=t}^{L-1} i^2 h_i. \end{aligned} \tag{12}$$

The mean and the standard deviations are defined as

$$\begin{aligned} &\mu_a=\frac{m_{1a}}{m_{0a}},\quad \mu_b=\frac{m_{1b}}{m_{0b}},\\ &\sigma_a=\frac{m_{2a}}{m_{0a}}-\mu_a^2,\quad \sigma_b=\frac{m_{2b}}{m_{0b}}-\mu_b^2. \end{aligned} \tag{13}$$

In Otsu's method,[12] the threshold is selected so as to maximize the class separability, which is based on the within-class variance, between-class variance, and total variance of gray levels. This method is nonparametric, unsupervised, and can be applied without *a priori* knowledge. This method has wide applicability and is often used as a standard algorithm with which other thresholding algorithms are compared.

$$n_{otsu}(t)=m_{0a}m_{0b}(\mu_a-\mu_b)^2. \tag{14}$$

In the minimum error method of Kittler and Illingworth,[13] the sets of pixels that are comprised of the object and the background are both assumed to be Gaussian distributed. A criterion function is constructed such that the threshold selected will minimize the average error in pixel classification.

$$n_{minerr}(t)=p_a\log\left(\frac{\sigma_a}{p_a}\right)+p_b\log\left(\frac{\sigma_b}{p_b}\right). \tag{15}$$

The maximum entropy method[14] selects the threshold that maximizes the entropy of the segmented portions of the histogram. The entropy of segmented portion is defined as

$$\begin{aligned} e_a&=\frac{1}{m_{0a}}\sum_{i=0}^{t-1} h_i\log(h_i)-\log(m_{0a}),\\ e_b&=\frac{1}{m_{0b}}\sum_{i=t}^{L-1} h_i\log(h_i)-\log(m_{0b}), \end{aligned} \tag{16}$$

$$n_{maxent}=e_a+e_b. \tag{17}$$

The minimum cross entropy method[15] selects the threshold that minimizes the entropy of the image and its segmented version. The criterion function is defined as

$$n_{croent}=-m_{1a}\log\left(\frac{m_{1a}}{m_{0a}}\right)-m_{1b}\log\left(\frac{m_{1b}}{m_{0b}}\right). \tag{18}$$

The target output of the modular expert network is given by

$$t = \arg\min E(t_i), \tag{19}$$

where t_i are the thresholds selected by the four algorithms.

5 Results and Discussions

In order to test the performance of the modular expert network, a small number of training histograms are generated to train the network and a large number of testing histograms are tested on the network. The small number of training histograms simulates typical situations where teaching data are limited. The large number of testing samples approximates the overall behavior of the network over as large a sample space as possible.

The training sample histograms are generated under the following conditions:

- ρ_1 is uniformly sampled from the interval (0.01, 0.99), $\rho_2 = 1 - \rho_1$
- μ_1 is uniformly sampled from the interval (71.5, 121.5)
- μ_2 is uniformly sampled from the interval (135.5, 185.5)
- σ_1 are uniformly sampled from the interval (5, 30), $\sigma_2 = \sigma_1$.

The testing histograms are generated under the following conditions:

- ρ_1 varies from 0.01 to 0.99 in steps of 0.01, $\rho_2 = 1 - \rho_1$
- σ_1 varies from 5 to 30 in steps of 0.252, $\sigma_2 = \sigma_1$
- μ_1 is uniformly sampled from the interval (71.5, 121.5)
- μ_2 is uniformly sampled from the interval (135.5, 185.5).

The testing samples consists of 99 different values of ρ and 99 different values of σ. For each set of values of ρ and σ, five sets of values for μ_1 and μ_2 are generated. Thus, the total testing sample set is comprised of 49005 ($99 \times 99 \times 5$) histograms.

The intervals for μ_1 and μ_2 are chosen such that μ_1 and μ_2 are separated from each other and away from the maximum and minimum gray values of 255 and 0. The intervals for standard deviations σ_1 and σ_2 are chosen to cover situations of minimal overlapping to high overlapping of gray levels between the foreground and the background. The proportions of the background against the foreground are in ratios ranging from 1:99 to 99:1, which should cover commonly occurring situations. The observed histogram noise is set as 0.01, which gives a moderate amount of noise to the histogram.

The performances of the different thresholding algorithms on these testing histograms are shown in Table 1 and Table 2. The results on the modular network approach were obtained by training the network with 1000 sample histograms. In order to compare the various algorithms' performances with the theoretical results, the lower bound of errors is calculated with the additional information of the individual distributions that generate the histograms. The lower bound on errors measures the amount of the overlap between the component distributions and thus represents the smallest misclassification error that can be achieved. The lower bound on errors can be calculated from the average of the minimum of misclassification errors in Eq. (7) using an exhaustive search. The upper bound on accuracy follows from its definition as a ratio as shown in Eq. (9).

Table 1 Average misclassification errors of different algorithms: (a) cross entropy, (b) maximum entropy, (c) minimum error, (d) Otsu's method, (e) proposed approach, and (f) lower bound on errors.

	(a)	(b)	(c)	(d)	(e)	(f)
$\bar{E}$(%)	12.2	6.2	11.7	11.5	4.9	4.7

Comparing the results on the average misclassification errors $\bar{E}$, the error of the neural network approach is significantly smaller than the error of any other major thresholding algorithms. In fact, the average misclassification error of the neural network approach is very close to the lower bound on the misclassification error.

Comparing the results on the average accuracy, the proposed method is very close to the optimal result, achieving an average of 96% accuracy. This average accuracy is much higher than the best performing classical algorithm, the minimum error method, which achieves an average of 69% accuracy. It is also of interest to note that the maximum entropy algorithm does not perform as well under the criterion of average accuracy than the average misclassification error. The reason is that the average misclassification error is easily dominated by sample histograms with large errors. The average accuracy defined in this paper normalizes the performances of thresholding algorithm so that each sample histogram has an even contribution to the average.

Another important property of the learning algorithm is the ability of the algorithm to generalize from a limited population of training samples. The modular network is trained with different numbers of training samples and tested with a large testing set. Figure 3 shows the average misclassification error of the network. The average error drops sharply with the first 50 training samples. There is further gradual improvement as the training samples increases to 1000 training samples. Figure 4 shows the average accuracy of the network. Similar to the results shown in the misclassification errors, only 50 to 100 training samples are needed to achieve a 90% average accuracy. The small training sample size required for training the network can be attributed to the random uniform sampling in parameter

Table 2 Average accuracy of different algorithms: (a) cross entropy, (b) maximum entropy, (c) minimum error, (d) Otsu, (e) proposed approach, and (f) upper bound on accuracy.

	(a)	(b)	(c)	(d)	(e)	(f)
$\bar{A}$(%)	58.0	61.0	69.0	62.5	96.1	100

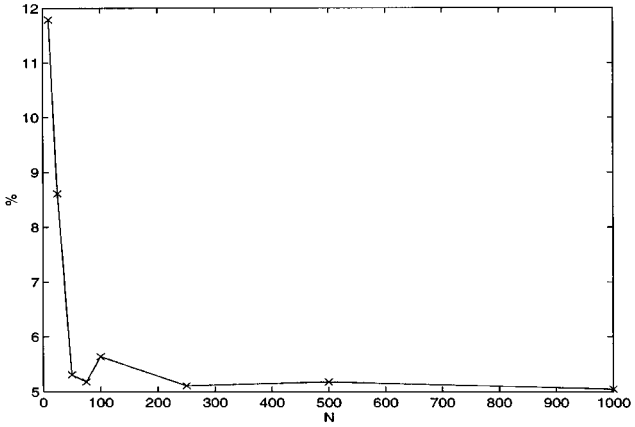

Fig. 3 Average misclassification error versus training sample size.

space of the Gaussian density. Such a sampling schedule generates a set of teaching samples with a wide spread of shapes and properties enabling a good generalization to be obtained.

5.1 *Applications to Real-World Images*

In this section, results are presented as an application of the modular network applied to the biomedical problem of quantitative measurement of cancer cells. Figure 5 shows two samples of cell images. The cell images represent tumor sections obtained from patients who have been *in situ* hybridized for tumor-related viruses. The gray values of the infected portion are approximately proportional to the amount of tumor-related virus. The aim of the analysis is to establish the amount of reaction in terms of the relative strength and the percentage of reacted population. Figure 6 shows the histograms of two images. Those gray values coming from the infected portion and the uninfected portion can be assumed to be approximated as Gaussian distributions. Each of the two portions is characterized by a different set of mean, proportion, and variance. In some cell samples, there are also white space regions where no cells reside. Such pixels can be easily removed from the histogram since they have distinctly high gray values.

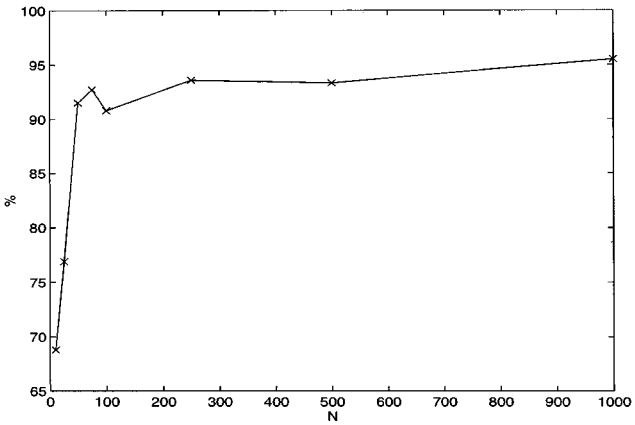

Fig. 4 Average accuracy versus training sample size.

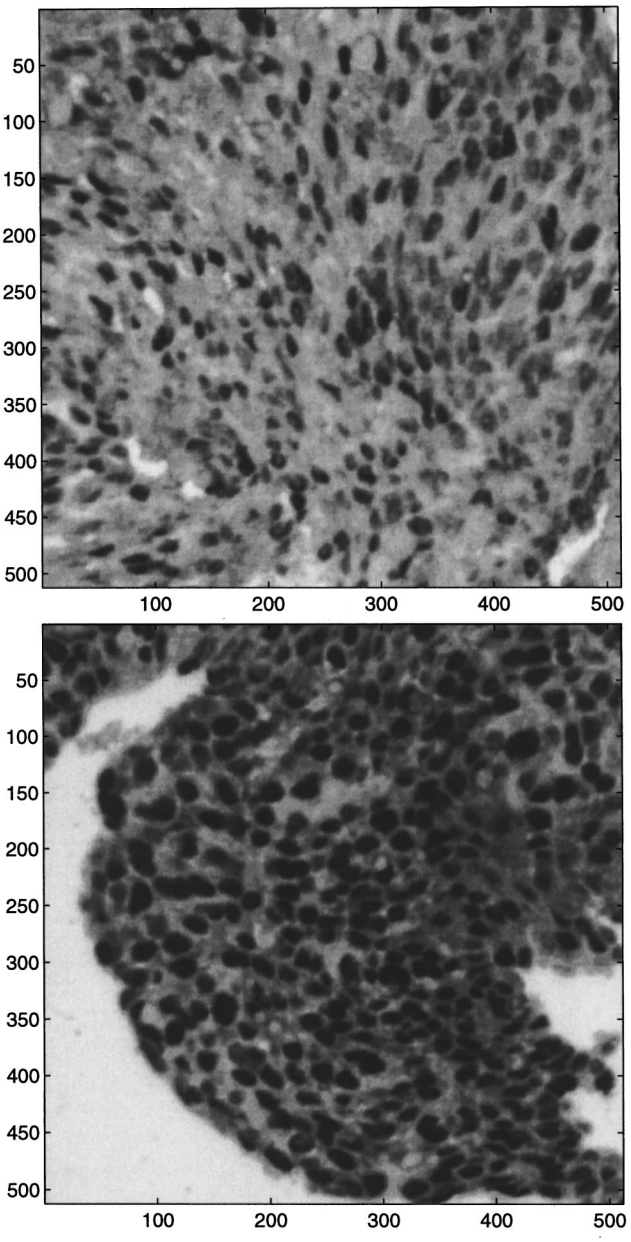

Fig. 5 Cell samples.

In theory, the gating network can be trained solely with cell histograms with thresholds selected manually by human experts. However, a good training set requires the selection of a large amount of data from a huge database to include a sufficient variety of samples for representation. Such a large database may not be available readily, and in any case, the selection and training processes are very time-consuming. Therefore, we propose to construct a hybrid training set using a combination of synthetic samples and actual samples. The synthetic samples are mixtures of Gaussian distributions selected to approximate a wide variety of histograms. These are used to complement the actual cell histograms for improving the generalization to untrained samples. To control the effect of the actual histogram as compared to the synthetic ones, we can include duplications of the actual samples in the training set. In the

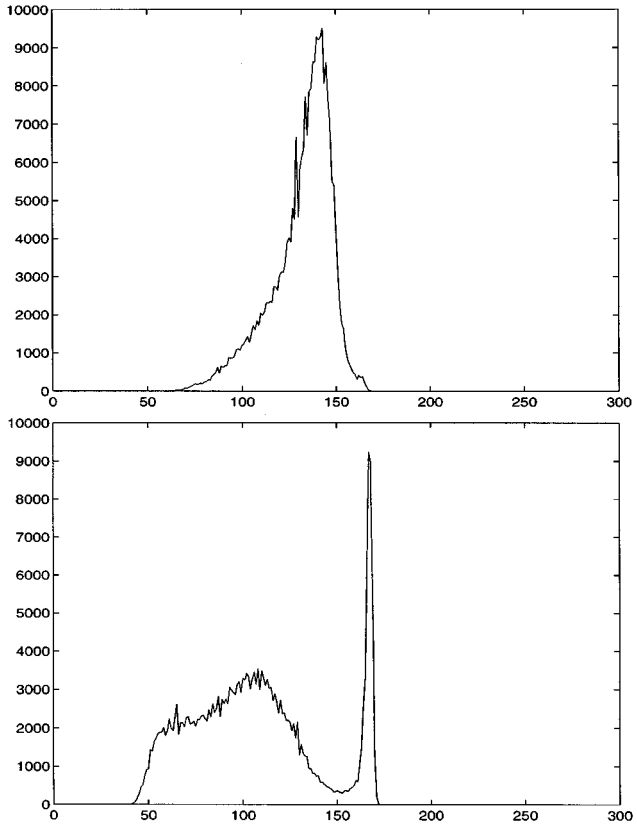

Fig. 6 Histograms of cell samples.

following investigations, 100 cell histograms are selected as the training samples and each of the cell histograms is duplicated ten times making a total of 1000 cell sample histograms. These 1000 cell histograms and 1000 synthetic Gaussian mixtures are supplied with shuffling as the training samples to the gating network. Figure 7 shows the result of the segmentation of the modular network. The nuclei and the cytoplasm are separated by black lines. The results agree with evaluation by human experts.

Further areas of applications include image thresholding in vision-based automated assembly lines and inspection systems. The ensembles of image samples in an assembly line can be modeled by specific distributions whose parameters are time-varying as a result of various factors: noise in sensors, mechanical tolerance, changes in ambient environment, imperfect workmanship. As the distribution is often unknown and time-varying, the use of the modular expert network can simplify the task of selection of algorithms and incremental adaptation to changes in the system can be easily incorporated using new teaching samples.

To conclude, the problem of histogram thresholding is tackled using a modular expert network in which classical thresholding algorithms are regarded as experts. The outputs from these experts are modulated by a trained gating network through teaching samples. For problems with *a priori* known distribution, teaching samples are obtained by sampling in the parameter space of the distribution model. For problems with real-world data sets, a hybrid training set consisting of samples from both the approximating distributions and the observed data sets are employed.

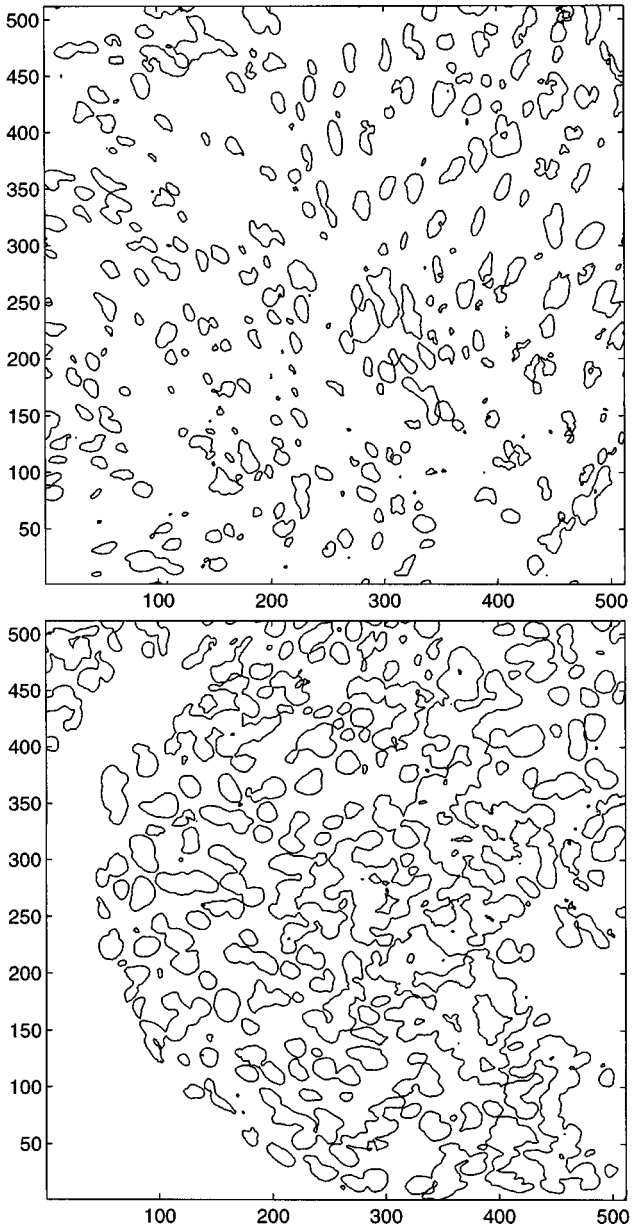

Fig. 7 Result of cell samples after thresholding by the modular network.

References

1. R. M. Haralick and L. G. Shapiro, *Computer and Robot Vision*, Addison-Wesley, Reading, MA (1992).
2. P. K. Sahoo, S. Soltani, and A. K. C. Wong, ''A survey of thresholding techniques,'' *Computer Vision, Graphics and Image Processing* **41**, 233–260 (1988).
3. C. A. Glasbey, ''An analysis of histogram-based thresholding algorithms,'' *CVGIP: Graphical Models and Image Processing* **55**(6), 532–537 (1993).
4. S. U. Lee, S. Y. Chung, and R. H. Park, ''A comparative performance study of several global thresholding techniques for segmentation,'' *Computer Vision, Graphics and Image Processing* **52**, 171–190 (1990).
5. S. J. Nowlan, R. A. Jacobs, M. I. Jordan, and G. E. Hinton, ''Adaptive mixtures of local experts,'' *Neural Computation* **3**, 79–87 (1991).
6. M. I. Jordan and R. A. Jacobs, ''Hierarchical mixtures of experts and the em algorithm,'' *Neural Computation* **6**, 181–214 (1994).
7. S. R. Waterhouse and A. J. Robinson, ''Classification using hierarchical mixtures of experts,'' in *Proc. 1994 IEEE Workshop on Neural Networks for Signal Processing IV*, pp. 177–186 (1994).
8. L. Breiman, J. Friedman, R. Olshen, and C. J. Stone, *Classification*

and Regression Trees, Wadsworth and Brooks/Cole, Belmont, CA (1984).
9. A. Papoulis, *Probability, Random Variables, and Stochastic Processes*, 3rd ed., McGraw-Hill, New York (1991).
10. S. Cho, R. Haralick, and S. Yi, ''Improvement of Kittler and Illingworth's minimum error thresholding,'' *Pattern Recognition* **22**(5), 609–617 (1989).
11. C. H. Li and P. K. S. Tam, ''Robustness analysis of histogram thresholding algorithms,'' in *Proc. Second Int'l. Conf. on Mechatronics and Machine Vision in Practice*, pp. 203–208 (1995).
12. N. Otsu, ''A threshold selection method from gray-level histogram,'' *IEEE Trans. Syst. Man Cybern.* **9**, 62–66 (1979).
13. J. Kittler and J. Illingworth, ''Threshold selection based on a simple image statistics,'' *Computer Vision, Graphics and Image Processing* **30**, 125–147 (1985).
14. A. K. C. Wong and P. K. Sahoo, ''A gray-level threshold selection method based on maximum entropy principle,'' *IEEE Trans. Syst. Man Cybern.* **19**, 866–871 (1989).
15. C. H. Li and C. K. Lee, ''Minimum cross entropy thresholding'' *Pattern Recognition* **26**, 617–625 (1993).

Chun Hung Li received his PhD in electronic engineering from the Hong Kong Polytechnic University in 1996. He is currently a research associate at the Department of Electronic Engineering in the Hong Kong Polytechnic University. His research interests include stochastic image models, image analysis, and pattern recognition.

Peter K. S. Tam received his BE, ME, and PhD degrees in 1971, 1973, and 1976, respectively, all in electrical engineering, from the University of Newcastle, Australia. From 1967 to 1980, he held a number of industrial and academic positions in Australia. In 1980, he joined the Hong Kong Polytechnic University as a senior lecturer. He is now an associate professor in the Department of Electronic Engineering, Hong Kong Polytechnic University. Dr. Tam is a member of the IEEE and has participated in organizing a number of conferences. His research interests include signal processing, automatic control, fuzzy systems, and neural networks.