MAXIMUM LIKELIHOOD NEURAL NETWORK BASED ON THE CORRELATION AMONG NEIGHBORING PIXELS FOR NOISY IMAGE SEGMENTATION

Thanh Minh Nguyen — nguyen1j@uwindsor.ca
Q.M. Jonathan Wu — jwu@uwindsor.ca
Electrical Engineering, University of Windsor, 401 Sunset , Windsor, ON, N9B3P4, Canada

ABSTRACT

In this paper, we will present a new algorithm which is extended from the standard Gaussian mixture model to segment the noisy image based on the correlation among neighboring pixels. Firstly, we use the correlation between each centre pixel and its neighboring pixels in 3x3 window in building the prior probability, and this centre pixel is used to construct the conditional density function. Finally, to estimate the posterior probabilities of each pixel, instead of using expectation maximization algorithm as usual, we present a new maximum likelihood neural network (MaxNet) to optimize the parameters by using the error back propagation. Extensive experimental results illustrate the better performance compared to mixture model based on Markov random fields.

Index Terms—Maximum Likelihood, Noisy Image Segmentation, Neural Network, and Standard Gaussian mixture model.

1. INTRODUCTION

Segmentation of images has found widespread applications in image processing and image recognition systems. A correct segmentation result provides more information for diagnosis. However, images corrupted by high levels of noise which results in hard work for accurate image segmentation. Many previous works have been carried out on image segmentation, in particular by the method of threshold [1]. However, it is not an easy task to choose a threshold value. This is also the large disadvantage to this algorithm. A bad choice of these thresholds could alternate the quality of the segmentation and leads to a worse interpretation.

The other idea developed in image segmentation is to use an artificial neural network in order to avoid this disadvantage. In [2], the authors clustered feature vectors extracted from an image using a neural network which minimized the distance between the feature vectors. Although this approach worked well in the examples shown, it led to sub-optimal image segmentation. This is because pixels in general are spatially correlated and the approach presented in this method did not incorporate any spatial information.

Current research into image segmentation has tended to produce algorithms which rely on model-based image segmentation techniques. Among the model-based techniques, standard Gaussian mixture model [3] based clustering method is a popular segmentation algorithm. In this approach, pixels are considered random variable whose possibility density function is a Gaussian mixture. However, the spatial relationship between neighboring pixels is not taken into account. So, the segmentation result is sensitive to noise. To make the segmentation result is not sensitive with noise, the Markov random fields and the Gibbs random fields [4], [5], [6] have received great attention for modeling and processing image data. Although this approach worked well in noisy image segmentation, it is too complex.

Our approach differs from those discussed above by the following statements. Firstly, we use the standard Gaussian mixture model in our proposed operator. Therefore, the image segmentation detector is simple. Secondly, it is well known that neighboring pixels in an image are similar in some sense. Hence, by using the local spatial interactions between neighboring pixels, our system is worked very well in noisy image. Thirdly, we use the neural network to optimize the parameters. Here, the excellent ability to learn the parameters is the advantage that our system has received from the neural networks. Finally, unlike method of threshold where decision thresholds have to be specified, our current operator represents the decision by comparing the probabilities of neighboring pixels without a threshold.

This paper consists of five parts. In section 2, the segmentation algorithm for noisy image is presented. Section 3 will show the structure of our proposed neural network. Learning algorithms for our system will be presented in Section 4. In section 5, we show experimental results. Finally, the conclusions are given in section 6.

2. MAXIMUM LIKELIHOOD NEURAL NETWORK BASE ON THE CORRELATION AMONG NEIGHBORING PIXELS

In this paper, a 3x3 window is scanned through the image.

Suppose that C, K denote the number of classes (segments), number of 3x3 window in the grayscale image, respectively. Classes will be denoted by $\Omega_1,\Omega_2,\ldots,\Omega_C$. The k-th window will be denoted by $x_k=(x_{1k}\ x_{2k}\ x_{3k};\ x_{4k}\ x_{5k}\ x_{6k};\ x_{7k}\ x_{8k}\ x_{9k})^T$. Where x_{5k} is the central pixel of the k-th window and x_{nk} (n=1,…,9; $n\neq 5$) are called the neighboring pixels of the x_{5k}.

For each window x_k, the segmentation problem can now be stated in a statistical framework: if we knew its posterior probability $p(\Omega_j|x_k)$ for all classes, a good decision could be made. That is "the centre pixel x_{5k} will belong to the class with the largest posterior probability".

$$\text{we assign } x_{5k} \text{ to class } \Omega_j \text{ IF } p(\Omega_j \mid x_k) \geq p(\Omega_c \mid x_k); \quad \forall j,c = 1,2,\ldots,C;\ j \neq c \tag{1}$$

where the posterior probability $p(\Omega_j|x_k)$ can express using Bayes' theorem in the form.

$$p(\Omega_j \mid x_k) = \frac{p(x_k \mid \Omega_j)\,p(\Omega_j)}{p(x_k)} \tag{2}$$

and $p(x_k)$ is given by

$$p(x_k) = \sum_{j=1}^{C} p(x_k \mid \Omega_j)\,p(\Omega_j) \tag{3}$$

$p(\Omega_j)$ is the prior probability of the data point having been generated from component Ω_j of the mixture. These priors are chosen to satisfy the constraints [7].

$$\sum_{j=1}^{C} p(\Omega_j) = 1; \quad 0 \leq p(\Omega_j) \leq 1 \tag{4}$$

Similarly, the component density functions $p(x_k|\Omega_j)$ are satisfy the following condition.

$$\int p(x_k \mid \Omega_j)\,dx_k = 1 \tag{5}$$

In this paper, we proposed the individual component densities are given by the following Gaussian distribution functions

$$p(x_k \mid \Omega_j) = \frac{1}{\sqrt{2\pi b_j^2}} \exp\left(-\frac{(x_{5k} - c_j)}{2b_j^2} \right) \tag{6}$$

where the parameters b_j and c_j will discuss in section 4.

In the standard Gaussian mixture model, the spatial relationship between neighboring pixels is not taken into account. So, the segmentation result is sensitive to noise. In this paper, a new approach is presented to fix this problem. The prior probability $p(\Omega_j)$ not only carried out subject to the constraints in (4) but also presented by the correlation between neighboring pixels.

$$p(\Omega_j) = \frac{\exp\left(-\sum_{i;\,i\neq 5}^{9} \alpha_{ij}(x_{5k} - x_{ik}) \right)}{\sum_{j=1}^{C}\left[\exp\left(-\sum_{i;\,i\neq 5}^{9} \alpha_{ij}(x_{5k} - x_{ik}) \right) \right]} = \frac{h(\Omega_j)}{\sum_{j=1}^{C} h(\Omega_j)} \tag{7}$$

the parameters α_{ij}, i = (1,2,…,9; $i\neq 5$), j = (1,2,…,C) are value that characterize the prior probability. The mixture density contains the following adjustable parameters: b_j, c_j and α_{ij} to maximum the likelihood function. The negative log-likelihood [7], [8] for the data set is given by

$$E = -\ln L = -\sum_{k=1}^{K} \ln\left\{ \sum_{j=1}^{C} p(x_k \mid \Omega_j)\,p(\Omega_j) \right\} \tag{8}$$

which can be regarded as an error function. Until now, maximizing the likelihood L is then equivalent to minimizing E. To minimize the error function E with respect the parameters b_j, c_j and α_{ij}, we propose a new maximum likelihood neural network in section 3.

3. THE STRUCTURE OF MAXNET

So far, our discussion has focused on probability estimation

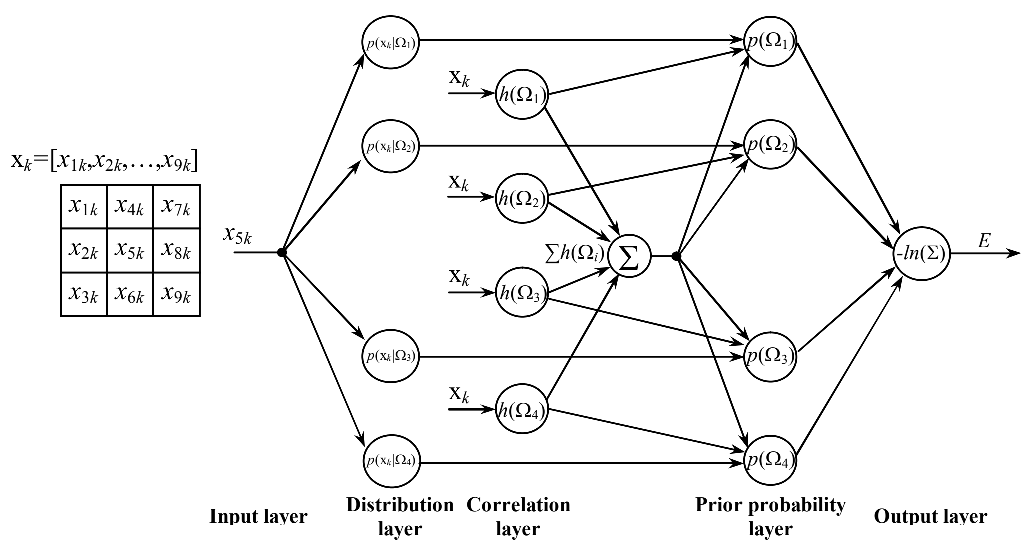


Figure 1: Schematic diagram of our maximum likelihood neural network (MaxNet) with 4 class (C=4).

in (1) and (2) to determine which class Ω_j the centre pixel x_{5k} should be assigned. To generalize the posterior probability $p(\Omega_j|\mathrm{x}_k)$, we need to adjustable parameters b_j, c_j and α_{ij} to maximum the likelihood function L or minimize the error function E with these parameters in Eq.(8). In this section, we propose MaxNet that can minimize the error function E in Eq.(8) by adjusting parameters by the learning algorithm. The proposed MaxNet structure consists of two visible layers (input and output layer) and three hidden layers (distribution layer, correlation layer, prior probability layer) as shown in Figure 1.

Input layer: this layer only transmits to the next layer directly. No computation is done in this layer. The output of each node in this layer is $O_{1n} = x_{nk}$ (n=1,2,…,9).

Distribution layer: Each node in this layer has 1 input from x_{5k}, and feeds its output to the node of the prior probability layer. We use the Gaussian distribution defined in Eq.(6) for each node in this layer. Here the parameters b_j, c_j (j=1,2,…,C) are constants that characterize the value of the component density functions $p(\mathrm{x}_k|\Omega_j)$. The optimal values of these parameters are determined by training to minimize the error function in (8), which will be discussed later. There are C nodes in this layer, each node has 2 parameters. The output of each node in this layer is $O_{2j} = p(\mathrm{x}_k|\Omega_j)$.

Correlation layer: There are C nodes of this layer. Each node in this layer represents the correlation between the centre pixels and its neighboring pixels. The fan-in of a node comes from x_k in the input layer. We choose the following function for each node $h(\Omega_j)$.

$$O_{3j} = h(\Omega_j) = \exp\left(-\sum_{i;i\neq 5}^{9}\alpha_{ij}(x_{5k} - x_{ik})\right) \tag{9}$$

The output of each node in this layer is $O_{3j} = h(\Omega_j)$. The parameters α_{ij}, i = (1,2,…,9;i≠5), j = (1,2,…,C) are discussed in section 4.

Prior probability layer: This layer performs the normalization operation. There are C nodes of this layer. The output O_{4j} of each node represents Eq.(7).

$$O_{4j} = O_{2j}\frac{O_{3j}}{\sum_{j=1}^{C} O_{3j}} \tag{10}$$

Output layer: this layer determines the negative log-likelihood (the error function E) in Eq.(8). There is 1 nodes of this layer. It also the value factor that we want to minimize.

$$O_{5j} = -\ln\left(\sum_{j=1}^{C} p(\mathrm{x}_k \mid \Omega_j)p(\Omega_j)\right) \tag{11}$$

Eq.(11) presents the error function in (8) for each input pattern x_k. Therefore, minimizing E in Eq.(8) is then equivalent to minimizing the output of ProNet in Eq.(11).

4. PARAMETERS LEARNING FOR MAXNET

To optimize the system performance, a two-phased hybrid parameter learning algorithm is applied with a given network structure. In hybrid learning, each iteration is composed of a forward pass and a backward one. In the forward pass, after the input vector is presented, we calculate the node outputs in the network. And then, we can compute the error for data pairs. In the backward pass the error signal propagates from the output towards the input nodes; the gradient vector is calculated and the nonlinear parameters updated by steepest descent method.

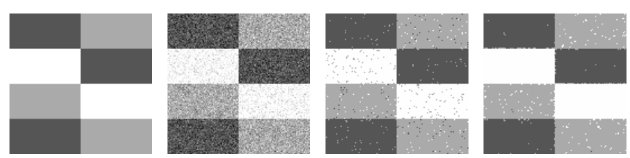
$$b_j(t+1) = b_j(t) - \eta\frac{\partial E}{\partial b_j};\ c_j(t+1) = c_j(t) - \eta\frac{\partial E}{\partial c_j} \tag{12}$$

$$\alpha_{ij}(t+1) = \alpha_{ij}(t) - \eta\frac{\partial E}{\partial \alpha_{ij}}$$

The learning step η of the nonlinear parameter update is adjusted using an adaptive approach. This process is repeated many times until the system converges. Detail descriptions of the parameter-tuning method can be found in [9], [10].

5. SIMULATION STUDIES

To test the effectiveness of our algorithm, we have obtained excellent results for many images. The results of checking our algorithm compared to mixture model based on Markov random fields in [5].



(a) (b) (c) (d)

Figure 2: 128x128 testing data, (a) Original image, (b) Corrupted original image with Gaussian noise (0 mean, 0.01 variance), (c) Mixture model method based on Markov random fields in [5] after 20 iterations, (d) Our proposed method.

Figure 2a shows the images with 3 classes used for the first simulation. This 128x128 pixel images can easily be generated by computer. Each square box in this image has a size of 32-by-64 pixels and the 2048 pixels contained within each box have the same luminance value (1/3, 2/3, 1). The image shown in Figure 2b is made from the original image by corrupting with Gaussian noise (0 mean, 0.01 variance). Figure 2c shows the final image using the method in [5] after 20 iterations. This method demonstrates a good performance with PSNR = 25.05dB. However, our method in Figure 2d achieves a higher PSNR = 26.93dB

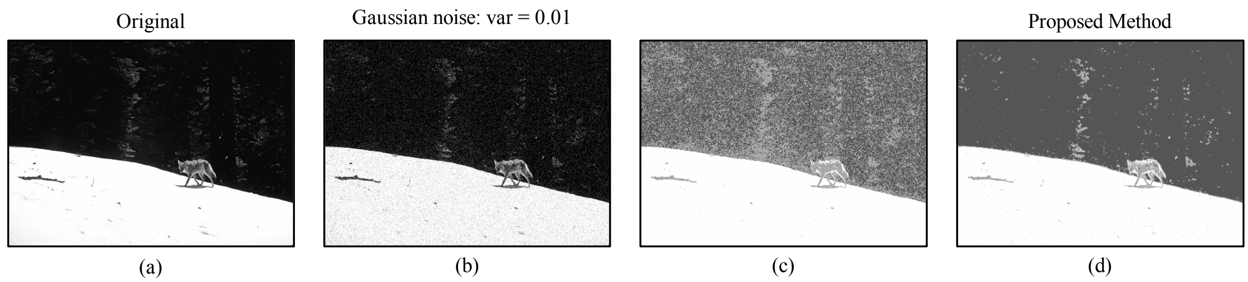


Figure 3 : 481x321 grayscale testing data, (a) Original image, (b) Corrupted original image with Gaussian noise (0 mean, 0.01 variance), (c) Mixture model method in [5] after 20 iterations, (d) Our proposed method.

after 10 iterations.

To emphasize the advantages of our proposed system compared with other. In the second experiment, the images from the Berkeley Segmentation Dataset (http://www.eecs.berkeley.edu/Research/Projects/CS/vision/grouping/segbench/) is used for testing. Figure 3a shows the images that we want to segment into 3 classes (Snow, Wolf, Tree). This image is corrupted by Gaussian noise (0 mean, 0.01 variance) as shown in Figure 3b. Clearly, our proposed system as shown in Figure 3d after 10 iterations could not only successfully segment the image when it is significantly degraded by high noise, but could also makes the effect of noise on the final segment image much less than the method in [5] as shown in Figure 3c after 20 iterations. The final parameters of our proposed method is shown in table 1.

Table 1: The final parameter of our proposed method for second simulation.

	j=1	j=2	j=3
b_j	0.2986	0.1416	0.2552
c_j	0.2005	0.0343	0.9530
α_{1j}	0.7612	0.0472	0.4394
α_{2j}	0.8815	0.0609	0.4279
α_{3j}	0.7709	0.0490	0.3749
α_{4j}	0.7881	0.0575	0.4233
α_{6j}	0.8213	0.0751	0.3837
α_{7j}	0.7317	0.0421	0.3965
α_{8j}	0.8380	0.0579	0.4627
α_{9j}	0.8043	0.0618	0.3661

6. CONCLUSION

In this paper, we propose present a new neural network extended from the standard Gaussian mixture model to segment the noisy image based on the correlation among neighbor pixels. Although we have shown only two experimental results of the proposed models, the MaxNet's ability has been sufficiently clarified by its high correct classification.

ACKNOWLEDGEMENTS

This research has been supported in part by the Canada Research Chair Program, AUTO21 NCE, and the NSERC Discovery grant.

REFERENCES

[1] Ritter G. X., "*Handbook of computer vision algorithms in image algebra*," Boca Raton, CRC Press, pages 125 - 141 and 271 - 273. 1996.

[2] S. C. Amartur, D. Piraino, and Y. Takefuji, "*Optimization neural networks for the segmentation of magnetic resonance images*," IEEE Transactions on Medical Imaging, vol. 11, pp. 215-220, 1992.

[3] G. McLachlan and D. Peel, "*Finite Mixture Models*," New York: Wiley,Nov. 2000.

[4] Blekas K., Likas A., Galatsanos N.P., Lagaris I.E., "*A spatially constrained mixture model for image segmentation*," IEEE Transactions on Neural Networks, Volume 16, Issue 2, March 2005 Page(s):494 - 498.

[5] Diplaros A., Vlassis N., Gevers T., "*A Spatially Constrained Generative Model and an EM Algorithm for Image Segmentation*," IEEE Transactions on Neural Networks, Volume 18, Issue 3, May 2007 Page(s):798 - 808.

[6] Sanjay Gopal S., Hebert T.J., "*Bayesian pixel classification using spatially variant finite mixtures and the generalized EM algorithm*," IEEE Transactions on Image Processing, Volume 7, Issue 7, July 1998 Page(s):1014 - 1028.

[7] C.M. Bishop, "*Neural Networks for Pattern Recognition*," Oxford University Press, Walton Street, Oxford, 1995, Page(s): 59 - 71.

[8] C.M. Bishop, "*Pattern Recognition and Machine Learning*," Springer, 2006.

[9] J.R. Jang, C. Sun, and E. Mizutani, "*Neuro-Fuzzy and Soft Computing*," Prentice-Hall, Englewood Cliffs, NJ, 1997.

[10] G. E. Hinton, "*Connectionist learning procedure*," Artificial Intelligence, vol. 40, no. 1, pp. 143-150, 1989.