Jinwen Ma* and Lei Li

[illegible], 100871, [illegible]
jwma@math.pku.edu.cn

Abstract. [illegible]

Keywords: [illegible]

* [illegible]

D.-S. Huang et al. (Eds.): ICIC 2008, LNCS 5226, pp. 569–576, 2008.

principal component of the covariance matrix of certain flat Gaussian of black pixels since the number of black pixels along it is always limited in the image. In such a way, these BYY harmony learning algorithms can learn the Gaussians from the image data automatically and detect the straight lines with the major principal components of their covariance matrices. On the other hand, from the BYY harmony learning on the mixture of experts in [16], a gradient learning algorithm was already proposed for the straight line or ellipse detection, but it was applicable only for some simple cases.

In this paper, we apply the fixed-point BYY harmony learning algorithm [15] to learning an appropriate number of Gaussians and utilize the major principal components of the covariance matrices of these Gaussians to represent the straight lines in the image. It is demonstrated well by the experiments that this fixed-point BYY harmony learning approach can e ciently determine the number of straight lines and locate these straight lines accurately in a binary image.

For the Gaussian mixture modeling, there have been several statistical learning algorithms, including the EM algorithm [17] and the k-means algorithm [18]. However, these approaches require an assumption that the number of Gaussians in the mixture is known in advance. Unfortunately, this assumption is practically unrealistic for many unsupervised learning tasks such as clustering or competitive learning. In such a situation, the selection of an appropriate number of Gaussians must be made jointly with the estimation of the parameters, which becomes a rather di cult task [19].

In fact, this model selection problem has been investigated by many researchers from di erent aspects. The traditional approach was to choose a best number k^* of Gaussians in the mixture via certain model selection criterion, such as Akaike's information criterion (AIC) [20] and the Bayesian Information Criterion (BIC) [21]. However, all the existing theoretic selection criteria have their limitations and often lead to a wrong result. Moreover, the process of evaluating a information criterion or validity index incurs a large computational cost since we need to repeat the entire parameter estimation process at a large number of di erent values of k. In the middle of 1990s, there appeared some stochastic approaches to infer the mixture model. The two typical approaches are the methods of reversible jump Markov chain Monte Carlo (RJMCMC) [22] and the Dirichlet processes [23], respectively. But these stochastic simulation methods generally require a large number of samples with di erent sampling methods, not just a set of sample data. Actually, it can e ciently solved through the BYY harmony learning on a BI-architecture of the BYY learning system related to the Gaussian mixture. Given a sample data set $\mathcal{S} = \{x_t\}_{t=1}^N$ from a mixture of k^* Gaussians, the BYY harmony learning for the Gaussian mixture modeling can be implemented by maximizing the following harmony function:

$$J(\Theta_k) = \frac{1}{N}\sum_{t=1}^{N}\sum_{j=1}^{k}\frac{\alpha_j q(x_t \mid \theta_j)}{\sum_{i=1}^{k}\alpha_i q(x_t \mid \theta_j)} ln[\alpha_j q(x_t \mid \theta_j)] \tag{4}$$

where $q(x \mid \theta_j)$ is a Gaussian mixture density given by Eq.(2).

For implementing the maximization of the harmony function, some gradient learning algorithms as well as an annealing learning algorithm were already established in [11,12,13,14]. More recently, a fast fixed-point learning algorithm was proposed in [15]. It was demonstrated well by the simulation experiments on these BYY harmony learning algorithms that as long as k is set to be larger than the true number of Gaussians in the sample data, the number of Gaussians can be automatically selected for the sample data set, with the mixing proportions of the extra Gaussians attenuating to zero. That is, these algorithms owns a favorite feature of automatic model selection during the parameter learning, which was already analyzed and proved for certain cases in [24]. For automatic straight line detection, we will apply the fixed-point BYY harmony learning algorithm to maximizing the harmony function via the following iterative procedure:

$$\alpha_j^+ = \frac{\sum_{t=1}^N h_j(t)}{\sum_{i=1}^i \sum_{t=1}^N h_i(t)}; \tag{5}$$

$$m_j^+ = \frac{1}{\sum_{t=1}^N h_j(t)} \sum_{t=1}^N h_j(t) x_t; \tag{6}$$

$$\Sigma_j^+ = \frac{1}{\sum_{t=1}^N h_j(t)} \sum_{t=1}^N h_j(t)(x_t - \hat{m}_j)(x_t - \hat{m}_j)^T, \tag{7}$$

where $h_j(t) = p(j|x_t) + \sum_{i=1}^k p(j|x_t)(\delta_{ij} - p(j|x_t)) ln[\alpha_i q(x_t|m_i, \Sigma_i)]$, $p(j|x_t) = \alpha_j q(x_t|m_j, \Sigma_j) / \sum_{i=1}^k \alpha_i q(x_t|m_i, \Sigma_i)$ and δ_{ij} is the Kronecker function. It can be seen from Eqs (5)-(7) that the fixed-point BYY harmony learning algorithm is very similar to the EM algorithm for Gaussian mixture. However, since $h_j(t)$ introduces a rewarding and penalizing mechanism on the mixing proportions [13], it di ers from the EM algorithm and owns the favorite feature of automated model selection on Gaussian mixture.

2.2 The Proposed Approach to Automatic Straight Line Detection

Given a set of black points or pixels $\mathcal{B} = \{x_t\}_{t=1}^N$ ($x_t = [x_{1,t}, x_{2,t}]^T$) in a binary image, we regard the black points along each line as one flat Gaussian distribution. That is, those black points can be assumed to be subject to a 2-dimensional Gaussian mixture distribution. Then, we can utilize the fixed-point BYY harmony learning algorithm to estimate those flat Gaussians and use the major principal components of their covariance matrices to represent the straight lines as long as k is set to be larger than the number k^* of the straight lines in the image. In order to speed up the convergence of the algorithm, we set a threshold value $\delta > 0$ such that as soon as the mixing proportion is lower than δ, the corresponding Gaussian will be discarded from the mixture.

With the convergence of the fixed-point BYY harmony learning algorithm on $\mathcal{B}$ with $k \geq k^*$, we get k^* flat Gaussians with the parameters $\{(\alpha_i, m_i, \Sigma_i)\}_{i=1}^k$ from the resulted mixture. Then, from each Gaussian $(\alpha_i, m_i, \Sigma_i)$, we pick up m_i and the major principle component $V_{1,i}$ of Σ_i to construct a straight line equation $l_i : U_{1,i}^T(x - m_i) = 0$, where $U_{1,i}$ is the unit vector being orthogonal to $V_{1,i}$, with the mixing proportion α_i representing the proportion of the number of points along this straight line l_i. Since the sample points are in a 2-dimensional space, $U_{1,i}$ can be uniquely determined and easily solved from $V_{1,i}$, without considering its direction. Hence, the problem of detecting multiple straight lines in a binary image has been turned into the Gaussian mixing modeling problem of both model selection and parameter learning, which can be e ciently solved by the fixed-point BYY harmony learning algorithm automatically.

With the above preparations, as $k(> k^*)$, the stop criterion threshold value $\varepsilon(> 0)$ and the component annihilation threshold value $\delta(> 0)$ are all prefixed, the procedure of our fixed-point BYY harmony learning approach to automatic straight line detection with $\mathcal{B}$ can be summarized as follows.

1. Let t = 1 and set the initial parameters Θ_0 of the Gaussian mixture as randomly as possible.
2. At time t, update the parameters of the Gaussian mixture at time $t-1$ by Eqs (5)-(7) to get the new parameters $\Theta_t = (\alpha_i, m_i, \Sigma_i)_{i=1}^{k}$;
3. If $|J(\Theta_t) - J(\Theta_{t-1})| \leq \varepsilon$, stop and get the result Θ_t, and go to Step 5; otherwise, let $t = t + 1$ and go to Step 4.
4. If some $\alpha_i \leq \delta$, discard the component $\theta_i = (\alpha_i, m_i, \Sigma_i)$ from the mixture and modify the mixing proportions with the constraint $\sum_{j=1}^{k} \alpha_j = 1$. Return to Step 2.
5. Pick up m_i and the major principle component $V_{1,i}$ of Σ_i of each Gaussian $(\alpha_j, m_j, \Sigma_j)$ in the resulted mixture to construct a straight line equation $l_i : U_{1,i}^T(x - m_i) = 0$.

It can be easily found from the above automatic straight line detection procedure that the operation of the fixed-point BYY harmony learning algorithm tries to increase the total harmony function on the Gaussian mixture so that the extra Gaussians or corresponding straight lines will be discarded automatically during the parameter learning or estimation.

x a

In this section, several simulation and practical experiments are conducted to demonstrate the e ciency of our proposed fixed-point BYY harmony learning approach. In all the experiments, the initial means of the Gaussians in the mixture are trained by the k-means algorithm on the sample data set $\mathcal{B}$. Moreover, the stop criterion threshold value ε is set to be $10 * e^{-8}$ and the component annihilation threshold value δ is set to be 0.08. For clarity, the original and detected straight lines will be drawn with red color, but the sample points along di erent straight lines will be drawn in black.

3.1 Simulation Results

For testing the proposed approach, simulation experiments are conducted on three binary image datasets consisting of di erent numbers of straight lines, which are shown in Fig.1(a),(b),(c), respectively. We implement the fixed-point BYY harmony learning algorithm on each of these datasets with k = 8. The results of the automatic straight line detection on the three image datasets are shown in Fig.1(d),(e),(f), respectively. Actually, in each case, some random noise from a Gaussian distribution with zero mean and a standard variance 0.2 is added to the coordinates of each black point. It can be seen from the experimental results that the correct number of straight lines is determined automatically to match the actual straight lines accurately in each image dataset.

3.2 Automatic Container Recognition

Automatic container recognition system is very useful for customs or logistic management. In fact, our proposed fixed-point BYY harmony learning approach

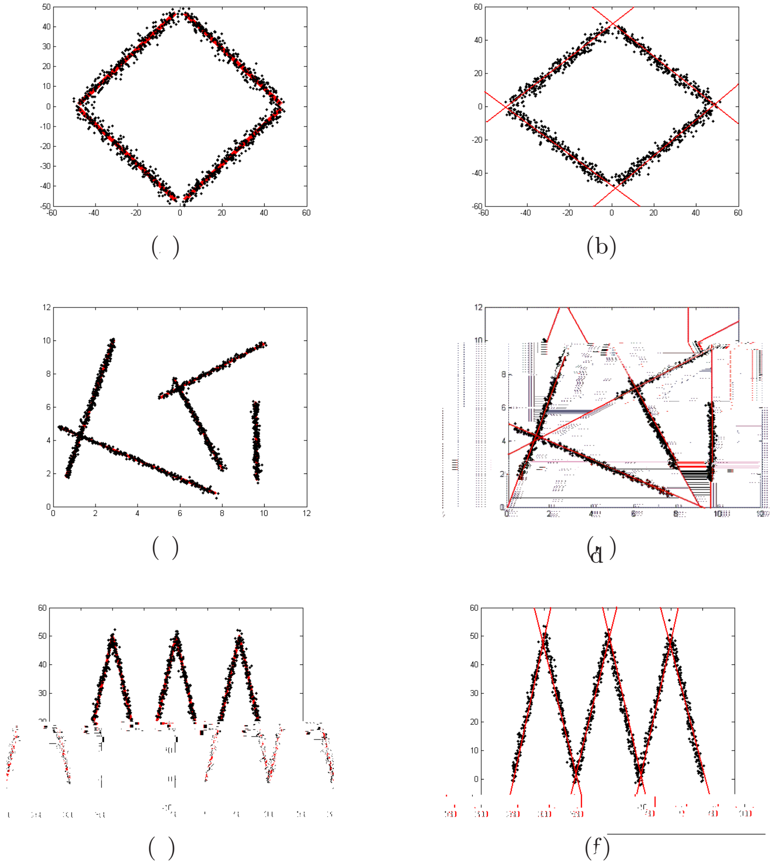

Fig. 1. ... ( ),(b),( ) ... ( ),( ),(f) ...

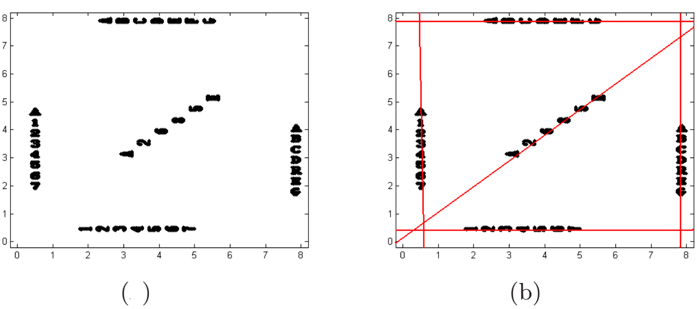

Fig. 2. ... ( ) ... (b) ...

can be applied to assisting to establish such a system. Container recognition is usually based on the captured container number located at the back of the container. Specifically, the container, as shown in Fig.2(a), can be recognized by the five series of numbers (with letters). The recognition process consists of two steps. The first step is to locate and extract each rectangular area in the raw image that contains a series of numbers, while the second step is to actually recognize these numbers via some image processing and pattern recognition techniques.

For the first step, we implement the fixed-point BYY learning algorithm to roughly locate the container numbers via detecting the five straight lines through the five series of the numbers, respectively. As shown in Fig.2(b), these five straight lines can locate the series of numbers very well. Based on the detected

7. , .: - : A s - b for s s or . 1995 . f. ro . 1995 . for ro ss (1995), o . 2, pp. 977–988 (1995)
8. , .: s - , s r b r of s rs. rs 18, 1167–1178 (1997)
9. , .: s r , A o o for r- s r s ss r s, r - r s - o s. o of r s s 11(1), 43–69 (2001)
10. , .: r , r r , f- r o s. r or s 15, 1231–1237 (2002)
11. , ., , ., , .: A r r ss r A o o . ro o 56, 481–487 (2004)
12. , ., o, ., , ., , .: r r s for r ss r A o o . o of Ar 19, 701–713 (2005)
13. , ., , .: r r : A r - A o s . r ro ss - rs 24(1), 19–40 (2006)
14. , ., , .: for for ss r o o s . 40, 2029–2037 (2007)
15. , ., , .: A s - r A for ss r A o o . rs 29(6), 701–711 (2008)
16. , ., , ., .: A r r A for r of r s for r . : r, ., , . ., r , . (s.) A 2005. o . 3578, pp. 250–257. r r, b (2005)
17. r, .A., r, . .: r s s, A for . A 26(2), 195–239 (1984)
18. , A. ., b s, . .: A for for s r , r , oo s (1988)
19. r , .A.: s r b rob s s r . : , . (.) s- s s r , pp. 45–72. A r ss, or (1977)
20. A , .: A oo s o . r s s A o ro A -19, 716–723 (1974)
21. r , .: s s of o . A s of s s 6, 461–464 (1978)
22. r , . .: rs b r o r o o s o r . o r 82(4), 711–732 (1995)
23. s ob r, . ., s , .: s s s f r s - r s. o of A r s A o 90(430), 577–588 (1995)
24. , .: A o o (A) r s: A or - A s s. : ro . 2006 f r or s (2006), o r, , 16-21, 2006, pp. 8255–8261 (2006)