Abstract - This paper introduces two novel approaches to estimate the clique potentials in binary and multilevel realizations of Gibbs Markov random field (GMRF) models. The first approach employs a genetic algorithm (GA) in order to arrive at the closest synthesized image that resembles the original “observed” image. The second approach is used to estimate the parameters of Gaussian Markov random field. Given an image formed of a number of classes, an initial class density is assumed and the parameters of the densities are estimated using the EM approach. Convergence to the true distribution is tested using the Levy distance. The segmentation of classes is performed iteratively using the ICM algorithm and a genetic algorithm (GA) search approach that provides the maximum a posteriori probability of pixel classification. During the iterations, the GA approach is used to select the clique potentials of the Gibbs-Markov models used for the observed image. The algorithm stops when a fitness function, equivalent to the maximum a posteriori probability, does not change. The approach has been tested on synthetic and real images and is shown to provide satisfactory results.

Keywords: Gibbs Markov random field (GMRF), Genetic Algorithms (GAs), Expectation maximization algorithm (EM), Lung, Chest, Computed tomography.

1 Introduction

The subject of image modeling involves the construction of models or procedures of the specification of images. These models serve a dual role in that they can describe images that are observed and also can serve to generate synthetic images from the model parameters. We will be concerned with a specific type of image model, the class of texture models. There are important areas of image processing in which texture plays an important area of image processing in which texture plays important role: for example, classification, image segmentation, and image encoding. Julesz [1] considers the problem of generation of familiar textures an important one from the theoretical and practical viewpoints. In addition, understanding texture is an essential part of understanding human vision [2]. These considerations have led to an increased activity in the area of texture analysis and synthesis.

Markov random field (MRF) models have been successfully used to represent contextual information in many ‘site’ labeling problems. A site labeling problem involves classification of each site (pixel, edge element, and region) into a certain number of classes based on an observed value (or vector) at each site. Contextual information plays an important role here because the true label of a site is assumed to be compatible with the labels of the neighboring sites. Markov random fields are appropriate modes of context because they can be used to specify this spatial dependency or spatial distribution. The class of MRF with exponential priors can be described, equivalently, by Gibbs models. Hence, for this vast class, the parameters of the MRF can be specified in terms of the clique potentials in the Gibbs distribution [3]. A Gibbs-Markov model is specified by the model order and the set of clique potentials. The flexibility of the Gibbs-Markov random fields generated considerable research interest in the past three decades. However, the problem of parameter estimation has remained to a large extent unsolved. Therefore, a systematic method for specification of these parameters is of significant interest. Several schemes have been proposed in the computer vision literature to estimate the parameters of an MRF [4-7]. For MRFs defined on pixel sites (e.g. texture modeling), these schemes have been applied with considerable success. For MRFs defined in edge sites [5] (line variable used to denote discontinuity between adjacent pixels), however, the available parameter estimation techniques [2-9] are difficult to apply because of the lack of true edge labels. Also the Least squares (LS) method is not accurate [10].

In this paper we introduce two novel unsupervised approaches to estimate GMRF parameters. In the first approach we use a genetic algorithm to minimize the error between the original image and regenerated image. In the second approach we estimate the model parameters that maximize posteriori probability of each pixel in given image. The MAP estimate is obtained using a combination of genetic search and deterministic estimation using the iterated conditional mode (ICM) approach of Besag (e.g.,
The desired estimate of the GMRF parameters is those corresponding to the MAP estimate.

2 Markov Random Field

The study of Markov random fields has had a long history, beginning with Ising thesis on ferromagnetism [12]. Although it did not prove to be a realistic model for magnetic domains, it is approximately correct for phase-separated alloys, idealized gases, and some crystals. The model has traditionally been applied to the case of either Gaussian or binary variables on lattice. Besag [4] allows a natural extension to the case of variables that have integer ranges, either bounded or unbounded. These extensions, coupled with estimation procedures, permit the application of the Markov random field to texture modeling. In this paper, we build on the models described in Geman and Geman 1984 [3], Dubes and Jain 1988 [13], Derrin and Elliott [6], and Farag andlp [14].

2.1 Definitions

Definition 1: A neighbor system \( \eta \) on the set of the sites \( S \) is a collection \( \eta = \{ \eta_s : s \in S \} \) of finite subset of \( S \) such that

1. \( s \not\in \eta_s \), and
2. \( s \in \eta_r \) if and only if \( r \in \eta_s, r, s \in S \).

The sites \( r \in \eta_s \) are called the neighbors of \( s \in S \).

Definition 2: A clique \( C \) over graph \( (S, \eta) \) is a subset of \( S \) which either consists of a single site or multiple sites, where every site is a neighbor of every other site in the set \( \eta \).

Definition 3: Consider a graph \( (S, \eta) \) of \( Q \) sites, and a random field \( G = \{ G_s, s \in S \} \) defined on \( S \). A site \( r \neq s \) is said to be neighbor of site \( s \) if and only if the conditional probability \( P(G_s = g_s | G_0, G_1…….,G_{s-1}, G_{s+1},G_{Q-1}) \) is dependent upon the variable \( G_r \).

Definition 4: The random field \( G \) is a Markov random field (MRF) with respect to a neighbor system \( \eta \) if

1. \( P(G = g) > 0 \) for all \( g \in \Omega \), and
2. \( P(G_s = g_s | G_r = g_r, r \neq s) = P(G_s = g_s | G_r = g_r, r \in \eta), \) for all \( s \in S \) and \( \{ G_r, s \in S \} \in \Omega \).

The structure of the neighborhood system determines the order of the MRF. For a first order MRF the neighborhood of a site consists of its four nearest neighbors. In a second order MRF the neighborhood of a site consists of the eight nearest neighbors. The clique structures for a 2nd MRF are illustrated in Figure 1. The order coding of the neighborhood up to order five is shown in Figure 2.

Definition 5: \( G \) is a Gibbs random field (GRF) with respect to the neighborhood system \( \eta = \{ \eta_s : s \in S \} \) if and only if

\[
P(G = g) = \frac{1}{Z} e^{-U(g)/T},
\]

where \( Z \) is a normalizing constant called the partition function, \( T \) is a control parameter called temperature and \( U \) is the energy function of the form:

\[
U(g) = \sum_{C \in \eta} V_C(g),
\]

where \( V_C \) is called a potential and is a function depending only on \( g_s, s \in C \). Only cliques of size 2 are involved in a pairwise interaction model. The energy function for a pairwise interaction model can be written in the form [4]:

\[
U(g) = \sum_{i=1}^{M} F(g_i) + \sum_{t=t_{ir}=1}^{K} H(g_t, g_{t+1}),
\]

where \( H(a, b) = H(b, a), H(a, a) = 0 \), and \( K \) depend on the size of the neighborhood around each site. Function \( F(.) \) is the potential function for single-pixel cliques, \( H(.,.) \) is the potential function for all cliques of size 2, and \( t = j + \text{Max}(t-1), \{ (j, j) : 1 \leq i \leq M, 1 \leq j < M \} \). For example, in the Derin-Elliott [4] model \( F(.) \), and \( H(.,.) \) are expressed as follows:

\[
F(g_i) = \alpha g_i \quad \text{and} \quad H(g_i, g_{i+1}) = \theta (g_i, g_{i+1}),
\]

where \( \theta (a, b) = -1 \) if \( a = b \) and 1 if \( a \neq b \).

Figure 1: Clique structure for the 2nd order neighborhoods

Figure 2: Neighborhood system for Gibbs-Markov random field.

2.2 Texture Simulation

The simulations in this paper are based on the Gibbs-sampler approach of Geman and Geman [2] (see also Derrin and Elliott [6] and Dubes and Jain [13]). A realization is generated as follows.

Algorithm I

1. Initialize the \( N \) by \( N \) lattice by assigning a color randomly from \( A = \{0, 1, 2, ..., G-1\} \) to each site. Call this initial coloring \( g \).
2. for \( s \) from 1 to \( M=N^2 \)
(a) Compute probabilities \( \{P_g\} \) for \( g=0, 1, \ldots, G-1 \).

(b) Set the color of site \( s \) to \( g \) with probability \( P_g \).

Repeat (2) \( N_{\text{iter}} \) times.

Figure 3: Examples of 64 X 64 realizations of Derin-Elliott model for various values of \( (\alpha, \theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6, \theta_7, \theta_8, \theta_9) \).

The convergence of this algorithm is assured if \( N_{\text{iter}} \) is large enough. Figure 3 shows realization of Derin-Elliott model generated with \( N_{\text{iter}} = 50 \), and using ten parameters coefficient corresponding to the second order neighborhood system. All are 64 X 64 images with \( \alpha = 1 \), and each image consist of two binary classes. The images shown in Figure 3 contain two binary classes. One can map each class to take gray level from 0 to 255 according to certain distribution (e.g., normal distribution \( N(\mu, \sigma^2) \)) as shown in Figure 4.

Figure 4: Examples of 64 X 64 realizations of Autonormal random process for various values of \( (\alpha, \theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6, \theta_7, \theta_8, \theta_9) \).

### 3 Model-based parameters estimation

#### 3.1 Parameter estimation for binary GMRF

In this section we introduce an approach to estimate the parameters for binary GMRF such as the images shown in Figure 3. The main idea of our approach is to use the GA to generate coefficient of \( U(g) \), and evaluate these coefficient through the fitness function. We will focus on first and second-order GMRF models, the approach can be readily generalized for higher order models as well. To build the genetic algorithm we define the following parameters (see Goldberg [15] for the terminologies of the genetic algorithm).

1) **Chromosome:** A chromosome is represented in binary digits and consists of representations for model order and clique coefficients. An example of a chromosome is shown in Figure 5. Each chromosome has 41 bits. The first bit represent the order of the system (we use digit “0” for first order and digit “1”for second order-GMRF). The remaining bits represent the clique coefficients, where each clique coefficient is represented by four bits (note that for first order system we estimate only five parameters, and the remaining cliques coefficient will be zero, but for the second order system we will estimate ten parameters).

<table>
<thead>
<tr>
<th>System</th>
<th>First Clique Order</th>
<th>Coefficient (( \theta_i ))</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 0 0 0</td>
<td>1</td>
<td>Last Clique Coefficient (( \theta_{10} ))</td>
</tr>
</tbody>
</table>

Figure 5: Example of a chromosome. Each chromosome corresponds to a possible solution of the problem.

2) **Fitness:** We defined the fitness of the individual as follow:

\[
\text{Error} = |G_{\text{image}}(i, j) \cdot O_{\text{image}}(i, j)|, \\
\text{fitness} = \begin{cases} 
1/\text{Error} & \text{Error} > 0 \\
10^{10} & \text{Error} = 0
\end{cases},
\]

where \( G_{\text{image}} \) is the regenerated image using the estimated parameters for the cliques, \( O_{\text{image}} \) is the original image.

3) **GA Process.** First, the initial population was randomly generated from a sequence of zeroes and ones. Next, the fitness of each individual was calculated. Then crossover and mutation is applied to generate the next population [15].

Algorithm II

(1) Generate the first generation which consists of 30 chromosomes.
(2) Use Algorithm I to generate image corresponding to each chromosomes, and use original image as an initial image.

(3) If the Fitness value is equal to $10^{10}$, then stop and the chromosome, which gives this value, is the desired solution. (If there are two chromosomes give the same fitness value we select the chromosomes which represent lower order system).

(4) If the fitness values for all chromosomes in the first generation do not satisfy the required condition, go to 2.

The above algorithm was applied on the six textures shown in Figure 3. Table 1 shows the original parameters and estimated parameters for each image. Figure 6 shows the original image and the regenerated image using the estimated parameters. From these results we note that the estimated parameters are very close to original parameters, also the regenerated images close to the original images. These results are more favorable than the least square approach results in [10] and Markov Chain Monte Carlo (MCMC) in [16]. We also note that our method estimates 10 parameters while the results in [10] and [16] were based on four parameters only. Table 2 shows comparison of our results with the results shown in [16]. On the first texture shown in Figure 3, the Genetic algorithm converged to the solution after 15 to 20 generation, which took two-three minutes on a PC with an Intel 2.2 GHz processor.

Table 1: Original and estimated parameters.

<table>
<thead>
<tr>
<th>NO.</th>
<th>Original Images</th>
<th>Regenerated Image using the estimated parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>[1 0.3 0.3 0.3 0.3 0.0 0.0 0.0]</td>
<td>[1 0.278 0.289 0.289 0.29 0.009 0.008]</td>
</tr>
<tr>
<td>2</td>
<td>[1 1 1 1 1 0 0 0 0 0]</td>
<td>[0.9 0.9 -0.99 0.9 0.0 0.0 0.0]</td>
</tr>
<tr>
<td>3</td>
<td>[1 2 2 1 1 1 0 0 0 0 0]</td>
<td>[1.99 1.97 -0.95 -0.94 0.0 0.0 0.0]</td>
</tr>
<tr>
<td>4</td>
<td>[1 1 1 1 1 1 1 -1 1]</td>
<td>[1 1 1 0.91 -0.99 0.9 0.98 0.931 0.99]</td>
</tr>
<tr>
<td>5</td>
<td>[1 1 -1 1 1 1 1 -1 1 1]</td>
<td>[0.97 0.97 0.99 0.99 0.98 -0.98 0.89 -0.99 0.99]</td>
</tr>
<tr>
<td>6</td>
<td>[1 1 1 1 1 1 1 1 1 -1 1]</td>
<td>[-0.99 1 1 1 1 1 1 1 -0.98]</td>
</tr>
<tr>
<td>7</td>
<td>[1 1 -1 1 1 1 1 1 1 1 -1]</td>
<td>[1 1 -0.95 1 1 1 1 1 -0.97]</td>
</tr>
<tr>
<td>8</td>
<td>[1 1 1 1 1 1 1 1 1 1 1 -1]</td>
<td>[1 1 1 1 1 1 1 1 -0.91]</td>
</tr>
</tbody>
</table>

3.2 Parameter estimation for Gaussian Markov Random Field

A typical outline for statistical-based image parameters estimation is as follows (e.g., [14], [17]): The observed image process $G$ is modeled as a composite of two random processes, a high level process $G^h$ and a low level process $G^l$, that is, $G = (G^h, G^l)$. Each of the three processes is a random field defined on the same lattice $S$. The high level process (the labeling or coloring process) $G^h$ is used to characterize the spatial clustering of pixels into regions. The low level process (pixel process) $G^l$ describes the statistical dependence of pixel gray level values in each region. Let the number of regions in the scene be $R$, the number of possible gray levels be $q$. The processes $G^h$ and $G^l$ are discrete parameter random fields with state spaces defined as follows:

$$
\Xi^h = \{ \xi^h : \xi^h \in [C_1, C_m] \},
$$

$$
\Xi^l = \{ \xi^l : \xi^l \in [0, q - 1] \},
$$

where $m$ is the number of regions (with colors or labels $C_1, C_2, \ldots, C_R$) and $q$ is the number of possible gray levels in a particular region (e.g., 256).
The observed image \( g \) can be described as follows: Consider a region type \( k \). The gray level value at pixel \( s \) in \( S \) of the observed image \( g \) equals that of region type \( k \), that is,

\[
g_s = g^l \quad \text{if} 
\]

\[
g^h = q_k, \quad s \in S, \quad k \in [1, m] .
\]

The maximum a posteriori (MAP) parameters estimation involves the determination of \( g^l \) that maximizes \( P(G^l = g^l | G = g) \) with respect to \( g^l \). By Bayes’ rule,

\[
P(G^h = g^h | G = g) = \frac{P(G = g | G^h = g^h) P(G^h = g^h)}{P(G = g)}. \tag{8}
\]

Since the denominator of Eq. (8) does not affect the optimization, the MAP parameters estimation can be obtained, equivalently, by maximizing the numerator of Eq. (8) or its natural logarithm; that is, we need to find \( g^h \) which maximizes the following criterion:

\[
\Gamma(G, G^h) = \ln P(G = g | G^h = g^h) + \ln P(G^h = g^h). \tag{9}
\]

The first term of Eq (9) is the likelihood due to the low level process and the second term is due to the high level process. Based on the models of the high level and low level processes, the MAP estimate can be obtained.

In order to carry out the MAP parameters estimation in Eq. (9), one needs to specify the parameters in the two processes. A popular model for the high level process is the Gibbs-Markov model, with the probability density function as shown in Eq. (1). In this paper we will assume the model for the low level process to a mixture of normal distributions which follow the following equation:

\[
\pi_i k + l = (1/n) \sum_{j=1}^{n} \delta_{jl} \quad \text{for} \ l = 1, \ldots, m, \tag{12}
\]

\[
\mu_k + l = \sum_{j=1}^{n} \delta_{jl} g_j / \sum_{l=1}^{m} \delta_{jl} ; \tag{13}
\]

\[
(\sigma^k_{l l l})^2 = \sum_{j=1}^{n} \delta_{jl} (g_j - \mu) k / \sum_{l=1}^{m} \delta_{jl} . \tag{14}
\]

3. The M-step: we compute the new mean, the new variance and the new proportion from the following equation:

\[
\rho(G^l) = \inf \{ \xi > 0 : \forall g, P_{emp}(g - \xi) - \xi \leq P_{ref}(g) \}
\]

For example, Figure 7 shows the empirical density function of the mixture and reference density function of the mixture. Figure 8 shows the empirical distribution of the mixture and reference distribution for each class. The Levy \( \rho (P_{emp}, P_{ref}) \) distance is defined as:

\[
\rho(P_{emp}, P_{ref}) = \inf \{ \xi > 0 : \forall g, P_{emp}(g - \xi) - \xi \leq P_{ref}(g) \}
\]

\[
\leq P_{emp}(g + \xi). \tag{16}
\]

4. Repeat steps 1 and 2 until the relative difference of the subsequent values of Eq. 11, Eq. 12, Eq. 13, and Eq. 14 are sufficiently small.

5. Compute the conditional expectation from the following equation

\[
Q(C) = \sum_{j=1}^{n} \sum_{l=1}^{m} \delta_{jl} \log(g_j | \Theta_j, C) . \tag{15}
\]

Repeat steps 2, 3, 4, and 5 if the conditional expectation \( Q(C) \) is still increasing, and increase the number of classes by one, otherwise stop and select the parameters which correspond to maximum \( Q(c) \).

We run the above algorithm in the first image in Figure 4. The results are presented in Table (3). As can be seen the last row of the table at \( m = 2 \) maximize the conditional expectation so we will select \( m = 2 \) for the mixture model. Also it is clear from Table (4) that the original parameters of the mixture are close to the estimated parameters.

In order to determine the convergence between the empirical distribution (which is computed from the low level image), and the reference distribution (Gaussian distribution, which their parameters are estimated using EM algorithm) we will compute the Levy distance between the empirical distribution (\( P_{emp} \)) and the reference distribution (\( P_{ref} \)) for each class. The Levy \( \rho (P_{emp}, P_{ref}) \) distance is defined as:

\[
\rho(P_{emp}, P_{ref}) = \inf \{ \xi > 0 : \forall g, P_{emp}(g - \xi) - \xi \leq P_{ref}(g) \}
\]

\[
\leq P_{emp}(g + \xi). \tag{16}
\]

Algorithm III

1. Assume the number of classes in the image are two
2. The E-step: given the current value of number of classes \( C_i \), compute \( \delta_{ji} \) as

\[
\delta_{ji}^{k+1} = \pi_i^k P(g_j | \Theta_i^k, C_i) / \sum_{l=1}^{m} \pi_i^k P(g_j | \Theta_i^k, C_i) , \tag{11}
\]

where \( g_j \) is the color at location \( j \), \( m \) is the number of classes, \( \pi_i^k \) is the proportion of class \( i \) at step \( k \), \( \Theta_i^k \) is estimated parameter for class \( i \) at step \( k \)
Note that the above simulation started with mixture of normal distributions that is why there is a perfect match between the model and estimated distributions. Same approach was used on various images where the distribution was not a priori known, and yielded satisfactory results.

Table 3: Comparison of the mixture fit for different values of classes

<table>
<thead>
<tr>
<th>Parameter</th>
<th>m = 2</th>
<th>m = 3</th>
<th>m = 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\mu_1$</td>
<td>31.4570</td>
<td>31.4570</td>
<td>21.5181</td>
</tr>
<tr>
<td>$\mu_2$</td>
<td>153.0938</td>
<td>143.2666</td>
<td>41.3176</td>
</tr>
<tr>
<td>$\mu_3$</td>
<td>-</td>
<td>163.5264</td>
<td>143.2666</td>
</tr>
<tr>
<td>$\mu_4$</td>
<td>-</td>
<td>-</td>
<td>163.5264</td>
</tr>
<tr>
<td>$\sigma_1^2$</td>
<td>153.4138</td>
<td>153.4138</td>
<td>59.2091</td>
</tr>
<tr>
<td>$\sigma_2^2$</td>
<td>157.5943</td>
<td>55.7180</td>
<td>52.9694</td>
</tr>
<tr>
<td>$\sigma_3^2$</td>
<td>-</td>
<td>59.5610</td>
<td>55.7180</td>
</tr>
<tr>
<td>$\sigma_4^2$</td>
<td>-</td>
<td>-</td>
<td>59.5610</td>
</tr>
<tr>
<td>$\pi_1$</td>
<td>0.5518</td>
<td>0.5514</td>
<td>0.2742</td>
</tr>
<tr>
<td>$\pi_2$</td>
<td>0.4482</td>
<td>0.2325</td>
<td>0.2773</td>
</tr>
<tr>
<td>$\pi_3$</td>
<td>-</td>
<td>0.2161</td>
<td>0.2324</td>
</tr>
<tr>
<td>$\pi_4$</td>
<td>-</td>
<td>-</td>
<td>0.2161</td>
</tr>
<tr>
<td>$Q$</td>
<td>-14163.56</td>
<td>-15831.98</td>
<td>-1555.05</td>
</tr>
</tbody>
</table>

Table 4 shows The Estimated, and the original parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>$\mu_1$</th>
<th>$\mu_2$</th>
<th>$\sigma_1^2$</th>
<th>$\sigma_2^2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Estimated</td>
<td>31.4570</td>
<td>153.0938</td>
<td>153.4138</td>
<td>157.5943</td>
</tr>
<tr>
<td>Original</td>
<td>31.9027</td>
<td>152.8616</td>
<td>149.8957</td>
<td>158.09</td>
</tr>
</tbody>
</table>

In order to estimate the parameters of the model of GMRF that maximizes Eq. 9, we will use the iterative conditional mode (ICM) approach [11], and genetic algorithm (GA). The ICM is a relaxation algorithm to find a global maximum. The algorithm assumes that the classes of all neighbors of a pixel $g$ are known. The high level process is assumed to be formed of $C$-independent processes each of the $C$ processes is modeled by Gibbs-Markov random field which follow Eq. 1. Then $g$ can be classified using the fact that $P(C_i|g)$ is proportional to $P(g|C_i)P(C_i|\eta_s)$; i.e.,

$$P(C_i|g) \propto P(g|C_i)P(C_i|\eta_s), \quad (17)$$

where $\eta_s$ is the neighbor set to site $S$ belonging to class $C_i$, $P(C_i|\eta_s)$ is computed from Eq. 1.

Bayes classifier can be used to get initial labeling image (High level process). The Classification is performed according to [18]: $g \in C_i$ if

$$P(C_i|g) > P(C_j|g) \quad \text{for all } j \neq i. \quad (18)$$

This can be rewritten in terms of class conditional densities, $P(g|C_i)$, as $g \in C_i$ if

$$P(C_i|g) P(C_i) > P(C_j|g) P(C_j) \quad \text{for all } j \neq i$$

where $P(C_i)$ is the a priori probability of class $C_i$.

To implement rule (19), there are two issues to be considered. The first issue is the a priori probability of each class. In this paper we assume that $P(C_i)$ is equal to the proportion ($\pi_i$), and it can be estimated from the data set using EM algorithm as shown in table 3. The second and major issue is to estimate the class conditional probability $P(g|C_i)$ for each class. In this paper we assume $P(g|C_i)$ is a Gaussian distribution which their parameters are estimated using EM.

In order to run ICM, first we must know the coefficient of potential function $U(g)$, so we will use GA to path the coefficient of $U(g)$, and evaluate these coefficient through the fitness function.

We will use the same structure of GAs that described in the section 3.1 and using Eq. 9 to be the fitness function of each chromosome.

Algorithm IV

1. Generate the first generation which consists of 30 chromosomes.
(2) Apply the ICM algorithm for each chromosome on each image and then compute the fitness function for each chromosome.

(3) If the fitness value for all chromosomes do not change from one population to another population, then stop and select the chromosome, which gives maximum fitness value (If there are two chromosomes give the same fitness value we select the chromosomes which represent lower order system). Otherwise go to step 2.

We run the previous algorithm on the six textures had shown in part IV. Table 5 shows the original parameters and estimated parameters for each image. Figure 9 shows the original image and the regenerated image using the estimated parameters. From these results we note that the estimated parameters are very close to original parameters, also the regenerated images close to the original images.

4 Application

We will apply the above approach on CT slices, in order to separate the lungs from other tissues in the chest cavity. Figure 10 shows a typical CT slice for the chest. The goal of this application is to separate the voxels corresponding to lung tissue from those belonging to the surrounding anatomical structures. We assume that each slice consists of two types: lung and other tissues (e.g., chest, ribs, liver).

Figure 10: A typical slice form a chest spiral CT scan

Also in this application we model each CT slice as a composite of two random processes, a high level process \( G^h \) and a low level process \( G^l \), that is, \( G = (G^h, G^l) \). Each of the three processes is a random field defined on the same lattice \( S \). The high level process (the labeling or coloring process) \( G^h \) is used to characterize the spatial clustering of pixels into regions. The low level process (pixel process) \( G^l \) describes the statistical dependence of pixel gray level values in each region.

We will select the model for the low level process to be a mixture of normal distributions. In order to determine the number of classes for low level image and estimate the mean and the variance for each class for low level image we will consider the low level process is a mixture of normal distribution and we will use the Expectation-Maximization (EM) algorithm to estimate the mean, the variance, and the proportion for each distribution as described above. After we run the EM algorithm we get the number of classes equal two, and the parameters are shown in Table 6.

Table 5: Original and estimated parameters.

<table>
<thead>
<tr>
<th>NO.</th>
<th>Original Images</th>
<th>Regenerated Image using the estimated parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>[1 0.3 0.3 0.3 0 0 0 0 0]</td>
<td>[0.99 0.25 0.27 0.289 0.21 0.01 0.007 0.00]</td>
</tr>
<tr>
<td>2</td>
<td>[1 1 1 1 1 0 0 0 0 0]</td>
<td>[0.86 0.96 0.91 0.9 0.08 0 0 0 0 0]</td>
</tr>
<tr>
<td>3</td>
<td>[1 2 2 1 -1 0 0 0 0 0]</td>
<td>[1.197 1.96 -0.93 -0.92 0 0 0 0 0 0]</td>
</tr>
<tr>
<td>4</td>
<td>[1 1 1 -1 -1 1 1 -1 1]</td>
<td>[0.9 0.89 0.91 0.88 -0.87 0.81 0.88 0.81 0.99]</td>
</tr>
<tr>
<td>5</td>
<td>[1 1 1 -1 1 -1 1 1 1]</td>
<td>[0.91 0.91 0.87 -0.92 1 -0.98 0.88 -0.99 0.99 1]</td>
</tr>
<tr>
<td>6</td>
<td>[1 -1 1 1 1 1 -1 1]</td>
<td>[1 -0.99 0.8 0.82 0.79 0.99 0.89 0.99 0.97 -0.98]</td>
</tr>
<tr>
<td>7</td>
<td>[1 1 1 1 1 1 1 -1 1]</td>
<td>[0.99 0.89 -0.95 0.89 0.87 0.87 0.99 0.98 0.99 -0.96]</td>
</tr>
<tr>
<td>8</td>
<td>[1 1 1 1 1 1 1 1 1]</td>
<td>[1 0.89 0.88 0.98 0.99 0.96 0.95 0.99 0.99 0.91]</td>
</tr>
</tbody>
</table>

Table 6 Estimated parameters for each class using EM

<table>
<thead>
<tr>
<th>Parameter</th>
<th>( \mu_1 )</th>
<th>( \mu_2 )</th>
<th>( \sigma^2_1 )</th>
<th>( \sigma^2_2 )</th>
<th>( \pi_1 )</th>
<th>( \pi_2 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Values</td>
<td>39.9</td>
<td>170.8</td>
<td>225.7</td>
<td>303.6</td>
<td>0.242</td>
<td>0.758</td>
</tr>
</tbody>
</table>

Figure 11 shows the empirical density function of the CT slices and reference density function of the CT slices. Figure 12 shows the empirical distribution of the CT slices and reference distribution for the CT slices. For these distributions the \( \rho (P_{emp}, P_{ref}) = 0.009 \). Since as \( \rho (P_{emp}, P_{ref}) \) approach to zero \( P_{emp}(g) \) converge weekly to \( P_{ref}(g) \), then above result indicate that the \( P_{emp}(g) \) converge to \( P_{ref}(g) \), and this proves that choosing of mixture of
normal distributions to be the model for the low level process is correct.

Figure 11  Empirical density function of CT slices $p_{emp}(g)$, and Reference density function of the CT slices $p_{ref}(g)$

Figure 12  Empirical Distribution $G(g)$, and Reference Distribution $F(g)$.

The next step of our algorithm is to choose a model of high level process. A popular model for the high level process is the Gibbs-Markov model, with the probability density function as shown in Eq. (1), and use will use Bayes classifier as discussed in section 5.2 to get initial labeling image. Then we run ICM and GA as illustrated in section 3.2 to determine the coefficients of potential function $U(g)$, we gets the following results, $(\alpha = 1, \theta_1 = 0.89, \theta_2 = 0.8, \theta_3 = 0.78, \theta_4 = 0.69, \theta_5 = 0.54, \theta_6 = 0.91, \theta_7 = 0.89, \theta_8 = 0.56, \theta_9 = -0.99)$. The results of segmentation for the image shown in figure 12 using these parameters is shown in Figure 13. Figure 13-a shows the results of proposed algorithm. Fig 13-b shows Output of ICM by selecting parameters randomly. As shown in Figure13 the accuracy of our algorithm is seems good.

Figure 13 (a) Segmented Lung Using the proposed algorithm (b) Output of ICM by selecting parameters randomly.

5  Conclusion

In this paper we introduced two novel approaches to estimate the clique potentials in Gibbs-Markov image models. First approach is used to estimate the clique potentials for binary Gibbs Markov random field (GMRF) by using genetic algorithms (GAs). Second approach is used to estimate the parameters of Gaussian Markov random field. The outline steps of the second algorithm are as follow. Given an image formed of a number of classes, an initial class density is assumed and the parameters of the densities are estimated using the EM approach. Convergence to the true distribution is tested using the Levy distance. The segmentation of classes is performed iteratively using the ICM algorithm though a genetic algorithm (GA) search approach that provides the maximum a posteriori probability of pixel classification. During the iterations, the GA approach is used to select the model order and the corresponding clique potentials of the Gibbs-Markov models used for the observed image. The approach has been tested on synthetic and real images and provides satisfactory results.

References


