SEGMENTATION OF COLOR IMAGES USING A COMBINATION BETWEEN MARKOW RANDOM FIELDS AND GRAPHS

Stefan GHERBEA, Aurel VLAICU
C.S. Hidroelectrica S.A. – S.H. Cluj, Cluj-Napoca, 1 Taberei st.,
Technical University of Cluj-Napoca, 15. C.Daicoviciu str.
Tel. 0788202770 Fax:(040) 0264-427797 , e-mail Stefan.Gherbea@hidroelectrica.ro
Aurel.Vlaicu@com.utcluj.ro

Abstract: The goal of image segmentation is partitioning the images into homogeneous and interpretable regions. Image interpretation is part of scene understanding. It can be viewed as the process of giving meaning to a 2D image by identifying and labeling significant objects or segments in the image. For example, we may first segment an image and then interpret each segment as being a road, river, building etc.

***Key words**: MRF, Gibbs distribution, Simulated Annealing, Graph, MST, probability*

I. INTRODUCTION

In image processing the probability theory plays an important role. In recent years has been developed several algorithms as stochastic search. This includes random search, recursive least square, stochastic approximation simulated annealing, evolutionary computation (including genetic algorithms) and reinforcement learning. Most of these algorithms have been developed in order to search global minimum of loss functions or energy functions.

In this article we concentrate on image interpretation schemes based on Bayesian approaches. The use of Bayesian approach results in encoding the domain knowledge in a probabilistic framework and hence reduce the dependence of the interpretation process on the domain knowledge.

Our experiment is based on Gibbs sampler and simulated annealing algorithm.

To segment the images using Gibbs sampler method the users have to know preliminarily the number of clusters (segments) for image and then they have to extract little regions from the image for each desired cluster (in most programs little rectangle region R_c). For each R_c, the algorithm detects the mean and variation. The algorithm uses information like mean and variation to detect, based on the energy function, to which segment belongs each pixel. The energy function is a quantitative measure of the global quality of the solution and a guide to the search for a minimal solution.

Because, the selections of R_c region is to the user's latitude, the method gives poor results when we have a lot of segments or when we do not have distinct clusters.

To avoid this, we use Gibbs sampler method to de-noise the image. In this case is not necessary to know the number of segments and we do not have to select regions like R_c. Farther more we know that the energy function guides to the minimal solution. This can be seen in figure 1. The left image represents a random image using a symmetric binary channel and the right image represents the result after we apply a Gibbs sampler.

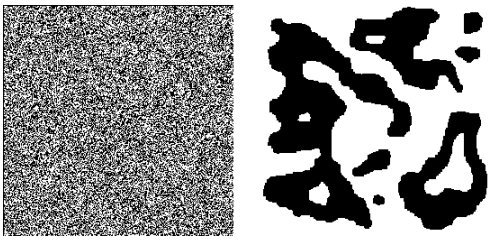

Figure 1

In the below diagram can be viewed that Gibbs sampler method and Simulated Annealing algorithm can be used either to directly segment an image or to de-noise

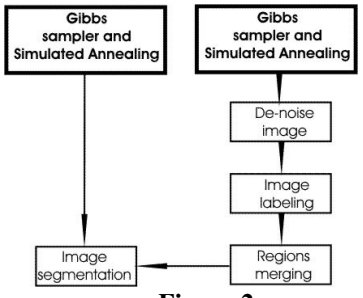

Figure 2

Manuscript received February 25, 2008; revised April 17, 2008

an image. In this article we propose a method for segmenting images. The Gibbs sampler and Simulated Annealing algorithm are used to de-noise image only. First the color image is converted into a black and white one using Gibbs sampler method and Simulated Annealing algorithm to de-noise image that results into homogeneous regions (see figure 1 for binary case).

Using an appropriate algorithm, each homogeneous region is labeled. The process computes for each region:

- the neighbors ,
- the mean color from the original color image,
- the pixels from each region.

In the third step we detect region-adjacency graph (RAG) from image and then compute the Minimum Spanning Tree of the graph. In this moment most of computations have been done. Based on this graph we chose which regions will be merged and which will not.

II. BRIEF REVIEW OF GIBBS SAMPLER THEORY AND SIMULATED ANNEALING

2.1 The Ising model:

Image modeling plays an important role in modern image processing. It is commonly used in image analysis for texture segmentation classification and image de-noising.

Due to the diversity of image types, it is impossible to have an universal model to cover all possible types.

Markov random fields belong to the statistical models of images. Each pixel of an image can be viewed as a random variable. Because each pixel is a random variable, the image will be considered a random field.

We consider a sequence 0,1,2,3,…,n on the line. For each point there is a spin which is oriented up or down. We define a probability measure on the set Ω of all possible configurations $x = (x_0, x_1, x_2, ..., x_n)$. Each spin is a function.

$$\delta_i(x) = \begin{cases} 1 & x_i = up \\ -1 & x_i = down \end{cases} \quad (1)$$

An energy U(x) is assigned to each configuration:

$$U(x) = -J\sum_{i,j}\delta_i(x)\delta_j(x) - mH\sum_i \delta_i(x) \quad (2)$$

The Ising model makes the assumption that only interactions of points with one unit apart need to be taken into account. The first term represents the energy caused by spins interactions. The constant J is a property of the material. If J>0 we have an "attractive" case between neighboring spins. If J<0 we have a "repulsive" case between neighboring spins. The second term represents the influence of an external magnetic field. H represents the intensity of the external field and m>0 is a property of material.

2.2 Markow random field and Gibbs distribution

MRF is mostly inspired by the Ising model. In order to generate a sample image from a given random field, we have to compute p(X), where X is the random field. Supposing that all the pixels are uncorrelated, p(X) can be computed based on the equation:

$$p(X) = \prod_{i=1}^{n} p(x_i) \quad (3)$$

In order to introduce correlation, we have to assume that all the pixels within the image are correlated and that all the pixels should be taken into account at the same time. In practice, this is a drawback, because the size of sampling space for a binary image is 2^{n*m} and for a k grey level image is k^{n*m}.

Definition:

X is a Markov random field (MRF) if and only if each pixel is given only by its neighbors.

The sites in S are related to one another via a neighborhood system. A neighborhood system for a regular lattice S is defined as: $N = \{N_i \forall i \in S\}$, where N_i is the set of sites neighboring i. The neighboring relationship has the following properties:

- a site is not neighboring to itself
- the neighboring relationship is mutual $i \in N_{i'} \Leftrightarrow i' \in N_i$

For a regular lattice, the neighboring set of i is defined as the set of nearby sites within a radius of r.

$$N_i = \{i' \in S \mid [dist(pixel_{i'}, pixel_i)]^2 \le r, i' \ne i\}$$

Where $dist(pixel_{i'}, pixel_i)$ is the Euclidian distance between $pixel_i$ and $pixel_{i'}$, and r takes an integer value.

Sites at or near the boundaries have fewer neighbors.

In the first order neighborhood system, (4-neighborhood system), every (interior) site has four neighbors, as shown in the figure bellow, where the black square denotes the considered site and the grey squares its neighbors.

In the second order neighborhood system (the 8-neighborhood system) there are eight neighbors for every (interior) site, as shown in bellow figure [1].

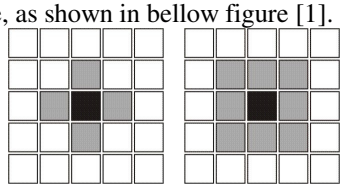

***Figure** 3 (First and second order* **neighborhood sites)**

The pair (S,N) constitutes a graph in the usual sense where S contains the nodes and N determines the links between the nodes according to the neighboring relationship. A site represents in our case a point in the image pixel.

The type of a clique for (S,N) of a regular lattice is determined by its size, shape and orientation. Fig. bellow shows clique types for the first and second order neighborhood systems for a lattice.

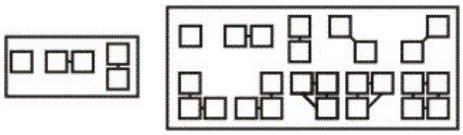

Figure 4 (*Clique for first and second order neighborhood sites*)

In our experiment clique is used for second order neighborhood system for a lattice.

If X has a finite configuration over a lattice S and $P(x=0)>0$, then X is a MRF with respect to a neighborhood N_i if and only if X is a Gibbs distributed. Gibbs distribution takes the following form [1]

$$P(X=x)=\frac{1}{Z}e^{\frac{-1}{T}U(x)} \tag{4}$$

Variable x is a realization of X, Z is the partition function (normalization constant), T is the temperature and U(x) is an energy term depending on the vicinity of x.

$$Z=\sum_{all\_config.clique} e^{-U(x)} \tag{5}$$

and U(x) is given by:

$$U(x)=\sum_{c\in C} V_c(x) \tag{6}$$

U(x) is a sum of *clique potentials* $V_c(x)$ over all possible cliques **C**. The value of $V_c(x)$ depends on the local configuration on the clique **C**.

An MRF is characterized by its local property (the Markovianity) whereas a GRF is characterized by its global property (the Gibbs distribution). The Hammersley-Clifford theorem establishes the equivalence of these two types of properties. The theorem states that **X** s an MRF on **S** with respect to N_i if and only if **X** is a GRF on **S** with respect to N_i.

2.3 Simulated Annealing Algorithm (SAA)

Let Θ the domain of allowable values θ. The main problem of interest is:

Find the value(s) of vector $\theta\in\Theta$ that minimize a scalar value of energy function $L(\theta)$. Simulated Annealing Algorithm (SAA) is designed to traverse local minima en route to global minima of energy function $L=L(\theta)$. The term annealing comes from analogies to the cooling of a liquid or solid. At high temperatures, molecules have much mobility, but as the temperature decreases, this mobility is lost and the molecules may tend to align themselves in a crystalline structure. This aligned structure is the minimum energy state for the system. The cooling must occur at a sufficiently slow rate. If the substance is cooled at too rapid rate, an amorphous state may be reached.

Figure 5
(Original image)

Figure 6
De-noise image

In optimization, the analogy to a minimum energy state for a system is a minimizing value of the energy function.

Gibbs probability distribution, describing the probability of a system having a particular discrete energy state: $P(energy=x)=1/Ze^{\frac{-1}{T}U(x)}$.

If a system is in some current energy state ε_{curr} and some system aspect are changed to make the system potentially achieved a new energy state ε_{new} then based on Metropolis simulation always the system go to the new state if $\varepsilon_{new}<\varepsilon_{curr}$. On the other hand, if $\varepsilon_{new}\geq\varepsilon_{curr}$ then probability of the system going to the new state is: $\exp(-(\varepsilon_{new}-\varepsilon_{curr})/T)$. Above expression is known like Metropolis criterion.

In figure 5 is the original image and in figure 6 is the result after we apply the Simulated Annealing algorithm using Gibbs sampler.

One of the most common such form has the temperature decaying at a rate proportional to $1/\log(k)$ (k is the iteration index).

2.4. SAA Algorithm with Noise Free Loss Mesurements

Step 0. (Initialization) Set the initial temperature T and initial parameter vector $\hat{\theta}_0=\theta_{curr}$; determine

Step 1. Relativ to the current value θ_{curr}, randomly determine a new value of θ, $\theta_{new}\in\Theta$, and determine $L(\theta_{new})$

Step 2. Compare the two L values based on Metropolis

criterion. Let $\delta = L(\theta_{new}) - L(\theta_{curr})$. If $\delta < 0$ accept θ_{new}. Alternatively, if $\delta \geq 0$ accept θ_{new} only if an uniform (0,1) rtandom variable U satisfies: $U \leq \exp[-\delta / cb * T]$ where cb is Boltzmann constant.

Step 3. Repeat step 1 and 2 for some period until either the budget of function evaluations allocated for that T has been used or the system reaches some state of equilibrium

Step 4. Lower T according to the annealing schedule and return to step 1.

Continue the process until the total budget for function evaluations has been used or some indication of convergence is satisfied.

III. LABELING IMAGE:

After de-noising the image, the result is a set of homogenous regions. For each region, the following parameters should be detected:

- the mean color
- the pixels from the image belonging to each region
- the neighbors of each region

Neighboring pixels are merged into regions, if their attributes (in our case, the gray level of the pixel), are sufficiently similar. This similarity is often represented by a homogeneity criterion. If a pixel satisfies the homogeneity criterion, then the pixel can be included in the region.

Figure 4

We note with V, the set of gray level pixels used to define connectivity. In a binary image we have $V = \{1\}$ for connectivity of pixels having the value 1. In a gray scale image, for connectivity of pixels within a range from 30 to 35 we have $V = \{30,31,32,33,34,35\}$

Neighbors of a pixel:

A pixel P having the (x,y) coordinates, has 4 vertical and horizontal neighbors whose coordinates are given by:

$$(x+1, y), (x-1, y), (x, y+1), (x, y-1)$$

This set of pixels is noted by $N_4(p)$.

The 4 diagonal neighbors of p have coordinates:

$$(x+1, y+1), (x+1, y-1), (x-1, y+1), (x-1, y-1)$$

This set of pixels is noted with $N_D(p)$ and, together with 4-neighbors, are called the 8-neighbors of p, denoted by $N_8(p)$.

There are 3 types of pixel connectivity for regular lattice S:

- 4-connectivity: two pixels p and q are 4 - connected if q is in the set $N_4(p)$
- 8-connectivity: two pixels p and q are 8 - connected if q is in the set $N_8(p)$
- m-connectivity (mixed connectivity) Two pixels p and q are m-connectivity if:
  i) q is in $N_4(p)$ or
  ii) q is in $N_D(p)$ and the set $N_4(p) \cap N_4(q)$ is empty

In our experiment for labeling regions, 4-connectivity and m-connectivity are used. Mixed connectivity is introduced to eliminate multiple path connections that often arise when 8-connectivity is used.

For each region, the mean color based on color image is computed (figure 7).

IV. MINIMUM SPANNING TREE

Based on the labeling algorithm, we detect homogeneous regions from the black and white de-noised image (the pixels from each homogenous region are detected and then, based on these pixels, the dominant color for each region is computed).

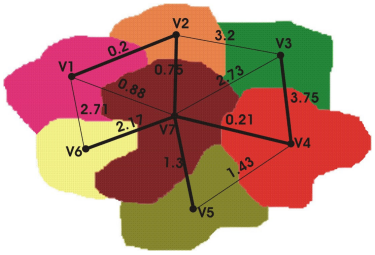

Figure 8

The region adjacency graph (RAG) is the data structure that is being used to represent the image.

Each region will represent the nodes of a graph G=(V,E), where V represents vertices (nod) of graph and E represents edge. Merging process is based on graph G=(V,E). Based on this graph we compute the minimum spanning tree (see figure 9). The thick rule represent minimum spanning tree.

A spanning tree is an acyclic subset of edges that connects all vertices together. In the computation process of MST we use Kruskal's algorithm [8], which generates the minimum spanning tree, denoted as T, by adding one edge at a time. Initially, the edges of G are sorted in a non-decreasing order of their weight. Then, the edges in the sorted list are examined one by one and checked whether adding the edge that is currently being examined creates a cycle with the edges that were already added to T. If it does not, it is added to the spanning tree of G.

In our project we use a perceptually uniform color space CIE L*a*b space. Each edge (u,v) of G has a certain weight, which is computed based on the Euclidian distance.

$$w(u,v) = \sqrt{(L_1 - L_2)^2 + (a_1 - a_2)^2 + (b_1 - b_2)^2} \quad (12)$$

Assuming that each edge (u,v) of G has a numerical weight or cost w(u,v), the cost of the spanning

Figure 10

tree T will be the sum of all edges in T. Otherwise, it is discarded.

The process is terminated when T contains n-1 edges. At the end of the process, T is the minimum

$$w(T) = \sum_{(u,v)\in T} w(u,v) \quad (13)$$

A minimum spanning tree (MST) is a spanning tree of minimum weight.

If we fix a threshold **T < 0,8,** regions connected with a cost less then 0,8 will be merged (see figure 9). The result of merged regions can be viewed in the figure 9.

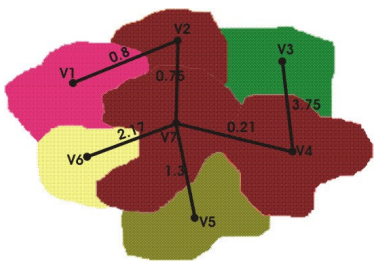

Figure 9

Using the minimum spanning tree and choosing a threshold the number of segments is reduced by merging similar segments. The results, after a threshold of 35,5 is applied can be seen in figure 10.

After the MST of an image is computed, the process is almost finished. In this moment it's easy to select different type of threshold and choose the optimal solution.

Using the minimum spanning tree and by choosing a threshold, similar regions will be merged and the number of segments will be reduced.

V. REGIONS MERGING BASED ON HOMOGENEITY CRITERION

In the first step the regions are merged using the global information. After this step, there are certain regions which can be merged too. Using a threshold value, the following graph can be obtained:

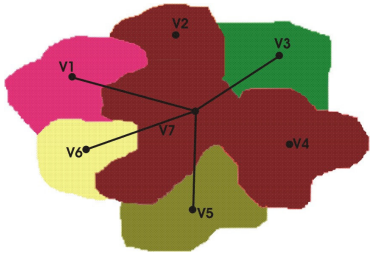

Figure 11

Further, based on the homogeneity criterion, we test if the regions **(v1,v7), (v3 ,v7), (v5,v7), (v6,v7)** and **(v1,v6)** can be merged, using local information. We convert the original color image to HSV color space and use the V component of this space to calculate the variance of each resulted region. The variance is a measure for homogeneity.

Let $\mu(R_i^{m_i})$ be the mean V value of region $R_i^{m_i}$ and $V(x,y)$ be the value of V in location (x,y) [10]. The variance of any region $R_i^{m_i}$, i=1,2,…,n after its m_i-th merge is defined as:

$$\sigma(R_i^{m_i}) = \frac{1}{|R_i^{m_i}|}\sqrt{\sum_{(x,y)\in R_i^{m_i}} (V(x,y) - \mu(R_i^{m_i}))^2}, \quad (14)$$

where $|R_i^{m_i}|$ is the size of region $R_i^{m_i}$. The change in the homogeneity of $R_i^{m_i}$ after m_i-th merge will be:

Figure 5

$$\Delta\sigma(R_i^{m_i}) = \left|\sigma(R_i^{m_i}) - \sigma(R_i^{m_i-1})\right| \quad (15)$$

Regions are merged until a single region is left and every region becomes inhomogeneous at a different merge operation.

`We argue that $R_i^{m_i}$ becomes inhomogeneous if

$$\Delta\sigma(R_i^{m_i}) > \beta \quad (16)$$

Where β is the mean value of $\Delta\sigma(R_i^{m_i})$.

$$\beta = \frac{1}{k}\sum \Delta\sigma(R_i^{m_i}) \quad (17)$$

The merge algorithm generating inhomogeneous region has to be canceled. We realize the test homogeneity and merge them based on MST path. The results can be seen in the figure below .

VI. CONCLUSIONS:

The goal of image segmentation is partitioning the image into homogeneous and interpretable regions. We propose in this article a method of image segmentation viewing the domain knowledge in a probabilistic manner and extracting the final result by using graphs.

In our experiment the probabilistic method was used only to reduce the noise from the original image and to create homogeneous grey level regions in it. Based on these regions we compute from the original color image the dominant color for each region and this color will become the nod for a graph. Based on this graph, the minimum spanning tree algorithm was computed, using the Kruskal's algorithm. By fixing a threshold, we can deduce which region will be merged and which not, based on Euclidian distance. The order in which the merging process takes place is based on the Kruskal algorithm [9] for finding minimum spanning tree in a graph.

The advantage of this method is that we do not have to know in advance the number of segments involved in the color image. If we compute the MST from the graph, we can vary the threshold and choose the optimal solution. Then, local information for each region is used, based on the original color image, in order to compute the variance. The original color image is converted in the HSV color space and the V component of the HSV color space is used to calculate the variance of each resulted region. The variance is a measure for homogeneity. Using this information we merge regions, provided they do not create inhomogeneous region.

VII. REFERENCE:

[1] Sunil K. Kopparapu, Uday B. Desai "Bayesian , Approach to image Interpretation", Kluwer Academic Publisher 2001

[2] Charles A. Bouman, "Approach to image Interpretation", 1995 IEEE International Conference on Image Processing 23-26 October 1995 Washington, D.C.

[3] Pedro F. Felzenszwalb, "Eficient Graph-Based Image Segmentation", IEEE Published in the Proceedings of CVPR , June 1999 Santa Barbara

[4] Pedro F. Felzenszwalb and Daniel P. Huttenlocher, "Image segmentation Using Local Variation", IEEE Published in the Proceedings of CVPR'98 , June 1998 Santa Barbara

[5] Ety Navon, Offer Miller, Amir Averbuch Color, "Eficient Graph-Based Image Segmentation", Image and vision computing, 23: 69-85, 2005

[6] H.D. Cheng, Ying Sun, "A Hierarchical Approach to Color Image Segmentation Using Homogeneity" IEEE Trans Image Process. 2000

[7] Youssef Chahir, "Searching Images on the Basis of Color Homogeneous Objects and their Spatial Relationship" Academic Press 2000

[8] Corman Thomas, Leiserson Charles, Rivest Ronald, "Introduction to Algorithms", the MIT press, 1990

[9] Saarinen, K. "Color image segmentation by a watershed algorithm and regiona djacency graph processing" Image Processing, 1994. Proceedings. ICIP-94., IEEE International Conference

[10] Ety Navon, Ofer Miller, Amir Averbuch "Color Image Segmentation Based on Adaptive Local Thresholds" IEEE Transactions on Circuits and Systems for Video 2002