Markov Random Fields (A Rough Guide)

Anil C. Kokaram

anil.kokaram@tcd.ie

Electrical and Electronic Engineering Dept.,
University of Dublin, Trinity College.

See www.sigmedia.tv for more information.
Markov Random Fields

- Kind of obvious intuition about how variables might interact in a local region
• Kind of obvious intuition about how variables might interact in a local region
Markov Random Fields

- Captures this idea: To find out things about my current site, I only need to know what is happening in the neighbourhood of that site and not anything else.

\[
p(b(x)|B) = p(b(x)|B(x \in \mathcal{N}))
\]

- Multidimensional version of a Markov Processes
- Established in Signal Processing (actually Control) since 1972 by John Woods and heavily used in image coding schemes,
- Popularised by the Geman brothers in 1987 as Gibbs Fields together with their work on Simulated Annealing.
- Leads to well understood optimisation schemes and behaviours
- Tends to go hand in hand with Bayesian inference as it is most useful as a prior
1. Image Analysis, Random Fields ... by Wilson


7. Bayesian estimation of motion vector fields, J. Konrad, E. Dubois, IEEE Transactions on

[1] is very good
Designing an MRF with a Gibbs Energy Function

Classic stuff: Let us say we are interested in finding all the sky in an image, or selecting some optimal routing protocol amongst the nodes of a network, or detecting a strange clump in a 3D MRI scan.

Say we call this variable $b(x)$ where $b(x) = 1$ means this site has sky, or this node has chosen protocol 1, or this voxel is part of a clump.

and $b(x) = 0$ means this site has NO sky, or this node has chosen protocol 2, or this voxel is NOT part of a clump. (I could have way more options here, it doesn’t matter)

$b(x)$ is a Random Field of random DISCRETE variables in several dimensions

It is clear from our knowledge of the world that in all these problems $b(x)$ tends to be smooth locally ... hence Markovianity is a good idea .. so
It is clear from our knowledge of the world that in all these problems $b(x)$ tends to be smooth locally ... hence Markovianity is a good idea .. so

This is what we want to happen :

$p(o=0|O) > p(o=1|O)$

$O(x)=1$ $O(x)=0$

$p(o=0|O) > p(o=1|O)$ $p(o=1|O) > p(o=0|O)$ $p(o=0|O) > p(o=1|O)$
Designing an MRF with a Gibbs Energy Function

It is clear from our knowledge of the world that in all these problems \( b(\mathbf{x}) \) tends to be smooth locally ... hence Markovianity is a good idea .. so

\[
p(b(\mathbf{x})|B^N) \propto \exp - \left( \Lambda \sum_{k=1}^{4} \lambda_k |b(\mathbf{x}) \neq b(\mathbf{x} + \mathbf{q}_k)| \right)
\]

If \( b \) was \( \pm 1 \) then this also works :

\[
p(b(\mathbf{x})|B^N) \propto \exp - \left( \Lambda \sum_{k=1}^{4} \lambda_k b(\mathbf{x}) b(\mathbf{x} + \mathbf{q}_k) \right)
\]
What does this mean?

\[
p(b(x)|B^N) \propto \exp - \left( \Lambda \sum_{k=1}^{4} \lambda_k |b(x) \neq b(x + q_k)| \right)
\]

\[= \exp - \Lambda (\lambda_1 |b(x) \neq b(x + [-1, 0])| + \lambda_2 |b(x) \neq b(x + [-1, -1])| + \ldots)
\]
Jargon: Potentials and Cliques

\[ b(x) = b(h, k) \]

\[ p(b(x) | B^N) \propto \exp - \left( \Lambda \sum_{k=1}^{4} \lambda_k V(b(x), b(x + q_k)) \right) \]
Designing an MRF with a Gibbs Energy Function

Does it do the job?

\[ b(x) = b(h, k) \]

\[ b(h, k-1) \]

\[ b(h+1, k) \]

\[ b(h, k+1) \]

\[ b(h, k-1) \]

\[ p(b(x)|B^N) \propto \exp - \left( \Lambda \sum_{k=1}^{4} \lambda_k |b(x) \neq b(x + q_k)| \right) \]

\[ p(o=0|O) > p(o=1|O) \]

\[ p(o=1|O) > p(o=0|O) \]

\[ p(o=0|O) > p(o=1|O) \]

\[ p(o=1|O) > p(o=0|O) \]

\[ p(o=0|O) > p(o=1|O) \]
Designing an MRF with a Gibbs Energy Function

What does this look like? Use the Gibbs sampler to generate samples

\[ p(b(\mathbf{x})|B^N) \propto \exp - \left( \Lambda \sum_{k=1}^{4} \lambda_k |b(\mathbf{x}) \neq b(\mathbf{x} + \mathbf{q}_k)| \right) \]

1. Generate random configuration of \( B \) as 1’s and 0’s say
2. At each site measure \( p(b = 1|B^N) \) and \( p(b = 0|B^N) \) and draw a random sample for \( b \)
3. Translation : measure \( E(0) = \Lambda \sum_{k=1}^{4} \lambda_k |0 \neq b(\mathbf{x} + \mathbf{q}_k)| \) and \( E(1) = \Lambda \sum_{k=1}^{4} \lambda_k |0 \neq b(\mathbf{x} + \mathbf{q}_k)| \)
4. \( p'(b = 1|B^N) = \exp -(E(1) - K) \) and \( p'(b = 1|B^N) = \exp -(E(0) - K) \). \( K \) is that clever scaling factor \( K = \min([E(0), E(1)]) \)
5. Normalise and sample
6. Keep going until you’re happy ....
Designing an MRF as a n-D Autoregressive Process

\[ I(x) = \sum_{k=1}^{P} a_k I(x + q_k) + e(x) \quad I \text{ is continuous e.g. grayscale, voltage, intensity} \]

Causal:

\[ I(h, k) = a_1 I(h - 1, k) + a_2 I(h - 1, k - 1) + a_3 I(h, k - 1) + e(h, k) \]

Semicausal:

\[ I(h, k) = a_1 I(h - 1, k) + a_2 I(h - 1, k - 1) + a_3 I(h, k - 1) + a_4 (h + 1, k - 1) + e(h, k) \]

\[ e(h, k) \sim \mathcal{N}(0, \sigma_e^2) \] Innovations sequence, or residual
Designing an MRF as a n-D Autoregressive Process

\[ I(x) = \sum_{k=1}^{P} a_k I(x + q_k) + \epsilon(x) \]

Very very elegant formulation (Woods 1972!), Can be considered as an IIR filter that generates \( I \), can be very efficient, recently being re-discovered by computer vision people.

Can generate using a filter! \( I \) turns out to be Gaussian distributed.

\[
p(I(x)|I) \propto \exp - \left( \frac{\left[ I(x) - \sum_{k=1}^{P} a_k I(x + q_k) \right]^2}{2\sigma^2} \right)
\]

See papers by Woods, Veldhuis, Godsill, O’Ruanaidh, Kokaram for extremely diverse treatment including efficient techniques for handling
Can model continuous fields with a Gaussian Markov Random Field i.e. use a Gibbs Energy function taking continuous variables. Loads possible.

Parameters for AR processes are easily estimated from the data using correlation measurements. Gibbs hyperparameters less so.

Parameter estimation for AR does not guarantee stable systems. Dunno how to handle stability for GMRFs.

AR processes are subsets of Gibbs RFs. Hard to model discrete fields with AR processes.
What do MRF’s look like? : The Gibbs Field

\[ \lambda = 20, \text{ 100 iterations, } \Lambda = 1 \text{ [See gibbsgenerator.m]} \]
An example: Image Segmentation

We have two classes of pixels: Class 1 has a colour $\sim \mathcal{N}(c_1, \sigma_1^2)$, and the other $\sim \mathcal{N}(c_2, \sigma_2^2)$.

We want to segment our field of pixels $I$ by assigning values to another image $B$ such that $b(x)$ is 1 at sites corresponding to pixels in $I$ that are from class 1 and $b(x)$ is 0 for the other class.

Classic place to start is to think about solving the problem one site at a time but let’s do it right. Proceed in a Bayesian fashion:

$$p(B|I) \propto p(I|B)p(B)$$

$$= p(I|B)p(b(x_1)|B)p(b(x_2)|B)p(b(x_3)|B)$$

We want to choose the best configuration for ALL of B which maximises $p(B|I)$ over ALL the image. That is one heck of a lot of variables for a 720x576 picture for instance.

Turns out that Graph Cuts optimisation routines solve this problem exactly for 2 variables.

We are going to simplify things first and say that we seek local optimisation moves that solve the global problem eventually. Consider ONE site

$$p(b(x)|I, B) \propto p(I|b(x))p(b(x)|B)$$
Design of the expressions

\[ p(b(x)|I,B) \propto p(I|b(x))p(b(x)|B) \]

Likelihood

\[
p(I(x)|b(x)) = \begin{cases} 
\frac{1}{Z_1} \exp - ([I - c_1]^T R_1^{-1} [I - c_1]) & b=1 \\
\frac{1}{Z_2} \exp - ([I - c_2]^T R_2^{-1} [I - c_2]) & b=2 
\end{cases}
\]
Design of the expressions

\[ p(b(x)|I, B) \propto p(I|b(x))p(b(x)|B) \]

Prior

\[ p(b(x)|B) = \begin{cases} 
\exp - \left( \Lambda \sum_{k=1}^{4} \lambda_k |1 \neq b(x + q_k)| \right) & \text{b}=1 \\
\exp - \left( \Lambda \sum_{k=1}^{4} \lambda_k |2 \neq b(x + q_k)| \right) & \text{b}=2 
\end{cases} \]
$p(b(x)|I, B) \propto p(I|b(x))p(b(x)|B)$

Prior

$$-\ln[p(b(x)|I, B)] = \begin{cases} \ln(Z_1) + [I - c_1]^T R_1^{-1} [I - c_1] + \Lambda \sum_{k=1}^{4} \lambda_k |1 \neq b(x + q_k)| & b=1 \\ \ln(Z_2) + [I - c_2]^T R_2^{-1} [I - c_2] + \Lambda \sum_{k=1}^{4} \lambda_k |2 \neq b(x + q_k)| & b=2 \end{cases}$$

$$= \begin{cases} \ln(Z_1) + E_l(1) + E_s(1) & b=1 \\ \ln(Z_2) + E_L(2) + E_s(2) & b=2 \end{cases}$$

$$= \begin{cases} E(1) & b=1 \\ E(2) & b=2 \end{cases}$$
ICM = Iterated Conditional Modes = Pick the best answer for each local conditional distribution and keep going

\[ p(b(\mathbf{x})|I, B) \propto p(I|b(\mathbf{x}))p(b(\mathbf{x})|B) \]

1. Initialise \( B \) randomly or with the likelihood only
2. At each site, measure \( E(1), E(2) \) pick the label that gives the LEAST energy at that site
3. Keep going

Not very efficient. Can use multigrid methods to help ALOT, can use GRAPH CUTS (but unclear the amount of memory needed), Can use Belief Propagation instead. ICM gives you a local optimum.

This optimisation is tricky because \( b \) is discrete, if it was continuous it would be easier.
ICM for finding sky

\[ \lambda = 1000, \text{ 1,3,5 iterations [See example.m]} \]
MRF for voxels or image sequences

$$\mathbf{q}_1 = [0, 0, -1] \quad \mathbf{q}_3 = [1, 0, -1] \quad \mathbf{q}_5 = [-1, 0, -1]$$

$$\mathbf{q}_2 = [0, -1, -1] \quad \mathbf{q}_4 = [0, 1, -1]$$

Frame $n-1$ Frame $n$

$\mathbf{d}_{n,n-1}$
Vector MRF ($\lambda \propto 1/r$)

$$p_d(d_{n,n-1}|S_n) \propto \exp\left(-\sum_{v \in S_n} \lambda(v)[d_{n,n-1} - v]^2\right)$$
Final Comments

- MRF’s are an intuitive way of capturing local dependency.
- Can be defined through Autoregressive processes or Gibbs energy functions
- Recently Conditional Random Fields have appeared
- Whether MRFs result in closed form solutions or not depends on the choice of formulation of the field
- High order models lead to more computations in general
- Graph Cuts, ICM, SA, good for solving discrete label problems while PDE’s or Gauss-Siedel may be better for continuous variable problems