Undirected Graphical Models

Raul Queiroz Feitosa
A Random Field

Images are represented by a graph $G = \{V,E\}$, where:

- $V$ is a set of vertices or nodes that correspond to image sites.

- $E$ is a set of edges connecting pairs of image sites that “interact” with each other. $E$ defines a neighborhood system.
An image site represents a point or a region in the Euclidean space such as an image pixel, a superpixel or an image patch.
**Neighborhood System**

**Examples** of *neighborhood systems* for a regular set of sites $S$:

4 - Neighborhood

8 - Neighborhood

Fully Connected

please, guess
Clique: any set of fully connected nodes

Maximal clique: a clique such that it is not possible to include any other node from the graph without it ceasing to be a clique.
A countable set of labels $\mathbf{x} = [x_1, \ldots, x_n]$, where

- $n$ is the number of sites,
- $x_i \in \{1, \ldots, M\}$,
- $M$ is the number of classes,
Markov Random Field

Let $x = \{x_i\}, i \in S$ be the set of variables at an input image, where $x_i$ corresponds to the $i^{th}$ site, $N_i$ are the set of $i$ neighbors, and $S$ is the set of image sites.

**Definition:** $x$ is said to be a **Markov random field** on $S$ w.r.t. a neighborhood system $N$, iff the following two conditions hold for all $x_i \in x$

$P(x_i) \geq 0,$  \hspace{1cm}  \text{(positivity)}

$P(x_i | x_{S-\{i\}}) = P(x_i | x_{N_i}),$  \hspace{1cm}  \text{(Markovianity)}
Gibbs Distribution

A set of random variables $\mathbf{x}$ has a **Gibbs distribution** on $S$ w.r.t. $N$ iff

$$P(\mathbf{x}) = \frac{1}{Z} \exp \left[ - \sum_{c \in C} E_c(x_c) \right]$$

where

- $C$ is the set of all maximal cliques,
- $E_c(x_c) \geq 0$ is the energy associated with the variables in clique $c$,
- $Z = \sum_{x_c} \exp \left[ - \sum_{c \in C} E_c(x_c) \right]$ is a normalizing constant called *partition function.*
$x$ is a Markov random field on $S$ w.r.t. a neighborhood system $N$, iff $P(x)$ follows the Gibbs distribution on $S$ w.r.t. a neighborhood system $N$. 
Gibbs Random Field

**Example:** from the graph below one can write

\[ P(\mathbf{x}) \propto \exp \left[ -E(\mathbf{x}) \right] \]

\[ E(\mathbf{x}) = E_{123}(x_1, x_2, x_3) + E_{234}(x_2, x_3, x_4) + E_{35}(x_3, x_5) \]

**Pairwise MRF:** for simplicity it may be restricted to edges rather than to maximal cliques (henceforth):

\[ E(\mathbf{x}) = E_{12}(x_1, x_2) + E_{13}(x_1, x_3) + E_{23}(x_2, x_3) + ... \\
+ E_{24}(x_2, x_4) + E_{34}(x_3, x_4) + E_{35}(x_3, x_5) \]
MAP-MRF framework

In the MRF framework, the posterior over the labels $\mathbf{x} = \{x_i\}$, the corresponding set of observed data $\mathbf{y} = \{y_i\}$, is expressed using the Bayes’ rule as,

$$P(\mathbf{x}|\mathbf{y}) \propto P(\mathbf{x})p(\mathbf{y}|\mathbf{x})$$

where the prior $P(\mathbf{x})$ over the labels is modeled as a MRF.

Usually, for tractability, the likelihood model $p(\mathbf{y}|\mathbf{x})$ is assumed to have the factorized form

$$p(\mathbf{y}|\mathbf{x}) = \prod_{i \in S} p(y_i|\mathbf{x}_i) = \exp \left[ - \sum_{i \in S} - \log p(y_i|\mathbf{x}_i) \right]$$
Pairwise MRF image model

Consequently, the posterior takes the form

\[
P(x|y) = \frac{1}{Z} \exp \left\{ - \sum_{i \in S} \left[ - \log p(y_i|x_i) + \sum_{j \in N} E(x_i, x_j) \right] \right\}
\]

**Inference**: the classification outcome \( \hat{x} \) is given by

\[
\hat{x} = \arg\max_x P(x|y)
\]
Example of MRF

Potts model: produces a smoothing effect

\[
E(x_i, x_j) = \begin{cases} 
0, & \text{for } x_i = x_j \\
\beta, & \text{for } x_i \neq x_j 
\end{cases}
\]

\[
E = \begin{bmatrix} 
0 & \beta & \beta \\
\beta & 0 & \beta \\
\beta & \beta & 0 
\end{bmatrix}
\]

(M=3)

Noisy Image

Restored Image (ICM)

Bayes' Theorem

label smoothing
Problems with MAP-MRF framework

1) The assumption $p(y|x) = \prod_{i \in S} p(y_i|x_i)$ is too restrictive for several natural image analysis applications!

2) The label interaction $E(x_i, x_j)$ does not depend on the data $y$. We may want to “turn-off” label smoothing between two adjacent nodes if there is a discontinuity in image intensity.

3) To model the posterior $P(x|y)$ we have first to model the likelihood $p(y_i|x_i)$. 
Conditional Random Field

CRF models the posterior $P(x|y)$ probability directly as an MRF without modeling the prior $P(x)$ and likelihood $p(y_i|x_i)$ individually.

$$P(x|y) = \frac{1}{Z} \exp \left\{ - \sum_{i \in S} \left[ - \log p(x_i|y) + \sum_{j \in N_i} I(x_i, x_j|y) \right] \right\}$$

local posterior given by some discriminative classifier

homogeneous CRF
MRF vs CRF

MRF: \[ P(x|y) = \frac{1}{Z} \exp \left\{ -\sum_{i \in S} \left[ -\log p(y_i|x_i) + \sum_{j \in N_i} E(x_i, x_j) \right] \right\} \]

CRF: \[ P(x|y) = \frac{1}{Z} \exp \left\{ -\sum_{i \in S} \left[ -\log p(x_i|y) + \sum_{j \in N_i} I(x_i, x_j|y) \right] \right\} \]

1) Association potentials are **likelihoods in MRF** and **local posteriors in CRF**.
2) Association potential is a **function of local data in MRF** and **of all data in CRF**.
3) Interaction potential is a **function of labels only in MRF** and **also of data in CRF**.
Example of Interaction Potential

A simple contrast sensitive Potts model: “turns off” smoothing at data discontinuities, e.g.,

\[ I(x_i, x_j | y_i, y_j) = \begin{cases} 
0, & \text{for } x_i = x_j \\
\beta \left(1 - \frac{D_{ij}}{\max D_{ij}}\right), & \text{for } x_i \neq x_j
\end{cases} \]

whereby discontinuity is given by

\[ D_{ij} = \|y_i - y_j\|^2 \]
Fully Connected CRF

The neighborhood becomes the whole image

\[ P(x|y) = \frac{1}{Z} \exp \left\{ - \sum_{i \in S} \left[ - \log p(x_i|y) + \sum_{j \in N_i} I(x_i, x_j|y) \right] \right\} \]

\[ P(x|y) = \frac{1}{Z} \exp \left\{ - \sum_{i \in S} - \log p(x_i|y) + \sum_{j, i \in S} I(x_i, x_j|y) \right\} \]
Fully Connected CRF

A contrast sensitive Potts model for fully connected CRF:

\[
I(x_i, x_j | y_i, y_j) = \mu(y_i, y_j) \left[ w_1 \exp \left( - \frac{||c_i - c_j||^2}{2\sigma^2_\alpha} - \frac{||y_i - y_j||^2}{2\sigma^2_\beta} \right) \right] \\
+ w_2 \exp \left( - \frac{||c_i - c_j||^2}{2\sigma^2_\gamma} \right)
\]

where

\[
\mu(y_i, y_j) = \begin{cases} 
1 & \text{for } y_i \neq y_j \text{ and zero otherwise,} \\
\end{cases}
\]

\[c_i\] represents pixel coordinates, and \[\sigma_\alpha, \sigma_\beta, \text{ and } \sigma_\gamma\] control the scale of Gaussian kernels.

Fully Connected CRF

A contrast sensitive Potts model for fully connected CRF:

\[ I(x_i, x_j | y_i, y_j) = \mu(y_i, y_j) \left[ w_1 \exp \left( -\frac{\|c_i - c_j\|^2}{2\sigma_\alpha} - \frac{y_i - y_j}{2\sigma_\beta} \right) \right] \]

\[ + w_2 \exp \left( -\frac{\|c_i - c_j\|^2}{2\sigma_\gamma} \right) \]

Parameters must be trained. Default for RGB images:

\[ \sigma_\alpha = 80 ; \sigma_\beta = 13 ; \sigma_\gamma = 3 \]

\[ w_1 = 1 (?) ; w_2 = 1 \]

MRF/CRF representation

A pairwise MRF/CRF model over a 4 neighborhood regular grid consists of:

1. Node structure (links)
2. Node potentials (unaries)
   \( M \) dimensional vectors
3. Edge potentials (binaries)
   \( M \times M \) matrices
MRF/CRF training

- Unaries are trained in a supervised way
- Binaries are estimated by cross-validation.
- In some cases binaries’ training also follows a supervised approach.
Inference

It is about computing \( \hat{x} = \arg\max_x P(x|y) \)

- For a grid containing \( a \times b \) nodes, whereby each node may belong to any of \( M \) classes, the number of possibilities is equal to \( M^{ab} \).

- Exact solution is only feasible for few particular problems.

- Efficient approximate algorithms are available\(^1\).

- To further reduce complexity superpixels instead of pixels are used.

\(^1\)P. Krähenbühl and V. Koltun, “Efficient inference in fully connected crfs with gaussian edge potentials,” in NIPS, 2011.
Some Examples

Semantic Segmentation of a hyperspectral image (Salinas)

Figure 13: Ground truth, input image and sample results for Salinas data set.
Some Examples

Semantic Segmentation of a hyperspectral image (Indian Pines)
Practical Assignment

Write a program that implements a fully connected CRF for semantic segmentation. It consists of the following of step:

1) Download the dataset ISPRSDataset. It comprises two images and the corresponding classification.

2) Use the training image to train a discriminative pixel-wise classifier.

3) Apply the classifier you trained in the previous step on the test image.

4) Assess the overall and average class accuracies.

5) Print the classification result as a label image.

6) Apply a fully connected CRF on the test image using the posteriors delivered by the classifier trained in step 3) and repeat steps 4) and 5)

7) You may play around with some CRF parameters to get an insight on how they affect the performance.

8) Report the experiments with emphasis in the conclusion section.

- **Hints:**
  - Use a Random Forest as discriminative classifier
  - For CRF implementation use the code available at https://github.com/Golbstein/Keras-segmentation-deeplab-v3.1/blob/master/utils.py
Books and Papers

References

Software


- Python code for fully connected CRF. Available at https://github.com/Golbstein/Keras-segmentation-deeplab-v3.1/blob/master/utils.py
END