CS840a
Learning and Computer Vision
Prof. Olga Veksler

Lecture 6

Markov Random Fields and Belief Propagation

Today

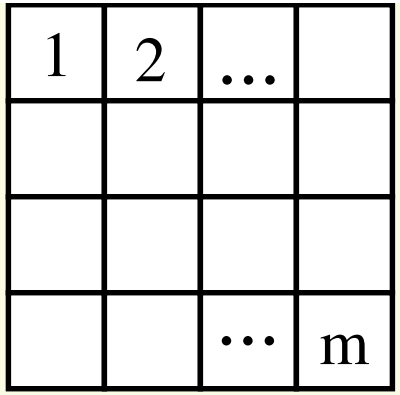
- Markov Random Fields
- Belief Propagation
- Next time paper:
  - “Object Class Recognition by Unsupervised Scale-Invariant Learning” by R. Fergus, P. Perona, A. Zisserman

Labeling Problem

Given

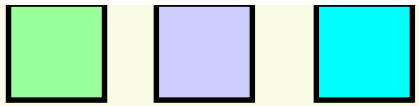
1 Image with m pixels

$$S = \{1, \cdots, m\}$$



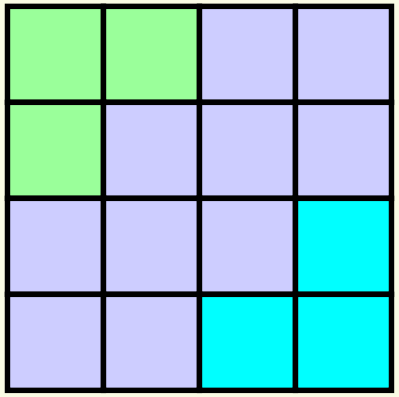
2 Set of labels

$$L = \{l_1, \cdots, l_k\}$$



Find

Labeling



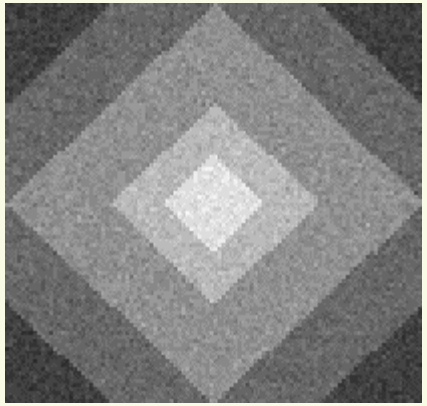
$$f = \{f_1, \cdots, f_m\}, \quad f_p \in L$$

pixel p is assigned l

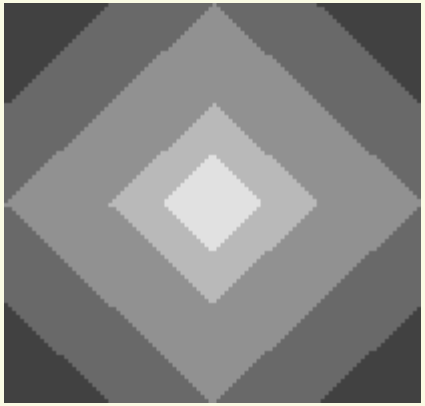
$$\Updownarrow$$

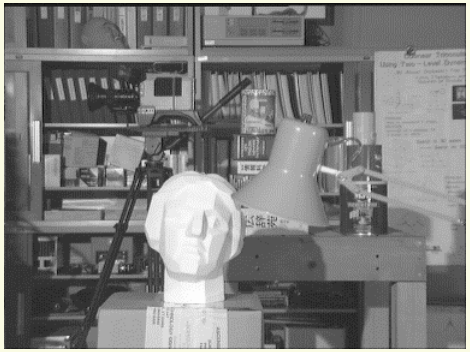
$$f_p = l$$

Examples



Restoration

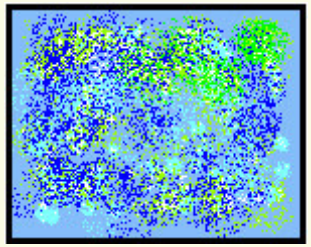




Stereo

A priori knowledge is frequently available

Unlikely answer



Better answer

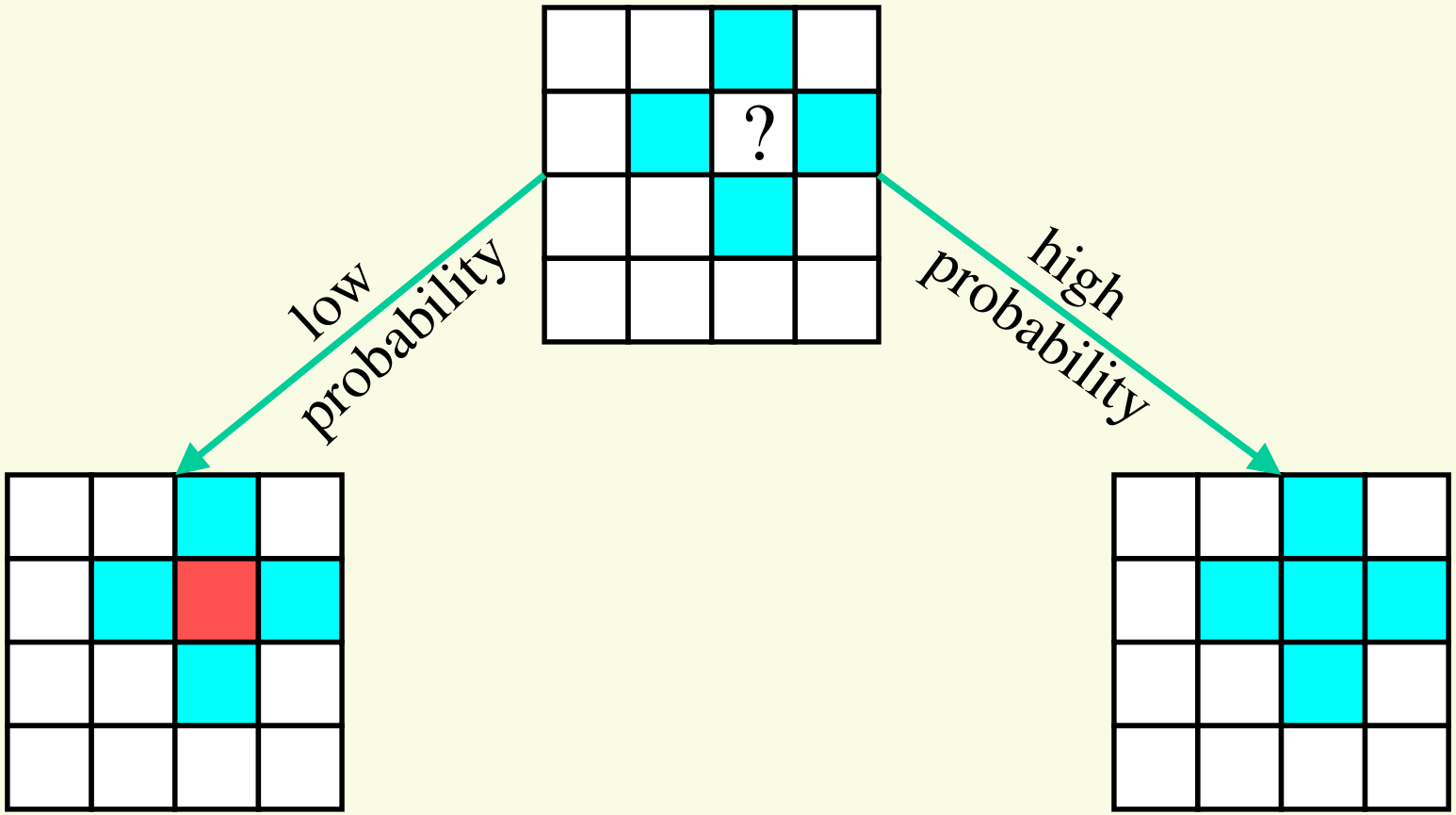
- Bayesian framework is suitable:
  assign prior probabilities *Pr(f)* to labelings

f_1	f_2	$\cdots$	
		$\cdots$	f_m

$$f = \{f_1, \ldots f_m\} \in L^m$$

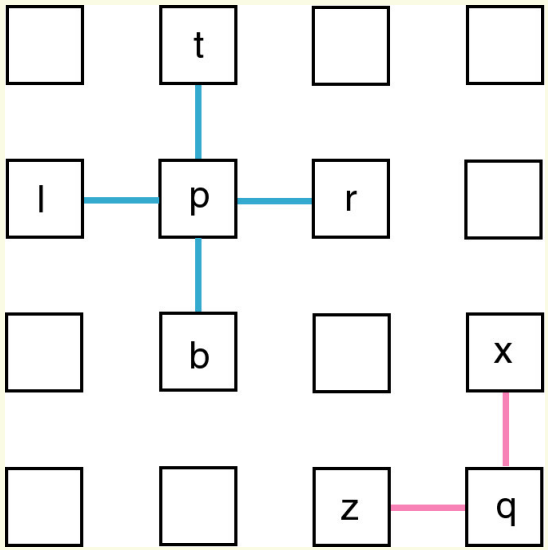
Desirable Properties of prior Pr(f)

Pixel's label depends on neighboring pixels



Markov Random Fields (MRF)

Each pixel p has a *neighborhood* $\boldsymbol{N}_p$



Examples:

$$\boldsymbol{N}_p = \{t, b, l, r\}$$

$$\boldsymbol{N}_q = \{x, z\}$$

Property: $q \in \boldsymbol{N}_p \Leftrightarrow p \in \boldsymbol{N}_q$

Markov Random Fields

- F is an MRF if:
- Markovianity:

$$Pr\left(f_p \middle| f_{S-\{p\}} \right) = Pr\left(f_p \middle| f_{N_p} \right)$$

- Positivity:

$$Pr\left(f \right) > 0, \quad \forall f \in L^m$$

Markov Random Fields

- Not obvious how to compute $Pr(f)$ from local conditional distributions $Pr\left(f_p \middle| f_{N_p}\right)$

- **Hammersley-Clifford** theorem: f is an MRF if and only if it has *Gibbs* distribution

$$Pr(f) \propto exp\left(-\sum_{c \in C} V_c(f)\right)$$

$$Pr(f) \propto exp\left(-\sum_{c \in C} V_c(f)\right)$$

- A *clique c* is a set of pixels
  $c = \{p_1, \ldots, p_k \mid p_i\text{'s are neighbors of each other}\}$
  C is the set of all cliques

- $V_c(f): L^m \rightarrow R$ is a *clique potential*

- *Clique potentials* $V_c(f)$ specify an MRF

MRF Example

- Only two-cliques are non-zero
- Two-clique potential $\{p,q\}$:

$$V_{pq}(f_p, f_q) = u_{pq} \cdot \delta(f_p \neq f_q)$$

$$\Downarrow$$

$$V_{pq}(f_p, f_q) = \begin{cases} 0 & \text{if } f_p = f_q \\ u_{pq} & \text{otherwise} \end{cases}$$

encourages piecewise-constant f

$$\boxed{Pr(f) \propto \exp\left(-\sum_{\{p,q\} \in N} u_{pq} \cdot \delta(f_p \neq f_q)\right)}$$

MAP-MRF estimation

- Given: observed data *d*, prior *Pr(f)* and likelihood function $Pr\left(d \middle| f\right)$

- *Maximize Posterior Probability (MAP)*
  $Pr(\ f \middle| d\ )$ over all $f \in L^m$

- By Bayes' law, $Pr(\ f \middle| d\ ) \propto Pr(\ d \middle| f\ )\, Pr(\ f\ )$

- MAP estimate:

$$\hat{f} = \underset{f \in L^m}{arg\ max}\ Pr(\ d \middle| f\ )\, Pr(\ f\ )$$

MAP-MRF estimation

- Likelihood function:

data for pixel p

$$Pr(d|f) = \prod_p D_p\left(d_p \middle| f_p\right)$$

- Maximization problem:

$$\underset{f \in L^m}{arg\ max}\ exp\left(\sum_p ln\, D_p\left(d_p \middle| f_p\right) - \sum_{\{p,q\}} u_{pq} \cdot \delta\left(f_p \neq f_q\right)\right)$$

- Equivalent to *posterior energy* minimization:

$$\boxed{\hat{f} = \underset{f}{arg\ min}\left(\sum_{\{p,q\} \in N} u_{pq} \cdot \delta\left(f_p \neq f_q\right) - \sum_p ln\, D_p\left(d_p \middle| f_p\right)\right)}$$

Comments:

- MAP-MRF approach for computer vision was first advocated by *Geman and Geman, 1984*

- High optimization cost. Standard methods, like simulated annealing, work very slow because typical image size is formidable

Comments:

- MAP-MRF approach for computer vision was first advocated by *Geman and Geman, 1984*
- Computation is only tractable in some special cases
- In general, high optimization cost. Standard methods, like simulated annealing, work very slow because typical image size is formidable

Belief Propagation (BP)

- Advocated by Pearl for tree-structured graphs in the context of Bayesian Nets
- Gives exact estimates in a tree graph
- Can be applied to general graphs, however
  - No optimality guarantee
  - May not converge
- There are 2 versions of BP
  - max-product, best to use if MAP estimate is needed
    - Map estimate is $\max_f Pr(f|d)$
  - sum-product, best to use if marginals at each node are needed
    - Marginal at node $\boldsymbol{p}$ is $Pr(f_p|d)$

Max-product BP

- Iterate until convergence:
  - Each pixel ***p*** (or node) sends a message vector (in parallel) to each of its neighobrs ***q***,

$$\lfloor m_{p\to q}(l_1) \quad m_{p\to q}(l_2) \quad .... \quad m_{p\to q}(l_k) \rfloor$$

where

$$m_{p\to q}(l) = min_{l'}\left[D_p(l') + V_{pq}(l,l') + \sum_{r\in N_p \setminus q} m_{r\to p}(l') \right]$$

- After "convergence" the final label at each pixel ***p*** is computed as:

$$arg\,min_l\left[D_p(l) + \sum_{q\in N_p} m_{q\to p}(l) \right]$$

Sum-product BP

- Iterate until convergence:
  - Each pixel ***p*** (or node) sends a message vector (in parallel) to each of its neighobrs ***q***,

$$\left[m_{p\to q}(l_1) \quad m_{p\to q}(l_2) \quad .... \quad m_{p\to q}(l_k) \right]$$

where

$$m_{p\to q}(l) = \sum_{l'} \left[D_p(l') V_{pq}(l', l) \prod_{r \in N_p \setminus q} \sum m_{r\to p}(l') \right]$$

- After "convergence" compute the belief for each label *l* at each pixel ***p***:

$$b_l(p) = D_p(l) \prod_{q \in N_p} m_{q\to p}(l)$$

BP

- For tree-structured graph, it can be shown that
  - max-product BP gives the MAP estimate of the MRF, that is $\max_f Pr(f \mid d)$
  - sum-product BP gives the correct posterior marginals that is $Pr(f_p|d) = b(p)$
- Start passing messages at the leafs moving up to the roof
- Compute the solution label (for max-product) or the belief vector (for sum-product) from the root down to the leafs

References for BP

- Several interesting speed-up techniques for max-product BP (and code) available from
  - "Efficient Belief Propagation for Early Vision", P.F. Felzenszwalb and D.Huttenlocher, CVPR2004
- "On the optimality of solutions of the max-product belief propagation algorithm in arbitrary graphs", Y. Weiss and W.Freeman, IEEE Transactions on Information Theory, 2001
- "Generalized Belief Propagation", J.S. Yedidia , W.T. Freeman, Y.Weiss to appear in NIPS 2000

Comments about BP on General Graphs

- Pros
  - Can be applied to any energy function and any neighborhood structure
  - Very easy to implement
- Cons
  - Not guaranteed to converge on (does go into infinite loops occasionally)
  - No optimality guarantees even if converges
  - Not very fast

Better Message Passing Algorithms

- Tree-reweighed message passing algorithms
  - "MAP estimation via agreement on hyper trees: Message-passing and linear-programming approaches", by Wainwright, Jaakkola, Willsky, in IEEE trans. On Infor. Theory, 2005
  - "Convergent tree-reweighted message passing for energy minimization", V. Kolmogorov, in International workshop on Artificial Intelligence and Statistics, 2005.