LECTURE 7

BAYESIAN METHODS

1. Introduction to Markov Random Fields and Gibbs Distribution
2. Optimization Methods
   - Simulated Annealing (SA)
   - Metropolis algorithm and Gibbs sampler
   - Iterated conditional modes (ICM)
   - Mean field Annealing (MFA)
3. MAP Motion Estimation
   - Basic Formulation
   - Discontinuity Models
   - Estimation Algorithms

MARKOV AND GIBBS RANDOM FIELDS

- MRFs are extensions of 1-D causal Markov chains to 2-D.
- MRFs were traditionally specified by local conditional probabilities which limited their usage. Recently it has been shown that every MRF can be described by a Gibbs distribution - hence the Gibbs random field (GRF).
- Bayesian estimation methods can be developed using GRFs as *a priori* signal models for complex image processing applications such as motion estimation and segmentation.
- Since Bayesian estimation requires global optimization of a cost function, we study a number of optimization methods including *simulated annealing* (SA), *iterative conditional mode* (ICM), and *highest confidence first* (HCF).

Definitions

- Let a random field $\mathbf{z} = \{z(\mathbf{x}), \mathbf{x} \in \Lambda\}$ be specified over a lattice Λ, and $\omega \in \Omega$ denote a realization of the random field $\mathbf{z}$.

The random field $z(\mathbf{x})$ can be continuous or discrete-valued, that is $\omega(\mathbf{x}) \in \mathbf{R}$ or $\omega(\mathbf{x}) \in \Gamma = \{0, 1, \ldots, L-1\}$, for all $\mathbf{x} \in \Lambda$, respectively.

- *A neighborhood system on* Λ.

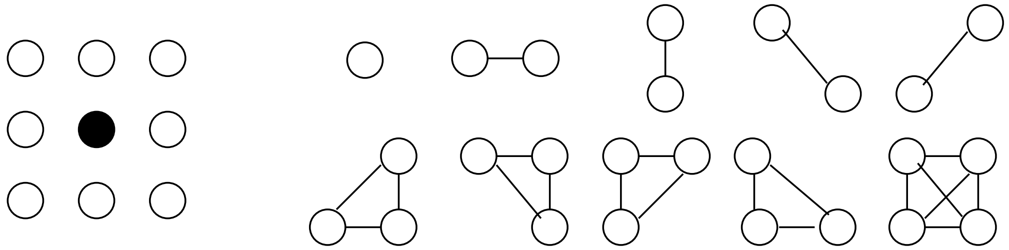
The set $\mathcal{N}_{\mathbf{x}}$ denotes the neighborhood of the site $\mathbf{x}$, and has the properties:
(i) $\mathbf{x} \notin \mathcal{N}_{\mathbf{x}}$, and
(ii) $\mathbf{x}_j \in \mathcal{N}_{\mathbf{x}_i} \leftrightarrow \mathbf{x}_i \in \mathcal{N}_{\mathbf{x}_j}$,
where $\mathbf{x}_i$ and $\mathbf{x}_j$ denote arbitrary sites in the lattice.

(In words, $\mathbf{x}$ does not belong to its own set of neighbors, and if $\mathbf{x}_j$ is a neighbor of $\mathbf{x}_i$, then $\mathbf{x}_i$ is a neighbor of $\mathbf{x}_j$, and vice versa.)

The neighborhood system over Λ is then defined as $\mathcal{N} = \{\mathcal{N}_x, \mathbf{x} \in \Lambda\}$

Examples of Neighborhood Systems

(a)



(b)

• *A clique* C is defined as $C \subseteq \Lambda$ such that all pairs of sites in C are neighbors. Further, $\mathcal{C}$ denotes the set of all cliques.

Markov Random Fields (MRF)

- The random field $\mathbf{z}$= { z($\mathbf{x}$) } is an MRF with respect to $\mathcal{N}$ if

$$p(\mathbf{z}) > 0, \quad \text{for all } \mathbf{z} = \omega,$$

and

$$p(z(\mathbf{x}_i) \mid z(\mathbf{x}_j), \forall \mathbf{x}_j \neq \mathbf{x}_i) = p(z(\mathbf{x}_i) \mid z(\mathbf{x}_j), \mathbf{x}_j \in \mathcal{N}_{\mathbf{x}_i}).$$

(In words, the first condition implies all realizations have non-zero pdf, while the second states that the conditional pdf at a particular site depends only on its neighborhood.)

- Difficulties with MRF models:

i) the joint pdf $p(\mathbf{z})$ cannot be easily related to local properties, and
ii) it is hard to determine when a set of functions

$$p(z(\mathbf{x}_i) \mid z(\mathbf{x}_j), \mathbf{x}_j \in \mathcal{N}_{\mathbf{x}_i}), \mathbf{x}_i \in \Lambda,$$

are valid conditional pdfs [Geman and Geman].

Gibbs Random Fields (GRF)

A GRF with a neighborhood system $\mathcal{N}$ and the associated set of cliques $\mathcal{C}$ is characterized by the joint pdf

- *discrete-valued*

$$p(\mathbf{z} = \omega) = \frac{1}{Q} \sum_{\omega} e^{-U(\mathbf{z}=\omega)/T} \delta(\mathbf{z} - \omega)$$

where

$$Q = \sum_{\omega} e^{-U(\mathbf{z}=\omega)/T}$$

- *continuous-valued*

$$p(\mathbf{z}) = \frac{1}{Q} e^{-U(\mathbf{z})/T}$$

where

$$Q = \int_{\omega} e^{-U(\mathbf{z})/T} d\mathbf{z}$$

and $U(\mathbf{z})$, the Gibbs potential (Gibbs energy) is defined by

$$U(\mathbf{z}) = \sum_{C \in \mathcal{C}} V_C(z(\mathbf{x}) \mid \mathbf{x} \in C).$$

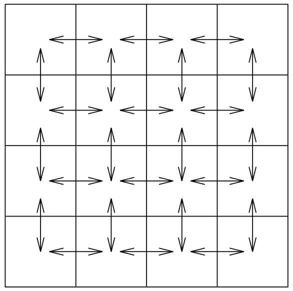
Example: Spatial smoothness constraint using GRF

Let us use a 4-point neighborhood system and the 2-pixel cliques. Over a 4×4 lattice, there are a total of 24 such cliques.

Let the 2-pixel clique potential be defined as

$$V_C(z(\mathbf{x}_i), z(\mathbf{x}_j)) = \begin{cases} -\beta & \text{if } \; z(\mathbf{x}_i) = z(\mathbf{x}_j) \\ +\beta & \text{otherwise} \end{cases} ,$$

where β is a positive number.



24 two-pixel cliques

(a)

2	2	2	2
2	2	2	2
2	2	2	2
2	2	2	2

V= -24 β

(b)

1	2	1	2
2	1	2	1
1	2	1	2
2	1	2	1

V = 24 β

(c)

Note that a lower potential means a higher probability.

Equivalence of GRF and MRF

Hammersley-Clifford (H-C) Theorem: Let $\mathcal{N}$ be a neighborhood system. Then $z(\mathbf{x})$ is an MRF with respect to $\mathcal{N}$ if and only if $p(\mathbf{z})$ is a Gibbsian with respect to $\mathcal{N}$.

- The H-C theorem provides us with a simple and practical way to specify MRFs through the Gibbs potentials.
- In general, the MRF is specified in terms of local conditional pdfs. Note that, there is no general method to obtain the joint pdf of an MRF from the local conditional pdfs [Besag].
- The Gibbs distribution gives the joint pdf of $\mathbf{z}$, which can be easily expressed in terms of the clique potentials which express the local interaction between pixels. They can be assigned arbitrarily.

Obtaining Local Conditional pdfs from Gibbs Potentials

◇ e.g., used in the Gibbs sampler method for optimization

- The local conditional pdf is defined as,

$$\begin{aligned} p(z(\mathbf{x}_i) \mid z(\mathbf{x}_j), \forall \mathbf{x}_j \neq \mathbf{x}_i) &= \frac{p(\mathbf{z})}{p(z(\mathbf{x}_j), \mathbf{x}_j \neq \mathbf{x}_i)} \\ &= \frac{p(\mathbf{z})}{\sum_{z(\mathbf{x}_i) \in \Gamma} p(\mathbf{z})}, \quad \forall \ \mathbf{x}_i \in \Lambda \end{aligned}$$

- After some algebra,

$$p(z(\mathbf{x}_i) \mid z(\mathbf{x}_j), \forall \mathbf{x}_j \neq \mathbf{x}_i) = Q_{\mathbf{x}_i}^{-1} e^{-\frac{1}{T} \sum_{C \mid \mathbf{x}_i \in C} V_C(z(\mathbf{x}) \mid \mathbf{x} \in C)},$$

where

$$Q_{\mathbf{x}_i} = \sum_{z(\mathbf{x}_i) \in \Gamma} e^{-\frac{1}{T} \sum_{C \mid \mathbf{x}_i \in C} V_C(z(\mathbf{x}) \mid \mathbf{x} \in C)}$$

OPTIMIZATION METHODS

• Many estimation/segmentation problems require the minimization of an energy function $E(\mathbf{d})$. We state the problem as

$$\hat{E} = min_{\mathbf{d}} E(\mathbf{d})$$

where $\mathbf{d}$ is some N-dimensional parameter vector. The value of $\mathbf{d}$ that results in the minimal E is denoted by

$$\hat{\mathbf{d}} = arg\,(min_{\mathbf{d}} E(\mathbf{d}))$$

• This minimization is exceedingly difficult for image processing applications due to both the dimensions of the vectors involved and the occurence of local minima because $E(\mathbf{d})$ is usually nonconvex.

Local vs. Global Minima

• *Gradient descent* suffers from a serious problem: its solution is strongly dependent on the starting point. If start in a "valley", it will be stuck at the bottom of that valley. We have no way of getting out of that local minimum to reach the "global" minimum.

• Here we look at several optimization methods that are capable of finding the global optimum.

A. Simulated annealing (stochastic relaxation)

Metropolis algorithm,
Gibbs sampler (by Geman and Geman).

B. Iterative conditional mode (ICM) (by Besag)

C. Mean field annealing (MFA) (by Bilbro *et al.*)

Simulated Annealing

• Simulated annealing, sometimes refered to as *stochastic relaxation*, belongs to the class of *Monte Carlo* methods.

• It enables us to find the global optimum of a nonconvex cost function of many variables.

• Here we describe two implementations,

- the original formulation of Metropolis and
- the Gibbs sampler proposed by Geman and Geman.

• The computational load of simulated annealing is usually significant especially when the number of elements in the unknown vector $\mathbf{d}$ and the number of values in the set Γ are large.

The Metropolis Algorithm

- We start at an arbitrary initial vector $\mathbf{d}$. At each iteration cycle, all components of $\mathbf{d}$ are perturbed one by one by assigning each another value in the set Γ randomly. Note that the order in which the components are perturbed is not important, as long as all components are perturbed in each iteration cycle. The change in the total energy, ΔE, due to the perturbation is computed after each perturbation to determine whether this perturbation is accepted.
- A perturbation is accepted with probability P given by

$$P = \begin{cases} exp(-\Delta E/T), & if\ \Delta E > 0 \\ 1, & if\ \Delta E \leq 0 \end{cases}$$

where T is the temperature parameter that controls the probability of our accepting positive changes in the energy. We always accept perturbations that lower the energy. The rationale behind accepting perturbations that increase the energy is to prevent the solution from settling in a local minimum.

- If T is relatively big, the probability of accepting a positive energy change is higher than when T is small, given the same ΔE. In the next iteration cycle, the temperature is lowered, and the components are revisited. The process continues until the temperature has been lowered to near zero.

- A temperature "schedule", expressing temperature as a function of the iteration number, is therefore an important component in the stochastic relaxation process. Geman and Geman proposed the following schedule

$$T = \frac{\tau}{\ln(k+1)},$$

where τ is a constant and k is the iteration cycle. This schedule is viewed as over conservative but guarantees a global minimum solution. Schedules that lower the temperature at a faster rate have been shown to work.

The Algorithm

1. Choose an initial value for $\mathbf{d} = \mathbf{d}^{(0)}$. Set $i = 0$ and $j = 1$.

2. Perturb the j^{th} component of $\mathbf{d}^{(i)}$ to generate the vector $\mathbf{d}^{(i+1)}$.

3. Compute $\Delta E = E(\mathbf{d}^{(i+1)}) - E(\mathbf{d}^{(i)})$.

4. Compute P from

$$P = \begin{cases} exp(-\Delta E/T), & if\ \Delta E > 0 \\ 1, & if\ \Delta E \leq 0 \end{cases}$$

5. If $P < 1$, then draw a random number that is uniformly distributed between 0 and 1. If the number drawn is less than P accept the perturbation.

6. Set $j = j + 1$. If $j \leq N$, go to 2. (N is the number of components of $\mathbf{d}$).

7. Set $i = i + 1$ and $j = 1$. Reduce T according to a temperature schedule. If $T > T_{min}$, go to 2. Otherwise terminate.

The Gibbs Sampler

- In Gibbs sampling, instead of making random perturbations and then deciding whether to accept or reject this perturbation, the new value is "drawn from" the distribution of $P(\mathbf{d})$ and is always accepted.
- First compute the conditional probability of the component $d(\mathbf{x}_i)$ to take each of the values in the set Γ given the present values of its neighbors using

$$P(d(\mathbf{x}_i) = \gamma \mid d(\mathbf{x}_j), \mathbf{x}_j \neq \mathbf{x}_i) = Q_{\mathbf{x}_i}^{-1} e^{-\frac{1}{T} \sum_{C|\mathbf{x}_i \in C} V_C(d(\mathbf{x})|\mathbf{x} \in C)},$$

where

$$Q_{\mathbf{x}_i} = \sum_{\gamma \in \Gamma} e^{-\frac{1}{T} \sum_{C|\mathbf{x}_i \in C} V_C(d(\mathbf{x})|\mathbf{x} \in C)}$$

- Then, the new value of the component $d(\mathbf{x}_i)$ is drawn from this conditional probability distribution.

• To clarify the meaning of "drawn from", suppose that the sample space $\Gamma = \{0, 1, 2, \text{ and } 3\}$, and it was found that

$$
\begin{aligned}
P(d(\mathbf{x}_i) = 0 \mid d(\mathbf{x}_j),\ \mathbf{x}_j \neq \mathbf{x}_i) &= 0.2, \\
P(d(\mathbf{x}_i) = 1 \mid d(\mathbf{x}_j),\ \mathbf{x}_j \neq \mathbf{x}_i) &= 0.1, \\
P(d(\mathbf{x}_i) = 2 \mid d(\mathbf{x}_j),\ \mathbf{x}_j \neq \mathbf{x}_i) &= 0.4, \ \textit{and} \\
P(d(\mathbf{x}_i) = 3 \mid d(\mathbf{x}_j),\ \mathbf{x}_j \neq \mathbf{x}_i) &= 0.3,
\end{aligned}
$$

A uniform random number, R, between 0 and 1 is generated.
If $0 \leq R \leq 0.2$ then $d(\mathbf{x}_i) = 0$, if $0.2 < R \leq 0.3$ then $d(\mathbf{x}_i) = 1$, if $0.3 < R \leq 0.7$ then $d(\mathbf{x}_i) = 2$, and if $0.7 < R \leq 1$ then $d(\mathbf{x}_i) = 3$.

• Properties of perturbations through Gibbs sampling:

(i) for any initial estimate, updating using the Gibbs sampler yields an asymptotically Gibbsian distribution. This result can be used to simulate a Gibbs random field with specified parameters.

(ii) for a specified temperature schedule, the maximum of the Gibbs distribution will be reached. Although this property is significant for MAP estimation, the specified temperature schedule may be too slow for use in practice.

Iterated Conditional Modes (ICM)

• ICM, also referred to as the greedy algorithm, is motivated by a need to reduce the computational load produced by stochastic relaxation or Gibbs sampling.

• Here, the sites are again visited one-by-one in some cyclic fashion, except there is no temperature change involved. The temperature T is set to zero, $T = 0$ for all iterations. Therefore, ICM is also refered to as the "instant freezing" case of simulated annealing.

• Refering to the equation of acceptance probability in SA, ICM only allows perturbations that provide negative ΔE, since $T = 0$ effectively gives a zero probability for accepting positive energy changes.

• Notice that due to this, solutions from ICM is likely to get trapped in local minima, and there is no guarantee that a global minimum can be reached.

• It can be shown that ICM converges to the solution that maximizes the local conditional probabilities

$$P(d(\mathbf{x}_i) = \gamma \mid d(\mathbf{x}_j), \mathbf{x}_j \neq \mathbf{x}_i) = Q_{\mathbf{x}_i}^{-1} e^{-\frac{1}{T} \sum_{C \mid \mathbf{x}_i \in C} V_C(d(\mathbf{x}) \mid \mathbf{x} \in C)},$$

where

$$Q_{\mathbf{x}_i} = \sum_{\gamma \in \Gamma} e^{-\frac{1}{T} \sum_{C \mid \mathbf{x}_i \in C} V_C(d(\mathbf{x}) \mid \mathbf{x} \in C)}$$

at each site. Thus, ICM is usually implemented as in Gibbs sampling but by choosing the value at each site that gives the maximum local conditional probability.

• ICM provides a much faster convergence than SA. Also, when the initial solution is a resonable estimate from other means rather than completely random, ICM reaches an acceptable solution in relatively few iterations. ICM produces good results for several applications that include image restoration [see Besag] and image segmentation [see Pappas].

Mean Field Annealing (MFA)

• Mean field annealing (MFA) originates from the "mean field approximation" idea in statistical mechanics.

• The main idea is that in describing the interaction between a pixel and its neighbors, we use the mean values of the neighboring pixels. Thus, MFA is an approximation to simulated annealing, and it enables replacing the random search with a deterministic gradient descent.

• The implementation of MFA is not unique. Details can be found in the references. In particular, [Snyder] is a good tutorial on several optimization methods.

• Other references: [Geman and Geman] discusses the GRF/MRF equivalence. Vigorous treatment of the statistical formulations can be found in [Besag] and [Spitzer].

BAYESIAN MOTION ESTIMATION

Let

$\mathbf{s}_k = \{s_k(\mathbf{x})\}, \quad \mathbf{x} \in \Lambda$, denote the kth frame of video,
$\mathbf{d}(\mathbf{x}) = [d_1(\mathbf{x}) \; d_2(\mathbf{x})]^T$ denote the displacement vector at site $\mathbf{x}$, and
$\mathbf{d}_1 = \{\mathbf{d}_1(\mathbf{x})\}$ and $\mathbf{d}_2 = \{\mathbf{d}_2(\mathbf{x})\}$ for $\mathbf{x} \in \Lambda$, denote the lexicographic ordering of the x_1 and x_2 components of the displacement field from frame $k-1$ to k, respectively

$$\text{i.e.,} \quad s_k(\mathbf{x}) = s_{k-1}(\mathbf{x} - \mathbf{d}(\mathbf{x})).$$

Then, the problem of motion estimation can be formulated as:

given $\mathbf{s}_k$ and $\mathbf{s}_{k-1}$, find an estimate of $\mathbf{d}_1$ and $\mathbf{d}_2$.

The maximum a posteriori probability (MAP) estimates of $\mathbf{d}_1$ and $\mathbf{d}_2$ are given by:

$$(\hat{\mathbf{d}}_1, \hat{\mathbf{d}}_2) = \arg\max_{\mathbf{d}_1, \mathbf{d}_2} p(\mathbf{d}_1, \mathbf{d}_2 | \mathbf{s}_k, \mathbf{s}_{k-1})$$

From Bayes formula

$$p(\mathbf{d}_1, \mathbf{d}_2 | \mathbf{s}_k, \mathbf{s}_{k-1}) = \frac{p(\mathbf{s}_k | \mathbf{d}_1, \mathbf{d}_2, \mathbf{s}_{k-1}) p(\mathbf{d}_1, \mathbf{d}_2 | \mathbf{s}_{k-1})}{p(\mathbf{s}_k | \mathbf{s}_{k-1})}$$

Since the denominator is not a function of $\mathbf{d}_1$ and $\mathbf{d}_2$,

$$(\hat{\mathbf{d}}_1, \hat{\mathbf{d}}_2) = \arg\max_{\mathbf{d}_1, \mathbf{d}_2} p(\mathbf{s}_k | \mathbf{d}_1, \mathbf{d}_2, \mathbf{s}_{k-1}) p(\mathbf{d}_1, \mathbf{d}_2 | \mathbf{s}_{k-1})$$

or

$$(\hat{\mathbf{d}}_1, \hat{\mathbf{d}}_2) = \arg\max_{\mathbf{d}_1, \mathbf{d}_2} p(\mathbf{s}_{k-1} | \mathbf{d}_1, \mathbf{d}_2, \mathbf{s}_k) p(\mathbf{d}_1, \mathbf{d}_2 | \mathbf{s}_k)$$

- The term $p(\mathbf{s}_k | \mathbf{d}_1, \mathbf{d}_2, \mathbf{s}_{k-1})$ is the conditional pdf, or the "consistency (likelihood) measure", that measures how well the estimates of $\mathbf{d}_1$, $\mathbf{d}_2$ explain the observations $\mathbf{s}_k$ given $\mathbf{s}_{k-1}$.
- The term $p(\mathbf{d}_1, \mathbf{d}_2 | \mathbf{s}_{k-1})$ is the *a priori* probability density that is modeled by a GRF, by specifying the clique potential functions according to the desired local properties of $(\hat{\mathbf{d}}_1, \hat{\mathbf{d}}_2)$.

Discontinuity Models

- Let us introduce two auxilary fields, the occlusion field $\mathbf{o}$, and the line field $\mathbf{l}$ to model the occlusion/uncovered areas, and the optical flow boundaries respectively, in order to improve the motion estimation results.

The occlusion field

$$\mathbf{o} = o(\mathbf{x}), \; \mathbf{x} \in \Lambda,$$

$$o(\mathbf{x}) = \begin{cases} 0 & \mathbf{d}(\mathbf{x}) \text{ is well defined} \\ 1 & \mathbf{x} \text{ is an occlusion point} \end{cases}$$

The line field

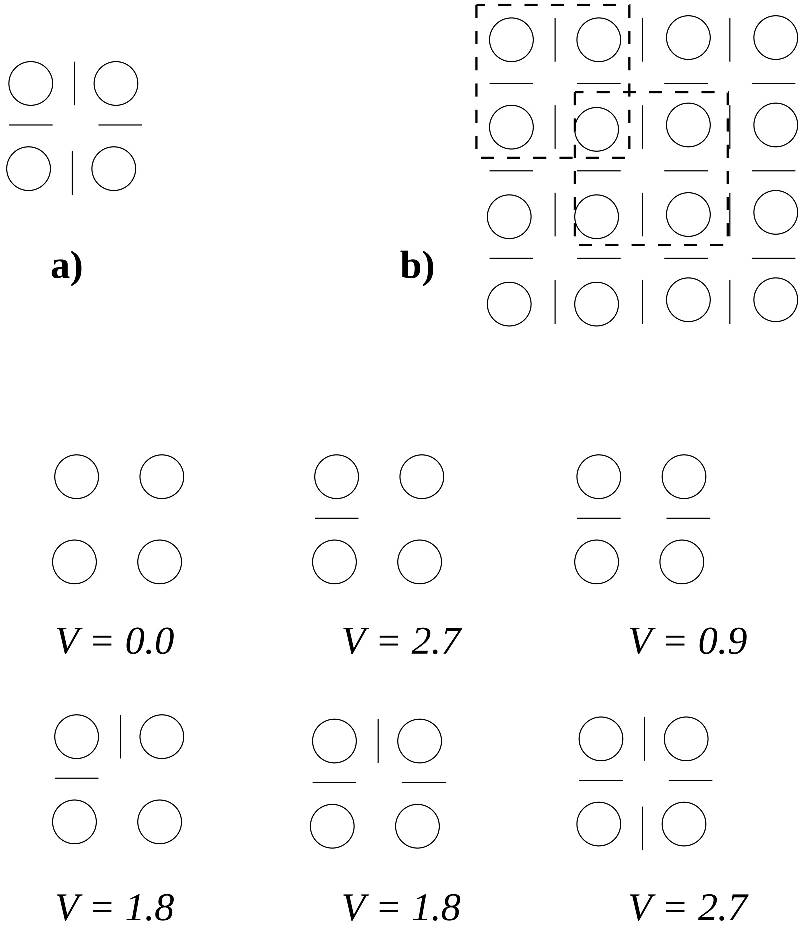
The *a priori* pdf $p(\mathbf{d}_1, \mathbf{d}_2 | \mathbf{s}_{k-1})$ is usually chosen to favor a globally smooth motion field. To allow for the presence of discontinuities in the motion field, we make use of the *line process.*

The line field $\mathbf{l}(\mathbf{x}_i, \mathbf{x}_j)$ models the horizontal and vertical discontinuities in the motion field (optical flow) between the sites $\mathbf{x}_i$ and $\mathbf{x}_j$ as

$$l(\mathbf{x}_i, \mathbf{x}_j) = \begin{cases} 1 & \text{if there is a discontinuity} \\ & \text{between } \mathbf{d}(\mathbf{x}_i)\text{and } \mathbf{d}(\mathbf{x}_j) \\ 0 & \text{otherwise.} \end{cases}$$

- The line process, $\mathbf{l}$ conceptually occupies the *dual lattice* which has sites for lines between every pair of pixel sites. The state of each line site can be either *ON* ($l = 1$) or *OFF* ($l = 0$), expressing the presence and absence of a discontinuity, respectively.
- Nonnegative potentials are assigned to each rotation invariant line clique configuration to penalize excessive use of the “ON” state.

Example: Line Process Clique Potentials

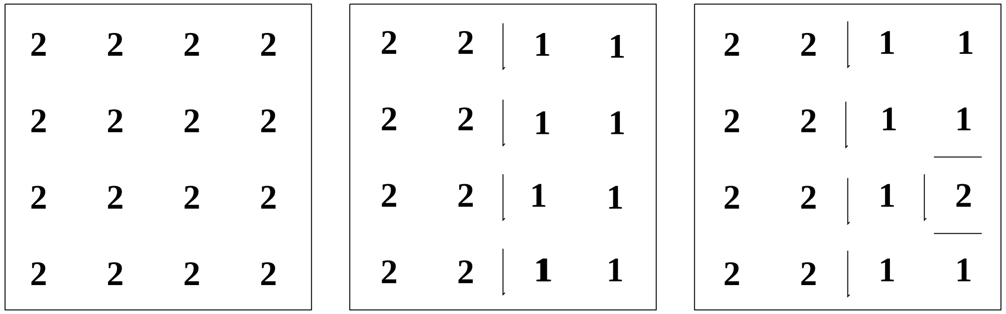


c)

An image with 4×4 pixel sites has 9 distinct 4-line cliques.

Example: Prior probabalities with and without the line field

The prior potentials slightly penalize straight lines ($V = 0.9$), penalize corners ($V = 1.8$) and "T" junctions ($V = 1.8$), and heavily penalize end of a line ($V = 2.7$) and "crosses" ($V = 2.7$).



The likelihood potential function puts no penalty on dissimilar pixel pairs if the line site in between is ON, and puts different amounts of penalty on different line configurations, reflecting our *a priori* expectation of their occurence.

• With the introduction of the auxiliary fields, the MAP estimate of $\{\mathbf{d}_1, \mathbf{d}_2, \mathbf{o}, \mathbf{l}\}$ is given by:

$$\{\hat{\mathbf{d}}_1, \hat{\mathbf{d}}_2, \hat{\mathbf{o}}, \hat{\mathbf{l}}\} = \arg\max_{\mathbf{d}_1, \mathbf{d}_2, \mathbf{o}, \mathbf{l}} p(\mathbf{d}_1, \mathbf{d}_2, \mathbf{o}, \mathbf{l} | \mathbf{s}_k, \mathbf{s}_{k-1})$$

Using the Bayes rule, and the symmetry of the expression

$$\{\hat{\mathbf{d}}_1, \hat{\mathbf{d}}_2, \hat{\mathbf{o}}, \hat{\mathbf{l}}\} = \arg\max_{\mathbf{d}_1, \mathbf{d}_2, \mathbf{o}, \mathbf{l}} p(\mathbf{s}_{k-1} | \mathbf{d}_1, \mathbf{d}_2, \mathbf{o}, \mathbf{l}, \mathbf{s}_k) p(\mathbf{d}_1, \mathbf{d}_2, \mathbf{o}, \mathbf{l} | \mathbf{s}_k)$$

Next, we discuss the likelihood (consistency) and the *a priori* probability models.

The Likelihood Model

- Assuming

-the change in the illumination from frame to frame is insignificant, and
- that there is no occlusion/uncovered areas,

the change in the intensity of a pixel along the motion trajectory is due to observation noise.

Modeling the observation noise as white, Gaussian, we have

$$p(\mathbf{s}_{k-1}|\mathbf{d}_1, \mathbf{d}_2, \mathbf{l}, \mathbf{s}_k) = C \ \exp\left\{-\sum_{\mathbf{x}\in\Lambda} \frac{(s_k(\mathbf{x}) - s_{k-1}(\mathbf{x} - \mathbf{d}(\mathbf{x})))^2}{2\sigma^2}\right\}$$

where C is some constant.

- Now, taking the occlusion points into account

$$p(\mathbf{s}_{k-1}|\mathbf{d}_1, \mathbf{d}_2, \mathbf{o}, \mathbf{l}, \mathbf{s}_k) =$$
$$C \ \exp\left[-\sum_{\mathbf{x}\in\Lambda} \frac{(1-o(\mathbf{x}))(s_k(\mathbf{x}) - s_{k-1}(\mathbf{x}-\mathbf{d}(\mathbf{x})))^2}{2\sigma^2}\right]$$

- This pdf can be expressed more compactly in terms of an "energy function"

$$p(\mathbf{s}_{k-1}|\mathbf{d}_1, \mathbf{d}_2, \mathbf{o}, \mathbf{l}, \mathbf{s}_k) = C \exp\left[-U(\mathbf{s}_k|\mathbf{d}_1, \mathbf{d}_2, \mathbf{o}, \mathbf{s}_{k-1})\right]$$

where

$$U(\mathbf{s}_k|\mathbf{d}_1, \mathbf{d}_2, \mathbf{o}, \mathbf{s}_{k-1}) = \frac{1}{2\sigma^2} \sum_{\mathbf{x}\in\Lambda} (1-o(\mathbf{x}))\left(s_k(\mathbf{x}) - s_{k-1}(\mathbf{x}-\mathbf{d}(\mathbf{x}))\right)^2$$

The Prior Model

The prior model incorporates the location of the optical flow boundaries and the occlusion/uncovered areas while dictating that the flow vectors vary smoothly within each optical flow boundary.

The *a priori* model can be expressed as

$$p(\mathbf{d}_1, \mathbf{d}_2, \mathbf{o}, \mathbf{l} | \mathbf{s}_k) = exp\left[-U(\mathbf{d}_1, \mathbf{d}_2, \mathbf{o}, \mathbf{l} | \mathbf{s}_k)\right]$$

where

$$\begin{aligned} U(\mathbf{d}_1, \mathbf{d}_2, \mathbf{o}, \mathbf{l} | \mathbf{s}_k) &= \lambda_d U(\mathbf{d}_1, \mathbf{d}_2 | \mathbf{l}) + \lambda_s U(\mathbf{o} | \mathbf{l}) + \lambda_l U(\mathbf{l} | \mathbf{o}) \\ &= \lambda_d \sum_{c \in \mathcal{C}_d} V_c(\mathbf{d}_1, \mathbf{d}_2 | \mathbf{l}) + \lambda_s \sum_{c \in \mathcal{C}_o} V_c(\mathbf{o} | \mathbf{l}) + \lambda_l \sum_{c \in \mathcal{C}_l} V_c(\mathbf{l} | \mathbf{s}_k) \end{aligned}$$

Here $\mathcal{C}_d$, $\mathcal{C}_o$ and $\mathcal{C}_l$ denote the sets of all cliques for the displacement, occlusion and line fields, respectively, $V_c(.)$ represent the corresponding clique function, and λ_d, λ_o and λ_l are positive constants.

ESTIMATION ALGORITHMS

The minimization of the overall potential is an exceedingly difficult problem,

- there are several hundreds of thousands of unknowns for a reasonable size image, and
- the criterion function is nonconvex.

For example, for a 256×256 image, there are 65,536 motion vectors (131,072 components), 65,536 occlusion labels, and 131,072 line field labels for a total of 327,680 unknowns.

An additional complication is that the motion vector components are continuous-valued, and the occlusion and line field labels are discrete-valued.

Three-step iteration of Dubois and Konrad:

1. Given the best estimates of the auxilary field $\hat{\mathbf{o}}$ and $\hat{\mathbf{l}}$, update the motion field $\mathbf{d}_k$ by minimizing

$$\min_{\mathbf{d}_1,\mathbf{d}_2} U_g(\mathbf{g}_k, \mathbf{d}_1, \mathbf{d}_2, \hat{\mathbf{o}}, \mathbf{g}_{k-1}) + \lambda_d U_d(\mathbf{d}_1, \mathbf{d}_2, \hat{\mathbf{l}}, \mathbf{g}_k)$$

This minimization can be done by Gauss-Newton optimization.

2. Given the best estimates of $\hat{\mathbf{d}}_1$, $\hat{\mathbf{d}}_2$ and $\hat{\mathbf{l}}$, update $\mathbf{o}$ by minimizing

$$\min_{\mathbf{o}} U_g(\mathbf{g}_k, \hat{\mathbf{d}}_1, \hat{\mathbf{d}}_2, \mathbf{o}, \mathbf{g}_{k-1}) + \lambda_o U_o(\mathbf{o}, \hat{\mathbf{l}}, \mathbf{g}_k)$$

An exhaustive search or the ICM method can be employed to solve this step.

3. Finally, given the best estimates of $\hat{\mathbf{d}}_1$, $\hat{\mathbf{d}}_2$ and $\hat{\mathbf{o}}$, update $\mathbf{l}$ by minimizing

$$\min_{\mathbf{l}} \lambda_d U_d(\hat{\mathbf{d}}_1, \hat{\mathbf{d}}_2, \mathbf{l}, \mathbf{g}_{k-1}) + \lambda_o U_o(\hat{\mathbf{o}}, \mathbf{l}, \mathbf{g}_k) + \lambda_l U_l(\mathbf{l}, \mathbf{g}_k)$$

Once all three fields are updated, the process is repeated until a suitable criterion of convergence is satisfied. This procedure has been reported to give good results.