# Lecture 12: Binocular Stereo and Belief Propagation

A.L. Yuille

March 11, 2012

## 1 Introduction

Binocular Stereo is the process of estimating three-dimensional shape (stereo) from two eyes (binocular) or two cameras. Usually it is just called stereo. It requires knowing the camera parameters (e.g., focal length, direction of gaze), as discussed in earlier lecture, and solving the *correspondence problem* – matching pixels between the left and right images. After correspondence has been solved then depth can be estimated by trigonometry (earlier lecture). The depth is inversely proportional to the *disparity*, which is the relative displacement between corresponding points. This lecture will discuss how to solve the correspondence to estimate the disparity can be formulated in terms of a probabilistic markov model which assumes that the surfaces of surfaces are piecewise smooth (like the weak membrane model).

The epipolar line constraint (see earlier lecture) means that points in each image can only match to points on a one-dimensional line in the second image (provided the camera parameters are known). This enables a simplified formulation of stereo in terms of a series of independent one-dimensional problems which can each be formulated as inference on a graphical model without closed loops. Hence inference can be performed by dynamic programming (see earlier lecture).

A limitation of these one-dimensional models is that they assume that the disparity/depth of neighboring pixels is independent unless they lie on the same epipolar line. This is a very restrictive assumption and it is better to impose dependence (e.g., smoothness) across the epipolar lines. But this prevents the use of dynamic programming. This motivates the use of the belief propagation (BP) algorithm which perform similarly to dynamic programming on models defined over graph structures without closed loops (i.e. are guaranteed to converge to the optimal estimate), but which also work well in practice as approximate inference algorithms on graphs with closed loops. The intuitive reasoning is that the one-dimensional models (which do dynamic programming) perform well on real images, hence introducing cross epipolar line terms will only have limited effects, and so belief propagation is likely to converge to the correct result (e.g., doing DP on the one-dim models which put you in the domain of attraction of the BP algorithm for the full model).

This lecture will also introduce the belief propagation (BP) algorithm. In addition (not covered in class) we will describe the TRW and related algorithms. Note that since stereo is formulated as a Markov Model we can use a range of other algorithms to perform inference (such as the algorithms described earlier). Similarly we can apply BP to other Markov Models in vision. (Note that BP applies to Markov Models with pairwise connections but this can be extended to generalized belief propagation GBP which applies to Markov Models with higher order connections).

Finally we point out a limitation of many current stereo algorithms. As known by Leonardo da Vinci, depth discontinuities in shapes can cause some points to be visible to one eye only, see figure (1). This is called half-occlusion. It affects the correspondence problem because it means that points in one eye may not have matching points in the other eye. Current stereo algorithms are formulated to be robust to this problem. However, the presence of unmatched points can be used as a cue to determine depth discontinuities and hence can yield information (this was done for one-dimensional models of stereo – Geiger, Ladendorf, and Yuille. Belhumeur and Mumford – which also introduced DP to this problem – except there was earlier work by Cooper and maybe Kanade?).

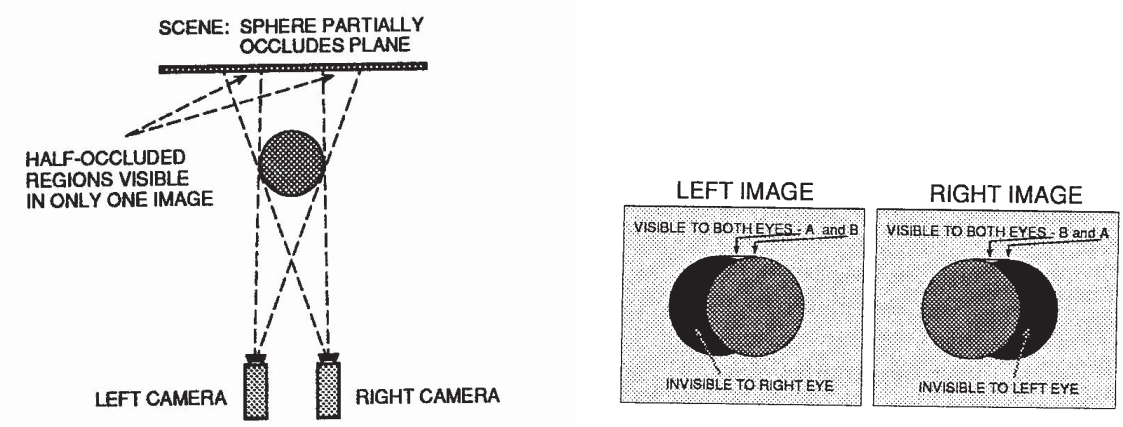


Figure 1: Half occluded points are only visible to one eye/camera (left panel). Hence some points will not have corresponding matches in the other eye/camera (right panel).

## 2 Stereo as MRF/CRF

We can model Stereo as a Markov Random Field (MRF), see figure (2), with input $\mathbf{z}$ and output $\mathbf{x}$. The input is the input images to the left and right cameras, $\mathbf{z} = (\mathbf{z}^L, \mathbf{z}^R)$, and the output is a set of disparities $\mathbf{x}$ which specify the relative displacements between corresponding pixels in the two images and hence determine the depth, see figure (3) (depth is inversely proportional to the disparity).

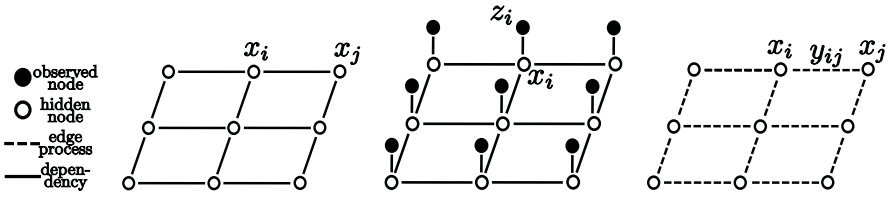


Figure 1: MRF graphs.

Figure 2: GRAPHS for different MRF's. Conventions (far left), basic MRF graph (middle left), MRF graph with inputs $z_i$ (middle right), and graph with lines processors $y_{ij}$ (far right).

We can model this by a posterior probability distribution $P(\mathbf{x}|\mathbf{z})$ and hence is a conditional random field [13]. This distribution is defined on a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where the set of nodes $\mathcal{V}$ is the set of image pixels $\mathcal{D}$ and the edges $\mathcal{E}$ are between neighboring pixels – see figure (2). The $\mathbf{x} = \{x_i : i \in \mathcal{V}\}$ are random variables specified at each node of the graph. $P(\mathbf{x}|\mathbf{z})$ is a Gibbs distribution specified by an energy function $E(\mathbf{x}, \mathbf{z})$ which contains unary potentials $U(\mathbf{x}, \mathbf{z}) = \sum_{i \in \mathcal{V}} \phi(x_i, \mathbf{z})$ and pairwise potentials $V(\mathbf{x}, \mathbf{x}) = \sum_{ij \in \mathcal{E}} \psi_{ij}(x_i, x_j)$. The unary potentials $\phi(x_i, \mathbf{z})$ depend only on the disparity at node/pixel $i$ and the dependence on the input $\mathbf{z}$ will depend on the application. For binocular stereo, we can set $\phi(x_i, \mathbf{z}^L, \mathbf{z}^R) = |f(\mathbf{z}^L)_i - f(\mathbf{z}^R)_{i+x_i}|$, where $f(.)$ is a vector-value filter and $|.|$ is the L1-norm, – $\psi_{ij}(x_i, x_j) = |x_i - x_j|$ – so that $\phi(.)$ takes small values at the disparities $x_i$ for which the filter responses are similar on the two images. The pairwise potentials impose prior assumptions about the local 'context' of the disparities. These models typically assume that neighboring pixels will tend to have similar disparities – see figure (3).

In summary, the model is specified by a distribution $P(\mathbf{x}|\mathbf{z})$ defined over discrete-valued random variables

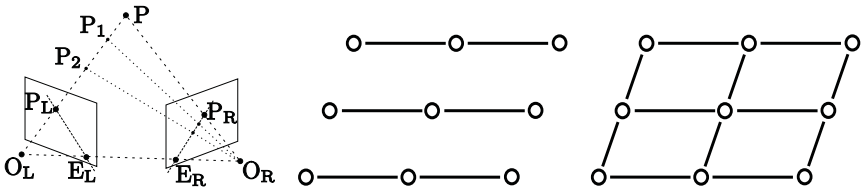


Figure 2: Stereo.

Figure 3: Stereo. The geometry of stereo (left). A point $P$ in 3-D space is projected onto points $P_L, P_R$ in the left and right images. The projection is specified by the focal points $O_L, O_R$ and the directions of gaze of the cameras (the camera geometry). The geometry of stereo enforces that points in the plane specified by $P, O_L, O_R$ must be projected onto corresponding lines $E_L, E_R$ in the two images (the epipolar line constraint). If we can find the correspondence between the points on epipolar lines then we can use trigonometry to estimate their depth, which is (roughly) inversely proportional to the disparity, which is the relative displacement of the two images. Finding the correspondence is usually ill-posed unless and requires making assumptions about the spatial smoothness of the disparity (and hence of the depth). Current models impose weak smoothness priors on the disparity (center). Earlier models assumed that the disparity was independent across epipolar lines which lead to similar graphic models (right) where inference could be done by dynamic programming.

$\mathbf{x} = \{x_i : i \in \mathcal{V}\}$ defined on a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$:

$$P(\mathbf{x}|\mathbf{z}) = \frac{1}{Z(\mathbf{z})} \exp\{-\sum_{i \in \mathcal{V}} \phi_i(x_i, \mathbf{z}) - \sum_{ij \in \mathcal{E}} \psi_{ij}(x_i, x_j)\}. \tag{1}$$

The goal will be to estimate properties of the distribution such as the MAP estimator and the marginals (which relate to each other),

$$\begin{aligned} \mathbf{x}^* &= \arg\max_{\mathbf{x}} P(\mathbf{x}|\mathbf{z}), \text{ the MAP estimate,} \\ p_i(x_i) &= \sum_{\mathbf{x}/i} P(\mathbf{x}|\mathbf{z}), \ \forall i \in \mathcal{V} \text{ the marginals.} \end{aligned} \tag{2}$$

This model can be simplified using the epipolar line constraint, see figure (3). We drop the vertical binary potentials and reformulate the graph in terms of a set of non-overlapping graphs corresponding to different epipolar lines (center panel). We write $\mathcal{V} = \bigcup_l \mathcal{V}_l$ and $\mathcal{E} = \bigcup_l \mathcal{E}_l$, where $\mathcal{V}_l$ denotes the pixels on corresponding epipolar lines $l$ and $\mathcal{E}_l$ denotes the edges along the epipolar lines – i.e. we enforce smoothness only in the horizontal direction (center panel) and not in both the horizontal and vertical directions (right panel). This gives us a set of one-dimensional models that can be used to estimate the disparities of each epipolar line independently, and hence inference can be performed by dynamic programming. The models are of form:

$$P(\mathbf{x}_l|\mathbf{z}_l) = \frac{1}{Z(\mathbf{z}_l)} \exp\{-\sum_{i \in \mathcal{V}_l} \phi_i(x_i, \mathbf{z}_l) - \sum_{ij \in \mathcal{E}_l} \psi_{ij}(x_i, x_j)\}, \tag{3}$$

and we can estimate $\mathbf{x}_l^* = \arg\max P(\mathbf{x}_l|\mathbf{z}_l)$ for each $l$ independently.

The advantages of exploiting the epipolar line constraint are computational (i.e. DP). But this simplification has a price. Firstly, it assumes that the epipolar lines are known exactly (while in practice they are only known to limited precision). Secondly, it assumes that there are no smoothness constraints across epipolar lines. Nevertheless, good results can be obtained using this approximation so it can be thought of a type of first order approximation which helps justify using approximate inference algorithms like BP. (Note we can use more sophisticated versions of DP – e.g. Geiger and Ishikawa).

# 3 Belief Propagation

We now present a different approach to estimating (approximate) marginals and MAPs of an MRF. This is called belief propagation BP. It was originally proposed as a method for doing inference on trees (e.g. graphs without closed loops) [15] for which it is guaranteed to converge to the correct solution (and is related to dynamic programming). But empirical studies showed that belief propagation will often yield good approximate results on graphs which do have closed loops [14].

To illustrate the advantages of belief propagation, consider the binocular stereo problem which can be addressed by using the first type of model. For binocular stereo there is the epipolar line constraint which means that, provided we know the camera geometry, we can reduce the problem to one-dimensional matching, see figure (3). We impose weak smoothness in this dimension only and then use dynamic programming to solve the problem [9]. But a better approach is to impose weak smoothness in both directions which can be solved (approximately) using belief propagation [18], see figure (3).

Surprisingly the fixed points of belief propagation algorithms correspond to the extreme of the Bethe free energy [20]. This free energy, see equation (9), appears better than the mean field theory free energy because it includes pairwise pseudo-marginal distributions and reduces to the MFT free energy if these are replaced by the product of unary marginals. But, except for graphs without closed loops (or a single closed loop), there are no theoretical results showing that the Bethe free energy yields a better approximation than mean field theory. There is also no guarantee that BP will converge for general graphs and it can oscillate widely.

## 3.1 Message Passing

BP is defined in terms of messages $m_{ij}(x_j)$ from $i$ to $j$, and is specified by the sum-product update rule:

$$m_{ij}^{t+1}(x_j) = \sum_{x_i} \exp\{-\psi_{ij}(x_i, x_j) - \phi_i(x_i)\} \prod_{k \neq j} m_{ki}^t(x_i). \tag{4}$$

The unary and binary pseudomarginals are related to the messages by:

$$b_i^t(x_i) \propto \exp\{-\phi_i(x_i)\} \prod_k m_{kj}^t(x_j), \tag{5}$$

$$\begin{aligned} b_{kj}^t(x_k, x_j) &\propto \exp\{-\psi_{kj}(x_k, x_j) - \phi_k(x_k) - \phi_j(x_j)\} \\ &\times \prod_{\tau \neq j} m_{\tau k}^t(x_k) \prod_{l \neq k} m_{lj}^t(x_j). \end{aligned} \tag{6}$$

The update rule for BP is not guaranteed to converge to a fixed point for general graphs and can sometimes oscillate wildly. It can be partially stabilized by adding a damping term to equation (4). For example, by multiplying the right hand side by $(1 - \epsilon)$ and adding a term $\epsilon m_{ij}^t(x_j)$.

To understand the converge of BP observe that the pseudo-marginals $b$ satisfy the *admissibility constraint*:

$$\frac{\prod_{ij} b_{ij}(x_i, x_j)}{\prod_i b_i(x_i)^{n_i - 1}} \propto \exp\{-\sum_{ij} \psi_{ij}(x_i, x_j) - \sum_i \phi(x_i)\} \propto P(\mathbf{x}), \tag{7}$$

where $n_i$ is the number of edges that connect to node $i$. This means that the algorithm re-parameterizes the distribution from an initial specification in terms of the $\phi, \psi$ to one in terms of the pseudo-marginals $b$. For a tree, this re-parameterization is exact (i.e. the pseudo-marginals become the true marginals of the distribution – e.g., we can represent a one-dimensional distribution by $P(\mathbf{x}) = \frac{1}{Z}\{-\sum_{i=1}^{N-1} \psi(x_i, x_{i+1}) - \sum_{i=1}^{N} \phi_i(x_i)\}$ or by $\prod_{i=1}^{N-1} p(x_i, x_{i+1}) / \prod_{i=2}^{N-1} p(x_i)$.

It follows from the message updating equations (4,6) that at convergence, the $b$'s satisfy the *consistency constraints*:

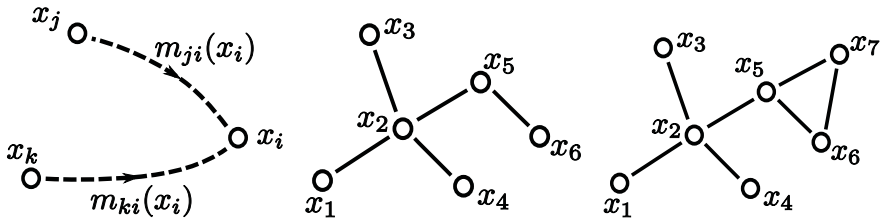


Figure 5: Belief propagation

Figure 4: Message passing (left) is guaranteed to converge to the correct solution on graphs without closed loops (center) but only gives good approximations on graphs with a limited number of closed loops (right).

$$\sum_{x_j} b_{ij}(x_i, x_j) = b_i(x_i), \quad \sum_{x_i} b_{ij}(x_i, x_j) = b_j(x_j). \tag{8}$$

This follows from the fixed point conditions on the messages – $m_{kj}(x_j) = \sum_{x_k} \exp\{-\phi_k(x_k)\} \exp\{-\psi_{jk}(x_j, x_k)\} \prod_{l \neq j} m_{lk}(x_k)$ $\forall k, j, x_j$.

In general, the admissibility and consistency constraints characterize the fixed points of belief propagation. This has an elegant interpretation within the framework of information geometry [11].

## 3.2 Examples

BP is like dynamic programming for graphs without closed loops. The sum-product and max-product relate to the sum and max versions of dynamic programming (see earlier lecture). But there are several differences. Dynamic programming has forward and backward passes which do different operations (i.e. the pass different types of messages). But BP only passes the same type of message. DP must start at either end of the graph, but BP can start anywhere and can proceed in parallel, see figure (5).

## 3.3 The Bethe Free Energy

The Bethe free energy [7] differs from the MFT free energy by including pairwise pseudo-marginals $b_{ij}(x_i, x_j)$:

$$\begin{aligned} \mathcal{F}[b; \lambda] = \sum_{ij} \sum_{x_i, x_j} b_{ij}(x_i, x_j) \psi_{ij}(x_i, x_j) + \sum_i \sum_{x_i} b_i(x_i) \phi_i(x_i) \\ + \sum_{ij} \sum_{x_i, x_j} b_{ij}(x_i, x_j) \log b_{ij}(x_i, x_j) - \sum_i (n_i - 1) \sum_{x_i} b_i(x_i) \log b_i(x_i), \end{aligned} \tag{9}$$

But we must also impose consistency and normalization constraints which we impose by lagrange multipliers $\{\lambda_{ij}(x_j)\}$ and $\{\gamma_i\}$:

$$\begin{aligned} \sum_{i,j} \sum_{x_j} \lambda_{ij}(x_j) \{ \sum_{x_i} b_{ij}(x_i, x_j) - b_j(x_j) \} \\ + \sum_{i,j} \sum_{x_i} \lambda_{ji}(x_i) \{ \sum_{x_j} b_{ij}(x_i, x_j) - b_i(x_i) \} + \sum_i \gamma_i \{ \sum_{x_i} b_i(x_i) - 1 \}. \end{aligned} \tag{10}$$

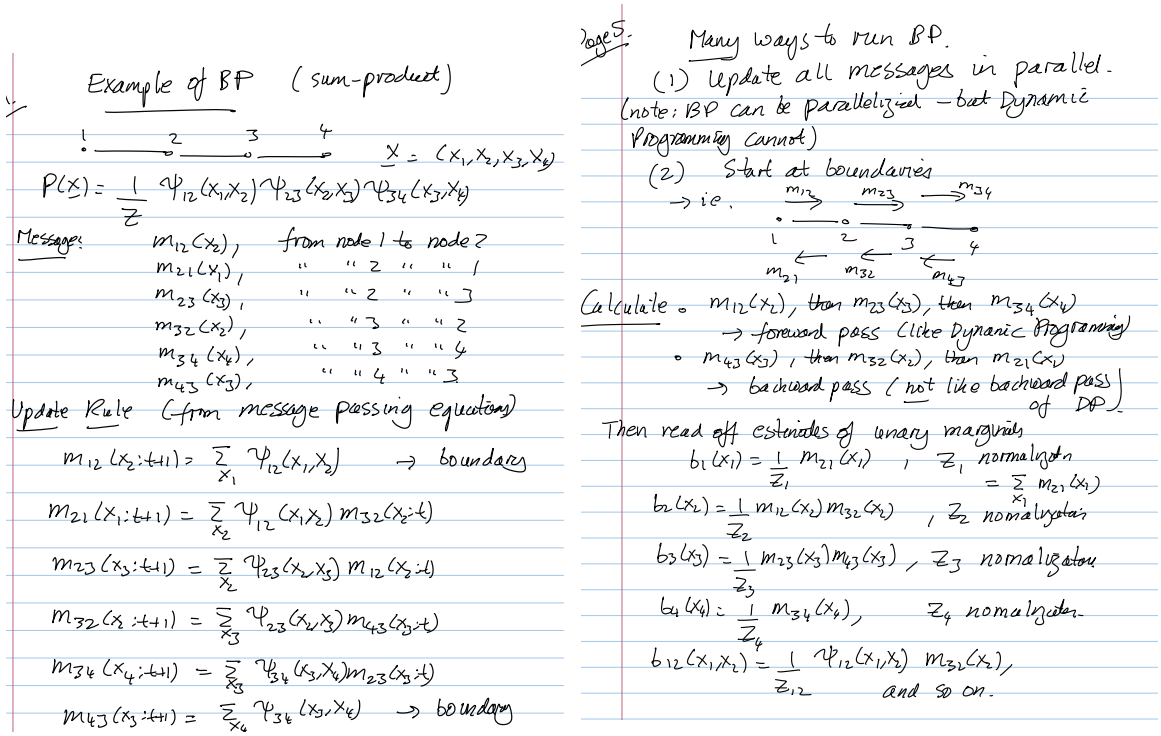


Figure 5: Example of message passing for a graph without closed loops. The messages can start at any node and can be updated in parallel. But the most efficient way is to start messages from the boundaries (because these messages are fixed and will not need to be updated, which causes the algorithm to converge faster.

It is straightforward to verify that the extreme of the Bethe free energy also obey the admissibility and consistency constraints – set the derivatives of the Bethe free energy to zero. Hence the fixed points of belief propagation correspond to extrema of the Bethe free energy.

### 3.4 Where do the messages come from? The dual formulation.

Where do the messages in belief propagation come from? At first glance, they do not appear directly in the Bethe free energy (Pearl has motivation for performing inference on graphs without closed loops). But observe that the consistency constraints are imposed by lagrange multipliers $\lambda_{ij}(x_j)$ which have the same dimensions as the messages.

We can think of the Bethe free energy as specifying a *primal problem* defined over *primal variables* $b$ and *dual variables* $\lambda$. The goal is to minimize $\mathcal{F}[b;\lambda]$ with respect to the primal variables and maximize it with respect to the dual variables. There corresponds a *dual problem* which can be obtained by minimizing $\mathcal{F}[b;\lambda]$ with respect to $b$ to get solutions $b(\lambda)$ and substituting them back to obtain $\hat{\mathcal{F}}_d[\lambda] = \mathcal{F}[b(\lambda);\lambda]$. Extrema of the dual problem correspond to extrema of the primal problem (and vice versa). Note: but there is a *duality gap* because the Bethe free energy is not convex except for graphs without closed loops. This means that the relationship between the solution of the primal and dual problems are complicated (revise this).

It is straightforward to show that minimizing $\mathcal{F}$ with respect to the $b$'s give the equations:

$$b_i^t(x_i) \propto \exp\{-1/(n_i-1)\{\gamma_i - \sum_j \lambda_{ji}(x_i) - \phi_i(x_i)\}\}, \tag{11}$$

$$b_{ij}^t(x_i,x_j) \propto \exp\{-\psi_{ij}(x_i,x_j) - \lambda_{ij}^t(x_j) - \lambda_{ji}^t(x_i)\}. \tag{12}$$

Observe the similarity between these equations and those specified by belief propagation, see equations (4). They become identical if we identify the messages with a function of the $\lambda$'s:

$$\lambda_{ji}(x_i) \quad = \quad - \sum_{k \in N(i)/j} \log m_{ki}(x_i). \tag{13}$$

There are, however, two limitations of the Bethe free energy. Firstly it does not provide a bound of the partition function (unlike MFT) and so it is not possible to using bounding arguments to claim that Bethe is 'better' than MFT (i.e. it is not guaranteed to give a tighter bound). Secondly, Bethe is non-convex (except on trees) which has unfortunate consequences for the dual problem – the maximum of the dual is not guaranteed to correspond to the minimum of the primal. Both problems can be avoided by an alternative approach, described in the next section, which gives convex upper bounds on the partition function and specifies convergent (single-loop) algorithms.

### 3.5 Extras about Belief Propagation

If the goal of belief propagation is to minimize the Bethe Free Energy then why not use direct methods like steepest descent or discrete iterative algorithms instead? One disadvantage is these methods require working with pseudomarginals that have higher dimensions than the messages (contrast $b_{ij}(x_i, x_j)$ with $m_{ij}(x_j)$). Discrete iterative algorithms have been proposed [21],[10] which are more stable than belief propagation and which can reach lower values of the Bethe Free Energy. But these DIA must have an inner loop to deal with the consistency constraints and hence take longer to converge than belief propagation. The difference between these direct algorithms and belief propagation can also be given an elegant geometric interpretation in terms of information geometry [11].

Belief propagation can also be formulated as re-parametrizing the probability distribution (Wainwright). This following from the admissibility constraint. We start off by expressing the probability distribution in terms of potentials (i.e. the way we formulate an undirected graphical model) and the output of BP – the unary and binary marginals – allow us to reformulate the probability distribution in terms of approximate conditional distributions. If the graph has no closed loops, then it is possible to express the probability distribution exactly in terms of conditional distributions – and DP (the sum version) will also enable us to do this "translation". This gives rise to an alternative formulation of BP which does not use messages, see figure (6).

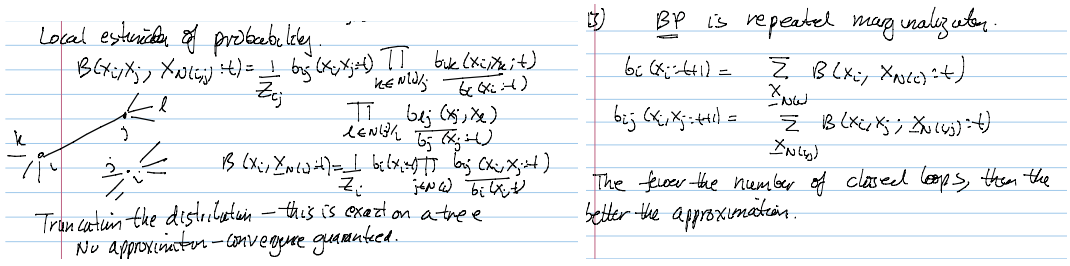


Figure 6: BP without messages. Use current estimates of unary and binary beliefs to define distributions over groups of nodes (left panel). Then re-estimate the unary and binary beliefs by marginalization (right panel).

There is also a relationship between BP and a form of Gibbs sampling [16] which follows from the formulation of BP without messages, see figure (6). In a sense, BP is like taking the expectation of the update provided by a Gibbs sampling algorithm that samples pairs of points at the same time. The relationship between sampling techniques and deterministic methods is an interesting area of research and there are successful algorithms which combine both aspects. For example, there are recent nonparametric approaches which combine particle filters with belief propagation to do inference on graphical models where the variables are continuous valued [17][12].

Work by Felzenszwalb and Huttenlocher [8] shows how belief propagation methods can be made extremely fast by taking advantage of properties of the potentials and the multi-scale properties of many vision problems. Researchers in the UAI community have discovered ways to derive generalizations of BP starting from the perspective of efficient exact inference [6].

# 4 Generalized Belief Propagation and the Kikuchi Free Energy

*Not covered in class.* It shows that BP can be generalized naturally to graphs where there are higher order interactions.

The Bethe free energy and belief propagation were defined for probability distributions defined over graphs with pairwise interactions (e.g. potential terms like $\psi_{ij}(x_i, x_j)$). If we have higher order interactions then we can generalize to the Kikuchi free energy and generalized belief propagation [7][20]

Let $R$ be the set of regions that contain basic clusters of nodes, their intersections, the intersections of their intersections, and so on. For any region $r$, we define the super-regions of $r$ to be the set $sup(r)$ of all of regions in $R$ that contain $r$. The sub-regions of $r$ are the set $sub(r)$ of all regions of in $R$ that lie completely within $r$. The direct sub-regions of $r$ are the set $sub_d(r)$ of all sub-regions of $r$ which have no super-regions which are also sub-regions of $r$. The direct super-regions $sup_d(r)$ are the super-regions of $r$ that have no sub-regions which are also super-regions of $r$.

Let $x_r$ be the state of nodes in region $r$ and $b_r(x_r)$ is the belief. $E_r(x_r)$ is the energy associated with the region (e.g. $-\sum_{(i,j)\in R}\psi_{ij}(x_i,x_j) - \sum_{i\in r}\psi_i(x_i)$). The Kikuchi Free energy is:

$$F_K = \sum_{r\in R} c_r\{\sum_{x_r} b_r(x_r)E_r(x_r)\} + \sum_{x_r} b_r(x_r)\log b_r(x_r) + L_k. \tag{14}$$

$c_r$ is an over-counting number defined by $c_r = 1 - \sum_{s\in sup(r)} c_s$. $L_k$ imposes the constraints. $\sum_r \gamma_r\{\sum_{x_r} b_r(x_r) - 1\} + \sum_r \sum_{s\in sub_d(r)}\lambda_{r,s}(x_s)\{\sum_{x}\in r/s\, b_r(x_r) - b_s(x_s)\}$, where $r/s$ is the complement of $s$ in $r$.

As before, there is a dual formulation of Kikuchi in terms of dual variables which motivates a generalized belief propagation (GBP) algorithm in terms of the dual variables which can be converted into messages. Double loop algorithms such as CCCP can also be directly applied to Kikuchi [**?**]. In short, all the properties of Bethe and belief propagation can be extended to this case.

For example, minimizing Kikuchi with respect to the beliefs $b$ gives solutions in terms of messages:

$$b_r(x_r) = \psi_r(x_r) \prod_{(u,v)\in M(r)} m_{uv}(x_v). \tag{15}$$

Generalized belief propagation can be defined by message updating:

$$m_{rs}(x_s : t+1) = m_{rs}(x_s : t)\frac{\sum_{x\in r/s}\exp\{-\psi_r(x_r)\}\prod_{(u,v)\in M(r)} m_{uv}(x_v)}{\exp\{-\psi_s(x_s)\}\prod_{(u,v)\in M(s)} m_{uv}(x_v)}. \tag{16}$$

# 5 Convex Upper Bounds and TRW

*Not covered in class. Advanced material. Bottom line - people should probably use TRW instead of BP.*

TRW is an alternative to the BP algorithm. It is theoretical much cleaner than BP and it yields better performance in practice (although how much to trust those types of comparisons – the improvements are not enormous). Instead of the Bethe free energy it starts with a convex free energy which is specified as an upper bound of the log partition function of the distribution (recall that Bethe is not a bound). This free energy becomes identical to the Bethe free energy for graphs without closed loops. There is also no duality gap between the free energy and its dual. Hence there are algorithms in dual space (similar to BP) which are guaranteed to converge to the global optimum (give some caveats!).

This section specifies an alternative class of convex free energies which are defined to be upper bounds of the log partition function. They lead to free energies which are similar to Bethe and to algorithms which

are similar to belief propagation. But they have several advantages. Firstly, they gives bounds so it is more easy to quantify how good they are as approximations. Secondly, they are convex so algorithms exist which can find their global minimum. Thirdly, they enable algorithms similar to belief propagation which are guaranteed to converge.

Firstly, we describe a duality relation – between the log partition function and the negative entropy – which is critical for deriving the convex free energy. This relationships gives two alternative ways of representing the probability function: (i) in terms of the coefficients of the potentials, and (ii) in terms of marginals of the distribution.

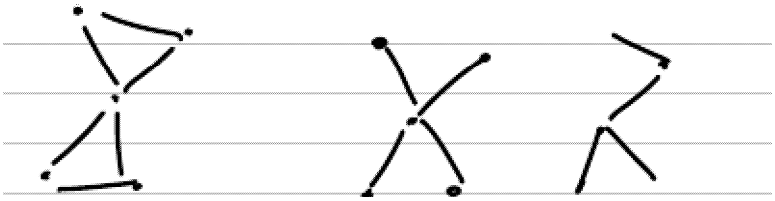

Figure 7: TRW; Graphs and Spanning Trees.

## 5.1 Exponential Form and Legendre Transforms

We formulate the probability distribution as

$$P(\vec{x}) = \exp\{\vec{\theta} \cdot \vec{\phi}(\vec{x}) - \Phi(\vec{\theta})\}, \tag{17}$$

where $\vec{\phi}(\vec{x})$ are potentials/statistics, $\vec{\theta}$ are coefficients, and $\Phi(\vec{\theta})$ is the logarithm of the partition function $Z(\vec{\theta}) = \sum_{\vec{x}} \exp\{\vec{\theta} \cdot \vec{\phi}(\vec{x})\}$.

We define $\vec{\mu}$ to be expected value of the potentials $\vec{\phi}(\vec{x})$:

$$\vec{\mu} = \sum_{\vec{x}} \vec{\phi}(\vec{x}) P(\vec{x} : \vec{\theta}) = \frac{\partial}{\partial \vec{\theta}} \Phi(\vec{x}). \tag{18}$$

There is a duality relationship between the coefficients $\vec{\theta}$ and the expected statistics $\vec{\mu}$. We define the Legendre transform of the log partition function $\Phi(\vec{\theta})$:

$$\psi(\vec{\mu}) = \sup_{\vec{\theta}} \{\vec{\theta} \cdot \vec{\mu} - \Phi(\vec{\theta})\}. \tag{19}$$

The supremum occurs at $\vec{\theta}^*$ such that $\vec{\mu} = \sum_{\vec{x}} P(\vec{x} : \vec{\theta}^*) \vec{\phi}(\vec{x})$. It can be checked that:

$$\psi(\vec{\mu}) = \sum_{\vec{x}} P(\vec{x} : \vec{\theta}^*) \log P(\vec{x} : \vec{\theta}^*) = -H(P(\vec{x} : \vec{\theta}^*), \tag{20}$$

where $-H(P(\vec{x} : \vec{\theta}^*)$ is the negative entropy of $P(\vec{x} : \vec{\theta}^*)$.

Similarly, we can recover the log partition function from the negative entropy by applying the inverse Legendre transform:

$$\Phi(\vec{\theta}) = \inf_{\vec{\mu}} \{\psi(\vec{\mu}) - \vec{\theta} \cdot \vec{\mu}\}. \tag{21}$$

Now suppose that the potentials form an over-complete set, as described below. Then the $\vec{m}u$ correspond to the unary and pairwise marginals of the distribution respectively. This gives two ways of representing the distribution. The first is by the coefficients $\vec{\theta}$ of the potentials. The second is by the unary and pairwise

marginals. As described earlier (where??), if the distributions are defined over a graph with no loops (i.e. a tree) then it is easy to translate from one representation to the other. This relationship will be exploited as we derive the convex free energy.

## 5.2 Obtaining the Upper Bound

The strategy to obtain the convex free is based on exploiting the fact that everything is easy on graphs with no loops (i.e. trees). So the strategy is to define a probability distribution $\rho(T)$ on the set of spanning trees $T$ of the graph – i.e. $\rho(T) \geq 0$ and $\sum_T \rho(T) = 1$. For each tree, we define parameters $\vec{\theta}(T)$ with the constraint that $\sum_T \rho(T)\vec{\theta}(T) = \vec{\theta}$. On each spanning tree we are able to compute the entropy and partition function of the distribution defined on the tree. We now show how this can be exploited to provide an upper bound on the partition function for the full distribution.

We first bound the partition function by the averaged sum of the partition functions on the spanning trees (using Jensen's inequality):

$$\Phi(\vec{\theta}) = \Phi(\sum_T \rho(T)\vec{\theta}(T)) \leq \sum_T \rho(T)\Phi(\vec{\theta}(T)). \tag{22}$$

We can make the bound tighter by solving the constrained minimization problem:

$$\sum_T \rho(T)\Phi(\vec{\theta}(T)) - \vec{\tau} \cdot \{\sum_T \rho(T)\vec{\theta}(T) - \vec{\theta}\}. \tag{23}$$

Here the $\vec{\tau}$ are Lagrange multipliers. In more detail, we define a set of over-complete potentials $\{\phi_s(x_s;k), \phi_{st}(x_s,x_t;k,l)\}$ where $s \in V \quad (s,t) \in E$ are the nodes and edges of the graph, $x_s$ are the state variables, and $k,l$ index the values that the state variables can take. We define corresponding parameters $\{\theta_s(k;T), \theta_{st}(k,l:T)\}$. Then the $\vec{\tau}$ can be written as $\{\tau_s(k), \tau_{st}(k,l)\}$.

Minimizing equation (23) with respect to $\vec{\theta}(T)$ gives the equations:

$$\rho(T)\sum_{\vec{x}} \vec{\theta}^*(T)P(\vec{x} : \vec{\theta}^*(T)) = \rho(T)\vec{\tau}. \tag{24}$$

Recall that each spanning tree $T$ will contain all the nodes $V$ but only a subset $E(T)$ of the edges. Equation (24) determines that we can express $P(\vec{x} : \vec{\theta}^*(T))$ in terms of the dual variables $\{\tau_s : s \in V\}, \{\tau_{st} : (s,t) \in E(T)\}$:

$$P(\vec{x} : \vec{\theta}^*(T)) = \prod_s \tau_s(x_s) \prod_{(s,t)\in E(T)} \frac{\tau_{st}(x_s,x_t)}{\tau_s(x_s)\tau_t(x_t)}. \tag{25}$$

Equation (25) is very insightful. It tells us that the $\vec{\tau}$, which started out as Lagrange multipliers, can be interpreted (at the extremum) as pseudo-marginals obeying the consistency constraints $\sum_{x_s} \tau_s(x_s) = 1$, $\sum_{x_t} \tau_{st}(x_s,x_t) = \tau_s(x_s)$, and so on. It tells us that the $\vec{\theta}(T)$ are related to the same $\vec{\tau}$ for all $T$, although for each $T$ some of the $\vec{\tau}$ terms will not be used (those corresponding to the missing edges in $T$). It also tells us that we can evaluate the terms $\Phi(\vec{\theta}^*)$ involved in the bound – they can be computed from the Legendre transform (ref backwards!!) in terms of $\vec{\tau} \cdot \vec{\theta}^*(T)$ and the negative entropy of $P(\vec{x} : \vec{\theta}^*(T))$, which can be easily computed from equation (25) to be:

$$\vec{\tau} \cdot \vec{\theta}^*(T) = \vec{\theta}^*(T) \cdot \vec{\tau} + \sum_s H_s + \sum_{(s,t)\in E(T)} I_{st}. \tag{26}$$

We compute the full bound by taking the expectation with respect to $\rho(T)$ (using the constraint $\sum_T \rho(T)\vec{\theta}^*(T) = \vec{\theta}$) yielding the free energy:

$$F_{\text{conbound}} = -\vec{\theta} \cdot \vec{\tau} + \sum_s H_s + \sum_{(s,t)\in E} \rho_{st} I_{st}. \tag{27}$$

Next we use the inverse Legendre transform to express the right hand side of equation (22) as a free energy which, by the derivation, is convex in terms of the pseudomarginals $\mu$:

$$\mathcal{F}(\mu : \rho, \theta) = -\sum_s H_s(\mu_s) + \sum_{(s,t)\in E} T_{st} I_{st}(\mu_{st}) - \vec{\mu} \cdot \vec{\theta}, \tag{28}$$

where $H_s(T_s) = -\sum_j T_{s:j} \log T_{s:j}$ and $I_{st}(T_{st}) = \sum_{ij} T_{st:jk} \log \frac{T_{st:jk}}{(\sum_k T_{st;jk})(\sum_j T_{st:jk}\theta^*_{st;jk})}$. This is the expectation with respect to the distribution $\rho(T)$ over trees of the entropies defined on the trees. The pseudomarginals must satisfy the consistency constraints which are imposed by lagrange multipliers.

## 5.3 TRW algorithm

The bound on the partition function can be obtained by minimizing the free energy $\mathcal{F}(\mu : \rho\theta)$ with respect to $\mu$, and the resulting pseudomarginals $\mu$ can be used as estimators of the marginals of $P(\vec{x})$. By minimizing the free energy with respect to $\mu$ we can express the pseudmarginals as messages (similar to Bethe): (REMOVE THE $*$'s!!):

$$\begin{aligned}
\mu_s(x_t) &= k \exp\{\phi_s(x_s : \theta^*_s)\} \prod_{\nu\in\Gamma(s)} |M_{\nu s}(x_s)|^{\mu_{\nu s}}, \\
\mu_{st}(x_s, x_t) &= K\phi_{st}(x_s, x_t : \theta^*) \frac{\prod_{\nu\in\Gamma(s)/t} |M_{\nu s}(x_s)|^{\mu_{\nu s}}}{|M_{ts}(x_s)|^{1-\mu_{st}}} \\
&\quad \times \frac{\prod_{\nu\in\Gamma(t)/s} |M_{\nu t}(x_t)|^{\mu_{\nu t}}}{|M_{st}(x_t)|^{1-\mu_{ts}}},
\end{aligned} \tag{29}$$

where the quantity $\phi_s(x_s; \theta^*_s)$ takes value $\theta^*_{st}$ when $x_s = j$ and analogously for $\phi_{st}(x_s, x_t : \theta^*_{st})$.

It can be checked that these pseudomarginals satisfy the *admissability* conditions:

$$\begin{aligned}
\theta^* \cdot \phi(x) + C = \\
\sum_{\nu\in V} \log \hat{\mu}(x_s) + \sum (s,t) \in E \rho_{st} \log \frac{\hat{\mu}_{st}(x_s, x_t)}{\hat{T}_s(x_s)\hat{T}_t(x_t)}.
\end{aligned} \tag{30}$$

The TRW algorithm updates the messages by the following rules. The fixed points occur at the global minimum of the free energy, but there is no guarantee that the algorithm will converge.

Initialize the messages $M^0 = \{M_{st}\}$ with arbitrary positive values. For iterations $n = 0, 1, ....$ update the messages by:

$$\begin{aligned}
M^{n+1}_{ts}(x_s) = K \sum_{x'_t \in X_t} \exp\{\frac{1}{\mu_{st}} \phi_{st}(x_s, x'_t; \theta^*_{st}) + \phi_t(x'_t : \theta^*_t)\} \\
\{\frac{\prod_{\nu\in\Gamma(t)/s} \{M^n_{\nu i}(x'_t)\}^{\mu_{\nu t}}}{|M^n_{st}(x'_t)|^{(1-\mu_{ts})}}\}.
\end{aligned} \tag{31}$$

### 5.4 A convergent variant

An alternative approach by Globerson and Jaakkola uses the same strategy to obtain a free energy function. They write a slightly different free energy:

$$-\mu \cdot \theta - \sum_i \rho_{0i} H(X_i) - \sum_{ij \in E} \rho_{i|j} H(X_i|X_j), \tag{32}$$

which agrees with the TRW free energy if the pseudomarginals $\mu$ satisfy the constraints but differs elsewhere. The advantage is that they can explicitly computer the dual of the free energy. This dual free energy is expressed as:

$$\sum_i \rho_{0i} \log \sum_{x_i} \exp\{\rho_{j|i}^{-1}\{\theta_{ij}(x_i, x_j) - \sum_{k \in N(i)} \lambda_{k|i}(i; \beta)\}\}. \tag{33}$$

where

$$\lambda j|i(x_i : \beta) = -\rho_{j|i} \log \sum_{x_j} \exp\{\rho_{j|i}^{-1}\{\theta_{ij}(x_i, x_j) + \delta_{j|i}\beta_{ij}(x_i, x_j)\}\}. \tag{34}$$

The $\beta$ variables are dual variables.

The explicit form of the dual (not known for the TRW free energy) makes it possible to derive an iterative algorithm that is guaranteed to converge. The update rule is specified by:

$$\beta_{ij}^{t+1}(x_i, x_j) = \beta_{ij}^t(x_i, x_j) + \epsilon \log \frac{\mu_{j|i}^t(x_j|x_i)\mu_i^t(x_i)}{\mu_{i|j}^t(x_i|x_j)\mu_j^t(x_j)}. \tag{35}$$

It can be checked that this update satisfies the tree re-parameterization conditions.

Note: CCCP will give a provably convergent algorithm but relies on a double loop (Nishihara et al).

## 6 Conclusions

This lecture described MRF methods for addressing the stereo correspondence problem and discussed how they could be solved using dynamic programming – exploiting the epipolar line constraint – or by belief propagation if we allow interactions across the epipolar lines.

We also introduced belief propagation and also the TRW algorithm (not covered in class).

## References


[1] Y. Amit. "Modelling Brain Function: The World of Attrcator Neural Networks". Cambridge University Press. 1992.

[2] C.M. Bishop. Pattern Recognition and Machine Learning. Springer. Second edition. 2007.

[3] M. J. Black and A. Rangarajan, "On the unification of line process, outlier rejection, and robust statistics with applications in early vision", Int'l J. of Comp. Vis., Vol. 19, No. 1 pp 57-91. 1996.

[4] S. Roth and M. Black. Fields of Experts. International Journal of Computer Vision. Vol. 82. Issue 2. pp 205-229. 2009.

[5] A. Blake, and M. Isard: The CONDENSATION Algorithm - Conditional Density Propagation and Applications to Visual Tracking. NIPS 1996: 361-367. 1996.

[6] A. Choi and A. Darwiche. A Variational Approach for Approximating Bayesian Networks by Edge Deletion. In Proceedings of the 22nd Conference on Uncertainty in Artificial Intelligence (UAI), pages 80-89, 2006.

[7] C. Domb and M.S. Green. Eds. Phase Transitions and Critical Phenomena. Vol.2. Academic Press. London. 1972.

[8] P. Felzenszwalb and D. P. Huttenlocher. Efficient Belief Propagation for Early Vision. Proceedings of Computer Vision and Pattern Recognition. 2004.

[9] D. Geiger, B. Ladendorf and A.L. Yuille. "Occlusions and binocular stereo". *International Journal of Computer Vision*. 14, pp 211-226. 1995.

[10] T. Heskes, K. Albers and B. Kappen. Approximate Inference and Constrained Optimization. Proc. 19th Conference. Uncertainty in Artificial Intelligence. 2003.

[11] S. Ikeda, T. Tanaka, S. Amari. "Stochastic Reasoning, Free Energy, and Information Geometry". Neural Computation. 2004.

[12] M. Isard, "PAMPAS: Real-Valued Graphical Models for Computer Vision," cvpr, vol. 1, pp.613, 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '03) - Volume 1, 2003.

[13] J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In: Proc. 18th International Conf. on Machine Learning, Morgan Kaufmann, San Francisco, CA. 2001.

[14] R.J. McEliece, D.J.C. MacKay, and J.F. Cheng. Turbo Decoding as an instance of Pearl's belief propagation algorithm. IEEE Journal on Selected Areas in Communication. 16(2), pp 140-152. 1998.

[15] J. Pearl. "Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference." Morgan Kaufmann, San Mateo, CA. 1988.

[16] M. Rosen-Zvi, M. I. Jordan, A. L. Yuille: The DLR Hierarchy of Approximate Inference. Uncertainty in Artificial Intelligence. 2005: 493-500.

[17] E. Sudderth, A.T. Ihler, W.T. Freeman, and A.S. Willsky. Nonparametric Belief Propagation. CVPR. pp 605-612. 2002.

[18] J. Sun, H-Y Shum, and N-N Zheng. Stereo Matching using Belief Propagation. Proc. 7th European Conference on Computer Vision. pp 510-524. 2005.

[19] M.J. Wainwright, T.S. Jaakkola, and A.S. Willsky. "Tree-Based Reparamterization Framework for Analysis of Sum-Product and Related Algorithms". *IEEE Transactions on Information Theory*. Vol. 49, pp 1120-1146. No. 5. 2003.

[20] J.S. Yedidia, W.T. Freeman, and Y. Weiss, "Generalized belief propagation". In *Advances in Neural Information Processing Systems 13*, pp 689-695. 2001.

[21] A.L. Yuille. "CCCP Algorithms to Minimize the Bethe and Kikuchi Free Energies: Convergent Alternatives to Belief Propagation". *Neural Computation*. Vol. 14. No. 7. pp 1691-1722. 2002.

[22] A.L. Yuille and Anand Rangarajan. "The Concave-Convex Procedure (CCCP)". *Neural Computation*. 15:915-936. 2003.