Some figures and materials courtesy of: Silvia Zuffi, Michael Black, MPI Tubingen
Pairwise Graphical Models

Nodes are continuous random variables \( x_s \in \mathbb{R}^d \)

- **Potentials** \( \psi(x) \) encode statistical relationships
- Edges indicate direct pairwise energetic interaction
- Facilitates efficient statistical inference

\[
p(x) \propto \prod_{s \in V} \psi_s(x_s) \prod_{(s,t) \in E} \psi_{st}(x_s, x_t)
\]

\( G(V, E) : \text{Vertices } s \in V, \text{ edges } (s, t) \in E \)
Example: Stereo Vision

Input Images

Depth Map

Grid Graph

Non-Gaussian continuous model:

\[ \psi_s(x_s, y_s) : \text{Likelihood scores disparity / obeys occlusion} \]

\[ \psi_{st}(x_s, x_t) : \text{Compatibility encourages similar disparities, allows discontinuity at image edges} \]

Can’t use particle filter, not a time series…
Discrete: BP updates well-defined, matrix-vector products.

Continuous: no general closed-form updates.

Solution: Approximate continuous messages with discrete particles.
Importance Sampling

- Draw samples from proposal distribution,
  \[ x^{(i)} \sim q(x), \quad \{x^{(1)}, \ldots, x^{(N)}\} \]

- Approximate expectation,
  \[ \mathbb{E}[g(x)] = \int_{\chi} g(x)p(x) \, dx \approx \sum_{i=1}^{N} g(x^{(i)})w(x^{(i)}), \]

- Importance weights account for proposal,
  \[ w(x) \propto \frac{p(x)}{q(x)}, \quad \sum_{i=1}^{N} w(x^{(i)}) = 1. \]

Main Idea: Approximate BP messages with importance sampling.
Particle Belief Propagation

» Rewrite BP message as expectation,

\[ m_{ts}(x_s) = \mathbb{E}_{qt} \left[ \psi_{st}(x_s, x_t) \frac{\psi_t(x_t)}{q_t(x_t)} \prod_{k \in \Gamma(t) \setminus s} m_{kt}(x_t) \right] \]

» Importance weighted expectation,

\[ m_{ts}(x_s) \approx \sum_{i=1}^{N} \psi_{st}(x_s, x_t^{(i)}) w_t(x_t^{(i)}) \triangleq \hat{m}_{ts}(x_s) \]

» Importance weights,

\[ w_t(x_t^{(i)}) = \frac{\psi_t(x_t^{(i)}) \prod_{k \in \Gamma(t) \setminus s} m_{kt}(x_t^{(i)})}{q_t(x_t^{(i)})} \]

Sample particles from BP marginal.
Stereo Vision Results

- **True Disparity**
- **Estimate**
- **Prediction Error**

Comparison to related Nonparametric BP

- $L_1$ error w.r.t. true beliefs via discretization
- Errors decrease at rate $\frac{1}{\sqrt{N}}$

A. Ihler and D. McAllester, AISTATS 2009
Maximum a Posteriori (MAP)

Maximizer of the posterior probability:

\[ x^* = \arg\max_x p(x \mid y) \]

Issues with continuous models:

- Analytically intractable posterior density
Maximum a Posteriori (MAP)

Maximizer of the posterior probability:

\[ x^* = \arg\max_x p(x \mid y) \]

Issues with continuous models:

- Analytically intractable posterior density
- Multiple local optima (these can be useful too…)

Data: \( y \)
Unknowns: \( x \)
Posterior: \( p(x \mid y) \)
MAP Estimate
Message Passing

Global MAP inference decomposes into local computations via graph structure...

$$\max_x p(x) \propto \max_x \psi_{12}(x_1, x_2) \psi_{23}(x_2, x_3) \psi_{24}(x_2, x_4)$$

$$= \max_{x_1} \max_{x_2} \psi_{12}(x_1, x_2) \left[ \max_{x_3} \psi_{23}(x_2, x_3) \right] \left[ \max_{x_4} \psi_{24}(x_2, x_4) \right]$$

$$m_{32}(x_2), m_{42}(x_2), m_{21}(x_1)$$
Max-Product (MP) Belief Propagation

Passing *messages* in a graphical model...

**Message**

\[ m_{ts}(x_s) \propto \max_{x_t} \psi_{st}(x_s, x_t) \psi_t(x_t) \prod_{k \in \Gamma(t)\setminus s} m_{kt}(x_t) \]

**Max-Marginal**

\[ q_t(x_t) \propto \psi_t(x_t) \prod_{k \in \Gamma(t)} m_{kt}(x_t) \]
Poses & Discrete Probabilities

Felzenszwalb & Huttenlocher, 2005

- Pairwise MRF with rigid geometry
- MAP estimate of pose via discrete max-product BP
- Discrete state space limits allowable deformations
then be parametrized by PCA coefficients. There is some pose variation among subjects, which could have been partially removed by the body pose assumption of the authors. However, it is likely that there are still unknowns, and these can be solved using a straightforward minimization technique.

Principal Component Analysis (PCA) is applied on the triangles of the template surface. The template mesh is obtained by aligning a mesh to the training set of instances. The registration technique employs markers in a first stage, where we can easily estimate the markers to overlap. Then, the triangles of the template are transformed to match the data. PCA is appropriate for modeling the main body pose and shape deformations in the training set. We assume that these vectors are generated from a simple linear subspace, which can be estimated by using our learned parameters of matrices $\{\mu_i \}_{i=1}^N$. The application of consecutive transformation matrices $\{Q_i \}_{i=1}^N$ results in the body shape deformation $S_i$. The first four principal components in the space of body shape parameters are shown in the diagram.

To map out the space of body shape deformations, we view the differences from the mean still looking very much like humans (see Fig. 5). We found that even shapes which are three standard deviations away from the mean are not given, and need to be estimated. We use the same idea as above, and solve directly for these unknowns, and therefore can be solved using a straightforward minimization.

SCAPE: Examples of muscle deformations that can be captured in the SCAPE pose model.
Deformable Structures

\[ p(x, y) \propto \prod_{s \in \mathcal{V}} \psi_s(x_s, y) \prod_{(s,t) \in \mathcal{E}} \psi_{st}(x_s, x_t) \]

Complicated Likelihood

Non-Gaussian Compatibility

Continuous state \( x_s \in \mathcal{X}_s \) for part shape, location, orientation and scale.
DS defines a joint probability from which we can sample human poses.
Max-Product Belief Propagation

**Discrete**

\[ x \in \{1, \ldots, N\}^D \]

Message Update:

\[ m_{ts} = \max_{x_t} \psi_{st} \psi_t \prod m_{kt} \]

Matrix-vector multiplication & discrete maximization

**Continuous**

\[ x \in \mathcal{R}^D \]

Message Update:

\[ m_{ts}(x_s) = \ldots \max_{x_t} \psi_{st}(x_s, x_t) \psi_t(x_t) \prod m_{kt}(x_t) \]

Nonlinear optimization
Regular Discretization

Approximate continuous max-product messages over regular grid of points

Example: Torso
- ~10 dimensions.
- 10 grid points per dimension
- 10 Million points!

Infeasible for high dimensional models.
Particle Max-Product (PMP)

Combine particle filter ideas with max-product more effectively.

Particle approximation of continuous max-product (MP) messages.
Sample new hypotheses at every node to grow particle set.
Particle Max-Product (PMP)

Update MP messages on augmented particles.

$$m_{ts}(x_s) \propto \max_{x_t} \psi_{st}(x_s, x_t) \psi_t(x_t) \prod_{k \in \Gamma(t) \setminus s} m_{kt}(x_t)$$
Particle Max-Product (PMP)

1. Augment Particles
2. Max-Product Update
3. Select Particles

Select subset of *good* particles & repeat

Need a particle selection method...
Synthetic Pose Estimation

Binary image of 4 silhouettes.

**Model** Truncated Gaussian pairwise potentials $\psi_{st}(x_s, x_t)$:

- PCA Shape
- Orientation
- “Spring”
- Likelihood $\psi_s(x_s)$ distance-map from silhouette contours.
Top-N Particle Max Product (T-PMP)

- Data-Driven
- Neighbor
- Random Walk
- ...

Augment:

MP Update → Select Top-N

- Keep N-best particles
- Sensitive to initialization
- Still too greedy; Selection reduces effective number of particles

Maintain diversity in particles.

[ Pacheco et al., ICML 2014 ]
Diverse Particle Selection

Particles selected to minimize max-product message distortion.

Integer Program (IP) solved with efficient greedy approximation:

- **Integer Program (IP)**: Optimal solution by brute force
- **Greedy**: Efficient approximation

**Linear Program (LP)** relaxation

- **LP**: Linear Program relaxation
Diverse Particle Selection

Minimize total message distortion:

\[
\text{minimize} \sum_{z} \sum_{s \in \Gamma(t)} \sum_{a=1}^{\alpha N} (m_{ts}(a) - \hat{m}_{ts}(a, z))
\]

subject to \( \|z\|_1 \leq N, \ z \in \{0, 1\}^{\alpha N} \)

× NP-hard

✓ Submodular

Good approximation qualities.
Avoids particle degeneracies by maintaining *ensemble of diverse solutions* near local modes.

- No explicit diversity constraint
- Objective encourages diversity
- Efficient *Lazy* greedy algorithm
- Bounds on optimality

Example Runs

[ Pacheco et al., ICML 2014 ]
Top 3 arm hypotheses MAP estimate, 2\textsuperscript{nd} and 3\textsuperscript{rd} modes for upper arm (magenta, cyan), lower arm (green, white).

➢ “Buffy” dataset [Ferrari et al. 2008].
➢ Detections versus number of ranked hypotheses.
➢ Baseline: Flexible Mixture of Parts (FMP) [Yang & Ramanan 2013; Park & Ramanan 2011]

[ Pacheco, Zuffi, Black & Sudderth, ICML 2014 ]
Real Images (Multiple People)

D-PMP Particles | Mode Estimates

Precision-Recall for multi-person frames:

**T-PMP**: High precision, low recall, particles on one figure

**D-PMP**: Outperforms FMP and other particle methods

Note: G-PMP not reported due to poor performance.

[ Pacheco, Zuffi, Black & Sudderth, ICML 2014 ]
Prior work fails to show improvement by incorporating motion model.

This is a failure of inference…
Articulated Pose Tracking

Data and Optical Flow

1. 2. 3. ... 4. 

Extension of the *Flowing Puppets* model [Zuffi et al., 2013]

Part Likelihood

**Gradients:** Encode object and motion boundaries via HOG / HOF.

**Appearance:** 2D histogram of A/B color channels in L*a*b* space. Luminance ignored.

Prior

**Part Motion:** Scale mixture captures heavy tailed statistics of motion between frames.

**Structural prior identical to DS.**
Many interesting models exhibit *cyclic* dependency structure...

**Loopy Max-Product BP:**
Iteratively update until converged.

Stay tuned later in the course for *reweighted* message passing...
VideoPose2 Experiments

Comparison on VideoPose2 dataset of ~2,000 video frames from TV shows [Sapp et al., 2011]
Pose Tracking Particles

Greater diversity in particles allows D-PMP to reason more globally

Colors

Both right arm hypotheses
Superior to single-frame estimates (---,--)

Clear improvement over Sapp et al. baseline

D-PMP superior to Flowing Puppets in close detection ranges. Looking at failure cases.
D-PMP for 3D Mesh Alignment

Figure 2. Example of particles initialization: note that particles are initialized from disconnected SP models randomly sampled. The red arm on the right belongs to the FAUST scan around which the random puppets are generated.

As an estimate of the global translation of the scan, we take the average value of the scan points. We make this point coincide with the centre of the torso of the SP models that we generate for initialization. Note that this simple assumption creates a bias between model and scan, but the optimization algorithm can deal with large uncertainty in the particles locations, also due to the fact that the scan data does not contain outliers.

Figure 2 shows the set of initial particles in an example where the optimization uses 30 particles. During optimization, we use an adaptive scheme for assigning the weights $\alpha$ and $\beta$ in the energy.

Figure 3. Stitched Puppet Model. To generate an instance of SP we start with the template body (top left), which is segmented into parts. We apply the intrinsic shape deformation to change the body shape (top right). We generate pose deformations for each body part (see text) (bottom left). The pose of the body is defined by the rotation and translation that stitches the parts together (bottom, middle and right).

Upper arms, lower arms, upper legs, lower legs, hands and feet (see color coding in Fig. 1(c)).

SP is a tree-structured graphical model in which each body part corresponds to an ode, with torsos as other root. Each part is represented by a triangulated 3D mesh in a canonical, part-centered, coordinate system. Let $i$ be a node index, with $i$ replicated for each node, and generate body parts for the template mesh with the desired intrinsic shape (Figure 3(b)). We then sample a vector of pose deformation variables for the torso. These define the pose of the torso: given in SP the torso part also includes the pelvis, poses with the torso bent or twisted with respect to the pelvis are modeled as pose deformations (Figure ??). We then assign a global rotation and generate the torso mesh in the global frame. Recursively in the tree, starting at torso, for each node $i$: we get the pose-dependent deformation variables of the parent, $d_{pa}(i)$; we condition the pairwise Gaussian $p_{pa}(i)$ with $d_{pa}(i)$, and marginalize the relative rotation vector $r_{pa}(i)$. This gives a Gaussian distribution over $d_i$; we sample this conditioned distribution to get part deformations, and generate the part mesh in the local frame. The effect of the part deformations applied to each body part is shown in Fig. 3(c). We finally compute the rotation and translation that stitches the parts together at their interface (Fig. 3(d,e)) using the orthogonal Procrustes algorithm. Figure 13 shows samples of bodies generated using this procedure. Note independent work by Zuffi & Black, appeared at CVPR 2015.
Motion Estimation: Optical Flow

Occlusion boundaries are crucial for accurate motion estimation.

Middlebury Benchmark: Ground Truth

Horn & Schunck (1981)

Occluded regions in black

Gaussian MRF

Need non-Gaussian models to capture natural motion statistics.
Optical Flow Estimation

- Robust MRF (Sun, Roth, Black, IJCV 2014), discretization needs \(~100,000\) flow vectors per pixel for good accuracy
- Low-level MRF often makes errors at occlusion boundaries, but D-PMP preserves true flow as secondary hypothesis
- Theory: Often have global MAP on particle set
Optical Flow Estimation

T-PMP  D-PMP  Ground Truth

Reweighted PMP comparison on a “superpixel” graph with ~10,000 nodes
Protein Structure Prediction


All information for predicting 3D structure encoded in amino acid sequence and physics
Protein Side Chains

Side chain prediction: Estimate side chains given fixed backbone.

20 Amino Acid Types

- Leu (L)
- Val (V)
- Phe (F)
- Trp (W)
Dihedrals and Rotamers

Dihedral Angles:
- Compact angular encoding
- 1D-4D continuous state

Rotamer discretization based on marginal statistics fails to capture fine details...

[Shapovalov & Dunbrack 2007]
Side Chain Prediction

\[ p(x) \propto \prod_{s \in V} \psi_s(x_s) \prod_{(s, t) \in \mathcal{E}} \psi_{st}(x_s, x_t) \]

Edges between amino acids within distance threshold.

[ Image: Harder et al., BMC Informatics 2010 ]
Side Chain Prediction

\[ p(x) \propto \prod_{s \in V} \psi_s(x_s) \prod_{(s,t) \in \mathcal{E}} \psi_{st}(x_s, x_t) \]

Rotamer Likelihood

Atomic Interaction

Statistical and physical potential functions.

[ Image: Harder et al., BMC Informatics 2010 ]
Continuous optimization of side chains:

- Captures non-rotameric side chains
- Conformational diversity
- Likelihood-based proposals
- Energy model used in FoldIt game
- Simulated annealing (SA) Monte Carlo
- Independent chains for multiple optima

Replace SA with D-PMP. Use Rosetta as black-box energy method.
Protein Side Chain Prediction

Log-probability of MAP estimate for…

20 Proteins (11 Runs)

370 Proteins

G-PMP, T-PMP, D-PMP, Rosetta simulated annealing [Rohl et al., 2004]

[ Pacheco et al., ICML 2015 ]
Protein Side Chain Prediction

Root mean square deviation (RMSD) from x-ray structure.

Oracle selects best configuration in current particle set.
Side chains don’t exist in a single conformation
Diversity in D-PMP particles captures multiple alternate states.
T-PMP particles get stuck in local optima
Protein Side Chain Prediction

T-PMP

Ground Truth
Protein Side Chain Prediction
Contributions

Reliable particle-based MAP inference for graphical models with continuous variables: object shape, articulation, position, ...

Validation: Inference of multiple poses, motions, protein conformations, ...