Markov Random Fields

Erkut Erdem

BIL717, April 2012

Energy Minimization

- Many vision tasks are naturally posed as energy minimization problems on a rectangular grid of pixels:

$$E(u) \;=\; E_{data}(u) \,+\, E_{smoothness}(u)$$

- The data term $E_{data}(u)$ expresses our goal that the optimal model u be consistent with the measurements.
- The smoothness energy $E_{smoothness}(u)$ is derived from our prior knowledge about plausible solutions.

- Recall Mumford-Shah functional

D. J. Fleet

Sample Vision Tasks

- **Denoising:** Given a noisy image $\hat{I}(x,y)$, where some measurements may be missing, recover the original image $I(x, y)$, which is typically assumed to be smooth.
- **Segmentation:** Assign labels to pixels in an image, e.g., to segment foreground from background.

- Stereo Disparity
- Surface Reconstruction
- …

D. J. Fleet

Markov Random Fields

- A Markov Random Field (MRF) is a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$.
- $\mathcal{V} = \{1, 2, ..., N\}$ is the set of *nodes*, each of which is associated with a random variable (RV), u_j , for $j = 1...N$.
- The neighborhood of node i, denoted $\mathcal{N}_i$, is the set of nodes to which i is adjacent; i.e. $j \in \mathcal{N}_i$ if and only if $(i, j) \in \mathcal{E}$.
- The Markov Random field satisfies

$$p(u_i \,|\, \{u_j\}_{j \in \mathcal{V} \setminus i}) \;=\; p(u_i \,|\, \{u_j\}_{j \in \mathcal{N}_i}) \qquad (1)$$

$\mathcal{N}_i$ is often called the Markov blanket of node i .

Markov Random Fields (cont'd)

- The distribution over an MRF (i.e., over RVs $u = (u_1, ..., u_N)$) that satisfies (1) can be expressed as the product of (positive) potential functions defined on maximal cliques of $\mathcal{G}$ *[Hammersley-Clifford Thm].*

- Such distributions are often expressed in terms of an *energy function* E, and clique potentials Ψ_c:

$$p(u) \;=\; \frac{1}{Z}\exp(-E(u,\theta))\;, \quad \text{where } E(u,\theta) \;=\; \sum_{c\in\mathcal{C}} \Psi_c(\bar{u}_c, \theta_c)$$

Markov Random Fields (cont'd)

$$p(u) \;=\; \frac{1}{Z}\exp(-E(u,\theta))\;, \quad \text{where } E(u,\theta) \;=\; \sum_{c\in\mathcal{C}} \Psi_c(\bar{u}_c,\theta_c)$$

- $\mathcal{C}$ is the set of maximal cliques of the graph (i.e., maximal subgraphs of $\mathcal{G}$ that are fully connected),
- The *clique potential* $\Psi_c,\, c \in \mathcal{C},$ is a non-negative function defined on the RVs in clique $\bar{u}_c,$ parameterized by θ_c .
- Z, the *partition function,* ensures the distribution sums to 1:

$$Z = \sum_{u_1 \ldots u_N} \prod_{c\in\mathcal{C}} \exp(-\Psi_c(\bar{u}_c,\theta_c))$$

- The partition function is important for learning as it's a function of the parameters $\theta = \{\theta_c\}_{c\in\mathcal{C}}$. But often it's not critical for inference.

Image Denoising

- Given a noisy image v, perhaps with missing pixels, recover an image u. that is both smooth and close to v.
- Classical techniques:
  - Linear filtering (e.g. Gaussian filtering)
  - Median filtering
  - Wiener filtering
- Modern techniques
  - PDE-based techniques
  - Non-local methods
  - Wavelet techniques
  - MRF-based techniques

Denoising/smoothing techniques that preserve edges in images

Denoising as a Probabilistic Inference

- Perform maximum a posteriori (MAP) estimation by maximizing the *a posteriori* distribution:

$$p(\text{true image} \mid \text{noisy image}) = p(u \mid v)$$

- By Bayes theorem:

likelihood of noisy image given true image

image prior

$$p(u \mid v) = \frac{p(v \mid u)\, p(u)}{p(v)}$$

normalization term

- If we take logarithm:

$$\log p(u \mid v) = \log p(v \mid u) + \log p(u) - \cancel{\log p(v)}$$

- MAP estimation corresponds to minimizing the encoding cost

$$E(u) = -\log p(v \mid u) - \log p(u)$$

Modeling the Likelihood

- We assume that the noise at one pixel is independent of the others.

$$p(v \mid u) = \prod_{i,j} p(v_{ij} \mid u_{ij})$$

- We assume that the noise at each pixel is additive and Gaussian distributed:

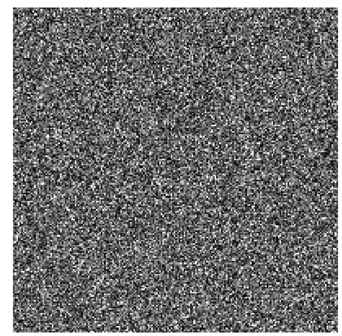
$$p(v_{ij} \mid u_{ij}) = G_\sigma(v_{ij} - u_{ij})$$

- Thus, we can write the likelihood:

$$p(v \mid u) = \prod_{i,j} G_\sigma(v_{ij} - u_{ij})$$

Modeling the Prior

- How do we model the prior distribution of true images?
- What does that even mean?
  - We want the prior to describe how probable it is (a-priori) to have a particular true image among the set of all possible images.



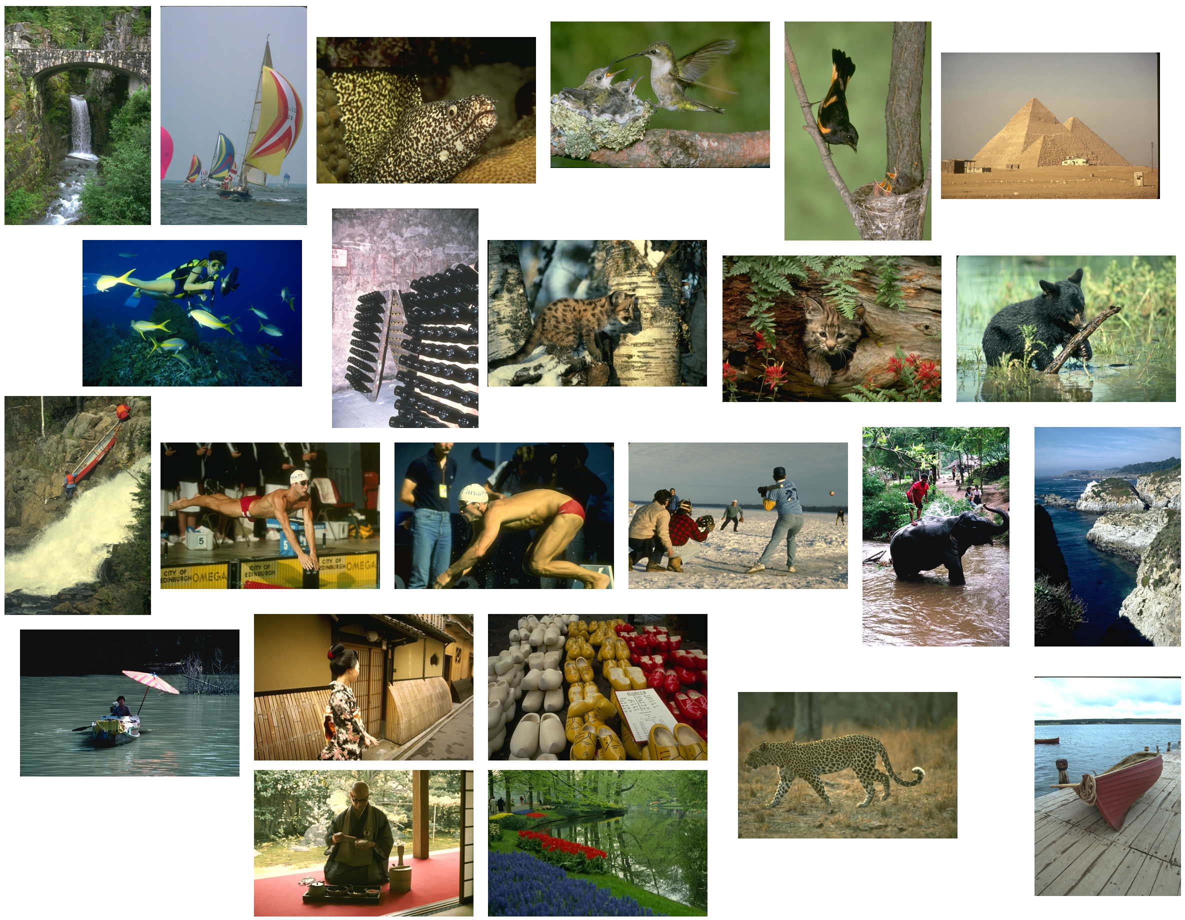
probable

improbable

S. Roth

Natural Images

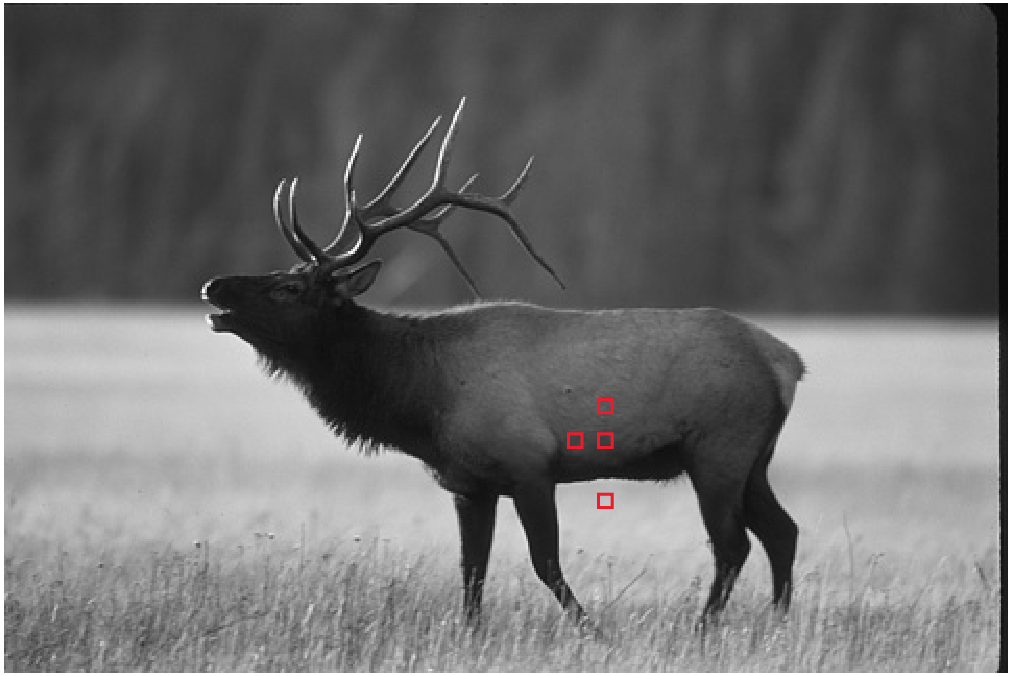
- What distinguishes "natural" images from "fake" ones?



S. Roth

Simple Observation

- Nearby pixels often have a similar intensity:

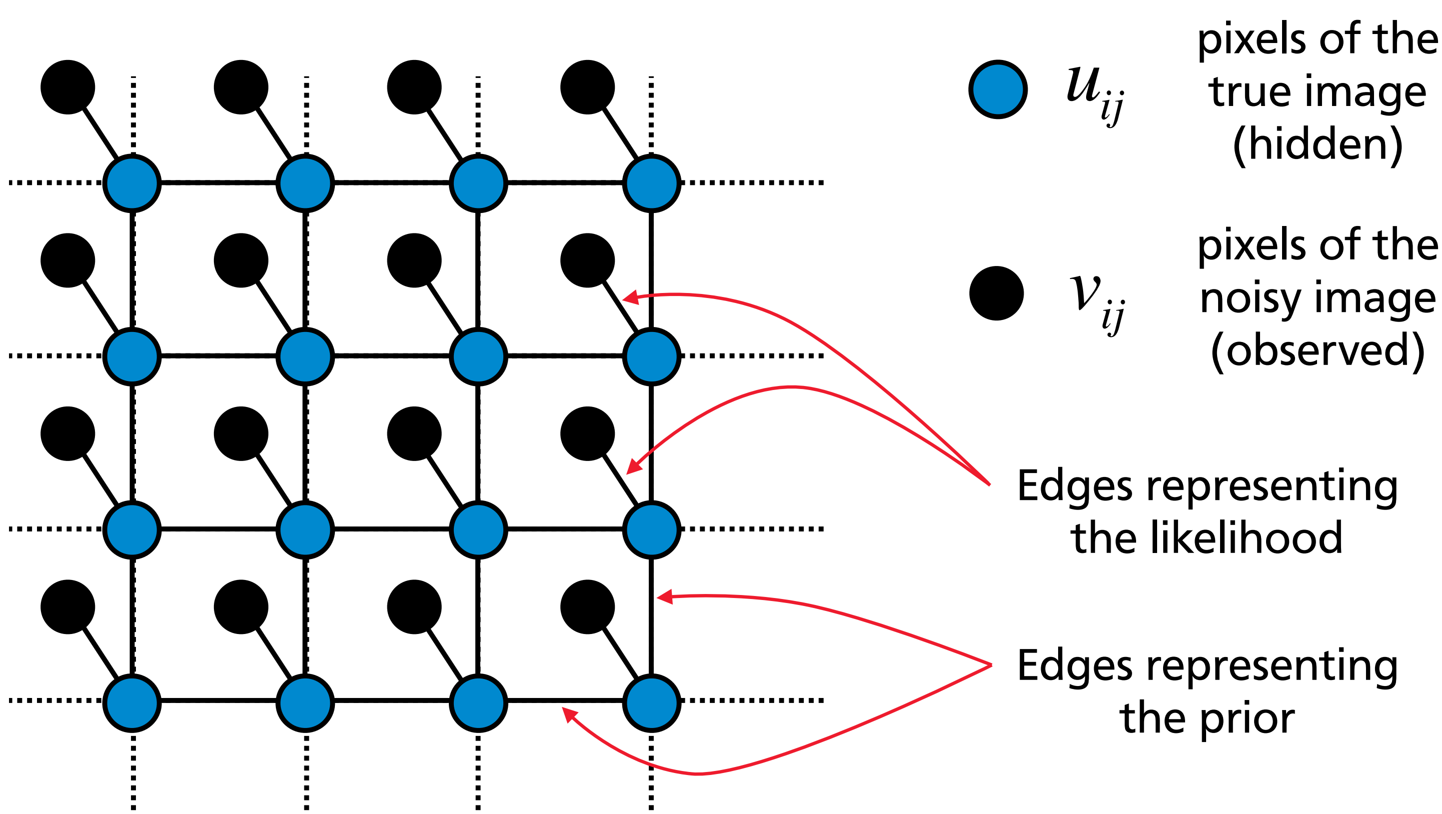


- But sometimes there are large intensity changes.

S. Roth

MRF-based Image Denoising

- Let each pixel be a node in a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with 4-connected neighborhoods.



S. Roth

Image Denoising

- The energy function is given by

$$E(u) \;=\; \sum_{i \in \mathcal{V}} D(u_i) \,+\, \sum_{(i,j) \in \mathcal{E}} V(u_i, u_j)$$

- Unary (clique) potentials D stem from the measurement model, penalizing the discrepancy between the data v and the solution u.

- Interaction (clique) potentials V provide a definition of smoothness, penalizing changes in u between pixels and their neighbors.

D. J. Fleet

Denoising as Inference

- **Goal:** Find the image u that minimizes $E(u)$
- Several options for MAP estimation process:
  - Gradient techniques
  - Gibbs sampling
  - Simulated annealing
  - Belief propagation
  - Graph cut
  - …

Quadratic Potentials in 1D

- Let v be the sum of a smooth 1D signal u and IID Gaussian noise e:
  where $u = (u_1, ..., u_N)$, $v = (v_1, ..., v_N)$, and
  $e = (e_1, ..., e_N)$.

- With Gaussian IID noise, the negative log likelihood provides a quadratic *data term*. If we let the *smoothness term* be quadratic as well, then up to a constant, the log posterior is

$$E(u) \ = \ \sum_{n=1}^{N} (u_n - v_n)^2 + \lambda \sum_{n=1}^{N-1} (u_{n+1} - u_n)^2$$

Quadratic Potentials in 1D

- To find the optimal u^*, we take derivatives of $E(u)$ with respect to u_n :

$$\frac{\partial E(u)}{\partial u_n} = 2\,(u_n - v_n) + 2\lambda\,(-u_{n-1} + 2u_n - u_{n+1})$$

and therefore the necessary condition for the critical point is

$$u_n + \lambda\,(-u_{n-1} + 2u_n - u_{n+1}) = v_n$$

- For endpoints we obtain different equations:

$$u_1 + \lambda\,(u_1 - u_2) = v_1$$

$$u_N + \lambda\,(u_N - u_{N-1}) = v_N$$

N linear equations in the N unknowns

D. J. Fleet

Missing Measurements

- Suppose our measurements exist at a subset of positions, denoted P. Then we can write the energy function as

$$E(u) \;=\; \sum_{n \in P} (u_n - v_n)^2 + \lambda \sum_{\mathrm{all}\, n} (u_{n+1} - u_n)^2$$

- At locations n where no measurement exists, we have:

$$-u_{n-1} + 2\, u_n - u_{n+1} \;=\; 0$$

- The Jacobi update equation in this case becomes:

$$u_n^{(t+1)} = \begin{cases} \frac{1}{1+2\lambda} \left(v_n + \lambda u_{n-1}^{(t)} + \lambda u_{n+1}^{(t)}\right) & \text{for } n \in P\,, \\ \frac{1}{2} \left(u_{n-1}^{(t)} + u_{n+1}^{(t)}\right) & \text{otherwise} \end{cases}$$

D. J. Fleet

2D Image Smoothing

- For 2D images, the analogous energy we want to minimize becomes:

$$E(u) \;=\; \sum_{n,m \in P} (u[n,m] - v[n,m])^2$$
$$+\, \lambda \sum_{\text{all}\, n,m} (u[n{+}1,m] - u[n,m])^2 + (u[n,m{+}1] - u[n,m])^2$$

where P is a subset of pixels where the measurements v are available.

Looks familiar??

Robust Potentials

- Quadratic potentials are not robust to *outliers* and hence they over-smooth edges. These effects will propagate throughout the graph.
- Instead of quadratic potentials, we could use a robust error function ρ:

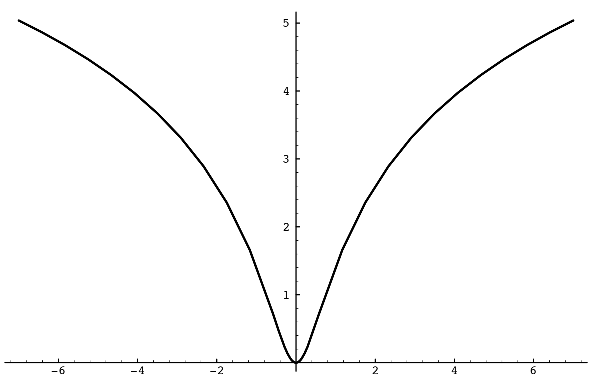
$$E(u) \;=\; \sum_{n=1}^{N} \rho(u_n - v_n,\, \sigma_d) + \lambda \sum_{n=1}^{N-1} \rho(u_{n+1} - u_n,\, \sigma_s)\,,$$

where σ_d and σ_s are scale parameters.

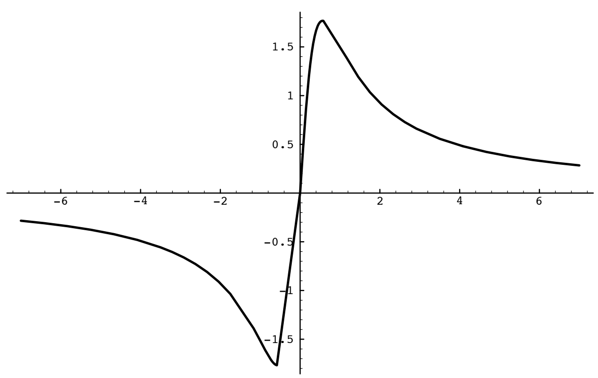
Robust Potentials

- **Example:** the *Lorentzian* error function

$$\rho(z,\sigma) = \log\left(1 + \frac{1}{2}\left(\frac{z}{\sigma}\right)^2\right), \quad \rho'(z,\sigma) = \frac{2z}{2\sigma^2 + z^2}.$$



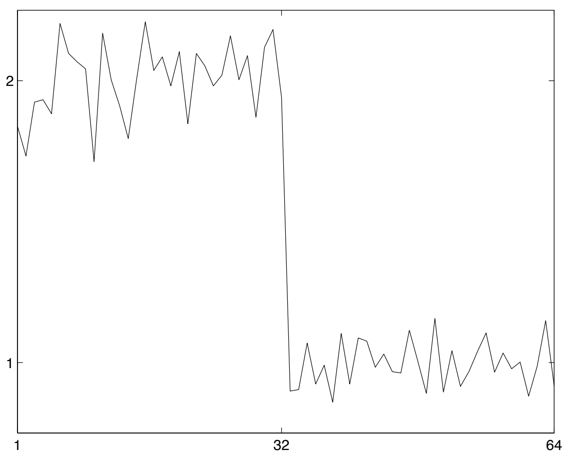
Error function



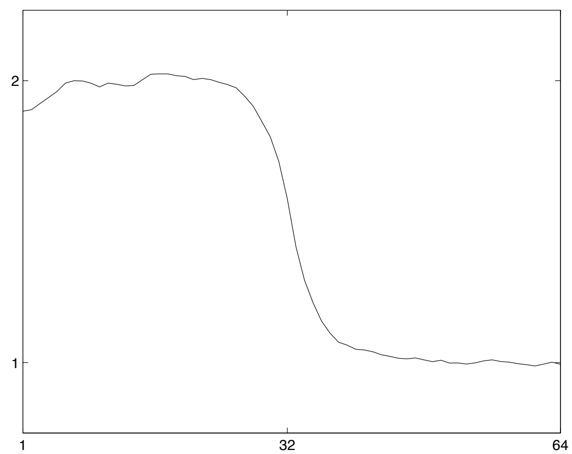
Influence function

Robust Potentials

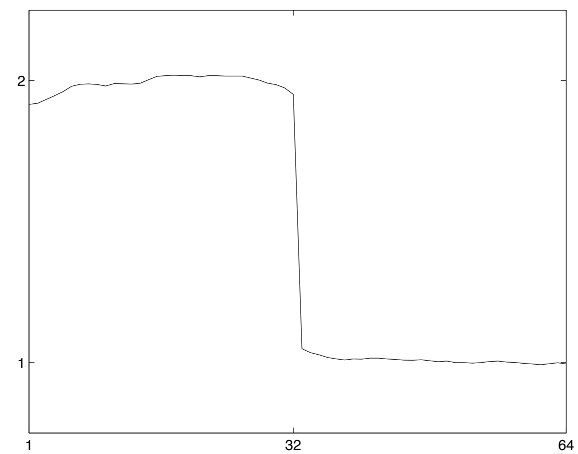
- **Example:** the *Lorentzian* error function
- Smoothing a noisy step edge



Noisy step



LS smoother



Lorentzian smoother

Robust Image Smoothing

- A Lorentzian smoothness potential encourages an approximately piecewise constant result:

Original image

Output of robust smoothing

D. J. Fleet

Robust Image Smoothing

- A Lorentzian smoothness potential encourages an approximately piecewise constant result:

Original image

Edges

D. J. Fleet

Higher-Order MRFs

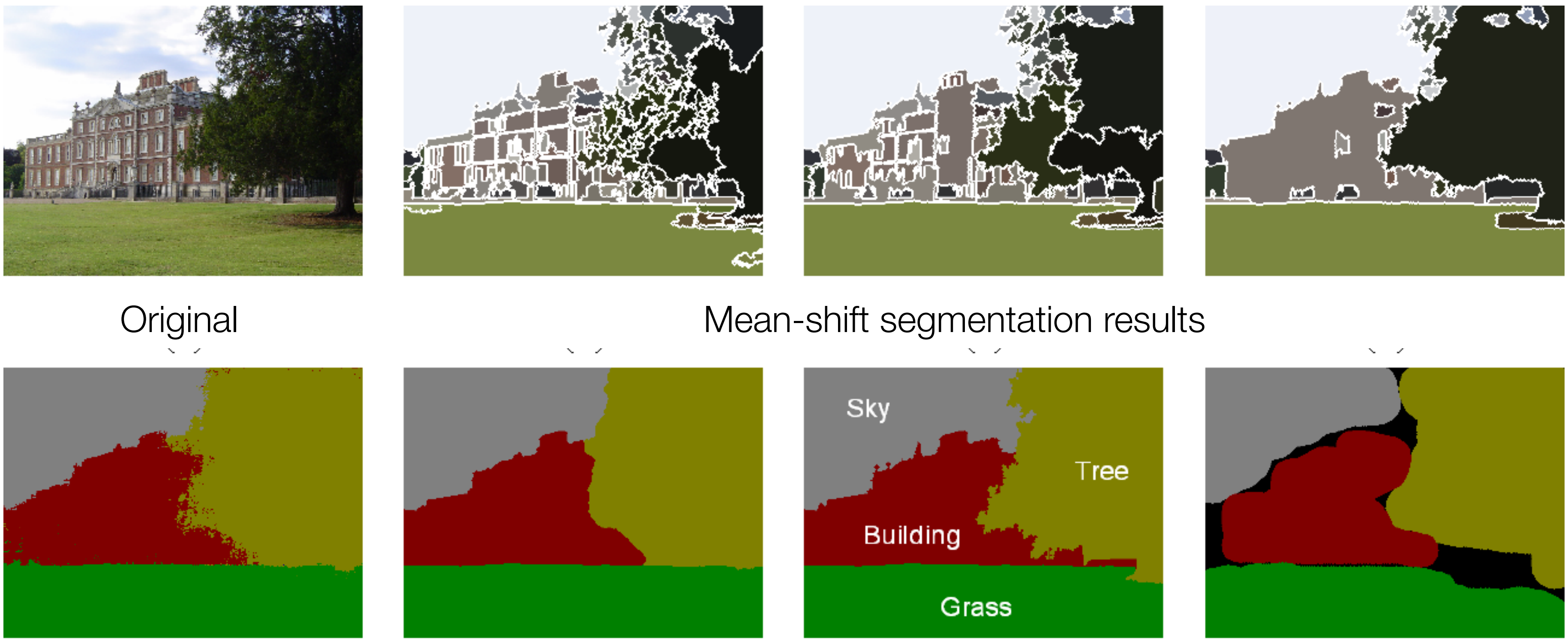
- Typical MRFs use unary and/or pairwise potentials:

$$E(\mathbf{x}) = \sum_{i \in \mathcal{V}} \psi_i(x_i) + \sum_{(i,j) \in \mathcal{E}} \psi_{ij}(x_i, x_j) + \boxed{\sum_{c \in \mathcal{S}} \psi_c(\mathbf{x}_c)}$$

higher-order term

- Employing higher-order potentials results in more expressive MRFs.
  – It enriches the interactions between nodes/pixels.

Higher-Order MRFs



Original

Mean-shift segmentation results

Unary potentials

Pairwise potentials

Higher-order potentials

Ground truth

Kohli et al., Int J Comput Vis (2009)

Higher-Order MRFs

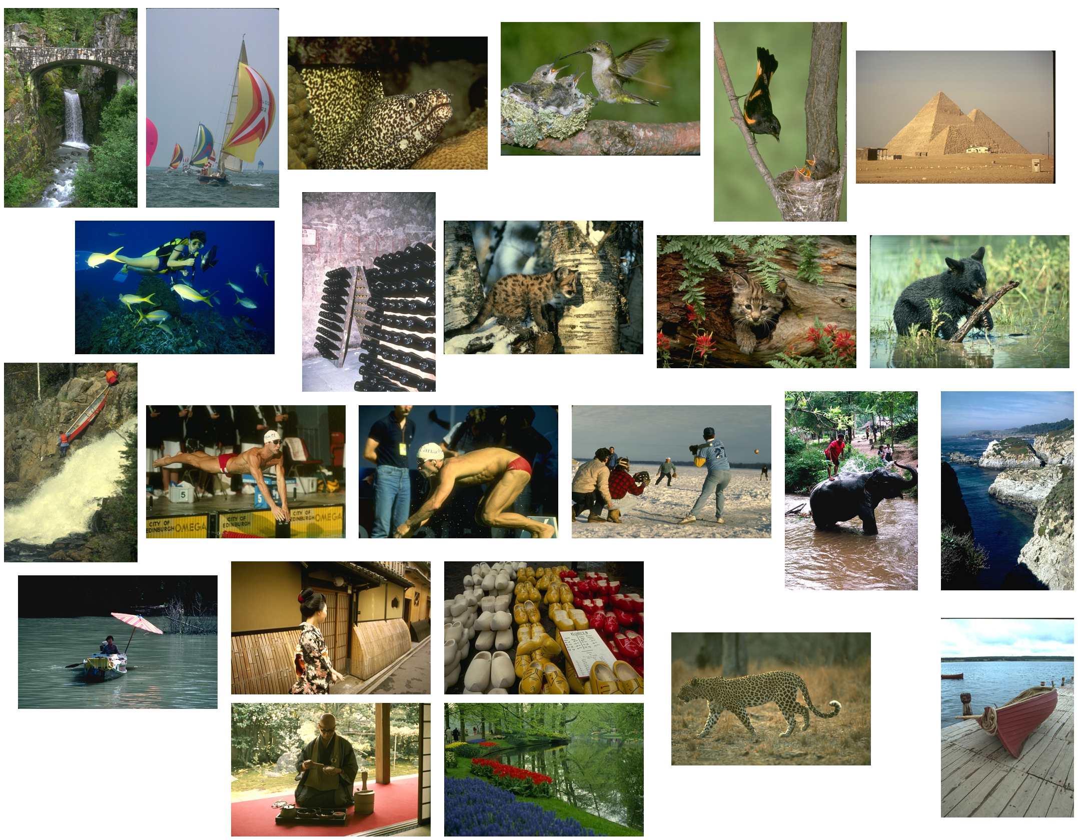
(a) Original (b) Pairwise CRF (c) Robust Pn Model (d) Ground Truth

higher-order potential

Kohli et al., Int J Comput Vis (2009)

Modeling the Potentials

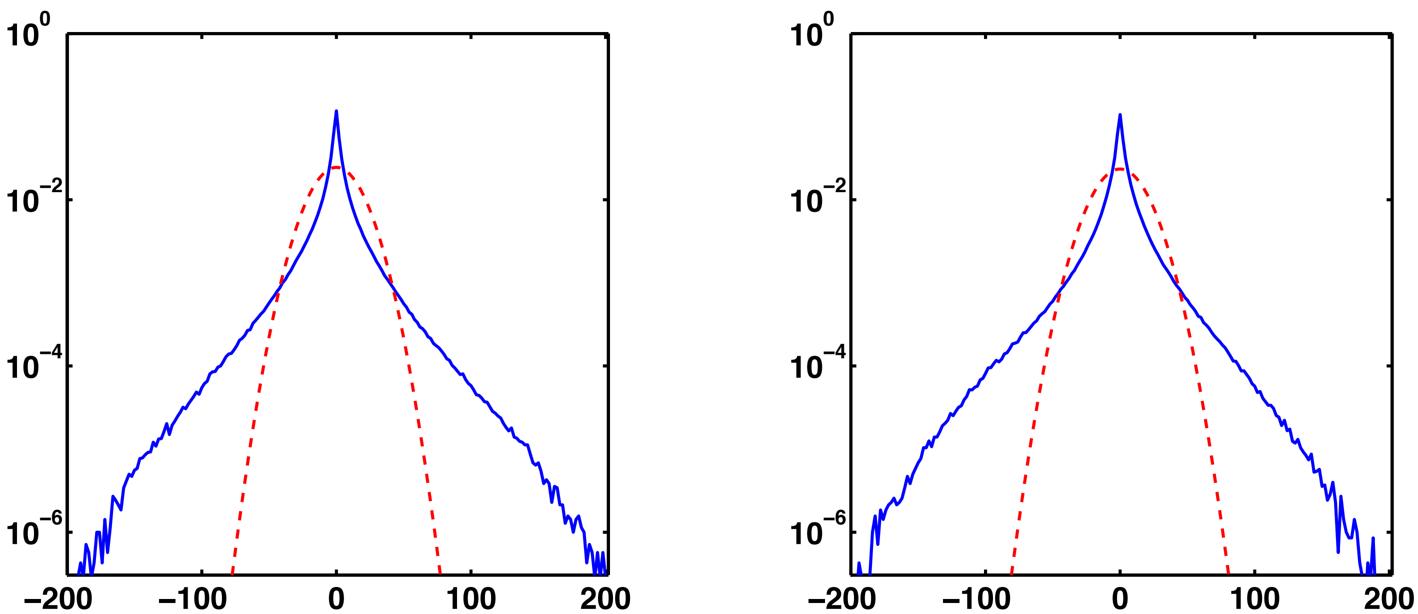
- Could the potentials (image priors) be learned from natural images?



Field of Experts (FoE),
S. Roth & M. J. Black,
CVPR 2005

Statistics of Natural Images

- Compute the image derivative of all images in an image database and plot a histogram:

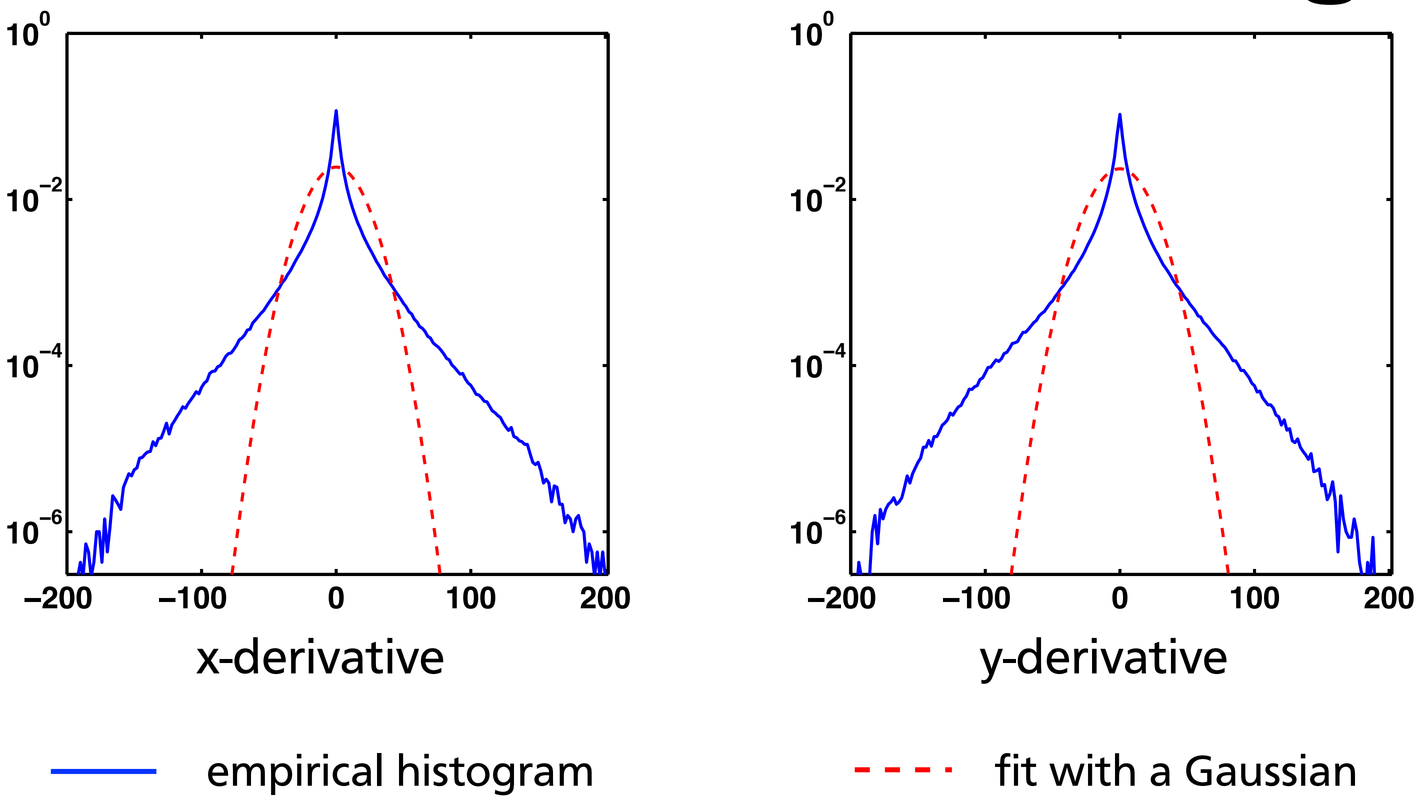


x-derivative   y-derivative

empirical histogram   fit with a Gaussian

S. Roth

Statistics of Natural Images



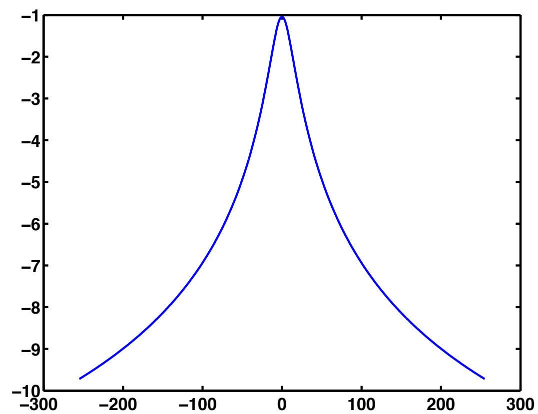
- **Sharp peak at zero:** Neighboring pixels most often have identical intensities.
- **Heavy tails:** Sometimes, there are strong intensity differences due to discontinuities in the image.

S. Roth

Statistics of Natural Images

- Gaussian distributions are inappropriate:
  - They do not match the statistics of natural images well.
  - They would lead to blurred discontinuities.
- Discontinuity-preserving potentials are needed:
- One possibility: Student-t distribution.

$$f_H(T_{i,j}, T_{i+1,j}) = \left(1 + \frac{1}{2\sigma^2}(T_{i,j} - T_{i+1,j})^2\right)^{-\alpha}$$



log-density

S. Roth

Fields of Experts (FoE) denoising results

original image

noisy image,
σ=20

PSNR 22.49dB
SSIM 0.528

denoised using
gradient ascent

PSNR 27.60dB
SSIM 0.810

- Very sharp discontinuities. No blurring across boundaries.
- Noise is removed quite well nonetheless.

S. Roth