Texture Based Image Segmentation

MIT COMPUTER SCIENCE AND ARTIFICIAL INTELLIGENCE LABORATORY

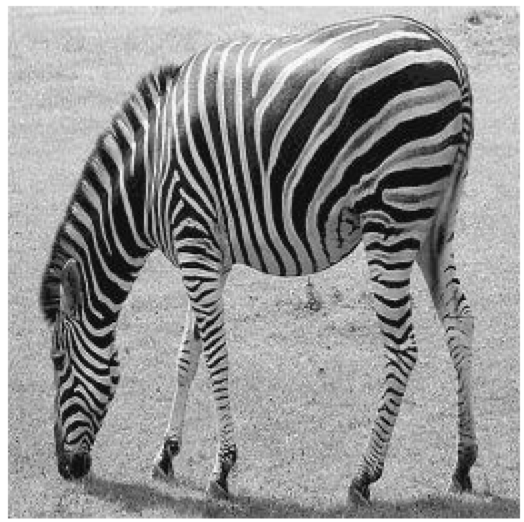
Jason Chang
Massachusetts Institute of Technology
Master's Degree Candidate
October 15, 2008

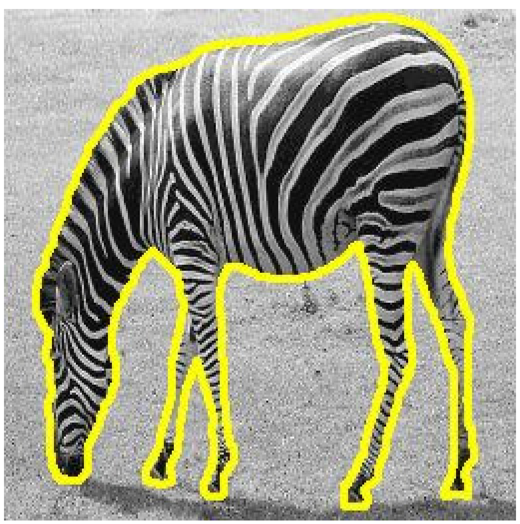
Outline

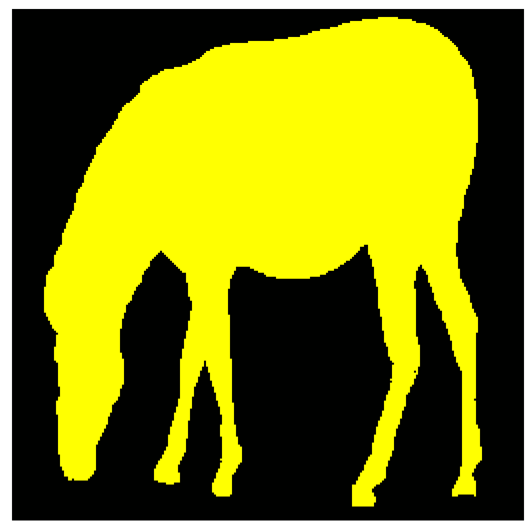
- The Goal
- Level Set Methods
- Previous Work
- Extensions / Improvements
  - Bias Field Estimation
  - Texture Based Segmentation

Image Segmentation

- Separate the image into separate regions
- Focus on Binary Segmentation (two regions, one curve)





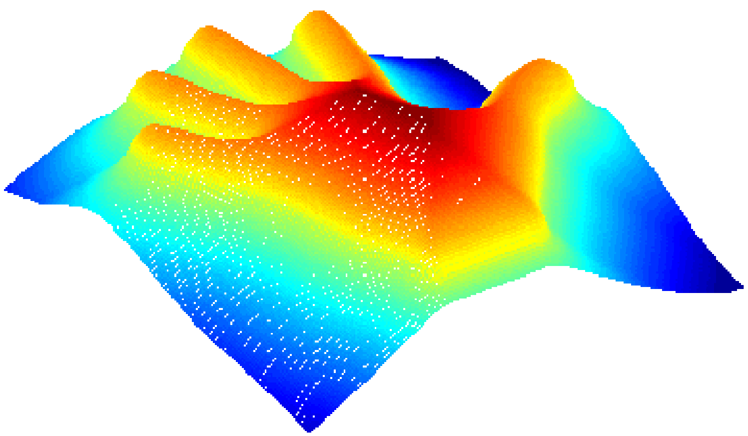


Level Set Methods

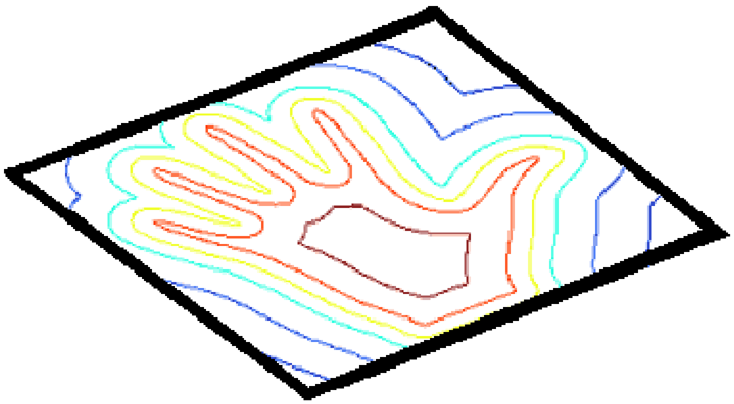
- Curve evolution is defined by an energy functional to minimize over
- Allows for easy manipulation
- Implicitly define a curve on the image with a surface in 3D

Level Set Methods

- Define a height at every pixel in the image



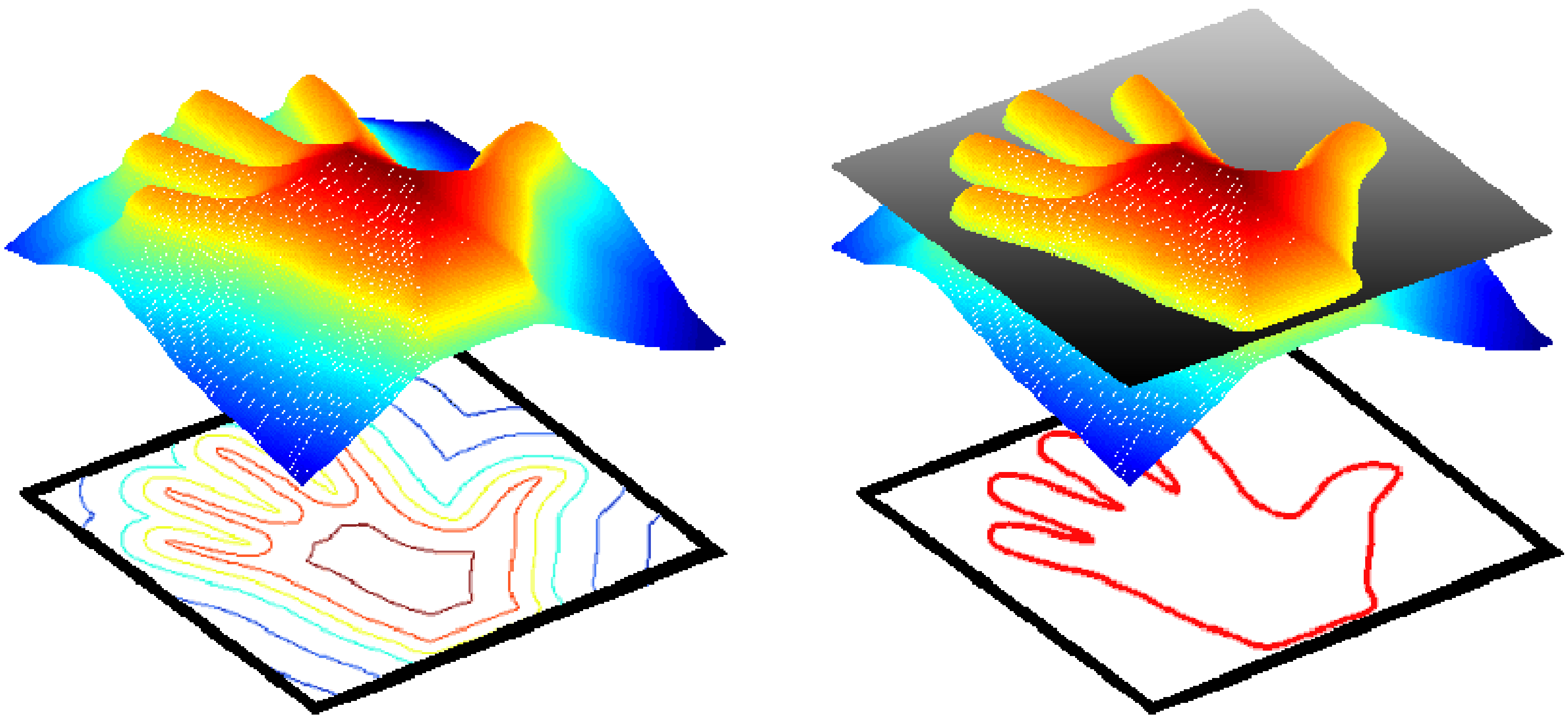
The Surface



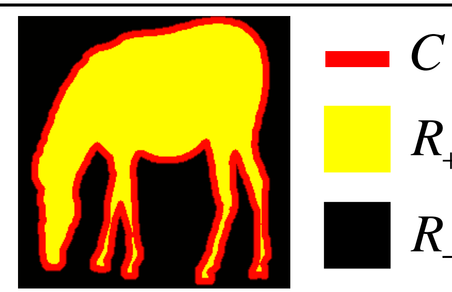
The Level Sets / Contours
of the Surface

Level Set Methods

- The zero level set represents the 2D curve



Some Notation

Notation	Meaning	Comments
j	The pixel location	$j \in \{1, 2, \dots, N\}$
$x(j) \equiv x_j$	The intensity value at pixel j	$x_j \in \{0, 1, \dots, 255\}$
$\phi(j) \equiv \phi_j$	The level set function at pixel j	
$L(j) \equiv L_j = \mathrm{sign}(\phi_j)$	The label assigned to pixel j	$L_j \in \{+1, -1\}$
$R_\pm = \{j \mid L_j = \pm 1\}$	The segmented regions	
$C = \{j \mid \phi_j = 0\}$	The curve that segments the image (zero level set)	

Segmentation Criterion

- Maximize mutual information between pixel intensity and labeling

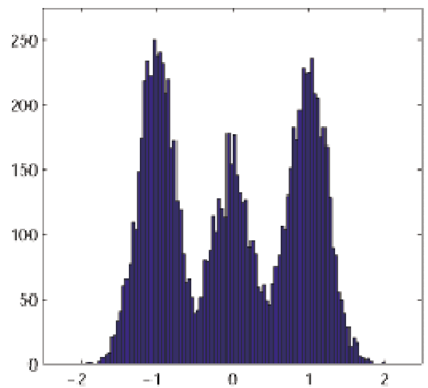
$$J \sim U\{1,\ldots,N\} \qquad I\left(x_J; L_J\right)$$

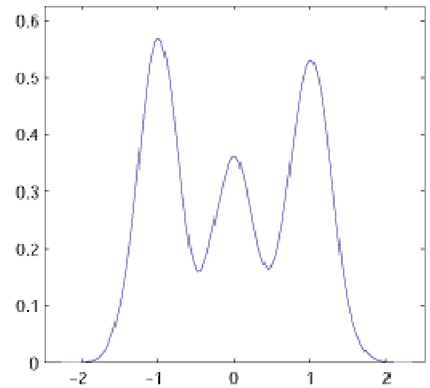
- Approximate MI by using a Kernel Density Estimate to find the PDFs

$$\hat{p}_{x_J|J\in R_\pm}\left(x_j\right) \equiv \hat{p}_x^\pm\left(x_j\right) = \frac{1}{h\left|R_\pm\right|}\sum_{s\in R_\pm} K\left(\frac{x_j - x_s}{h}\right)$$

- Use the Gaussian Kernel

$$K\left(x\right) = \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}}$$





J. Kim, J.W. Fisher III, A. Yezzi, M. Cetin, and A.S. Willsky. A nonparametric statistical method for image segmentation using information theory and curve evolution. *Image Processing, IEEE Transactions on Image Processing*, 14(10):1486-1502, Oct. 2005.

Junmo's Algorithm

- Minimize the energy functional

Curve Length Penalty

$$E(C) = -N \cdot \hat{I}(x_J; L_J) + \boxed{\alpha \oint_C ds}$$

- Gradient flow that minimizes the energy $\forall j \in C$

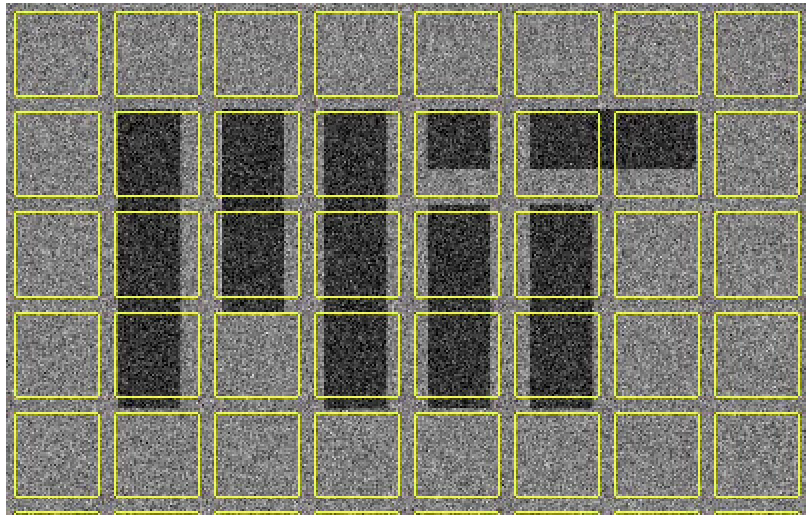
$$\frac{\partial \phi_j}{\partial t} = \left[\log \frac{\hat{p}_x^+(x_j)}{\hat{p}_x^-(x_j)} + \underbrace{\frac{1}{|R_+|} \int_{R_+} \frac{K(x_i - x_j)}{\hat{p}_x^+(x_j)} di - \frac{1}{|R_-|} \int_{R_-} \frac{K(x_i - x_j)}{\hat{p}_x^-(x_j)} di}_{\text{Computationally Intensive}} \right] \vec{N} \boxed{- \alpha \kappa \vec{N}}$$

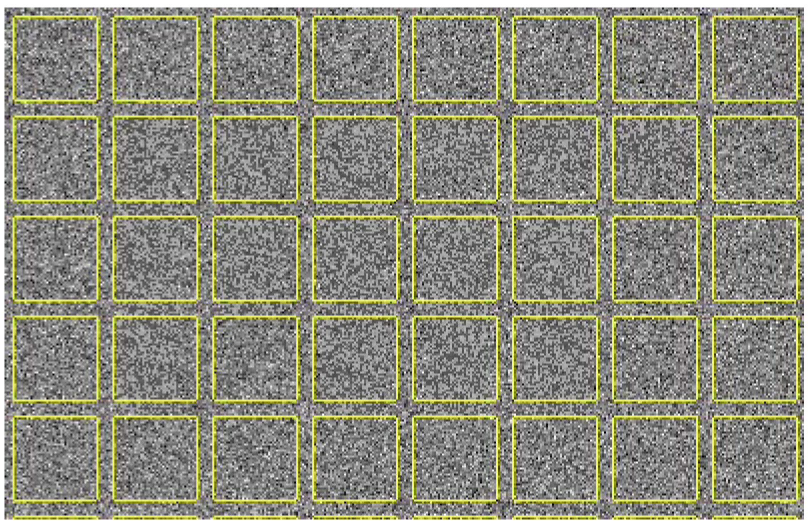
- Approximate Gradient Descent $\forall j \in C$

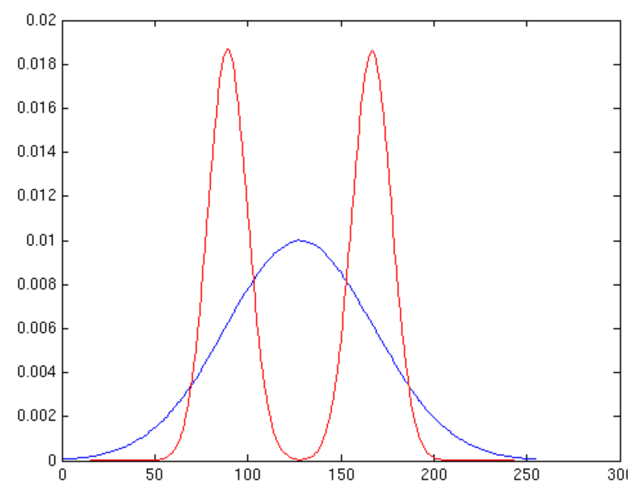
$$\frac{\partial \phi_j}{\partial t} \approx \log \frac{\hat{p}_x^+(x_j)}{\hat{p}_x^-(x_j)} \vec{N} - \alpha \kappa \vec{N}$$

J. Kim, J.W. Fisher III, A. Yezzi, M. Cetin, and A.S. Willsky. A nonparametric statistical method for image segmentation using information theory and curve evolution. *Image Processing, IEEE Transactions on Image Processing*, 14(10):1486-1502, Oct. 2005.

Junmo's Algorithm



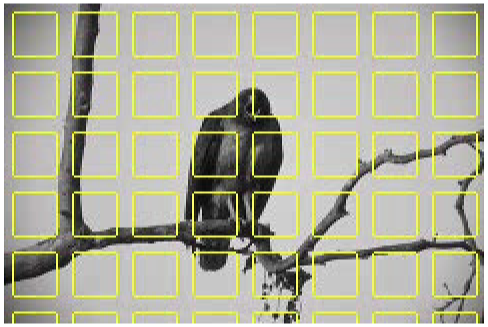


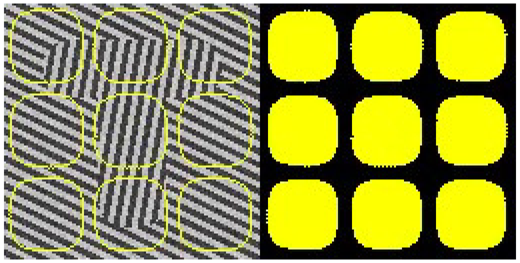


J. Kim, J.W. Fisher III, A. Yezzi, M. Cetin, and A.S. Willsky. A nonparametric statistical method for image segmentation using information theory and curve evolution. *Image Processing, IEEE Transactions on Image Processing*, 14(10):1486-1502, Oct. 2005.

Comments

- Why is the Scalar Segmentation Algorithm Good?
  - Segmentation based on non-parametric densities
  - No training required
- What needs to be improved?
  - Does not perform well on images with lighting effects
  - Only supports grayscale image segmentation
  - Textured images can not be segmented





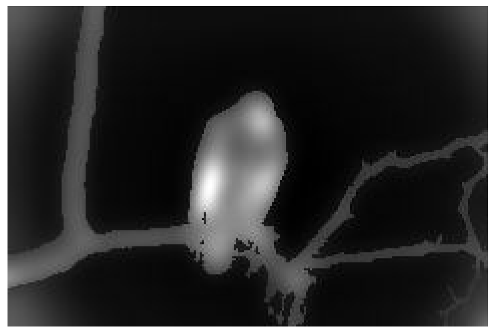
Bias Field Estimation

- Assume the observed image is the product of an intrinsic image and a multiplicative gain field

Basic Intrinsic Image

x

Gain Field



=

Observed Image

$$\mathbf{b} \times \mathbf{g} = \mathbf{x}$$

Bias Field Estimation

- Bias field is the log of the Gain field

$$x_j = b_j \times g_j$$
$$y_j = \log(x_j) = \log(b_j) + \log(g_j) = \log(b_j) + \beta_j$$

- Assume that the intrinsic image pixels, b_j, are i.i.d. conditioned on knowing the regions $R_\pm$

- Find the MAP estimate of $\boldsymbol{\beta}$ for a given segmentation

$$\hat{\boldsymbol{\beta}}_{\mathrm{MAP}} = \Lambda_\beta \mathbf{f}(\boldsymbol{\beta}), \quad \left[\mathbf{f}(\boldsymbol{\beta})\right]_j = \frac{\sum_i \Pr\left[L_j = i\right] \frac{\partial}{\partial \beta_j}\left[\hat{p}_y\left(y_j \mid \beta_j, L_j\right)\right]}{\sum_i \Pr\left[L_j = i\right] \hat{p}_y\left(y_j \mid \beta_j, L_j\right)}$$

- Use a fixed-point iteration to find β

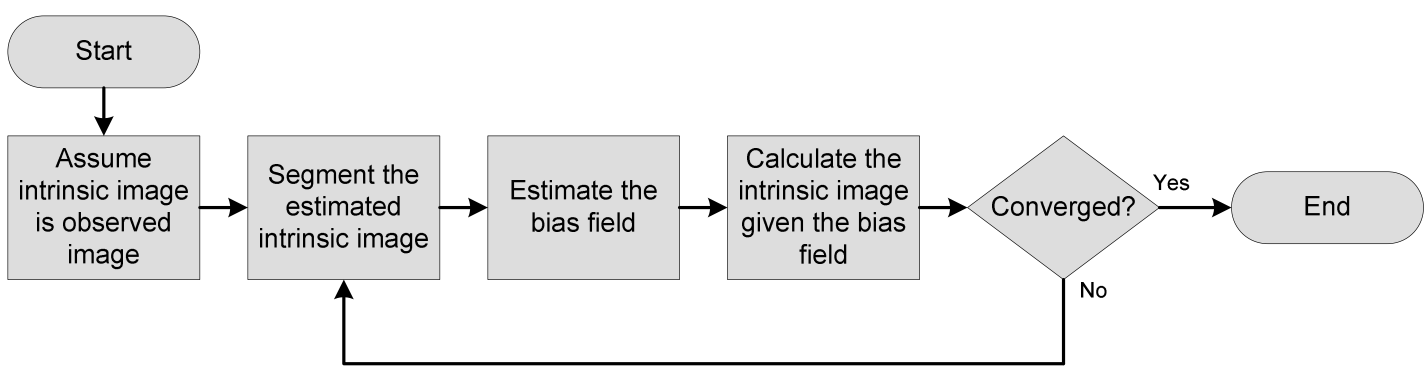
$$\hat{\boldsymbol{\beta}}^{(k+1)} = \Lambda_\beta \mathbf{f}\left(\hat{\boldsymbol{\beta}}^{(k)}\right)$$

W.M. Wells, III, W.E.L. Grimson, R. Kikinis and F.A. Jolesz. Adaptive segmentation of MRI data. *IEEE Transactions on Medical Imaging*, Vol. 15, pp. 429-442, 1995.

Bias Field Estimation

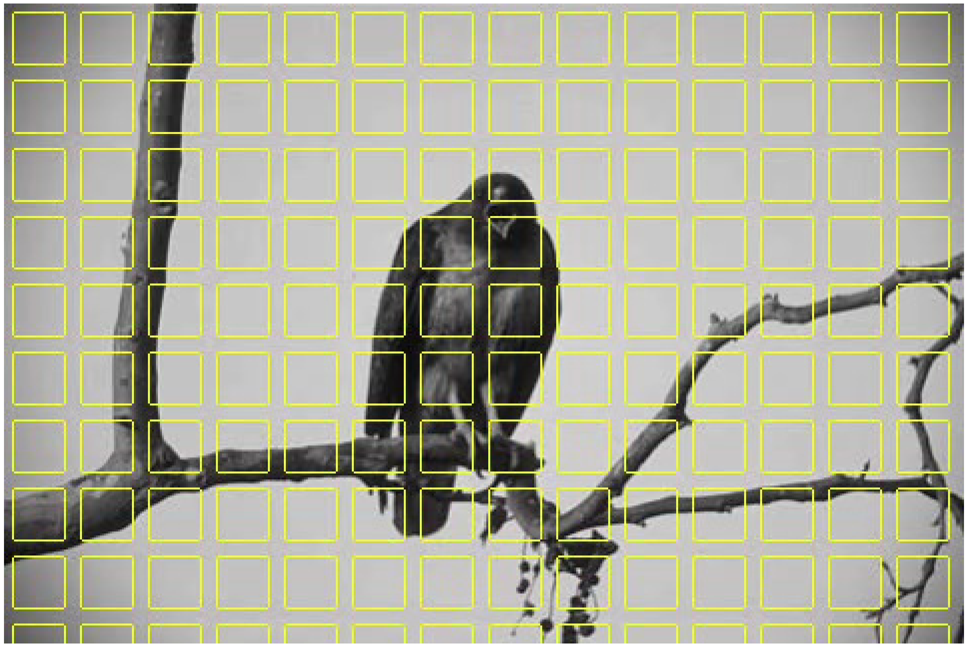
Segmentation Algorithm with Bias Field Estimation

1. Assume that the bias field, $\boldsymbol{\beta}$, is zero and the intrinsic image is just the observed image
2. Segment the estimated intrinsic image
3. Estimate the bias field, $\boldsymbol{\beta}$, given the current segmentation
4. Find the estimated intrinsic image, $\mathbf{b}$, from the estimated bias field
5. Repeat from Step 2 until convergence



Bias Field Estimation

- Alternate between segmentation and bias field estimation



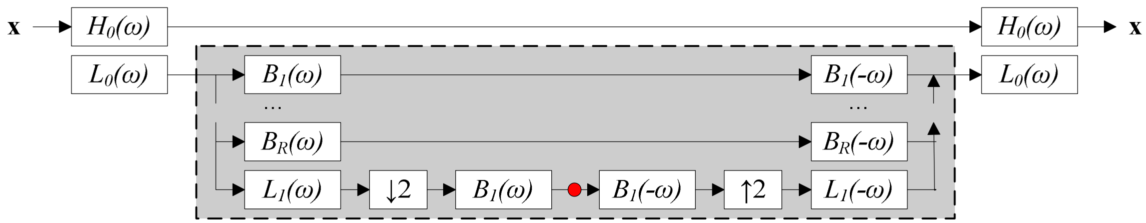
Vector Segmentation

- Extending the formulation to vector values (notice the **bold** vector $\mathbf{r}_i$, instead of the scalar x_i)

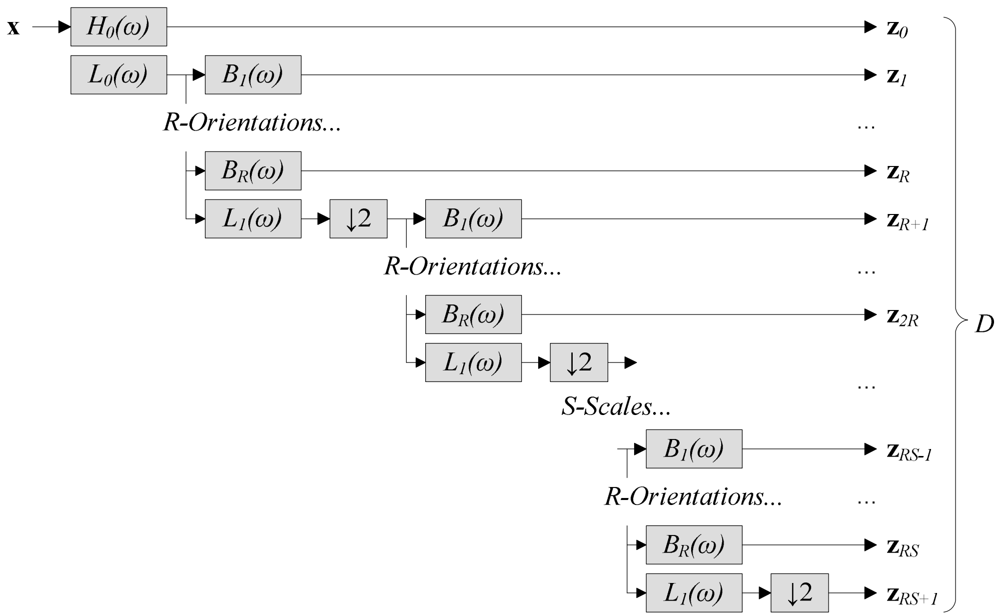
$$\frac{\partial \phi_j}{\partial t} \approx \left[\log \frac{\hat{p}_x^+(\mathbf{v}_j)}{\hat{p}_x^-(\mathbf{v}_j)} \right] \vec{N} - \alpha \kappa \vec{N}$$

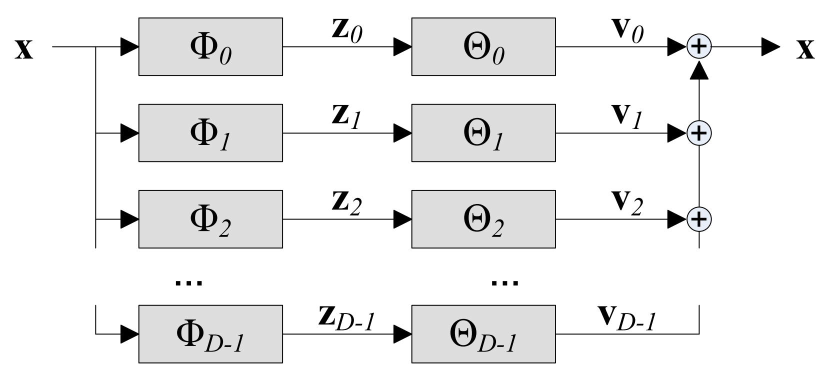
- Vector-valued images can be segmented
  - Color images are segmented using $\mathbf{v}_j = [R, G, B]$
  - Texture images can be segmented by representing each pixel with a texture vector

Steerable Pyramid



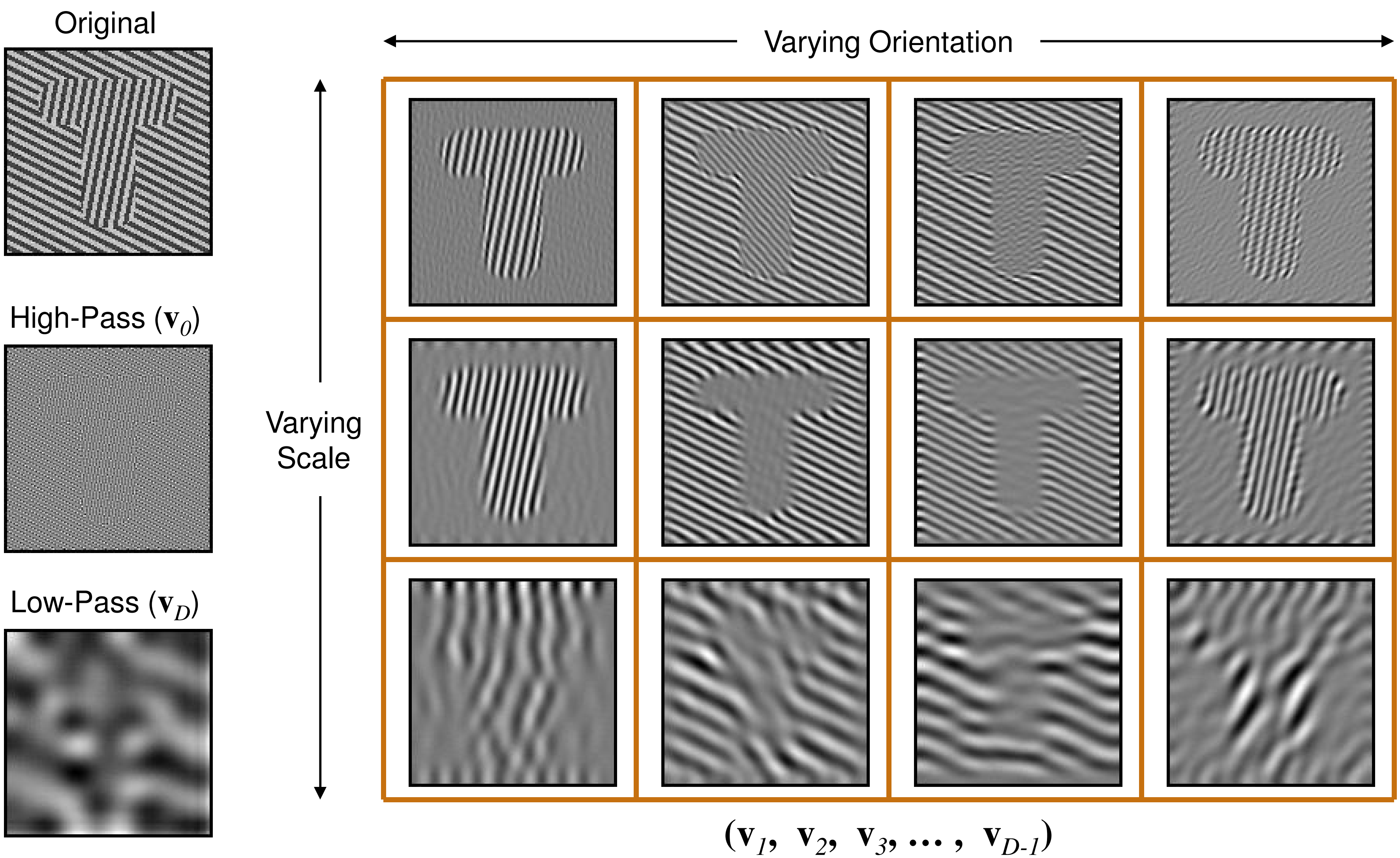
Recursively replace the gray box at the red dot





Eero P. Simoncelli and William T. Freeman. The steerable Pyramid: A flexible architecture for multi-scale derivative computation. In *International Conference on Image Processing*, volume 3, pages 444-447, 23-26 Oct. 1995, Washington DC, USA, 1995

Steerable Pyramid



Dimensionality Reduction

- Estimating PDFs using the Improved Fast Gauss Transform (Yang et al.) has complexity:

$$O\left(D^c\left(M+N\right)\right)$$

D – dimensionality M – # target points N – # source points

- Problems with high dimensionality
  - Takes ~1 hour to do a KDE on 14 dimensions
  - Sparse data in 14 dimensions provide for poor estimate

C. Yang, R. Duraiswami, N. Gumerov, and L. David. Improved fast gauss transform and efficient kernel density estimation. *IEEE International Conference on Computer Vision*, pages 464-471, 2003.

Dimensionality Reduction

- We can reconstruct the original image from the outputs

$$\mathbf{x} = \sum_{i=0}^{D} \Theta_i \mathbf{z}_i = \sum_{i=0}^{D} \mathbf{v}_i$$

- Approximately reconstruct $\mathbf{x}$ by using a subset of the outputs

$$\hat{\mathbf{x}} = \sum_{i=0}^{D-1} \mathbf{v}_i u_i \qquad \mathbf{u} \in \{0,1\}^D$$

- Define the error of the reconstruction as the MSE

$$e(\mathbf{x},\hat{\mathbf{x}}) = \|\mathbf{x}-\hat{\mathbf{x}}\|_2^2 = (\mathbf{x}-\hat{\mathbf{x}})^T (\mathbf{x}-\hat{\mathbf{x}})$$

- Perform the following optimization

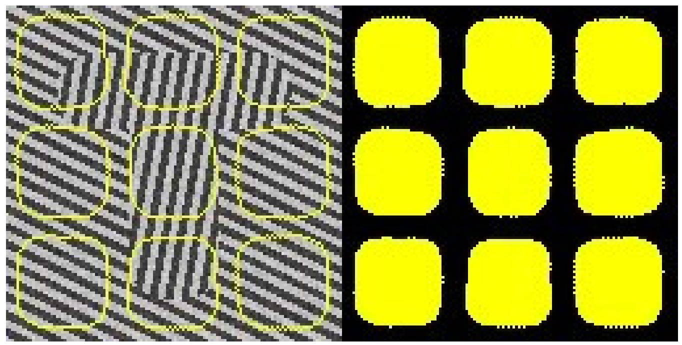
$$\mathbf{u}^* = \arg\min_{\mathbf{u}\in\{0,1\}^D} \left[e(\mathbf{x},\hat{\mathbf{x}}) \right]$$

$$s.t.\ |\mathbf{u}| = \sum_{i=0}^{D-1} u_i = d$$

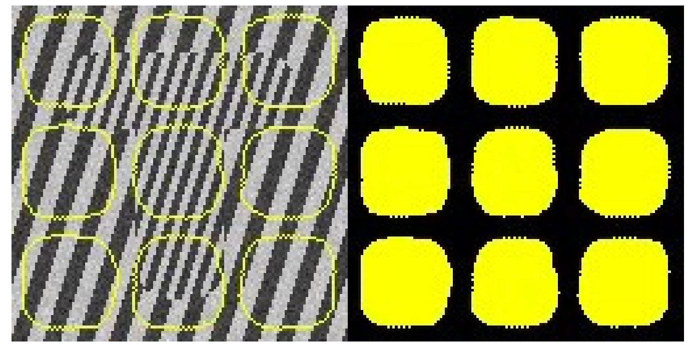
Pyramid Subset Results

Ground Truth

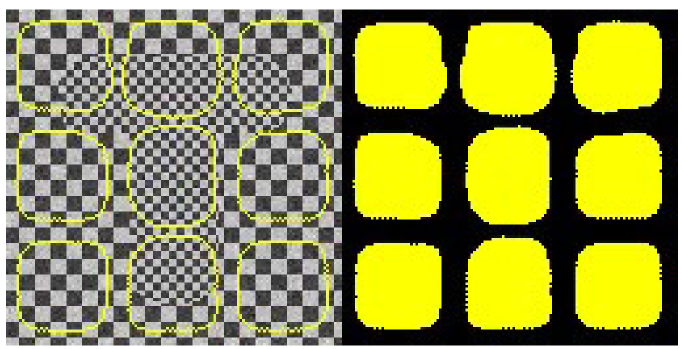
Oriented Stripes



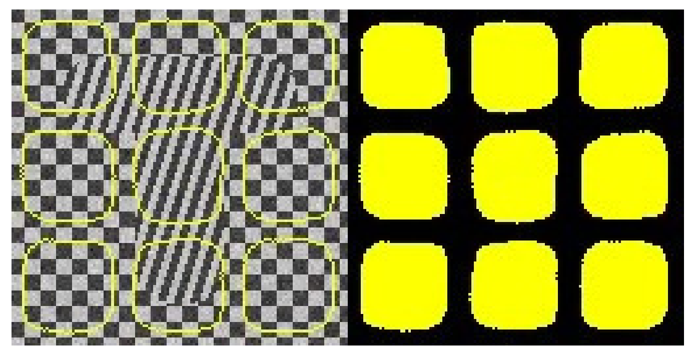
Scaled Stripes



Scaled Checkerboard



Different Textures

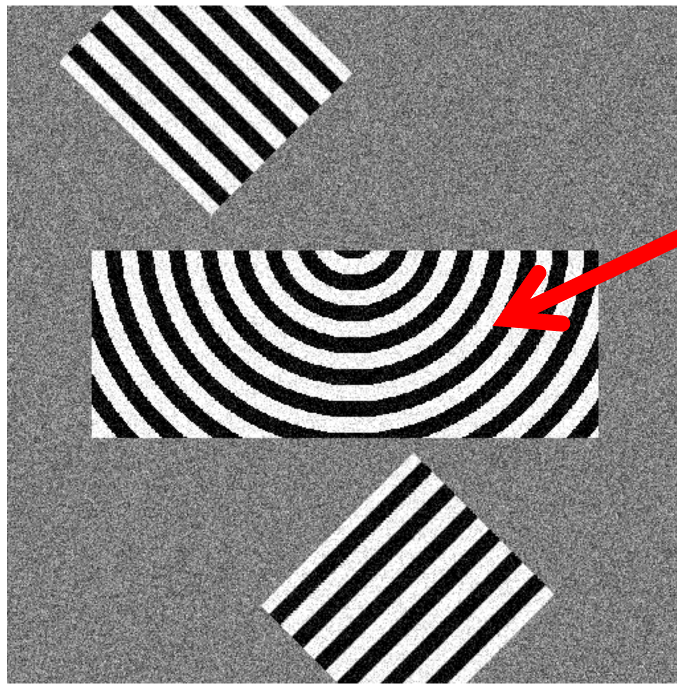


(Using d=3)

Smoothly Varying Textures

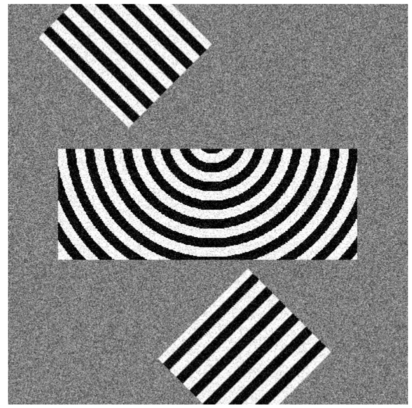
- Try to capture a texture that varies smoothly in orientation and scale

Smoothly Varying Textures

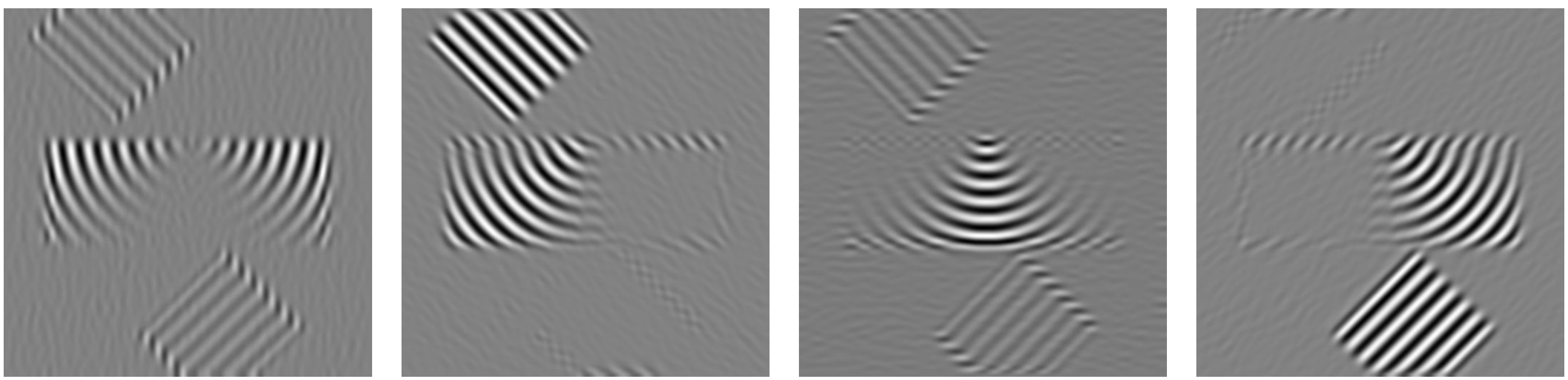


Smoothly changing orientation

Smoothly Varying Textures



Outputs at 4 orientations and 1 scale



Smoothly Varying Textures

Junmo's Scalar Segmentation

Vector Segmentation (Pyramid Subset)

Vector Segmentation (Smoothly Varying Textures)

Thanks!

MIT COMPUTER SCIENCE AND ARTIFICIAL INTELLIGENCE LABORATORY

CSAIL

Questions / Comments?