Mathematical and Perceptual Models for Image Segmentation

Thrasos Pappas

Electrical & Computer Engineering Department

Northwestern University

pappas@ece.northwestern.edu

http://www.ece.northwestern.edu/~pappas

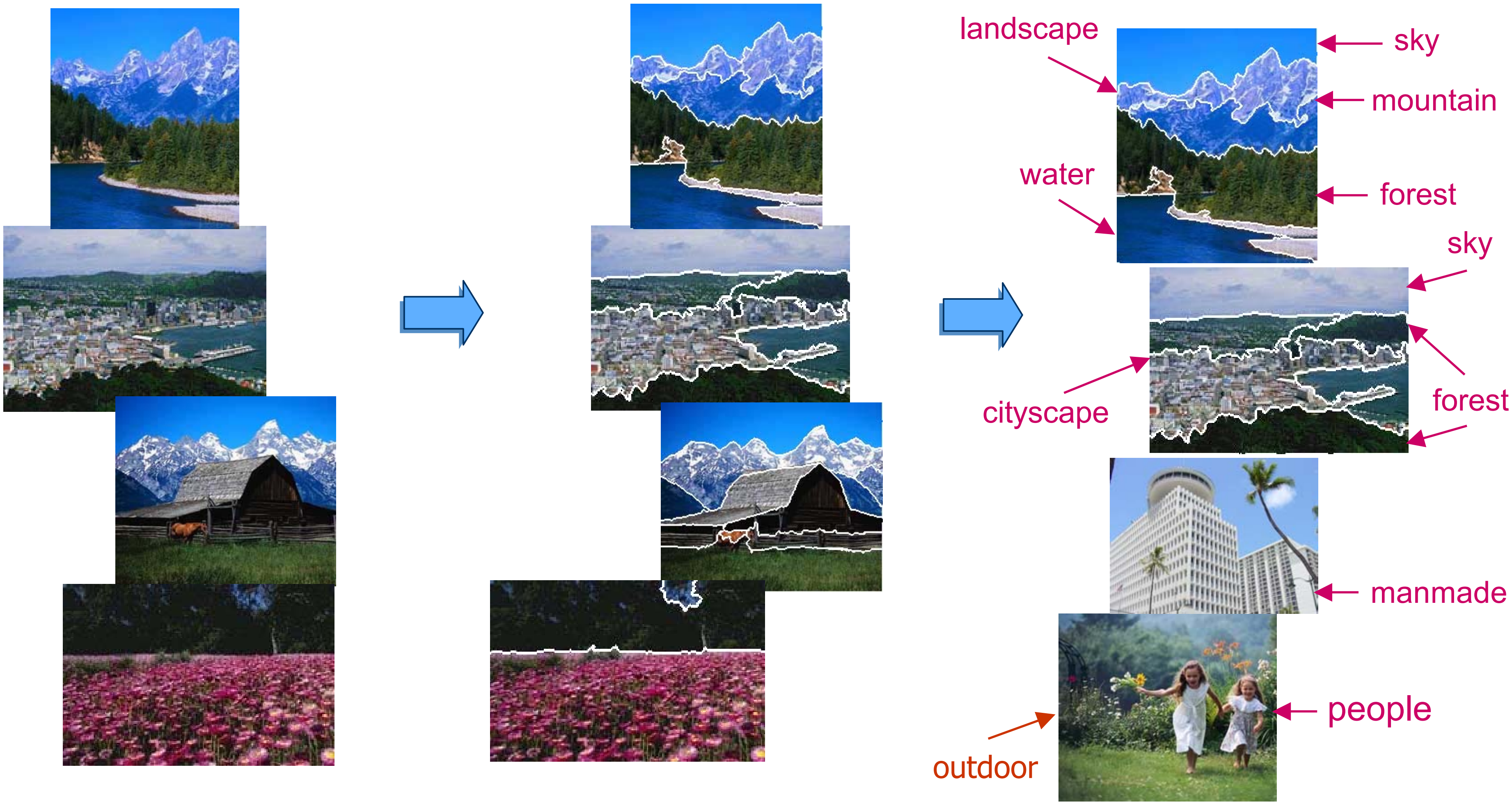
Banff, July 27, 2005

People

- Junqing Chen, Unilever Research
- Dejan Depalov, Northwestern University
- Aleksandra Mojsilovic, IBM T.J. Watson Research Center
- Bernice Rogowitz, IBM T.J. Watson Research Center
- Dongge Li, Motorola Labs
- Bhavan Gandhi, Motorola Labs

Problem

Images | "Ideal" Segmentations | Semantic Categories



Semantic Information Extraction

- Motivation
  - Proliferation of image and video acquisition devices (digital still and video cameras, image and video phones, PDAs)
  - World rich in digital visual content
  - Large personal repositories (consumer market)
  - Increasing processing capabilities
- Goal: Intelligent content management
  - Semantic labeling
  - Content organization
  - Efficient retrieval
- Techniques
  - Image and video segmentation
  - Extracting semantically related features
  - Relating features to semantic categories

Challenges

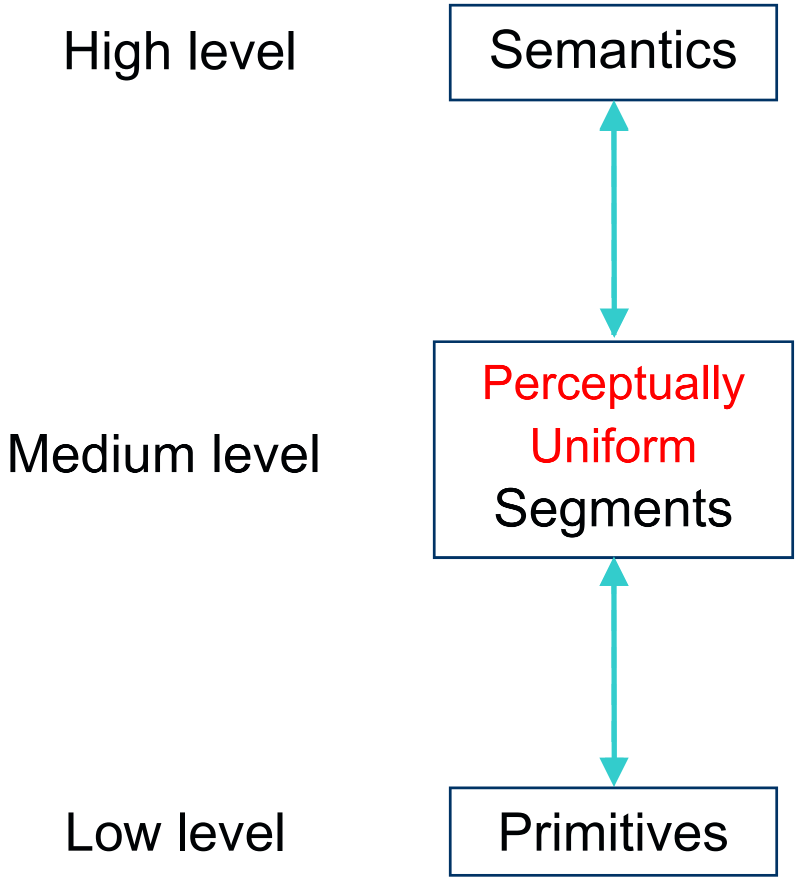
- What are the important semantic categories?
- How to link the low-level features to semantically important categories?

Semantic Categories

- Recent perceptual experiments by Mojsilovic and Rogowitz identified important semantic categories that humans use for image classification

- Conjecture: Semantic categories can be derived from combinations of low-level image features

Bridging the Semantic Gap



Use segment descriptors and statistical techniques to relate segments (first) and scenes (later) to semantic categories/labels

Incorporate knowledge of human perception and image characteristics into feature extraction and algorithm design

Adaptive Clustering Algorithm

Adaptive Clustering Algorithm


Original Image


K-means Class Labels


ACA Class Labels

Adaptive Clustering Algorithm (ACA)

- K-means clustering (LBG)
  - Based on image histogram
  - No spatial constraints
  - Each cluster is characterized by constant intensity
- Add spatial constraints
  - Region model: Markov/Gibbs random field
- Make it adaptive
  - Cluster centers spatially varying
  - Texture model: spatially varying mean + WGN
- MAP estimates of segmentation x given observation y

$$p(x \mid y) \propto p(y \mid x)\, p(x)$$

ACA

- K-means minimizes

$$\sum_s (y_s - \mu^{x_s})^2$$

- Adaptive clustering maximizes

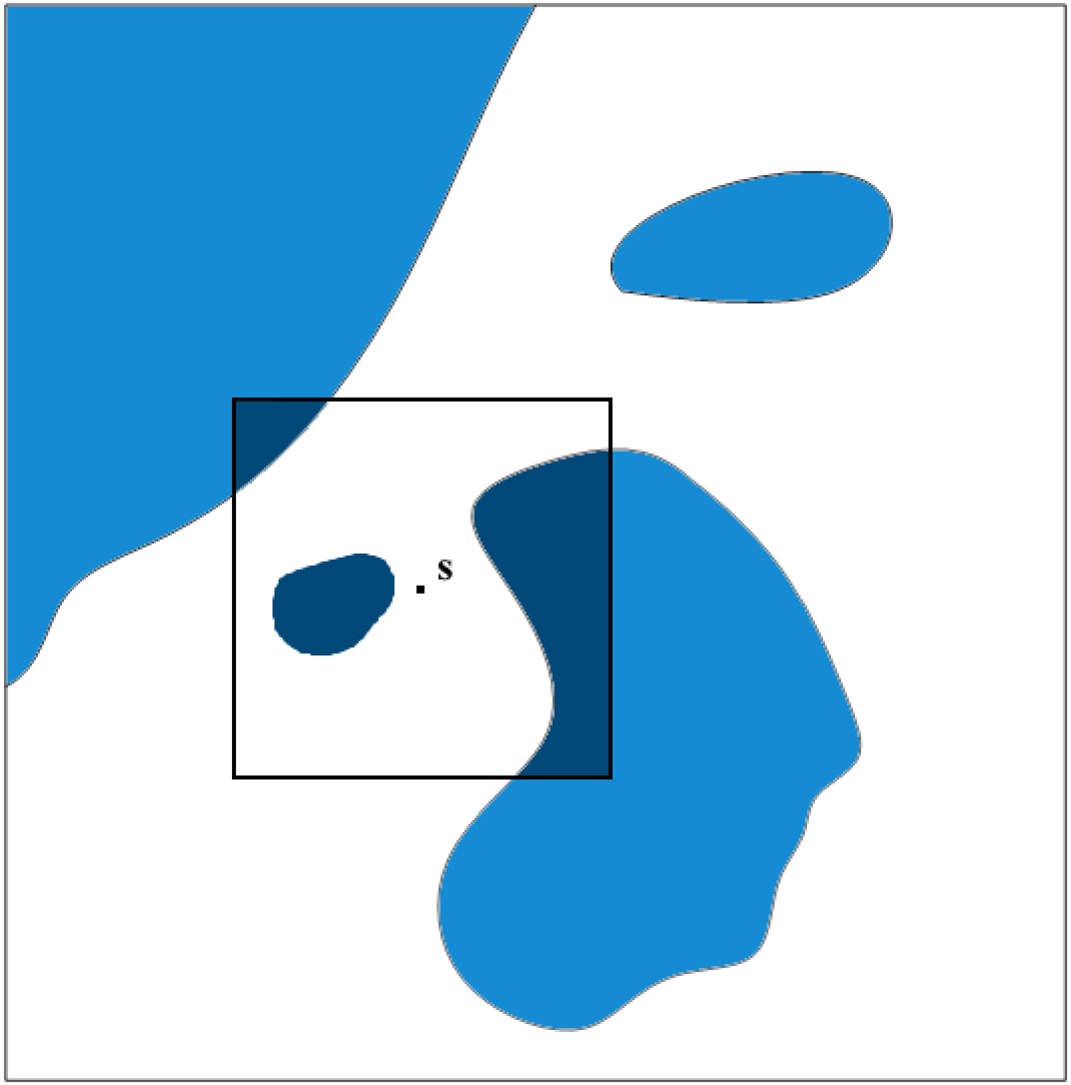
$$p(x \mid y) \propto \exp\left\{-\sum_s \frac{1}{2\sigma^2}(y_s - \mu_s^{x_s})^2 - \sum_C V_C(x)\right\}$$

- Or, minimizes

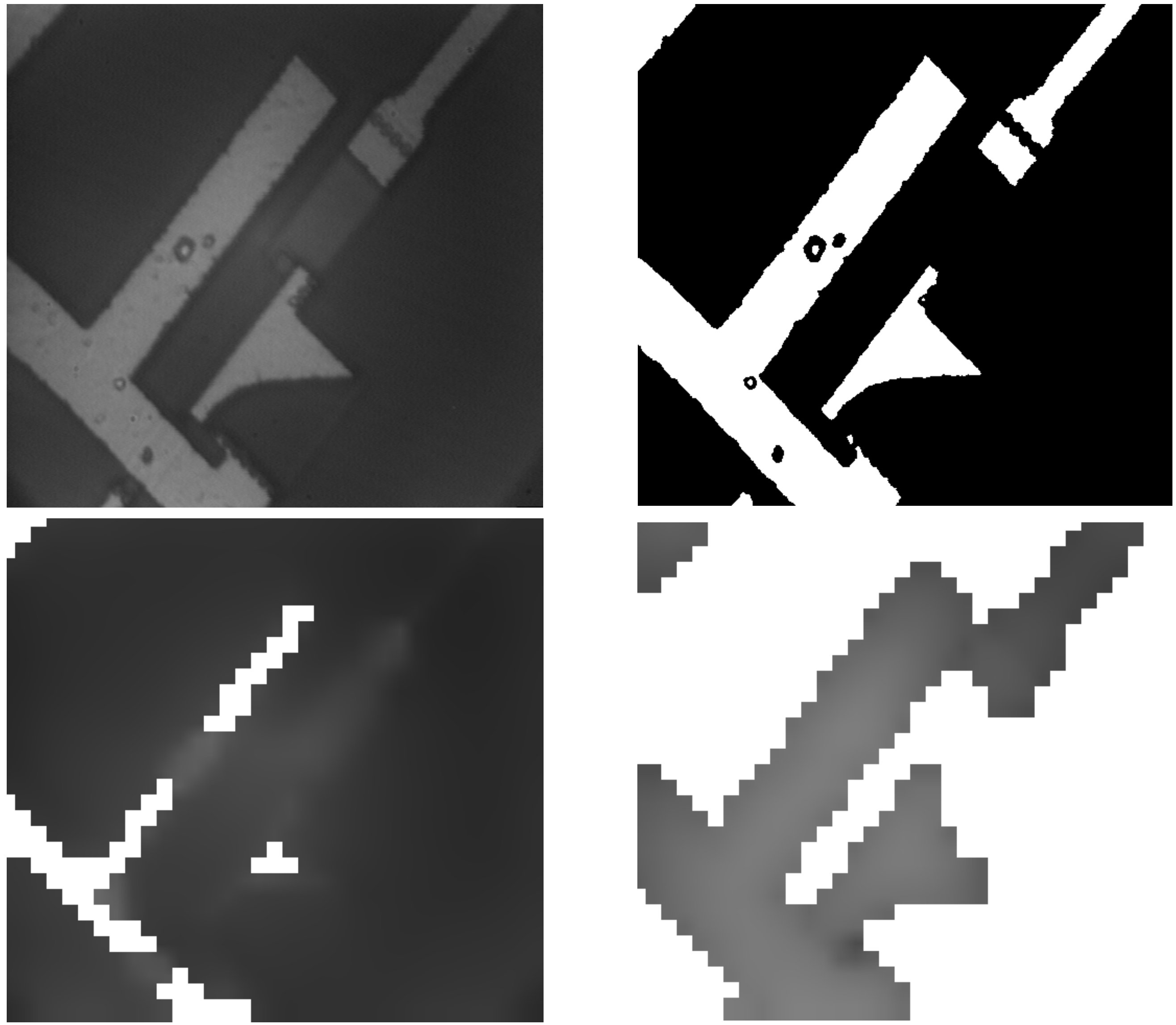
$$\sum_s \frac{1}{2\sigma^2}(y_s - \mu_s^{x_s})^2 + \sum_C V_C(x)$$

ACA: Local Intensity Function Estimation

- Given x, segmentation into classes
- Estimate $\mu_s^{x_s}, \forall x_s, s$
  Intensity function for each class at each point in the image
- Use hierarchy of window sizes



ACA

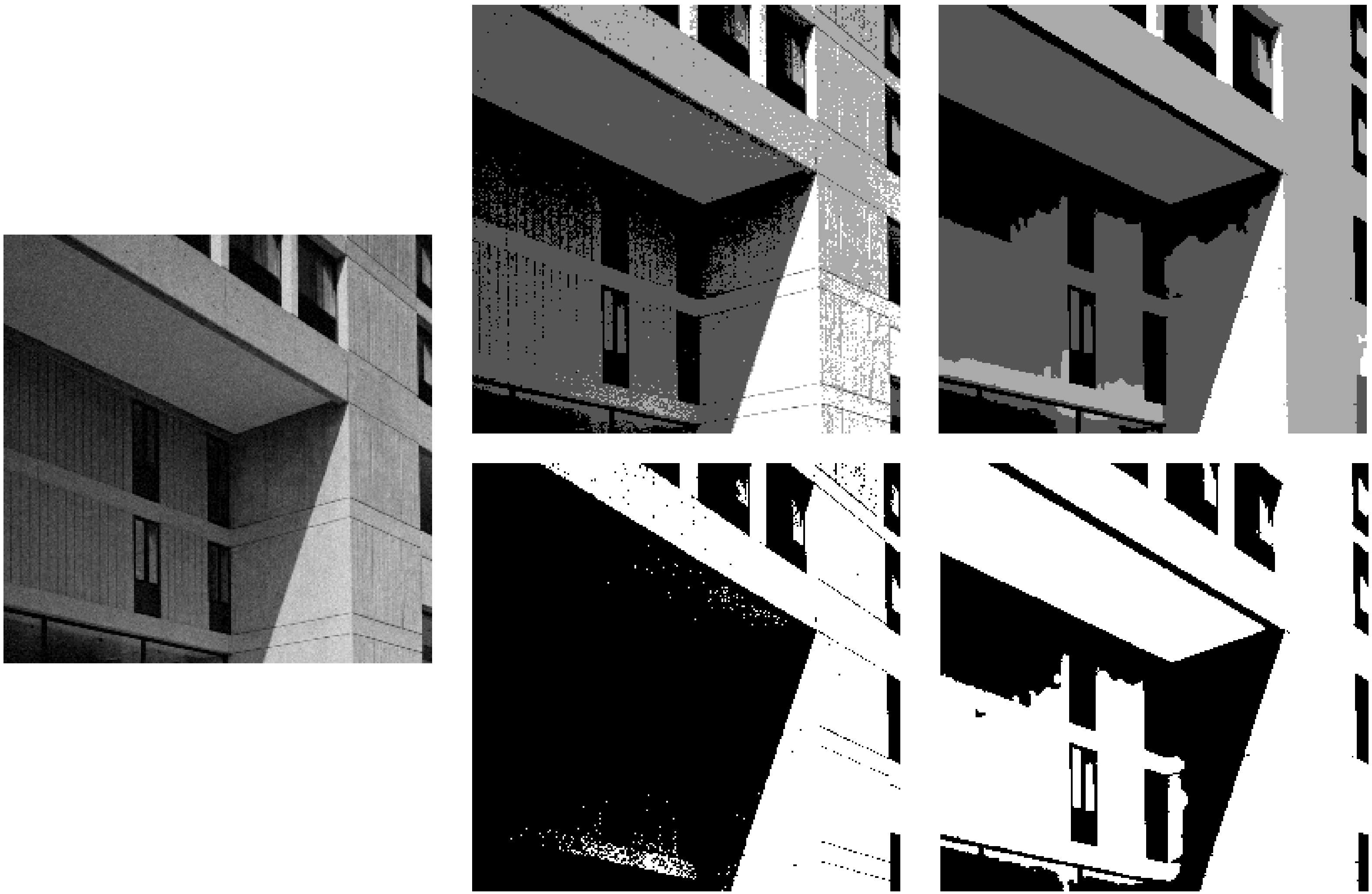


ACA: Region Estimation

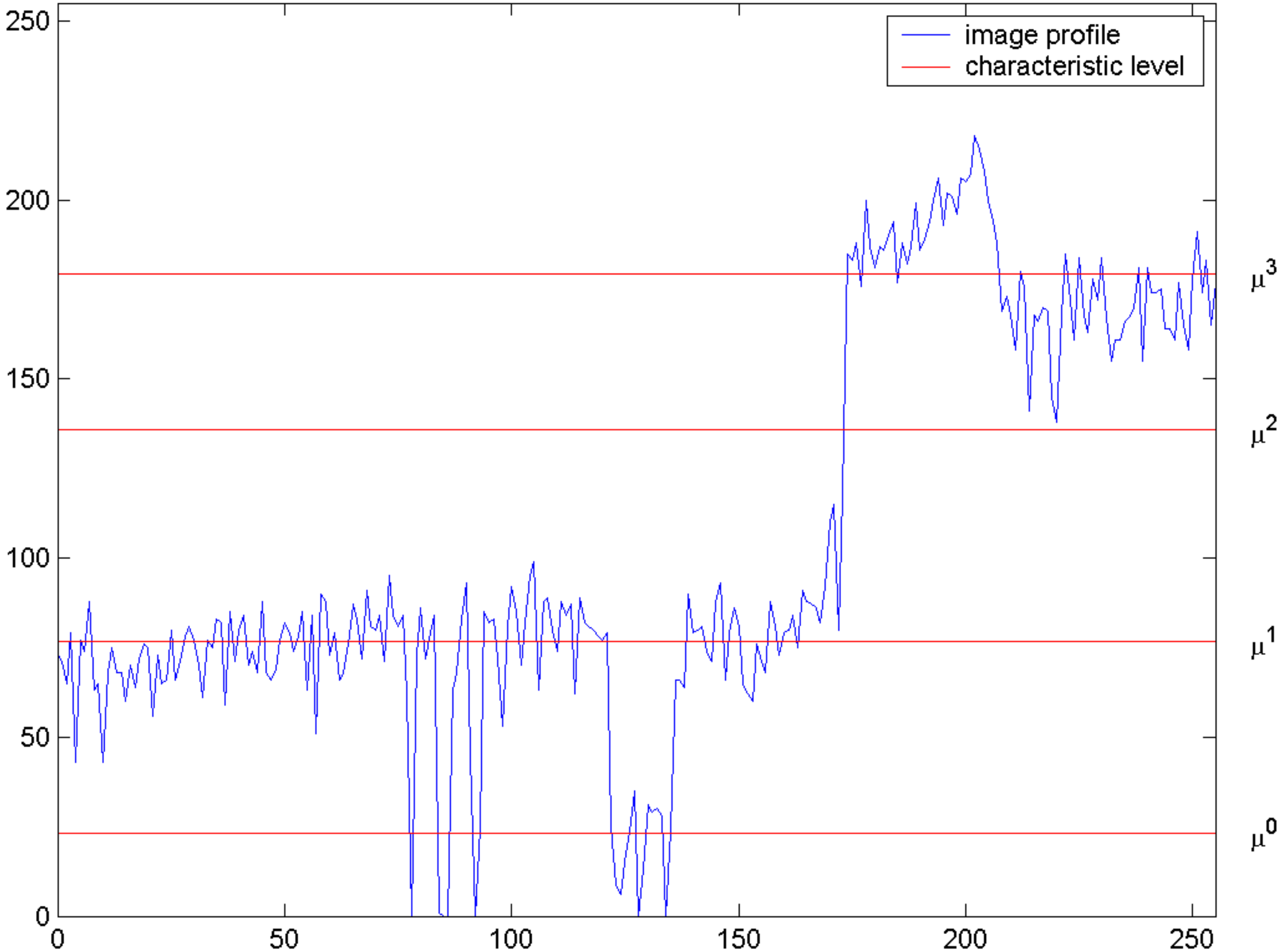
- Given $\mu_s^{x_s}, \forall x_s, s$
- Maximize $p(x \mid y)$ (too difficult)
- Maximize marginal densities (Iterated Conditional Modes)

$$p(x_s \mid y, x_q, \forall q \neq s) = p(x_s \mid y_s, x_q, q \in N_s)$$

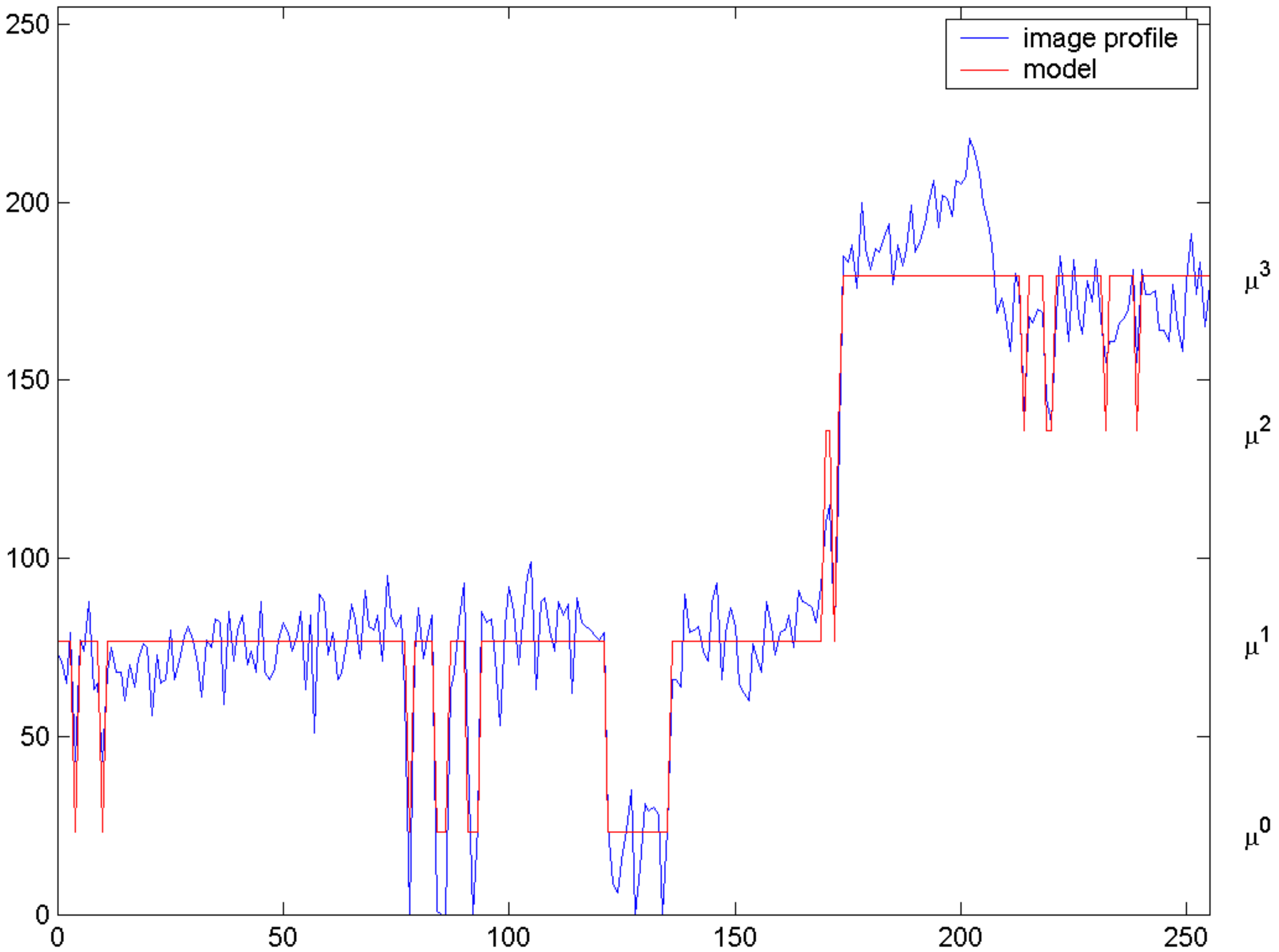
K-means vs. ACA



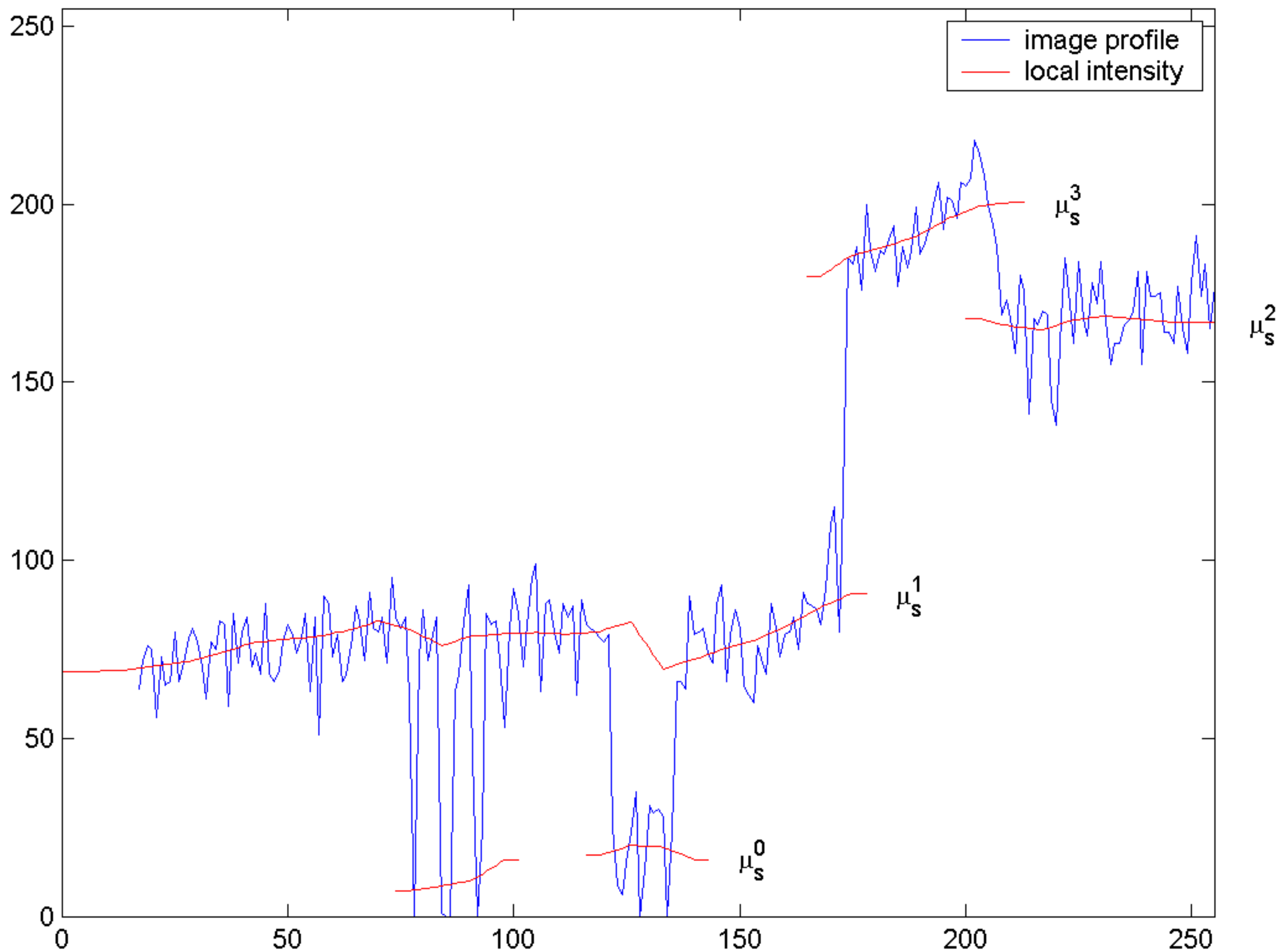
K-means Clustering



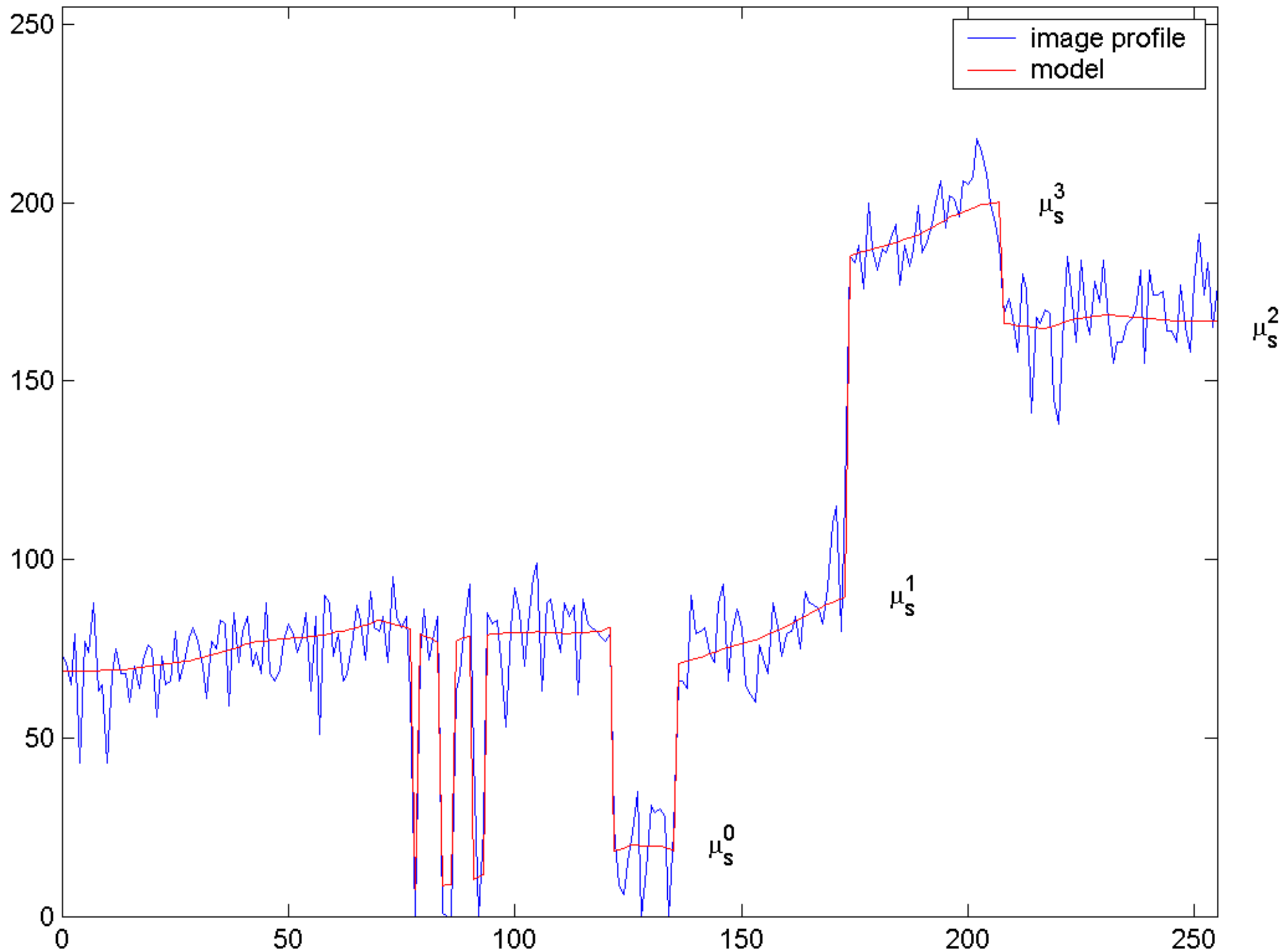
K-means Clustering



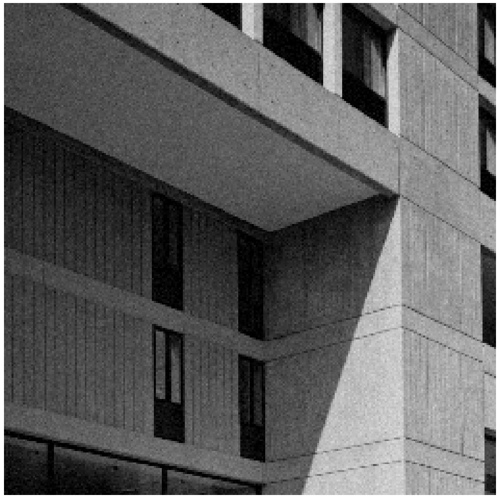
ACA: Local Intensity Functions (15x15)



ACA: Model (15x15)



Adaptive Clustering Algorithm

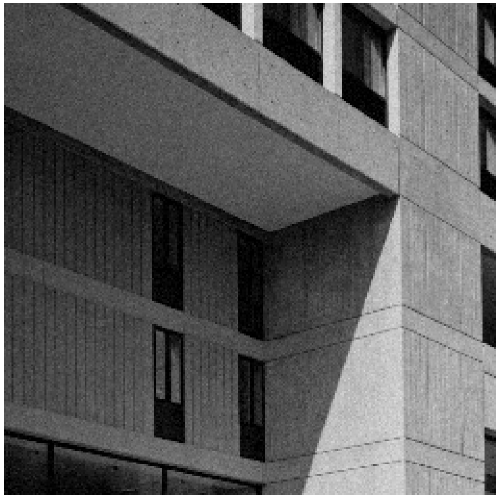


Original Image

ACA Class Labels

ACA Model (7x7)

Adaptive Clustering Algorithm

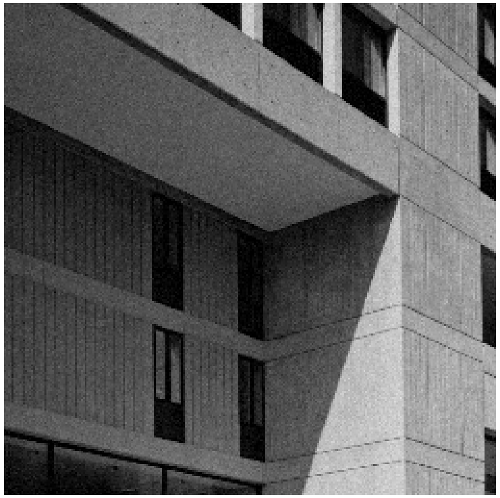


Original Image

ACA Class Labels

ACA Model (15x15)

Adaptive Clustering Algorithm



Original Image

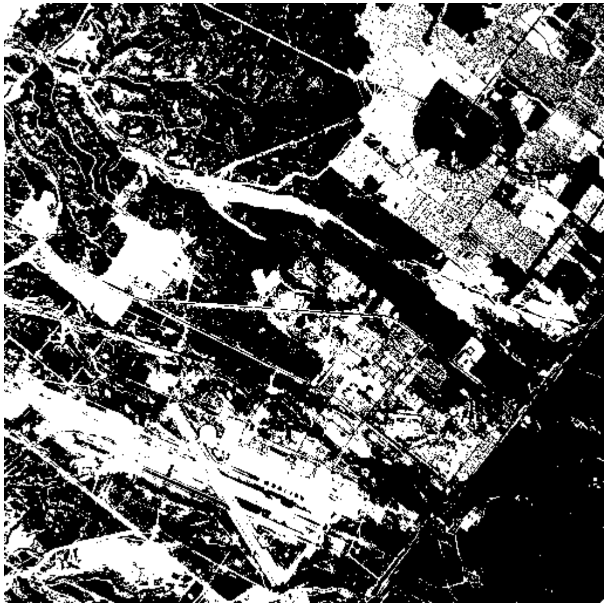
ACA Class Labels

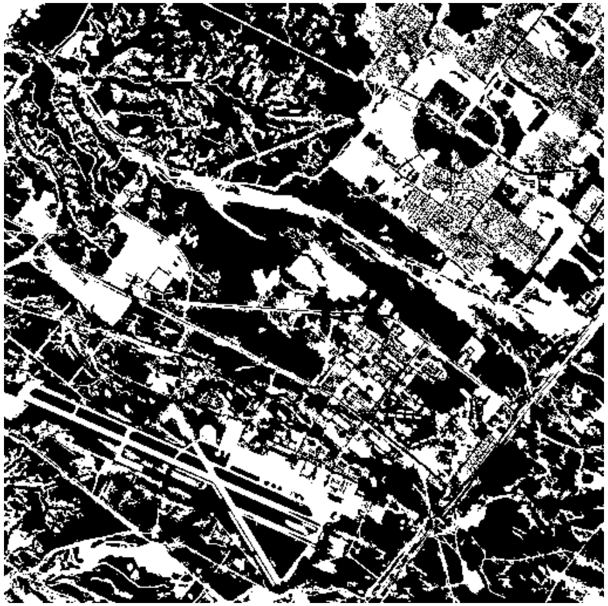
ACA Model (31x31)

Image Restoration Models

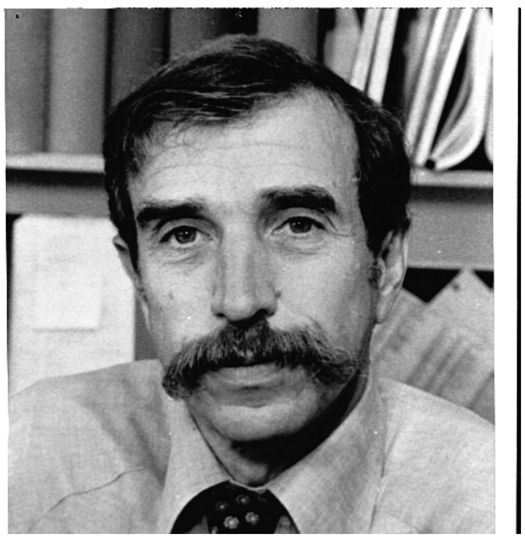
- Simple space varying image model [Kuan et al.` 85]
  - Space-varying mean + white Gaussian noise
- Spatially-adaptive LMMSE estimator
  - Use local sample mean and local sample variance
- No explicit model for region boundaries
  - Computes sample mean/variance across boundaries

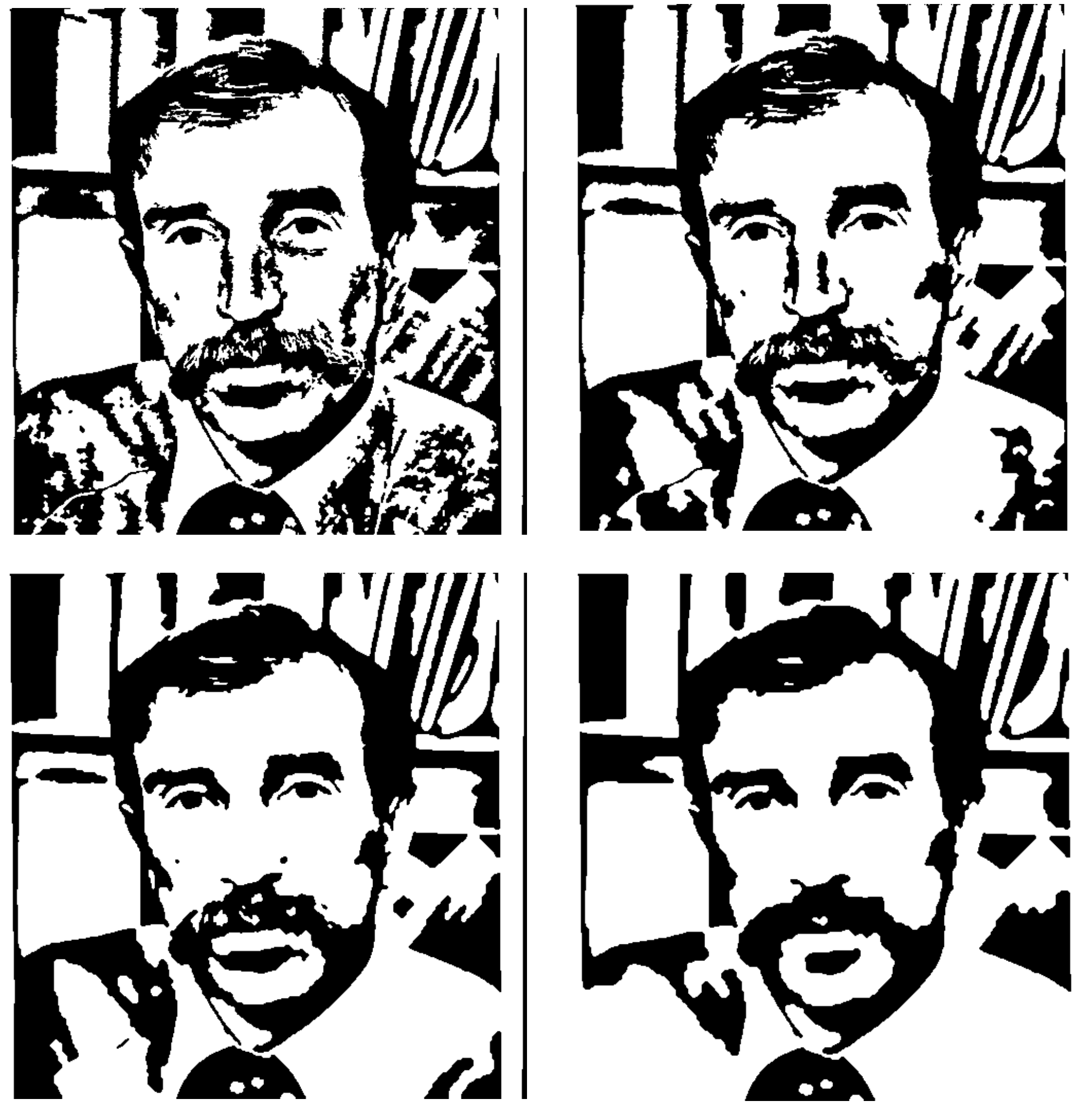
K-means vs. ACA





ACA





Adaptive Perceptual Color-Texture Segmentation

Natural Textures

- Combine color composition, spatial characteristics
- Non-uniform statistical characteristics (lighting, perspective)
- Perceptually uniform
- Need spatially adaptive features
- Small number of parameters

Texture Synthesis [Portilla-Simoncelli'00]

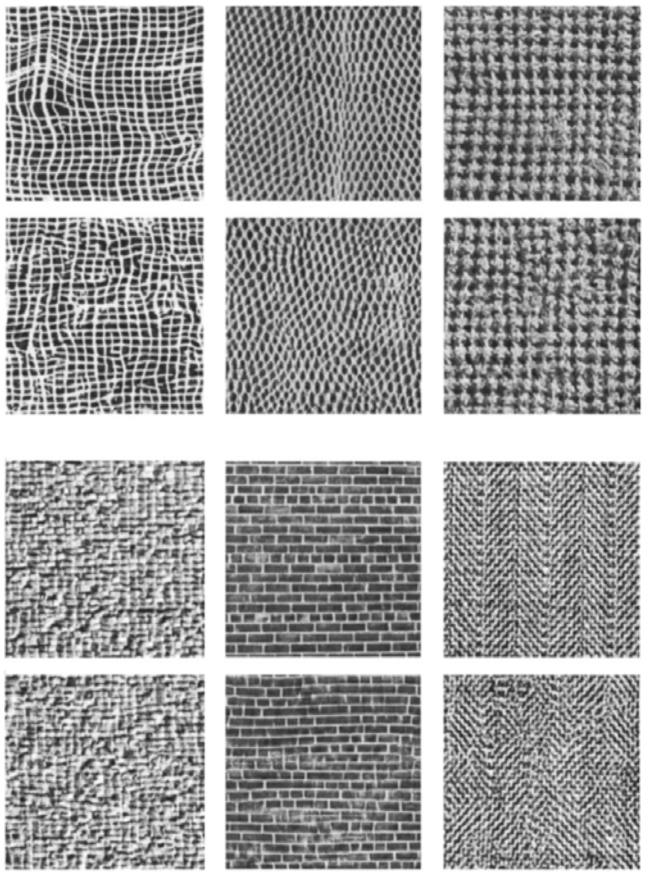


Figure 14. Synthesis results on photographic pseudo-periodic textures. See caption of Fig. 12.

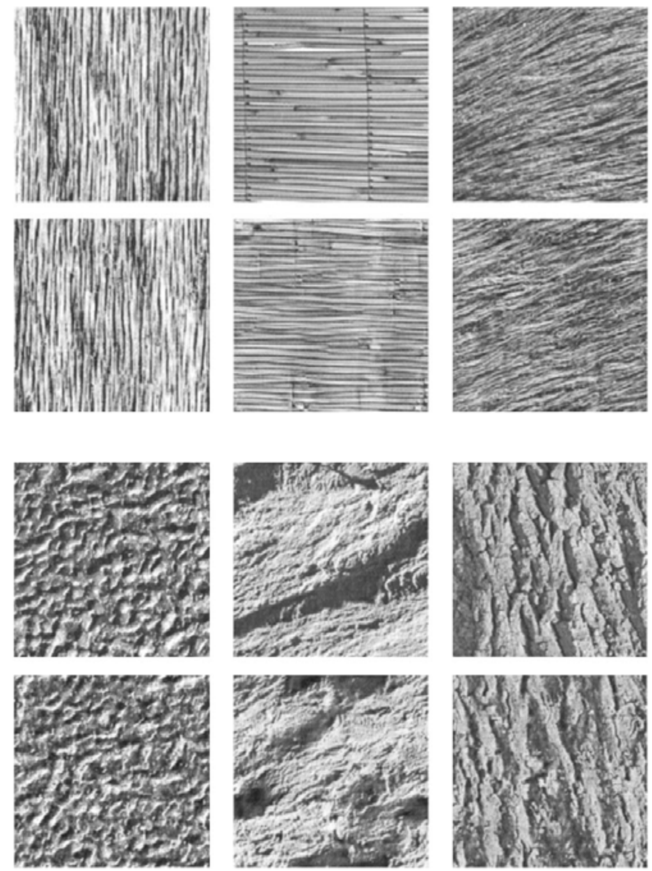
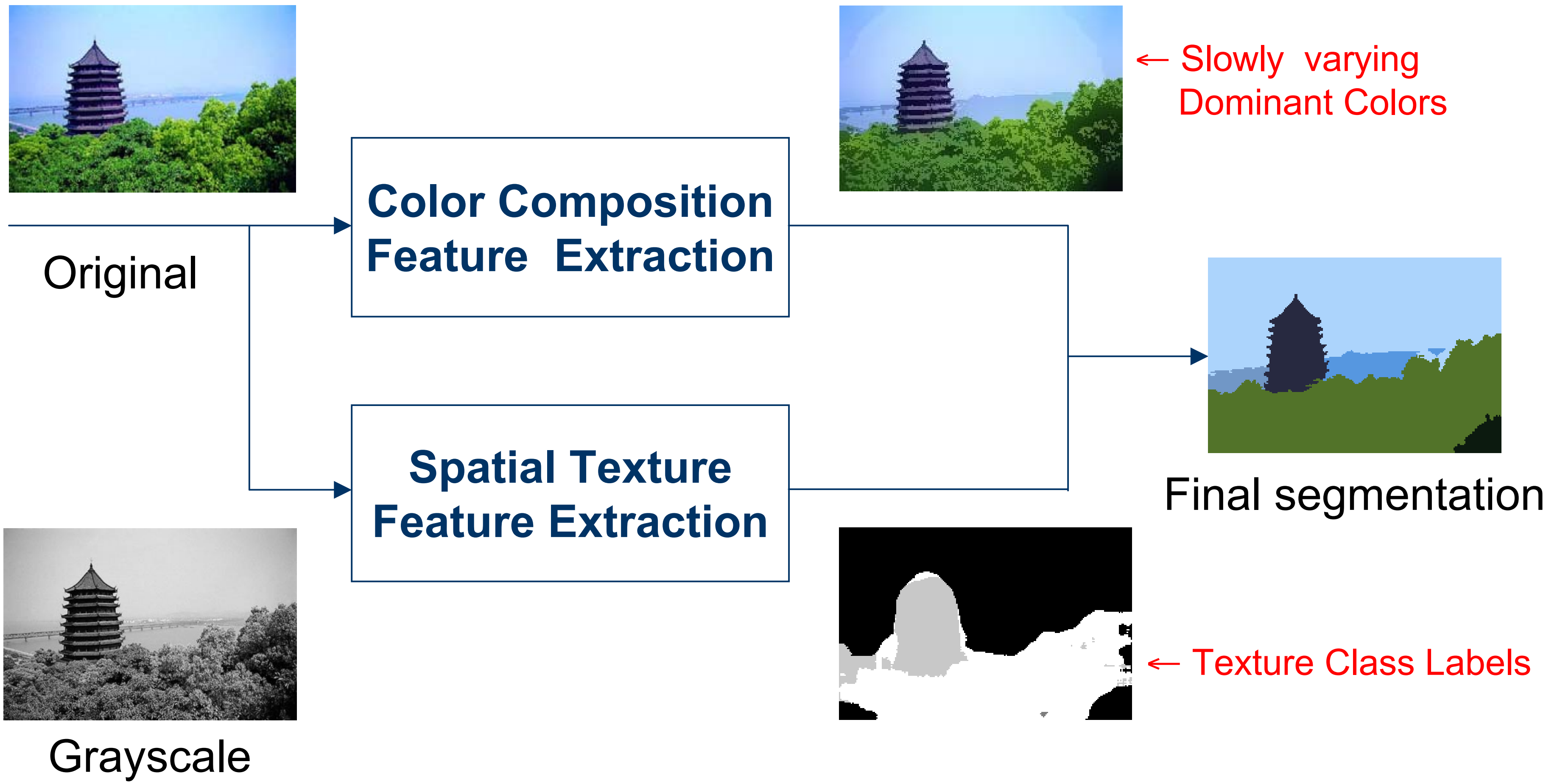


Figure 15. Synthesis results on photographic aperiodic textures. See caption of Fig. 12.

Adaptive Perceptual Color-Texture Segmentation



Dominant Colors

- Human eye cannot simultaneously perceive a large number of colors
  - Even though, under appropriate adaptation, it can distinguish more than 2M colors
- Small set of color categories
  - Efficient representation
  - Easier to capture invariant properties of object appearance
- Color categories are related statistical structure of perceived environment
  - K-means clustering to compute color categories [Yendrikovskij'00]

Spatially Adaptive Dominant Colors

- Dominant colors [Ma'97, Mojsilovic'00]
  - For class of images
  - For a given image
- Current approaches to extract dominant colors:
  - K-means (VQ) [LBG'80];
  - Mean-shift [Comaniciu-Meer'97];

  Assumption: constant dominant colors
- Proposed approach:
  - Spatially adaptive dominant colors
  - Use ACA

Comparison with Mean-Shift

Original Image

ACA

← 4 colors

under-segmentation

over-segmentation

quantization

Color Composition Feature

- Constant Dominant Colors:

$$f_c = \{(c_i, p_i), i = 0, \ldots, n, p_i \in [0,1]\}$$

c_i: color

p_i: percentage

- Spatially Adaptive Dominant Colors:

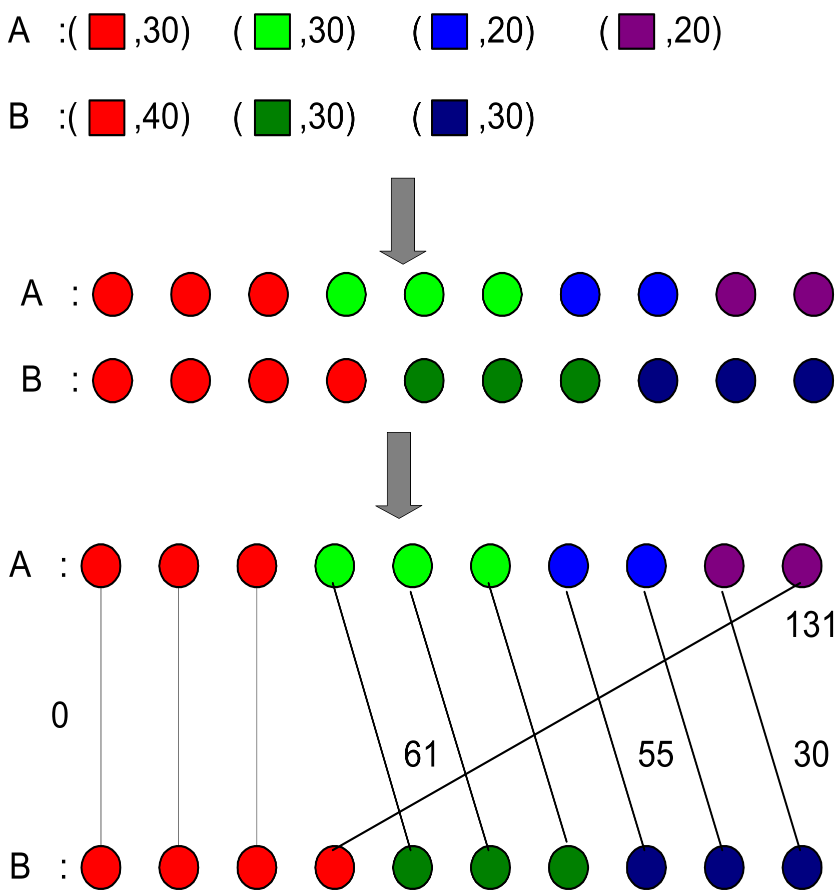
$$f_c(s, N_s) = \{(c_i, p_i), i = 0, \ldots, n, p_i \in [0,1]\}$$

- ACA adapts to local characteristics.
- Dominant colors relatively constant in small neighborhood: Can approximate with intensity at center of window.

Color Feature Similarity Metric

- Optimal Color Composition Distance (OCCD) [Mojsilovic'00]
  - Quantize color component based on percentage
  - Find best color correspondence
  - Then compute distance as sum of distances between matched colors (in a given colorspace)

Illustration of OCCD computation



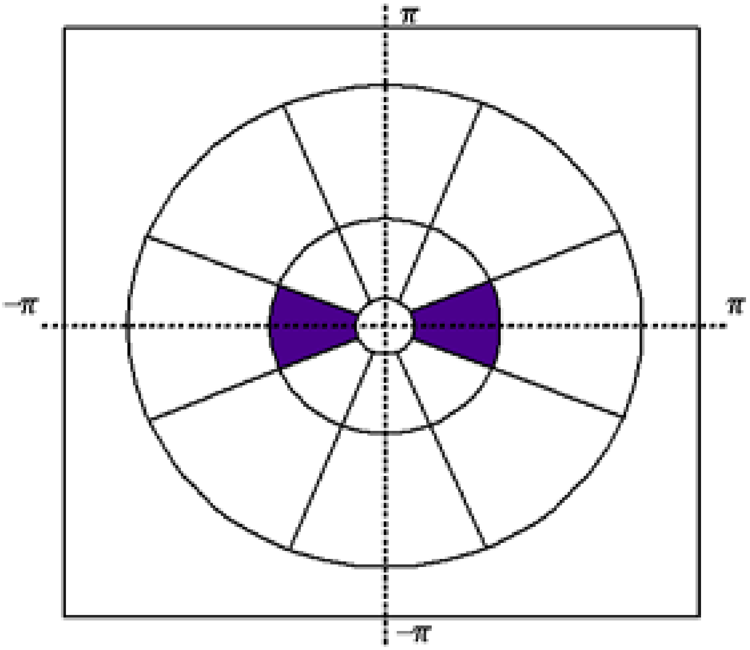
OCCD dist = 61*.3+55*.2+30*.1+131*.1=45.4

- Color Quantization unit p = 10
- Weight of the link is C_{max}-cost (color distance in Lab color space, C_{max} =376)
- Solve **maximum** graph matching problem using Gabow's algorithm.
- Apply color metric to resulting graph.

Spatial Texture Features

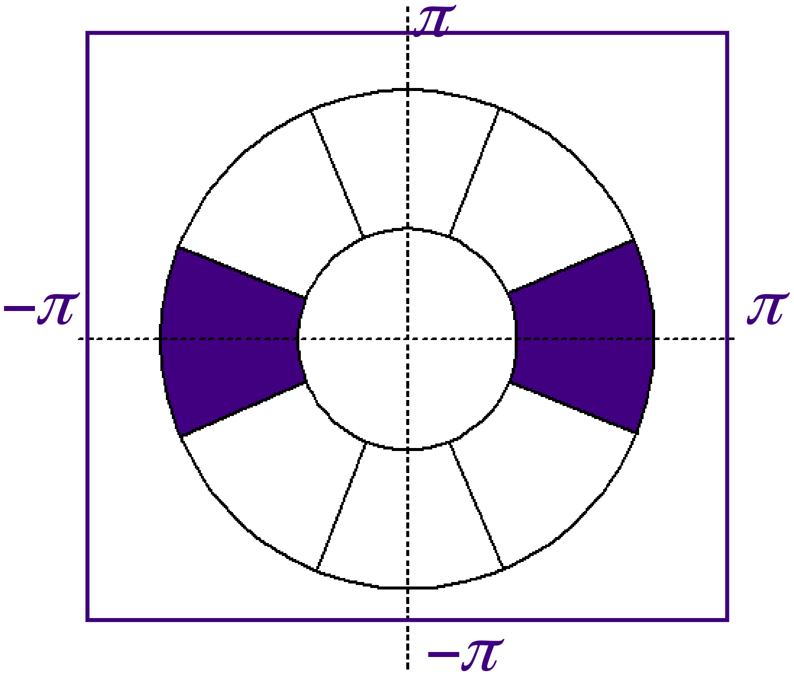
- Grayscale image component
  (vs. achromatic pattern map)
- Multiscale frequency decomposition
  – DWT (9/7 Daubechies)
  – Steerable filters [Freeman-Adelson'91]
  – Gabor filters [Daugman'86]
- Energy of subband coefficients is sparse
  – Use local median energy

Steerable Pyramid Decomposition



Ideal spectrum

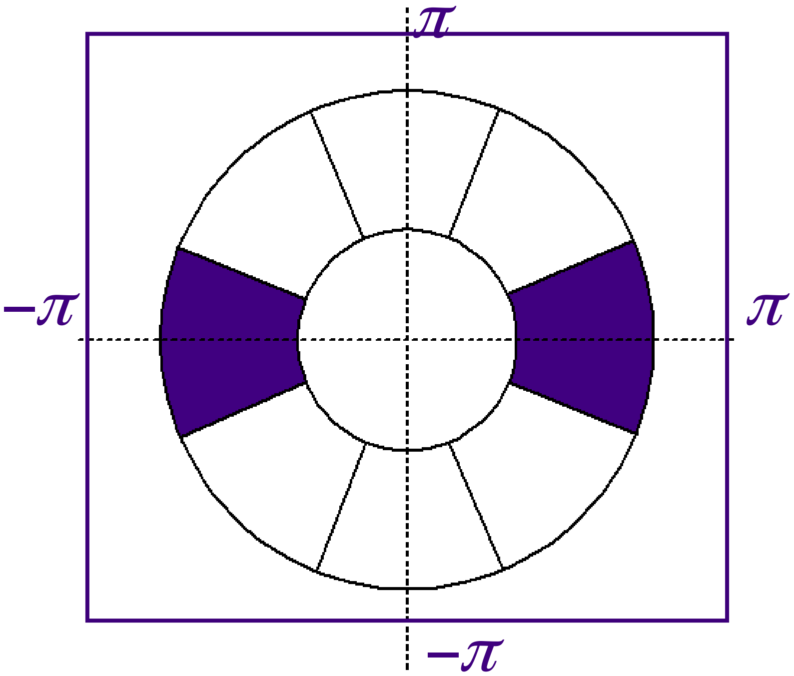
2-level decomposition



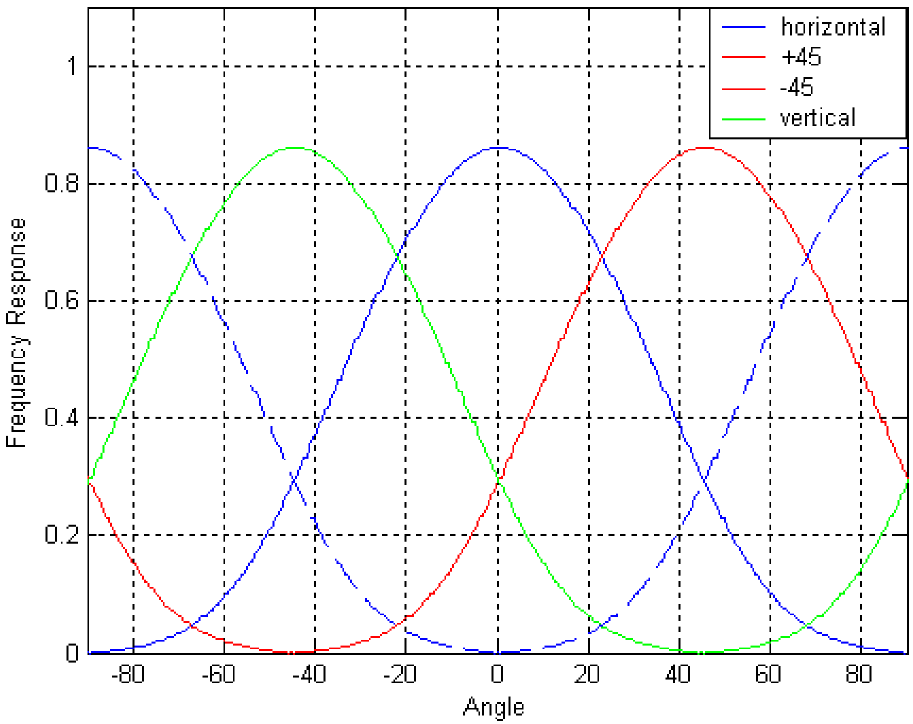
Ideal spectrum

1-level decomposition

Steerable Pyramid Decomposition



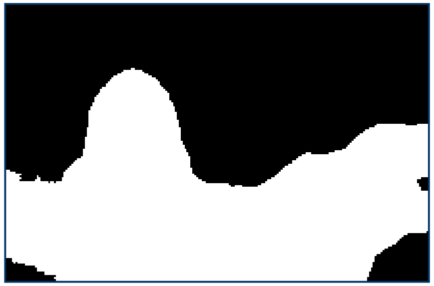
Ideal spectrum



Actual spectrum

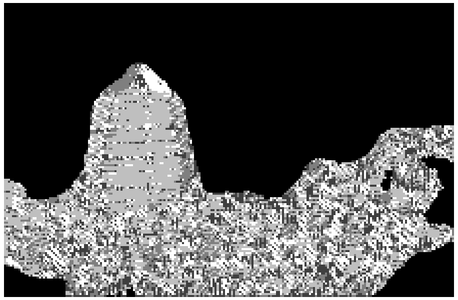
Smooth vs. Non-smooth Classification

- For each pixel:
  - S_{max} = Maximum of 4 subband responses
  - S_i = Index of maximum coefficients
  - Local median energy extraction on S_{max}
  - 2-level K-means on local median
    (Check validity of smooth/non-smooth cluster)
  - Use threshold provided by subjective test

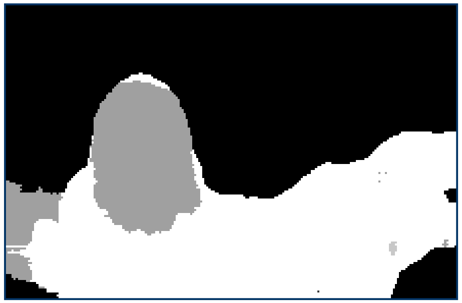


Classification of Non-smooth Regions

- Construct local histogram of $\boldsymbol{S_i}$
- "Complex": no dominant orientation, i.e., no index dominates (1st and 2nd maximum of histogram are close, or maximum is not large enough)
- Otherwise classify according to dominant orientation (max index) as "horizontal," "vertical," "+45," "-45."
- Can be used with any multiscale frequency decomposition



Max Indices $\boldsymbol{S_i}$



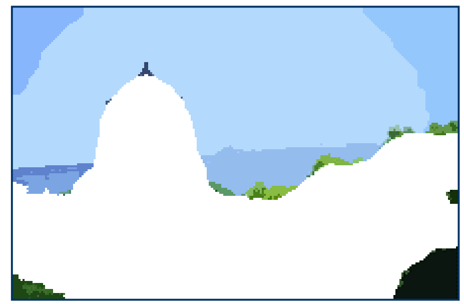
Texture classes

Multi-scale Texture Classification

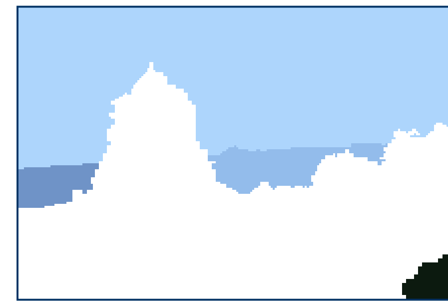
- Apply texture classification at each scale
- Combine texture classes from different scales based on the following rules:
  - "smooth": "smooth" at all scales
  - "Vertical," "Horizontal," "+45^o," "-45^o": consistent texture classification across all scales. Note: "complex" or "smooth" is consistent with any single direction
  - "complex": none of above satisfied

Image Segmentation

- "Smooth" regions:
  - Based on ACA
  - Merge based on color difference along border of each region pair
  - Small border regions merged with non-smooth

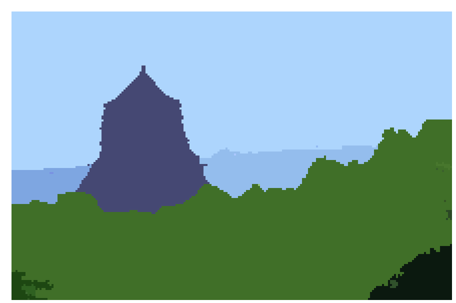


Before Merge



After Merge

- "Texture" regions:
  - Initial segmentation by region growing
  - Iterative border refinement



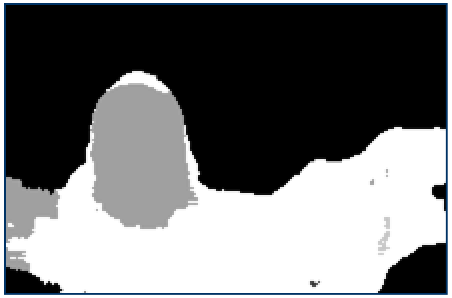
Crude segmentation

Final segmentation

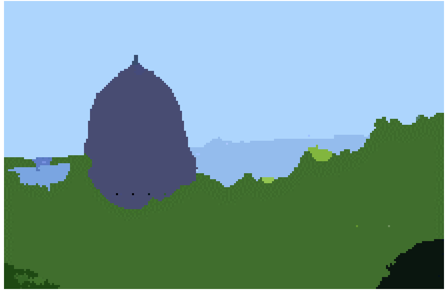
Initial Segmentation by Region Growing

- Starting from any pixel in the textured regions, grow by adding nearby pixels with similar color features (in the OCCD sense).
- Use higher threshold if pixels belong to same texture class; lower threshold if pixels belong to different texture classes
- Hierarchical grid approach
- Paint the resulting segment with average color of that region.

ACA image



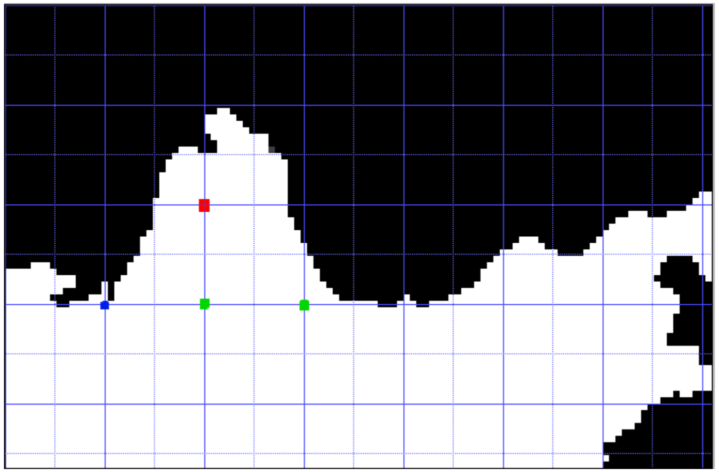
Texture classes



Crude segmentation

Hierarchical Grid Approach

- Do initial region growing on coarse grid using OCCD
- Reduce grid spacing (half)
- Find OCCD to the classified neighbors. If close to none, create new texture class.
- Add simple spatial constraints (MRF-type) to OCCD distance
- Repeat until all pixels are classified.
- Faster without loss of accuracy



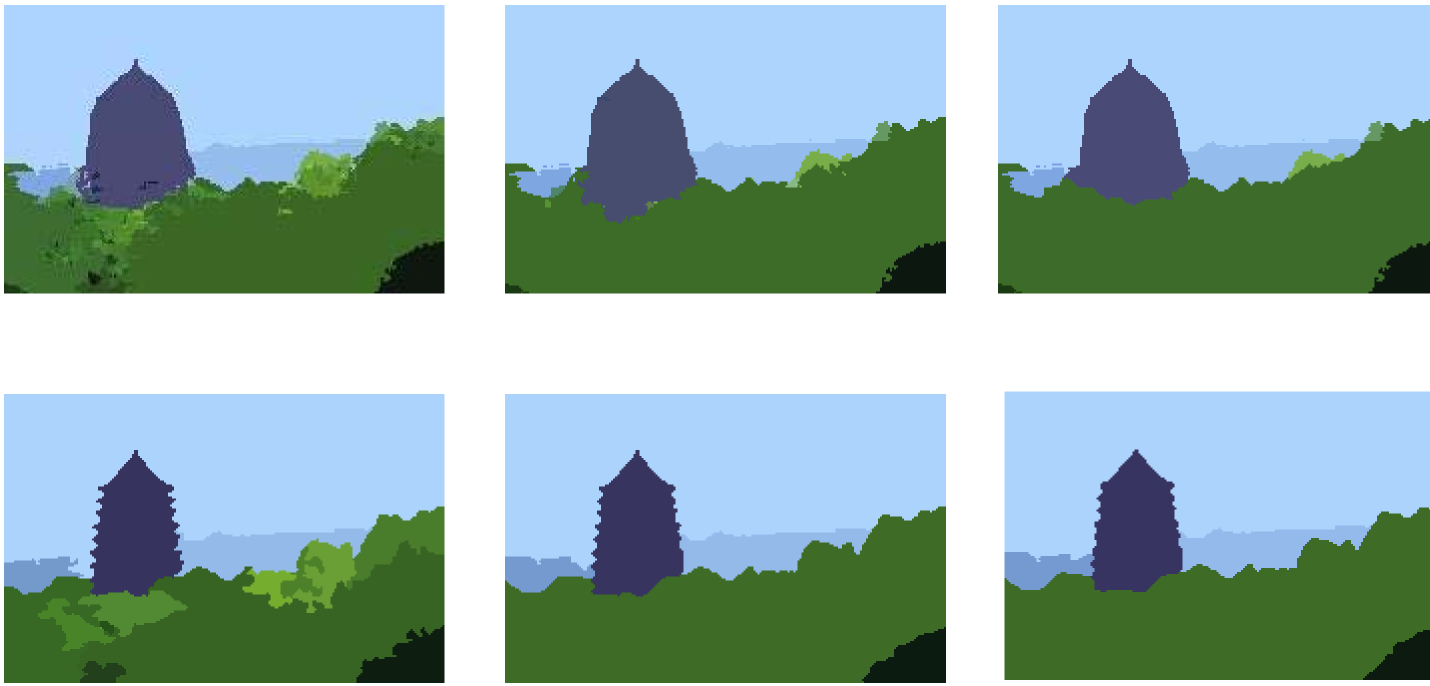
Black: non-texture region

White: textured region

Why MRF Constraints Are Necessary

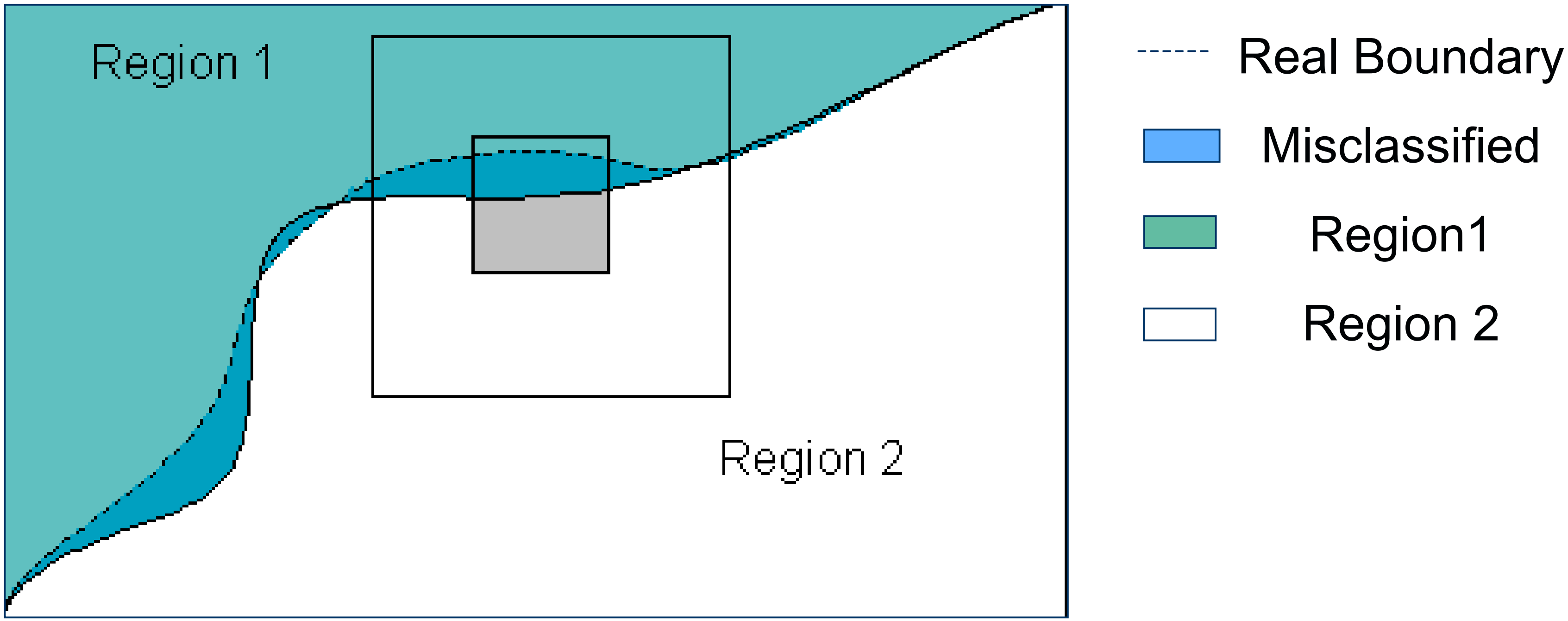
Crude:

Final:



$\beta=0$ $\beta=0.5$ $\beta=1.0$

Iterative Border Refinement



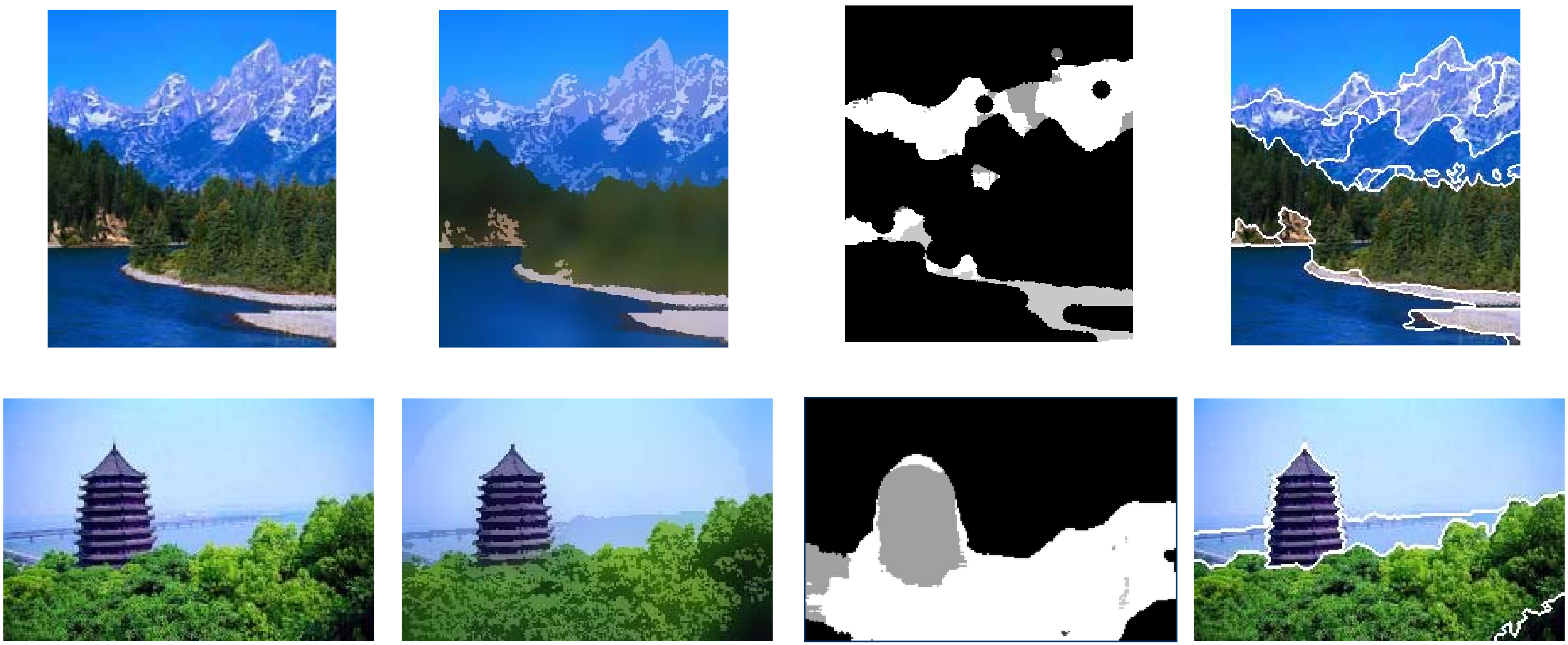
Color features in inner window represent local features

Color features in outer window represent region-wide characteristics

Window pairs used: {35/11, 21/9, 11/5, 11/3}

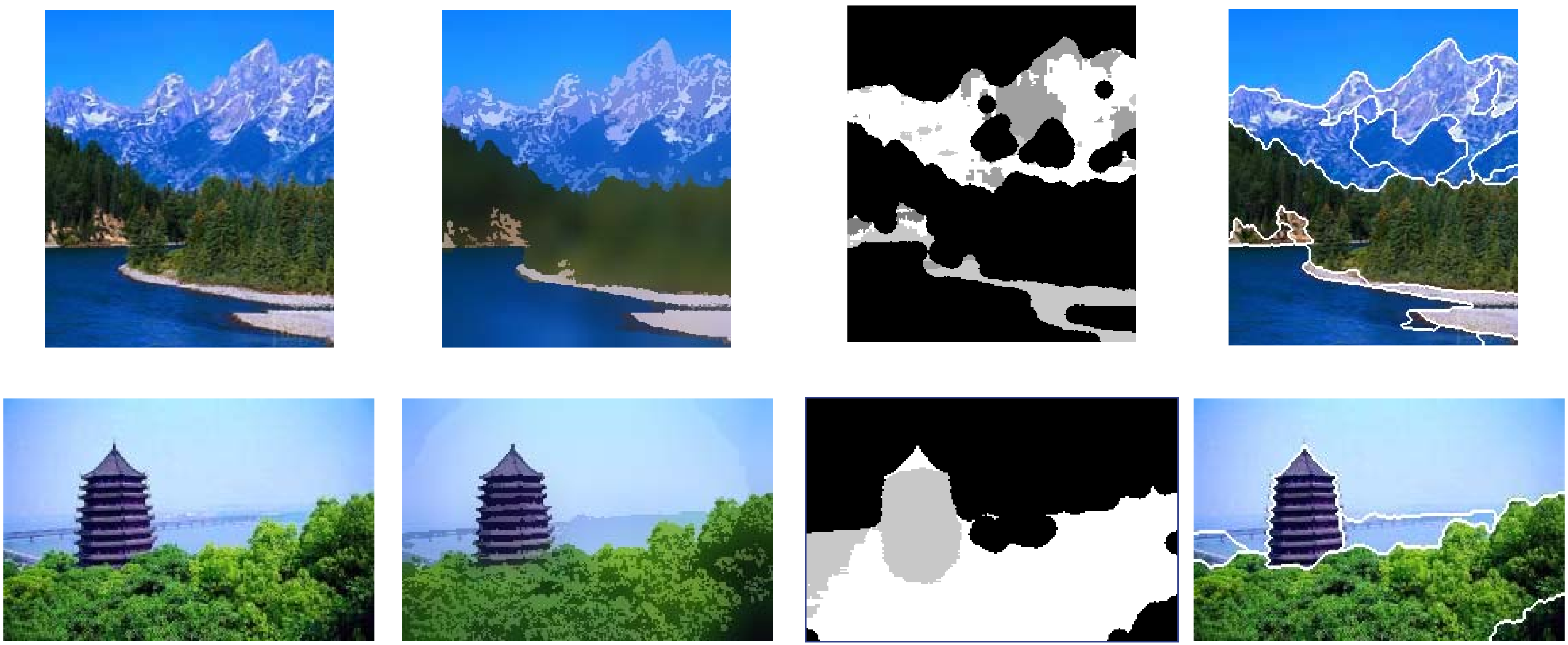
Results with steerable filters
without Perceptual Tuning

Original | ACA | Texture Classes | Segmentation



Results with steerable filters
with Perceptual Tuning

Original | ACA | Texture Classes | Segmentation

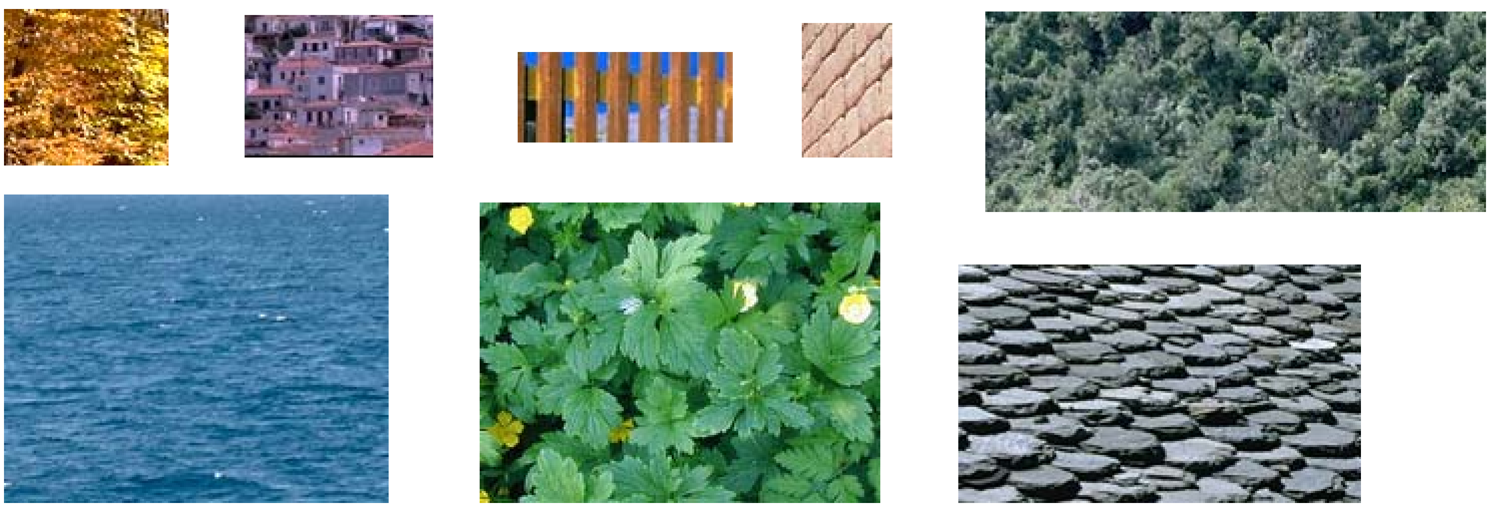


Perceptual Tuning

- Smooth vs. non-smooth classification
- Thresholds for Dominant Orientation
  - Horizontal, vertical, +45, -45, complex classification
- Threshold for color feature similarity
- Texture window size
  - Varies with scale

Texture Discrimination Test*

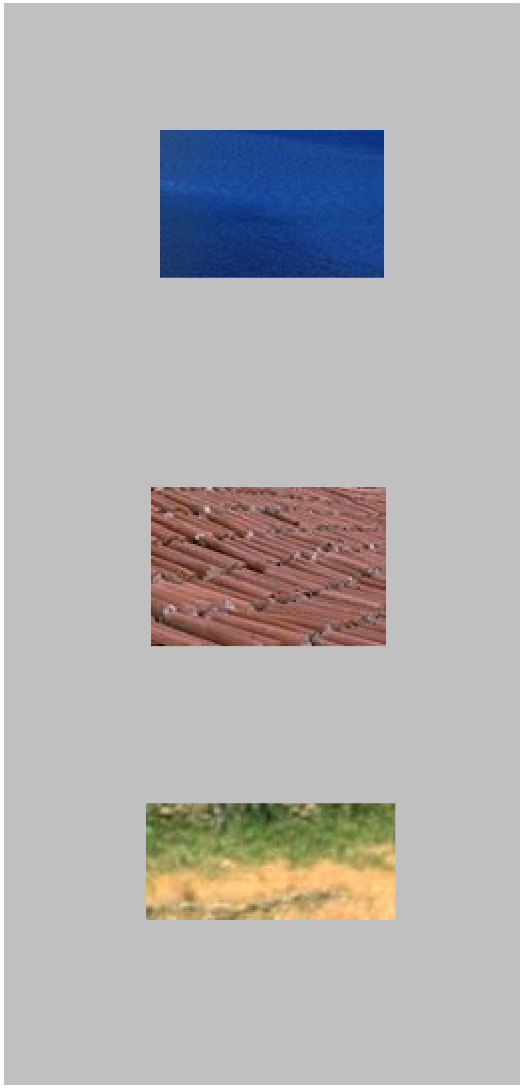
- Setup:
  - Viewing distance: about 2 feet;
  - Subjects with normal vision (corrected), normal color vision
  - 37 texture images from photo CD at 4-5 scales



* http://www.ece.northwestern.edu/~pappas/research/texture_perception_test/

Test I: Texture Classification

- Classify image into:
  - **SMOOTH:** Uniform or slowly varying image intensity; no objects or sharp boundaries present.
  - **TEXTURE:** Approximately uniform texture patterns; may be slowly varying (further classification into horizontal, vertical, +45, -45, complex categories)
  - **OTHER:** None of the above, e.g., non-uniform texture, multiple regions, multiple objects



Test II: Texture Similarity

- Similarity scores:
  - 0: dissimilar
  - 1: somewhat similar
  - 2: similar
  - 3: same texture

Segmentation Results










Segmentation Evaluation Metric

Human Segmentation Examples

- No “ground truth” for natural image segmentation
- The segmentations of different people are consistent.

Segmentation Evaluation Metric
[Martin'01]

- Quantify the consistency between segmentations of different granularities; allow mutual refinements
- Local error measure (asymmetric):

$$E(S_1, S_2, p_i) = \frac{\left| R(S_1, p_i) \setminus R(S_2, p_i) \right|}{\left| R(S_1, p_i) \right|}$$

- Local Consistency Error (LCE):

$$LCE(S_1, S_2) = \frac{1}{n} \sum_i \min \left\{ E(S_1, S_2, p_i), E(S_2, S_1, p_i) \right\}$$

- Global Consistency Error(GCE):

$$GCE(S_1, S_2) = \frac{1}{n} \min \left\{ \sum_i E(S_1, S_2, p_i), \sum_i E(S_2, S_1, p_i) \right\}$$

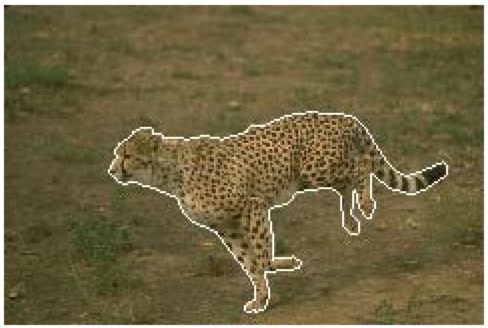
- GCE ≥ LCE

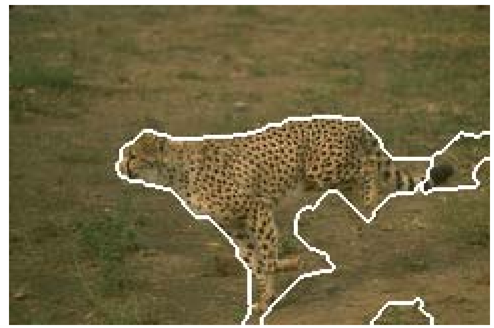
Comparison with JSEG Segmentation

Human Segmentation	Proposed Approach	JSEG (merge=0.4)
		
	GCE=0.04 LCE=0.02	GCE=0.33 LCE=0.28
		
	GCE=0.04 LCE=0.04	GCE=0.08 LCE=0.07

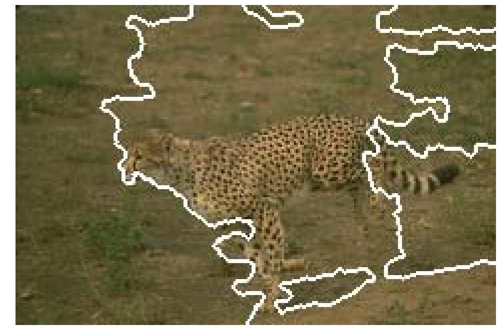
Comparison with JSEG Segmentation

Human Segmentation | **Proposed Approach** | **JSEG (merge=0.4)**





GCE=0.1 LCE=0.07



GCE=0.26 LCE=0.17

GCE=0.09 LCE=0.04

GCE=0.11 LCE=0.08

Segment Classification

Semantic Information Extraction at Segment Level

Segments as Medium Level Descriptors

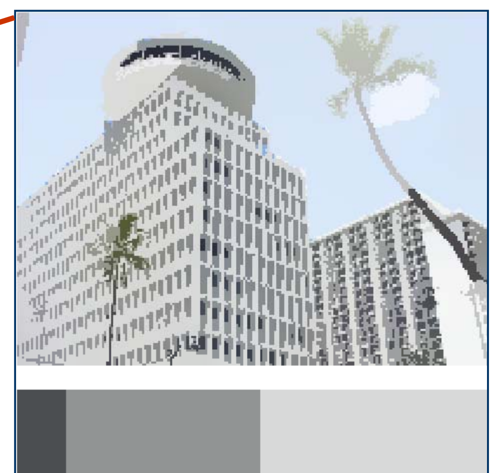
original

Dominant Colors (ACA)

segment 1

segment 2

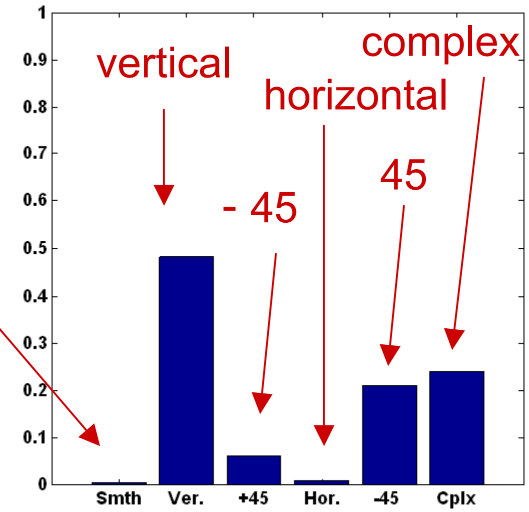
segment 3



Dominant Colors & Percentages

quantize

smooth



Spatial Texture

Plus:
- Location
- Shape
- Size

Color Naming Syntax

Hue primary	Hue secondary	Lightness	Saturation	Achromatic
red orange brown yellow green blue purple pink beige magenta olive	reddish brownish yellowish greenish bluish purplish pinkish	grayish moderate medium strong vivid	blackish very-dark dark medium light very-light whitish	black gray white

267 quantization points (NBS, Mojsilovic'02)

Eleven Colors That Are Almost Never Confused (Boynton'89)

Labels

Segment

Man Made

- Building
- Bridge
- Cityscape
- Car
- Boat
- Airplane
- Pavement
- Other Man Made

Natural

Vegetation

- Flower
- Grass
- Woods/Bushes
- Forest

Sky

- Day-sky
- Night-sky
- Sun
- Clouds
- Sunrise/Sunset

Landform

- Water
- Ground
- Mountain
- Snow

People

- Face
- Person
- Crowd

Animal

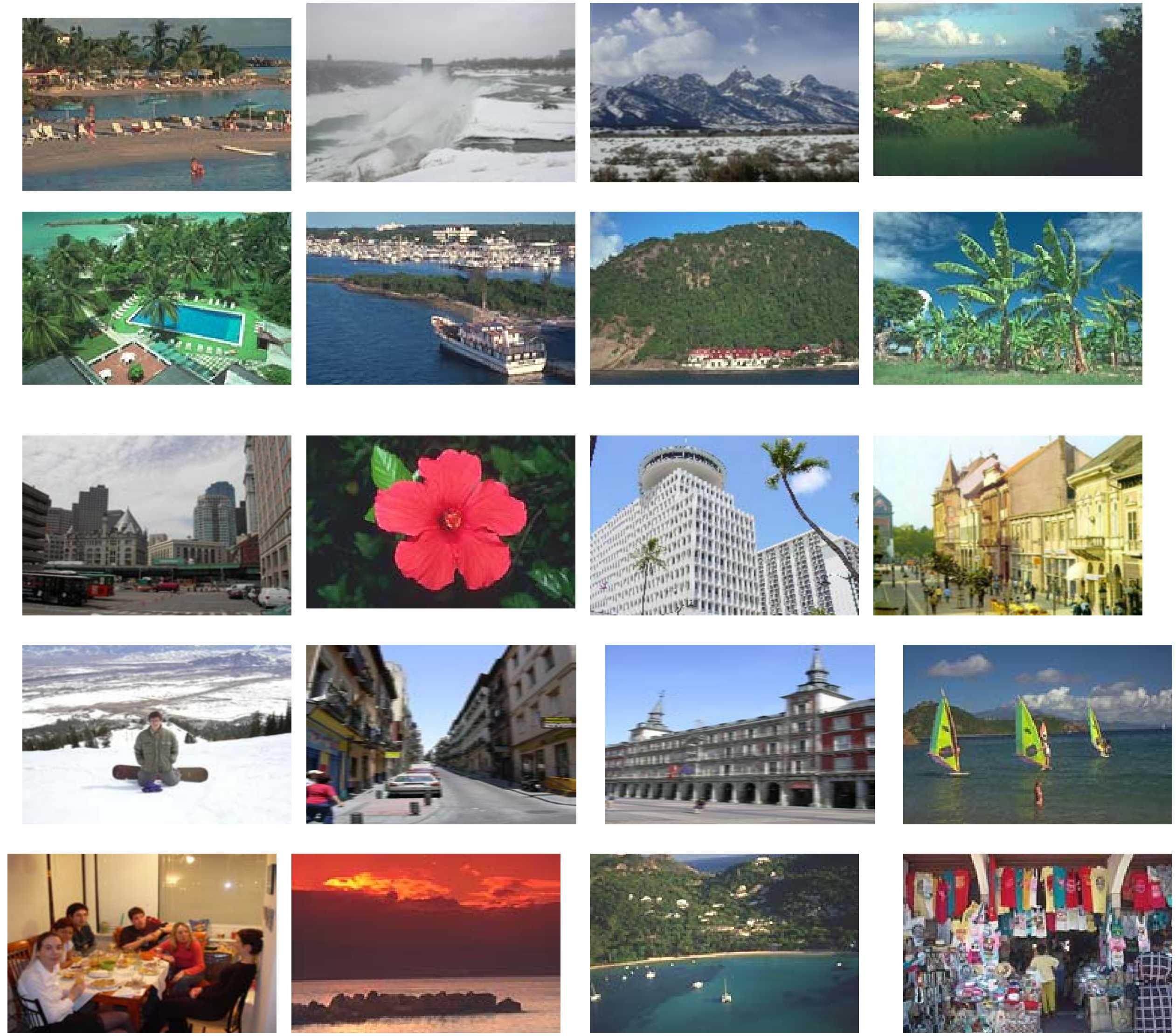
Scene

Indoor **Outdoor:** Street, skyline, beach, garden, night scene, day scene

Database

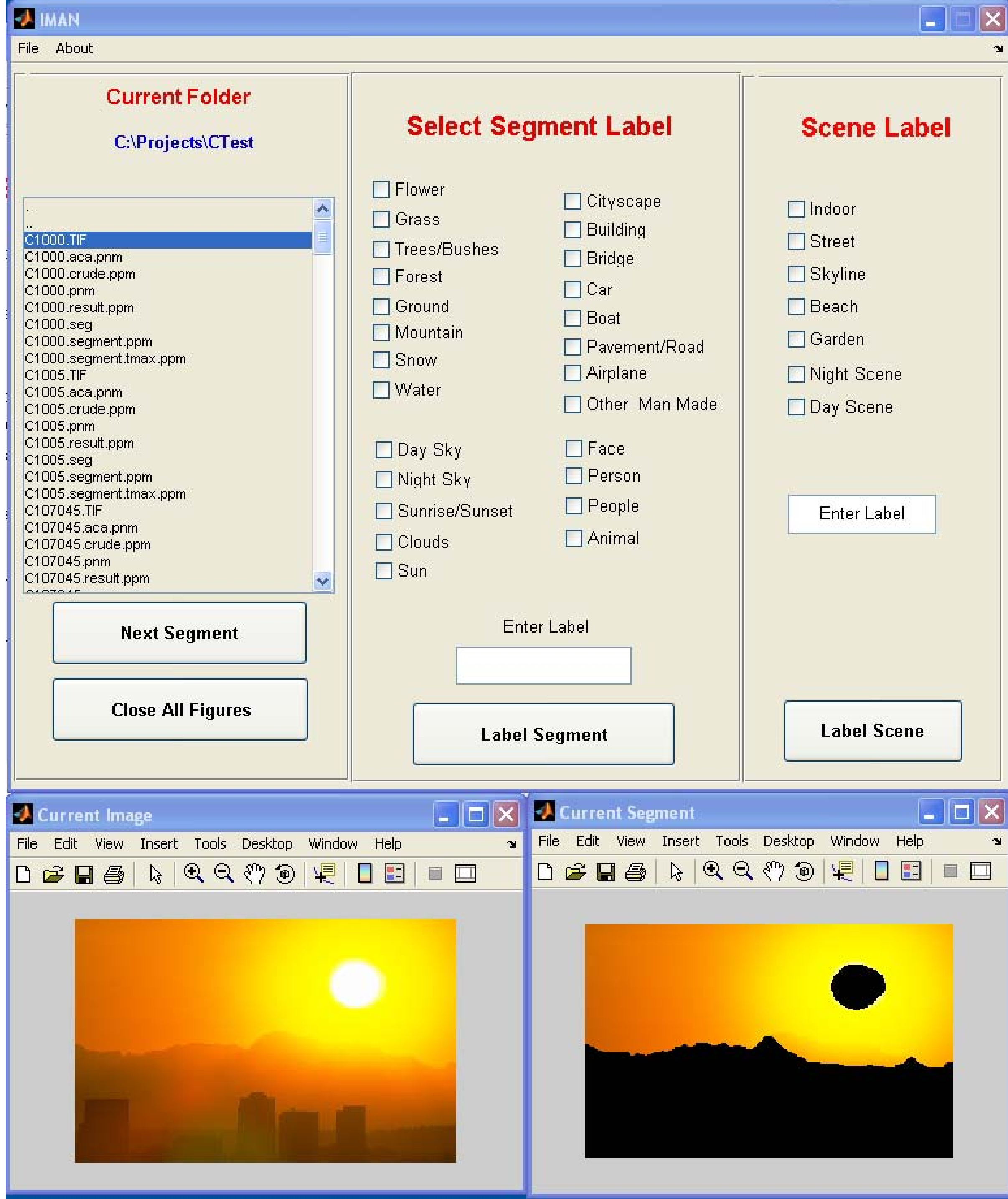
- Training
- Testing

- Corel:12,000
- Key Photos: 2,000
- Other: 600
- Corbis



Annotation Aide

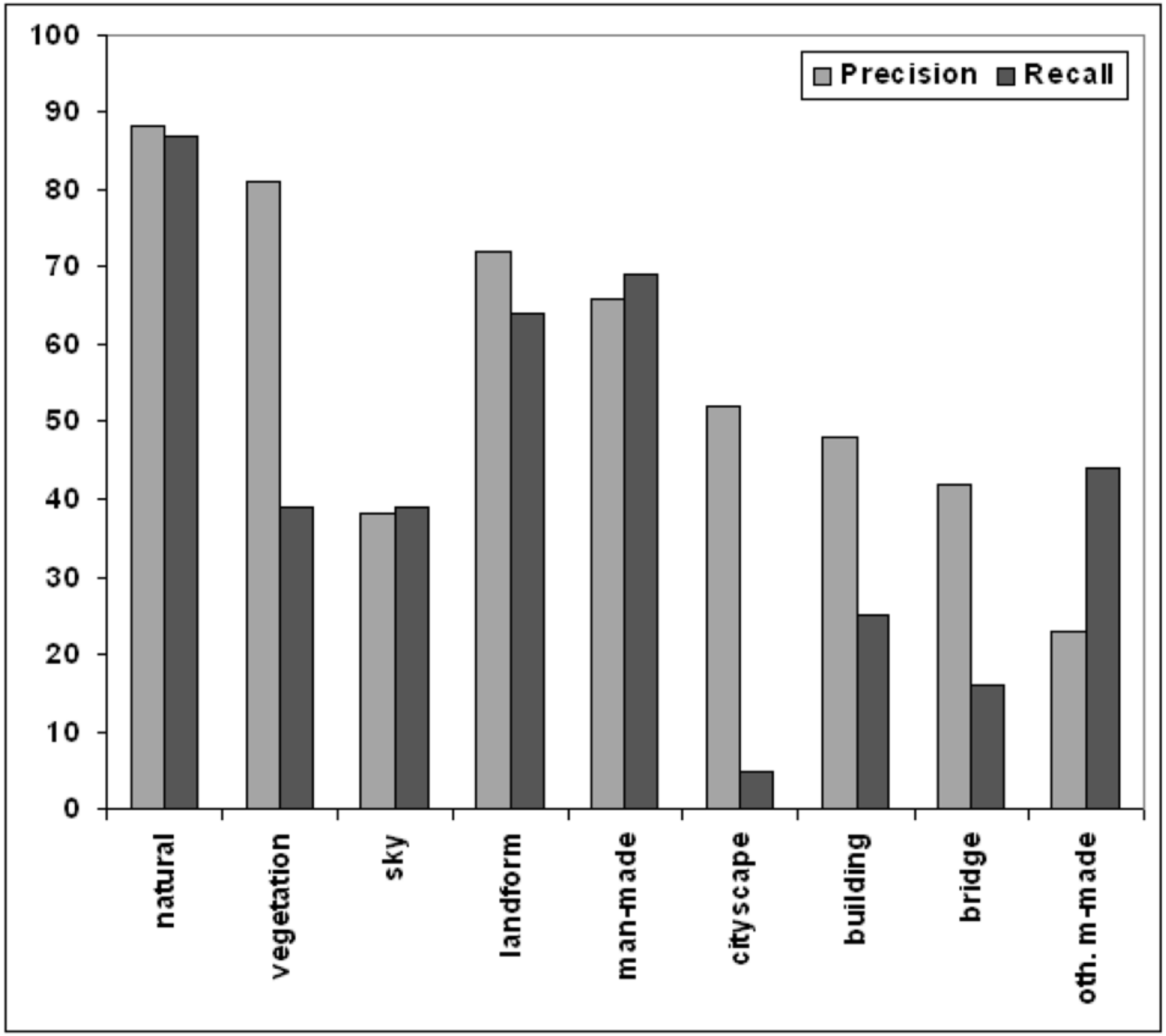
- XML output



Results

- 1600 photos
- No humans or animals
- 4000 manually labeled segments
- 80% training 20% testing
- Fisher Linear Discriminant method
- 14 colors, 6 textures

Results



- **Recall:** correctly labeled / total relevant segments
- **Precision:** correctly labeled / total assigned to label by algorithm