Beyond Mere Pixels:

How Can Computers Interpret and Compare Digital Images?

Nicholas R. Howe

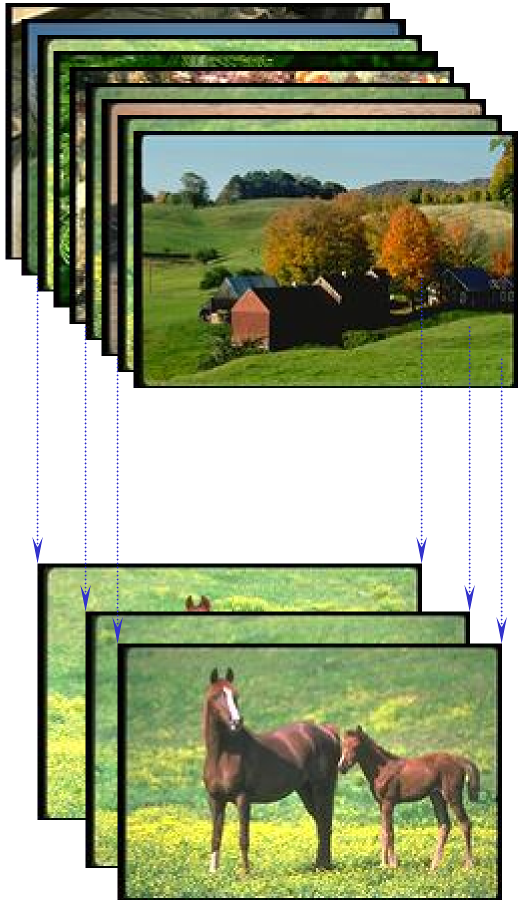
Cornell University

Why Image Retrieval?

- World Wide Web:
  - Millions of hosts
  - Billions of images
- Growth of video libraries
- Photography: going digital

Image Retrieval Framework

- Collection of diverse images
- User supplies a *query image*.



- System returns most similar images from collection.

Image Similarity & Retrieval

- Measuring the similarity between two images is a difficult test of image understanding.

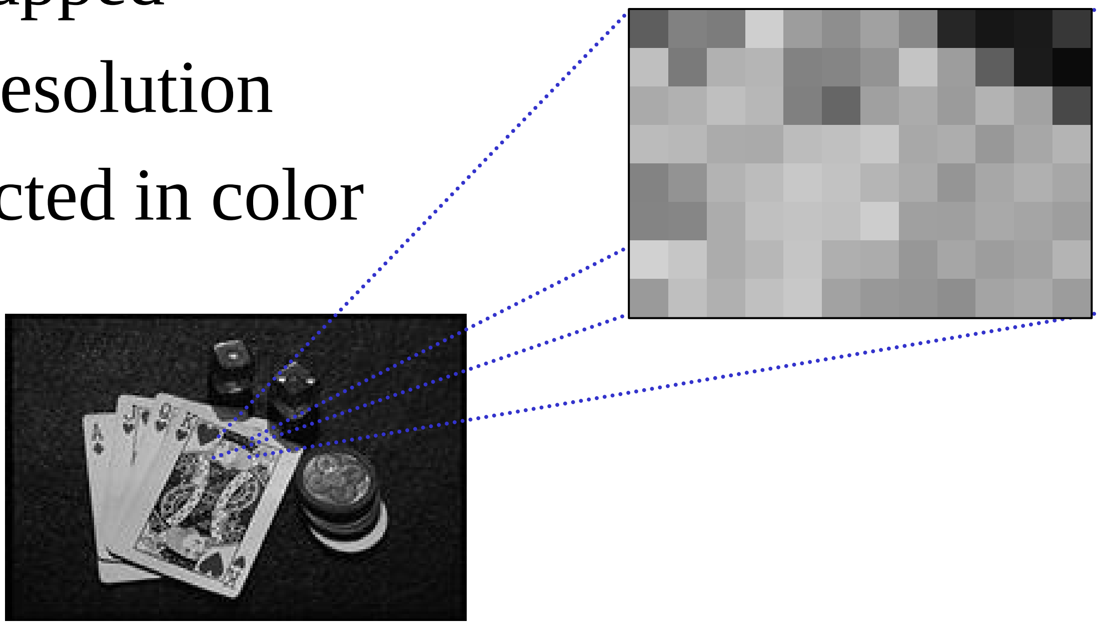
Q. Given a bunch of images, which are most similar to one I'm interested in?

Q. What's This?

94	129	124	207	157	142	161	136	38	22	26	55
191	122	177	181	130	133	147	196	157	94	27	11
170	177	191	183	128	102	160	171	155	179	162	72
187	184	171	170	188	192	200	168	173	152	167	180
131	147	179	188	200	194	181	171	149	167	176	167
132	134	172	192	195	192	205	160	159	169	165	158
209	198	172	183	197	175	172	151	166	157	162	180
154	191	176	192	200	162	152	149	142	164	169	156

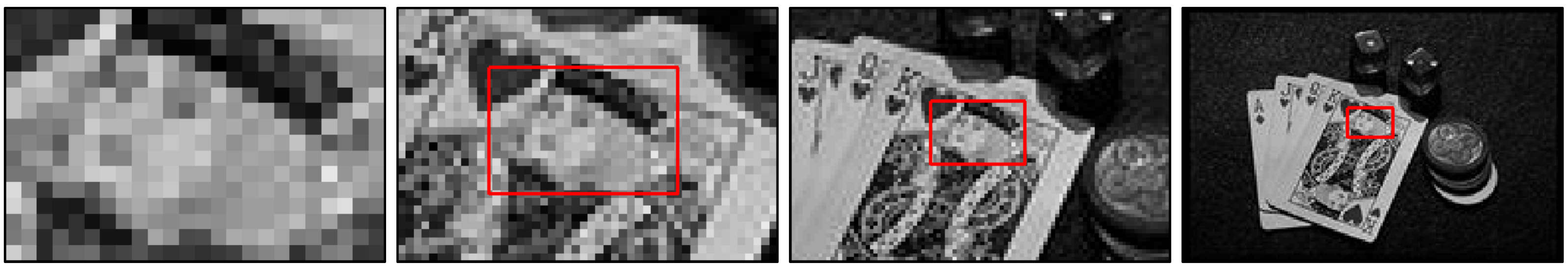
Difficulties With Digital Images

- Digital photographs are…
  - Bit-mapped
  - Low-resolution
  - Restricted in color



The Amazing Brain

- The brain sees more than just pixels.



- Aggregation into objects and larger regions is automatic & unconscious.
- Visual memory plays a role.

The Amazing Brain (2)

- The brain synthesizes diverse sources of information:

Shape

Texture/Shading

Multiple cues

- Seemingly effortless… (?)

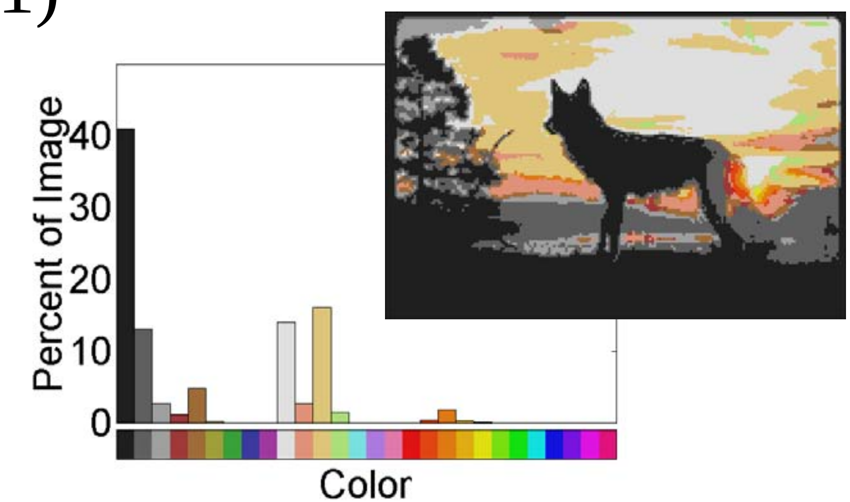
Brain-Like Computers?

- What we need is computers that can work more like brains.
  - Analysis of shapes, region, and texture
  - Semantic labeling of image content
  - Association with known models of objects, materials, and scenes

Increasing Difficulty

“Dumb” Stuff That Works

- Some simple (reliable) statistics work better than cognitively plausible (unreliable) ones.
- Classic example: color histograms for similarity (Swain & Ballard 1991)



Color Histograms Work…

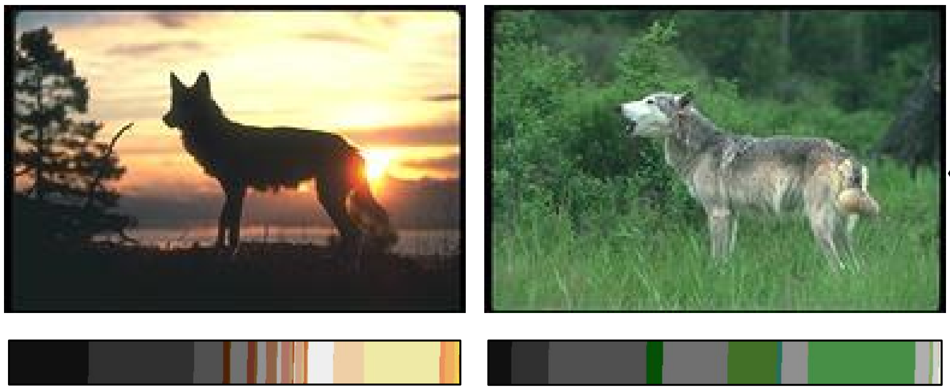
- Comparisons on more than 20,000 images:

Given this

These images are the most similar:

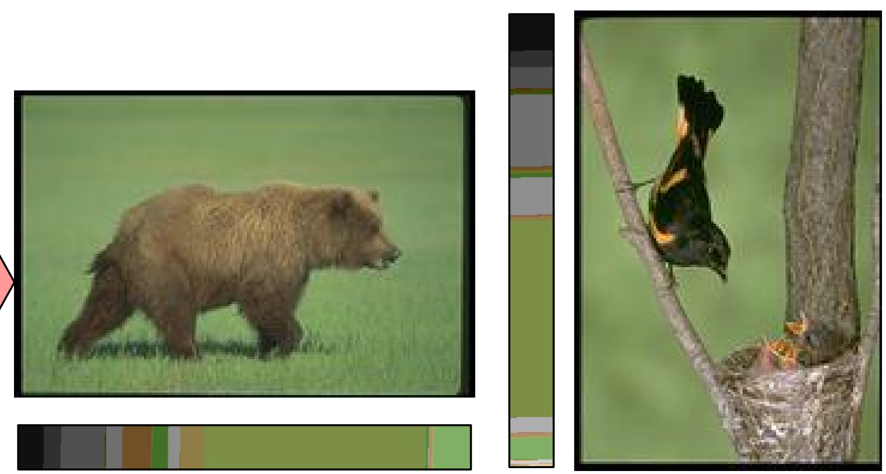
- Color histograms form the core of most working systems today.

…But Not Always



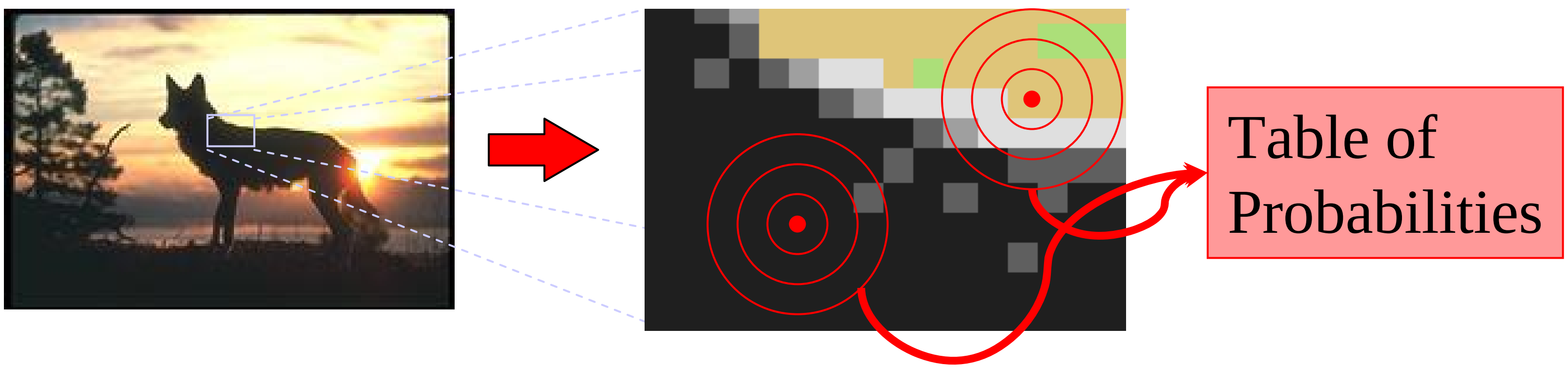
- Some related images have very different histograms.

- Some unrelated images have nearly the same histogram.



Variation: Color Correlograms

- Other statistical measures of image properties improve on color histograms.



- Correlograms (Huang et. al., 1997) have been particularly successful.

Correlogram Details

- Correlograms consist of a table of probabilities.

$$C(x,y) = P\Big(color(b) = x \Big| \big(color(a) = x\big) \wedge \big(\|a-b\| = y\big)\Big)$$

	Red	Orange	Yellow	etc…
1 pixel	0.32	0.0	0.06	0.14
3 pixels	0.16	0.0	0.04	0.0
5 pixels	0.08	0.0	0.03	0.0

"Given a pixel of color *x*, the probability that a pixel chosen distance *y* away is also color *x*"

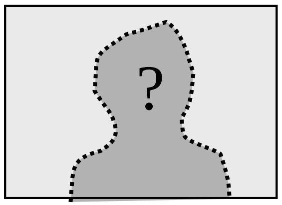
- Correlograms can be compared like vectors.

We Can Do Better

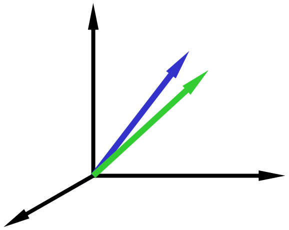
- How can we do something smarter?
  - Must incorporate spatial information & objects
  - Must employ multiple cues
  - Must adapt to lighting, etc.

Approach

1. Segmentation
   – Identifies spatial patterns and objects.



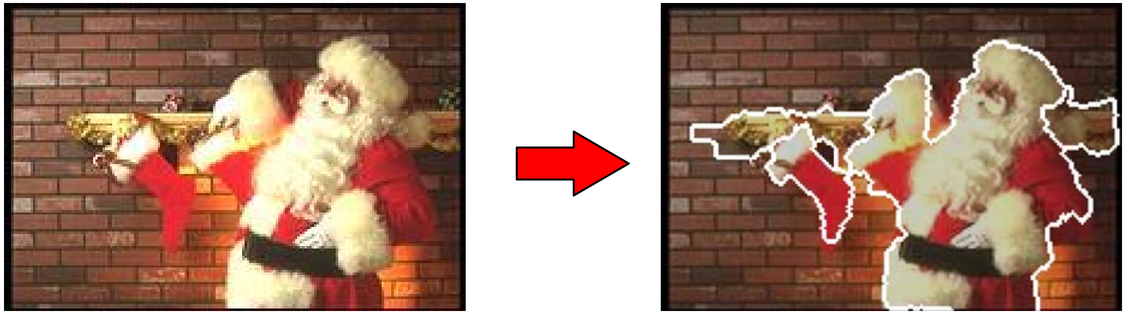
2. Vector Representation
   – Includes color, texture, and location cues.



3. Vector Comparison
   – Allows adaptation for varying conditions.
4. Focus on Objects

Segmentation

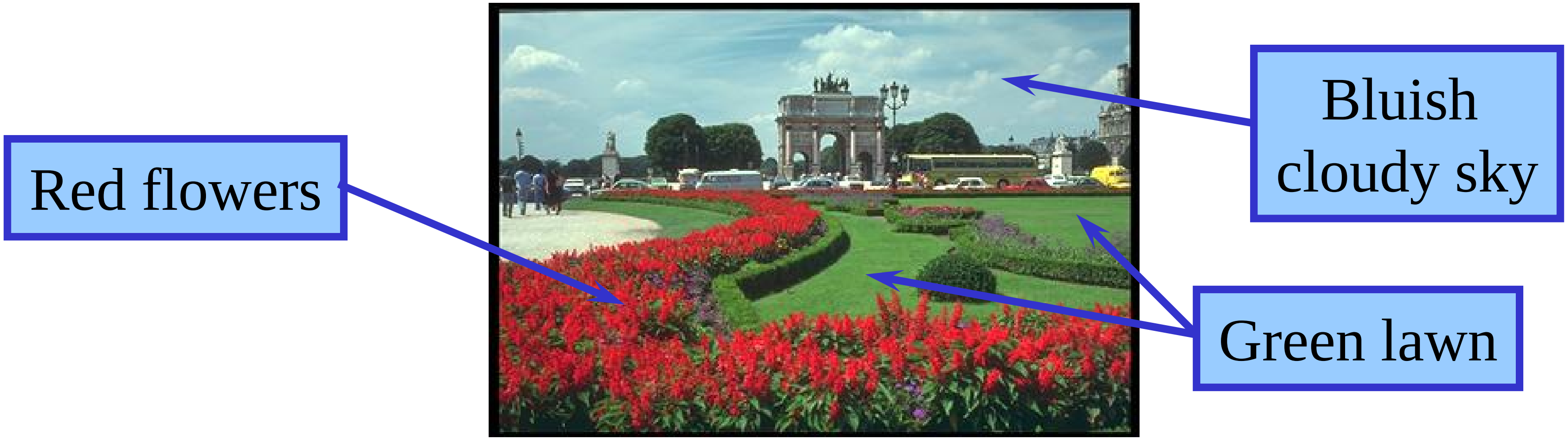
- Segmenting an image means dividing it into regions that "belong together."



Q. What's a sensible way to segment any given picture?

Characterizing Regions

- When humans segment an image, they can explain why each region hangs together.

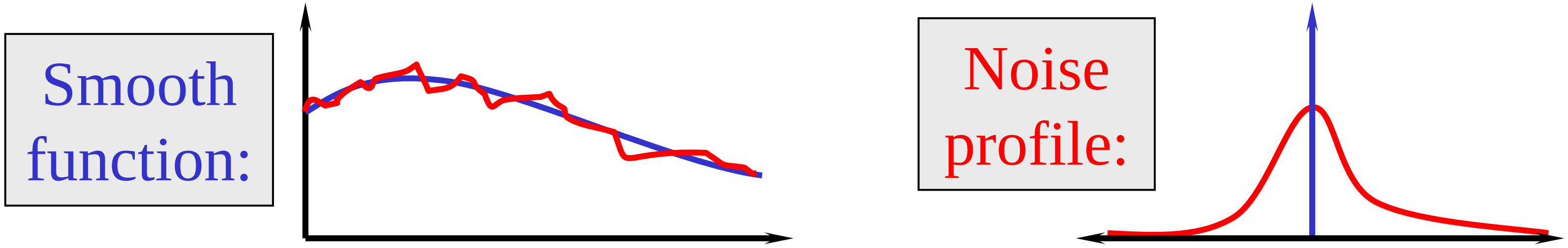


⇒ *Models motivate the grouping into regions.*

Mathematical Models of Regions

- Model regions as smooth functions + noise:

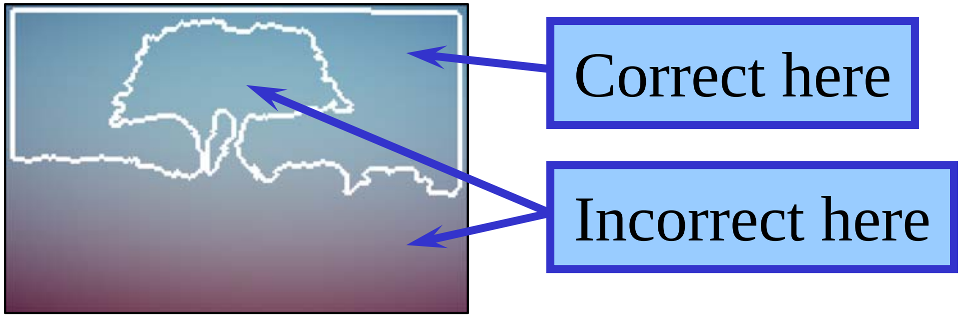
- 2D example:



Models of Regions (2)

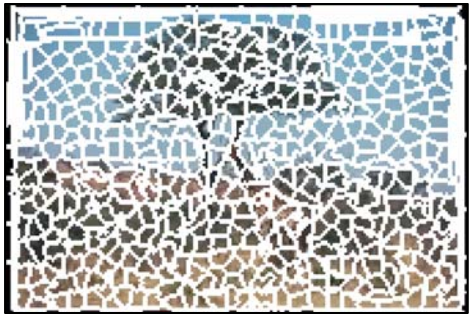
- Each model tries to predict the image.
- Successful models are rare.

Bad model



Good model

Outline of Segmentation Process



1. Start with small local regions.

   (Felzenszwalb & Huttenlocher 1998)
2. Create a pool of potential models.
3. Measure fit between all models & local regions.
4. Select a small number of models that fit many local regions well.

   (Details on the next slide)

Goal:

Segmentation Details

- Best segmentation found via energy minimization:

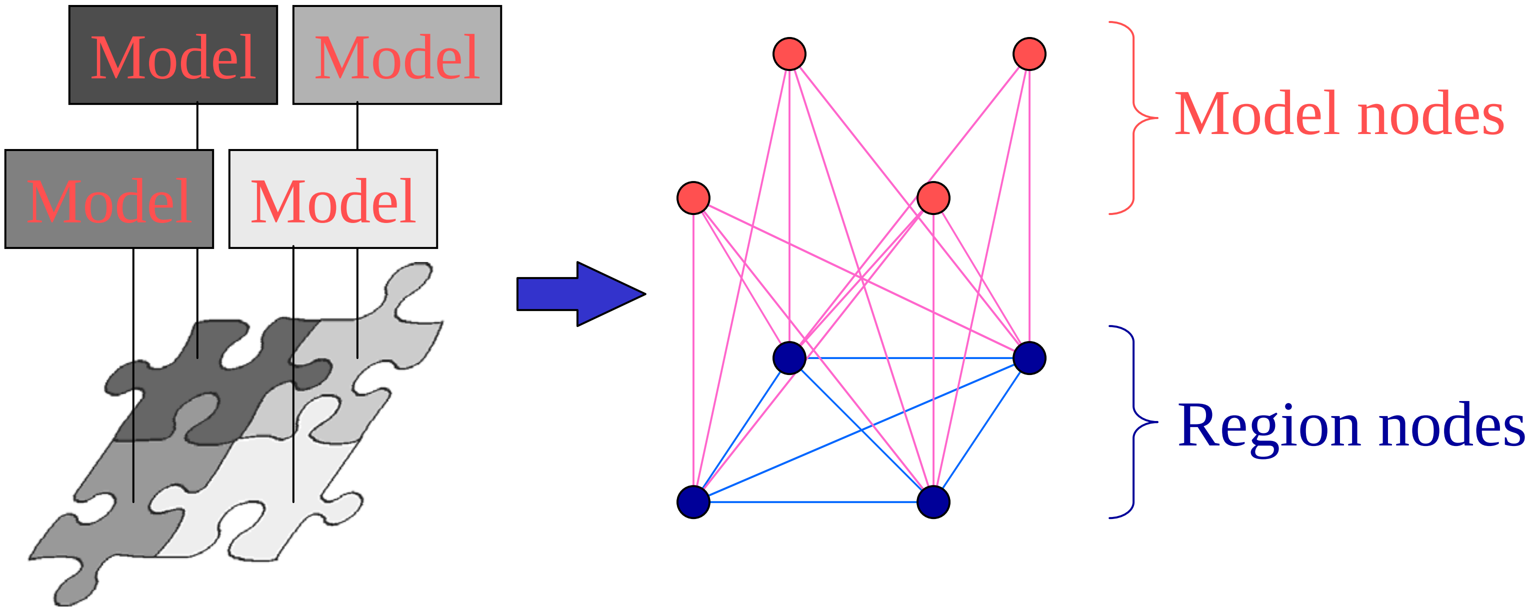
$$\boxed{E(R) = \sum_{r \in R} Fit(r, M_r) + \sum_{r_1 \in R} \sum_{r_2 \in R} \Delta(r_1, r_2)}$$

"The energy of a segmentation into regions R is equal to the fit of each region with its model plus a penalty to discourage excess regions."

- Minimum energy is difficult to compute in general.

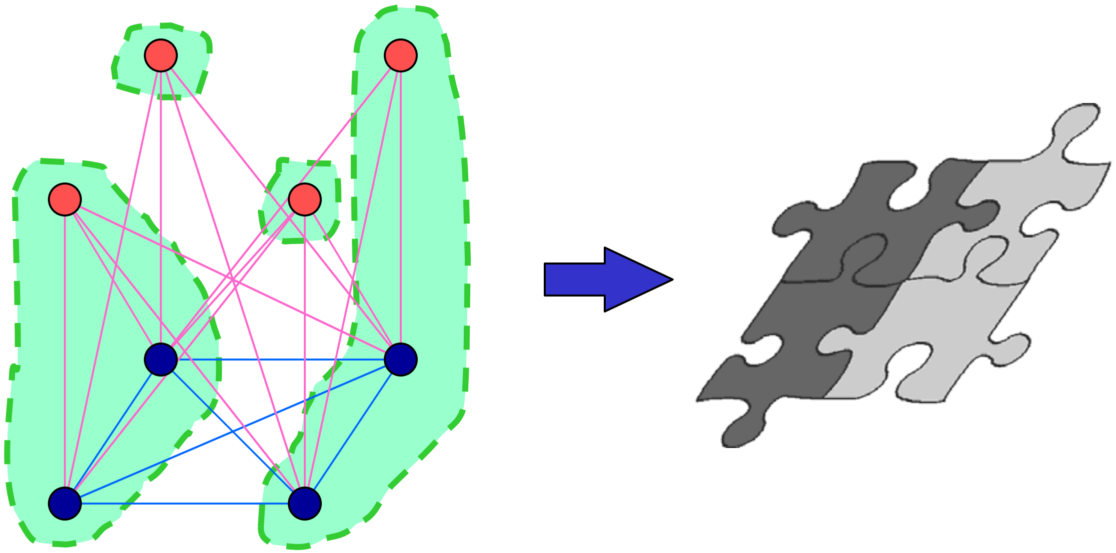
Graph Formulation

- Minimum energy = minimum graph cut
  (compare with Boykov, et. al., 1998)



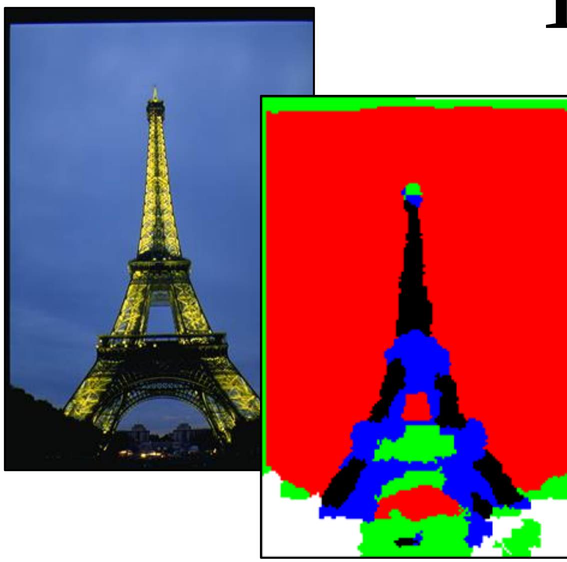
Graph Formulation (2)

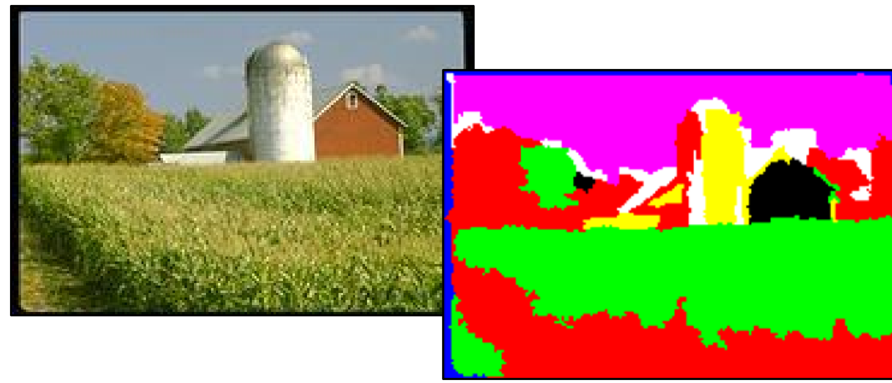
- Minimum graph cut = best segmentation

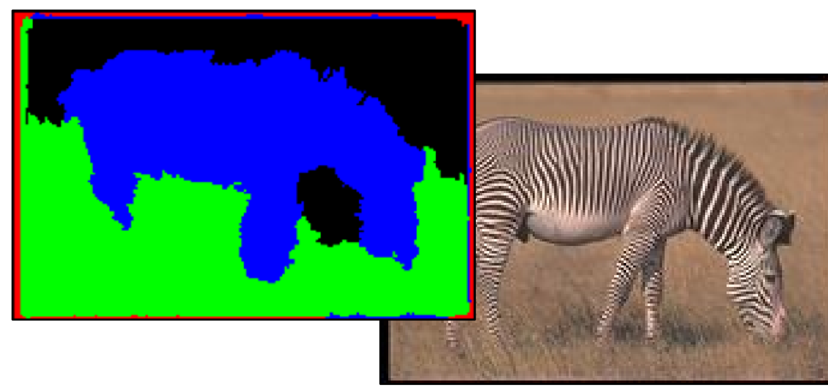


- Running time bound: quadratic in # of nodes
- Quality bound: Energy found is ≤ 2 × optimal.

Examples



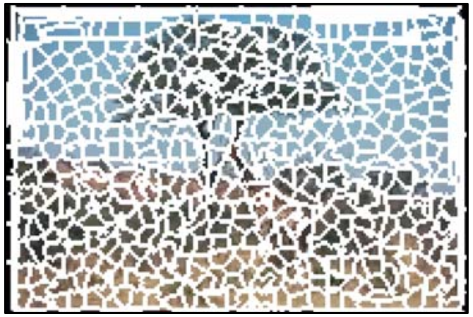




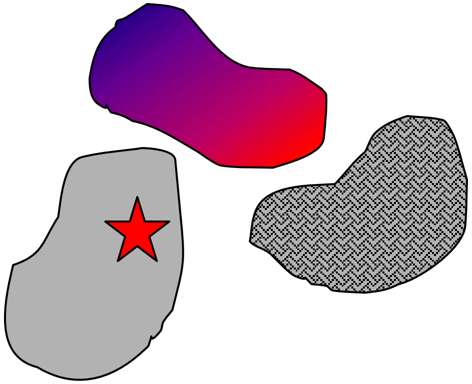
Related Work

- Stereo Vision & Energy Minimization
  (Boykov, Vexler & Zabih, 1998)
- Normalized Cuts
  (Shi & Malik, 1997)
- JSEG
  (Deng, Manjunath, & Shin, 1999)

A Region-Based Representation



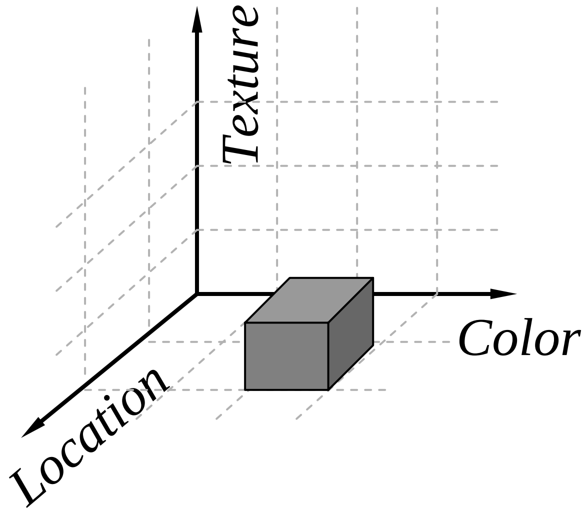
- Begin with segmentation.
  (Provides locality.)
- Describe each patch using multiple features.
  – Color
  – Texture
  – Location



- Combine such that each piece is preserved.

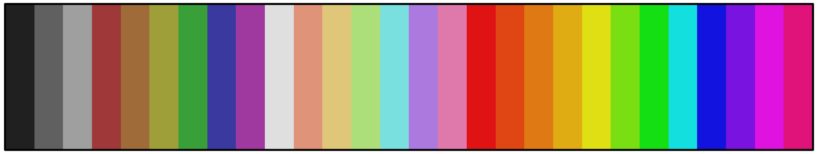
Describing an Image by its Parts

- Discretize the range of each feature.
  (Color, texture, and location)
- Count area in image described by each combination of features.
  - Blue-Smooth-TopLeft: 5,
    Blue-Smooth-TopMiddle: 1,
    …
    Green-Smooth-TopLeft: 0, etc.



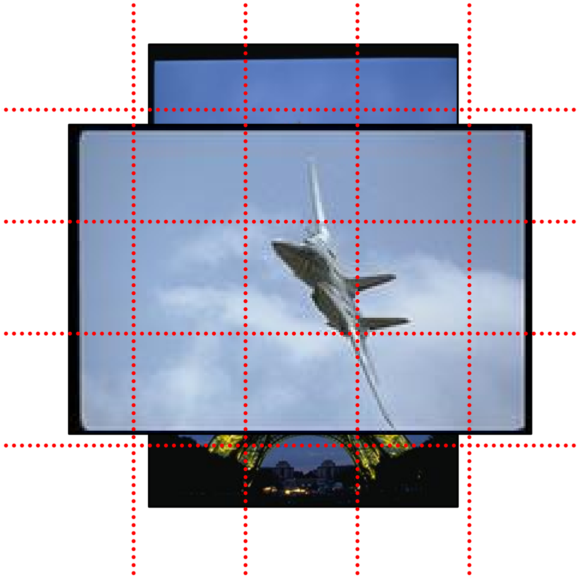
Discretization

- Color: 28 bins



- Texture: 3 bins

  (smooth, textured, rough)
- Location: 25 bins



- Total: 28×3×25

  = 2100 combinations

Vector Representation

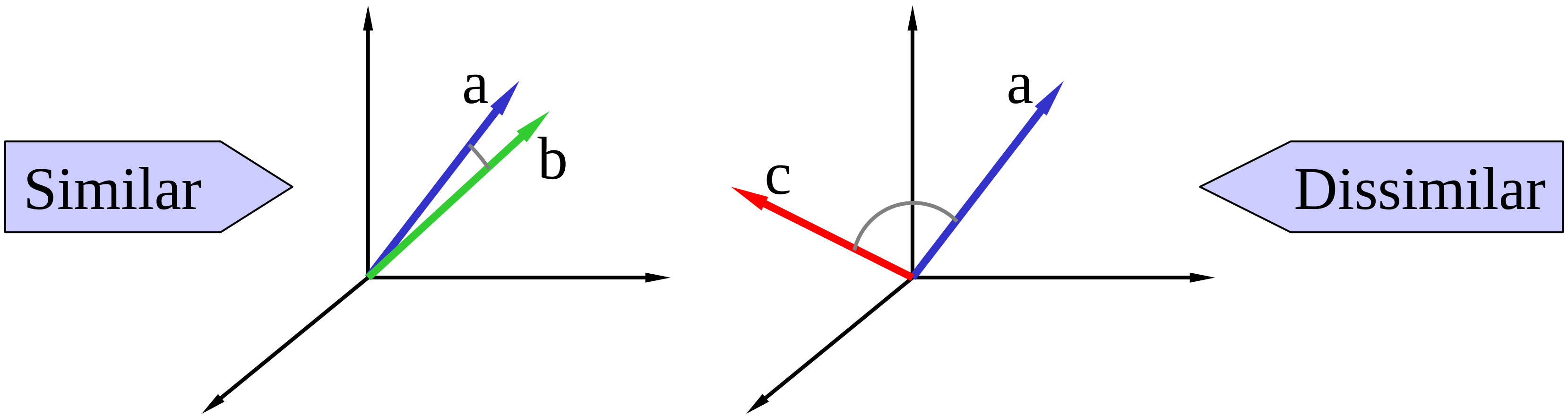
- Final representation of image is a vector with 2100 dimensions.

$$\mathbf{v} = \left\langle v_{c_1 t_1 l_1}, v_{c_1 t_1 l_2}, \ldots, v_{c_1 t_1 l_{25}}, v_{c_1 t_2 l_1}, \ldots, v_{c_{28} t_3 l_{25}} \right\rangle$$

- Each dimension records how much of a particular type of material is present.
  - e.g., how much smooth blue in the top left corner?

Comparison

- Vectors are points in space.
- Images with similar composition will have similar (normalized) vectors.
- Angle between similar vectors will be small.



Comparison (2)

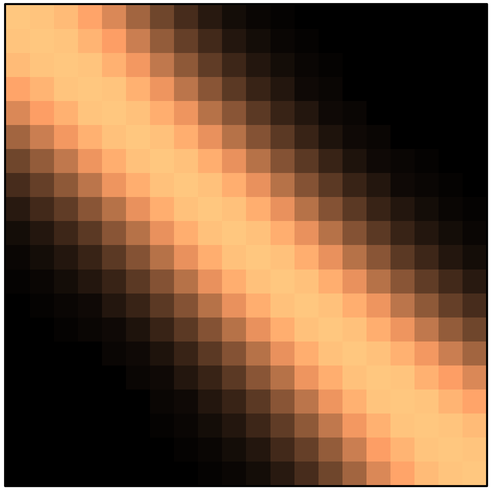
- Compare two images using a cosine metric:

$$D(\mathbf{v_1}, \mathbf{v_2}) = \cos^{-1}\left(\frac{\mathbf{v_1^T S v_2}}{\sqrt{(\mathbf{v_1^T S v}_1)(\mathbf{v_2^T S v_2})}}\right)$$

- Note generalization using **S** matrix:
  - **S** = **I** is standard cosine metric.
  - Other values of **S** allow adjustments to metric.

Comparison: Match Coefficients

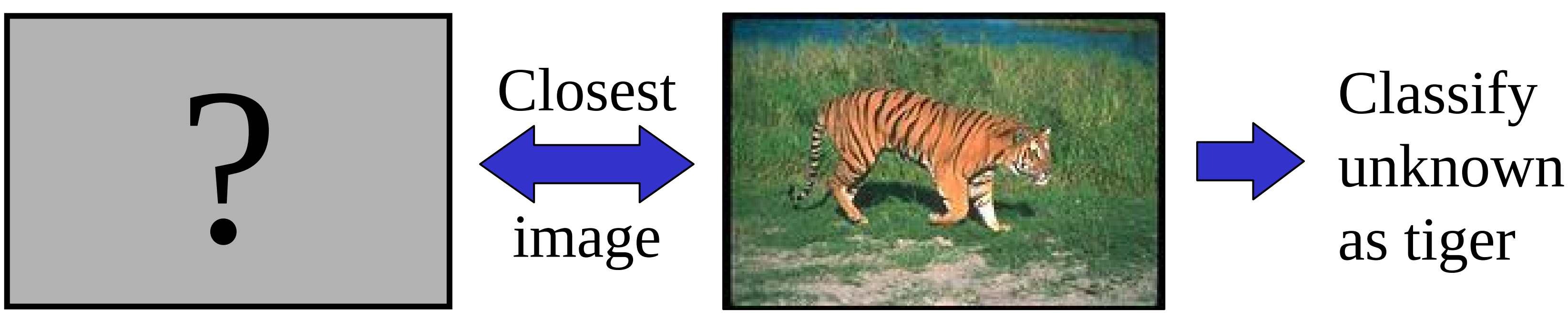
- Discretization of features loses some similarity information.
  - e.g., ***Blue*** is closer to ***Green*** than to ***Orange***.
- Such partial matches may be encoded in off-diagonal terms of **S**.



(lighter ⇔ more similar)

Evaluating the Vector Method

- Two sets of test images:
  - 12 and 16 categories of ~100 images each
- Classification task
  - Most similar known image is used to classify unknown images.

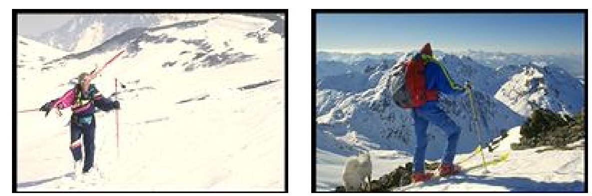


Sample Categories

Airshows

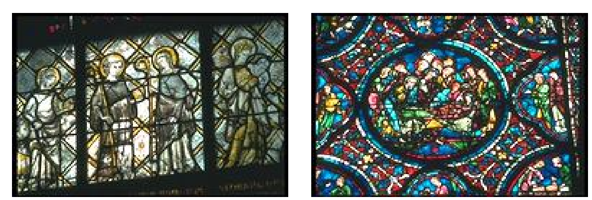
Caves

Elephants



Skiers

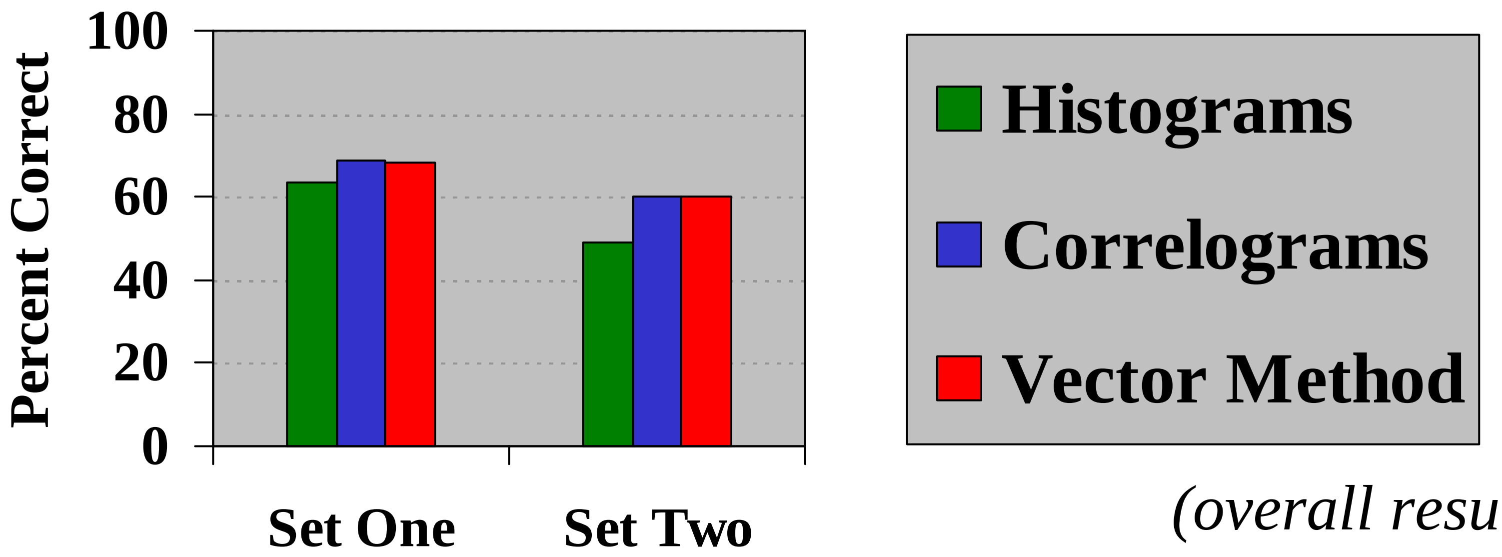
Polar Bears



Stained Glass

Classification Results

Comparison with histograms and correlograms:



(overall results)

- Outperforms baseline (histogram, green).
- Competitive with advanced image metric (correlogram, blue).

Object Queries

- Something we've wanted to do all along:

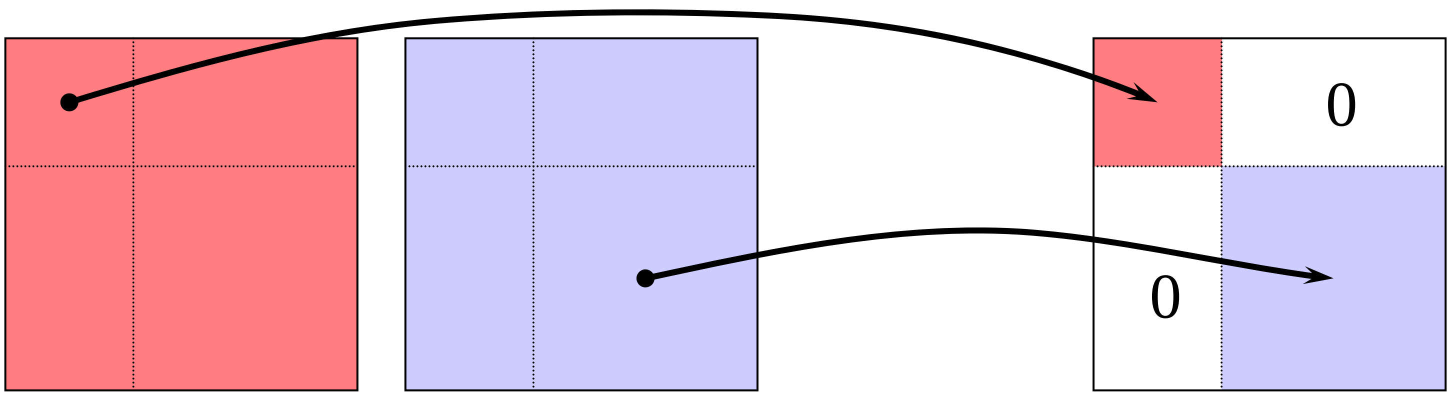
Search for objects, not whole images.

Rank 60
(of 19,000)

Rank 1
(of 19,000)

How Object Queries Work

$$S(i,j) = \begin{cases} S_{object}(i,j) & \text{if i and j appear in the target object.} \\ S_{background}(i,j) & \text{if neither i nor j appear in the target.} \\ 0 & \text{otherwise.} \end{cases}$$

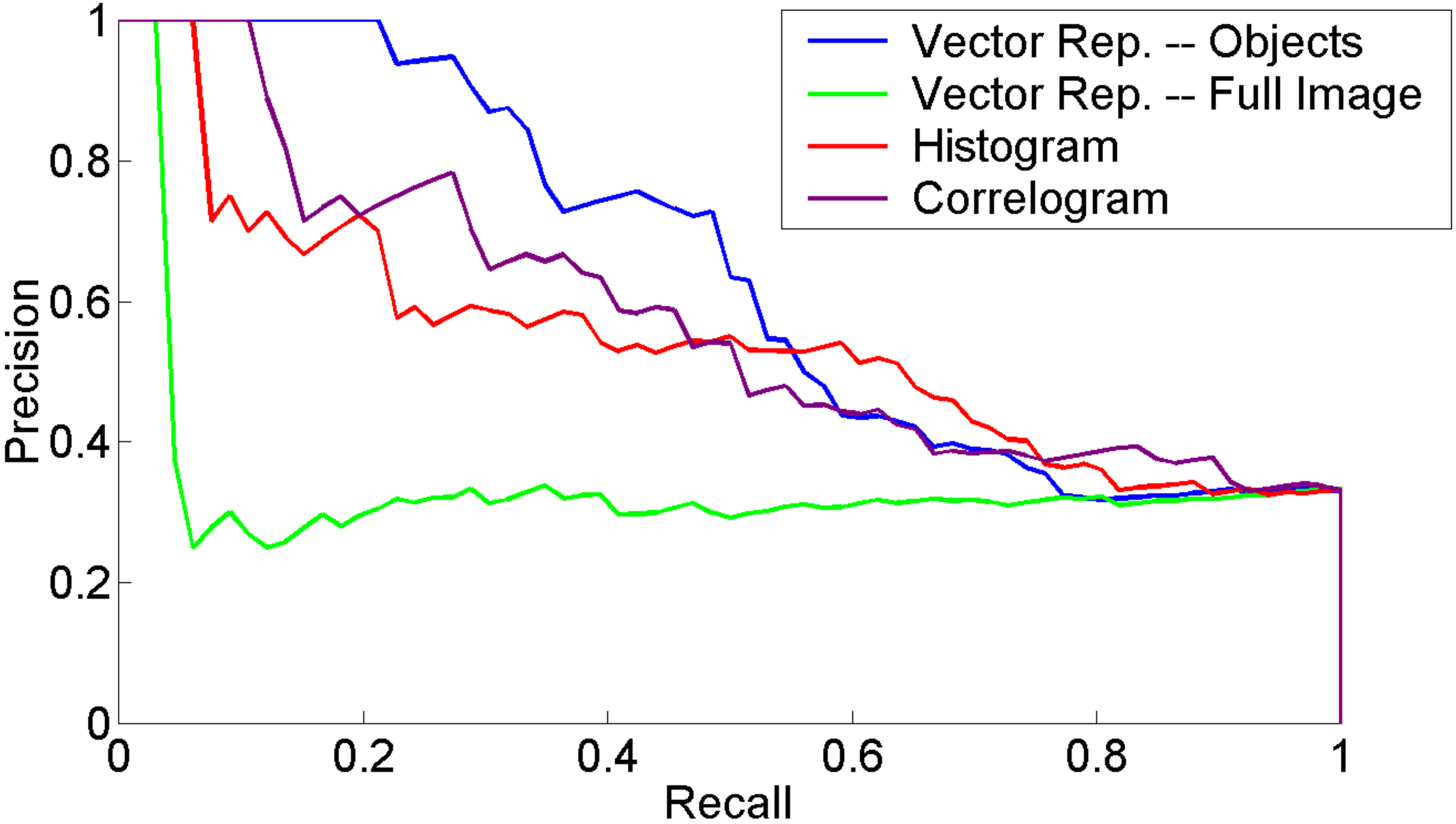


S_{object} (strict match) $S_{background}$ (loose match)

Testing Object Queries

- 200 images of cars
  - Visual context is irrelevant.
  - Classes are colors of car.

Sample Result (Red Cars)



Summary

- Vector representation preserves critical image features.
- Retrieval with vector representation is competitive with other techniques on full images.
- Flexible use of regions allows search for objects & arbitrary figures of interest.

Related Work

- Vector Representation
  - Howe & Huttenlocher, 2000; Howe, 2000; Howe 1998
- Earth Mover’s Distance
  - Cohen, 1999
- Blobworld (UC Berkeley)
  - Carson et. al., 1999; Belongie et. al., 1997
- Netra (UCSB)
  - Deng & Manjunath 1999; Ma & Manjunath, 1997

The Future

- Moving away from absolutism

  “OK, we can find red cars. Can we find *cars*?”

  – Relational encodings:
    - *White fur <u>next to</u> red velvet*
    - *A piece of <u>all the same</u> color*

- Interplay between segmentation, similarity, and compression/coding

  e.g., Color & texture from segment model

Challenges

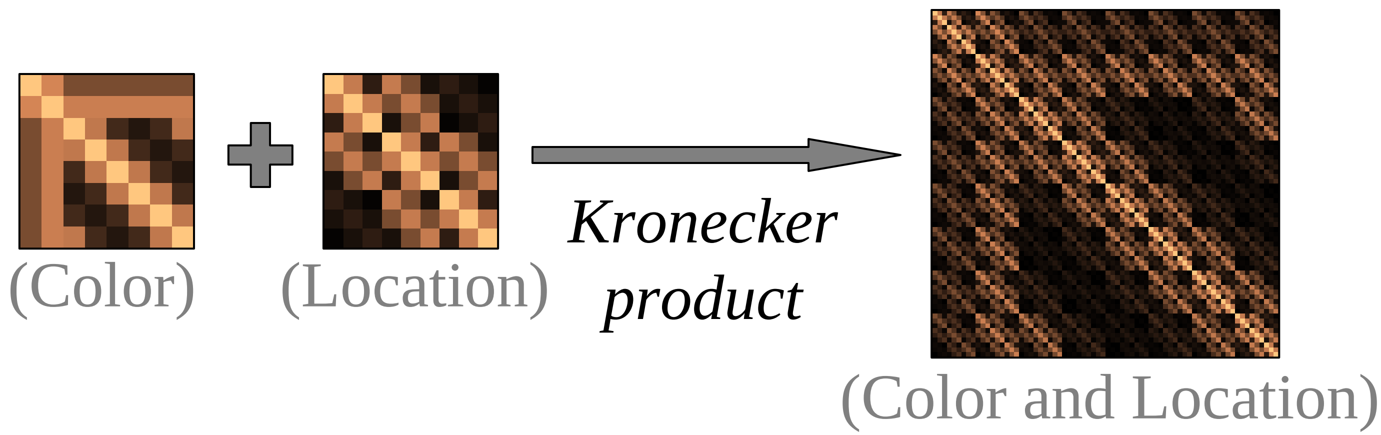
- Assumption: parts that belong together should look alike…

  …not always true!

- More sophisticated region models may help.

Generating the **S** Matrix

- **S** assembled from matrices $\mathbf{S}_j$ for each feature

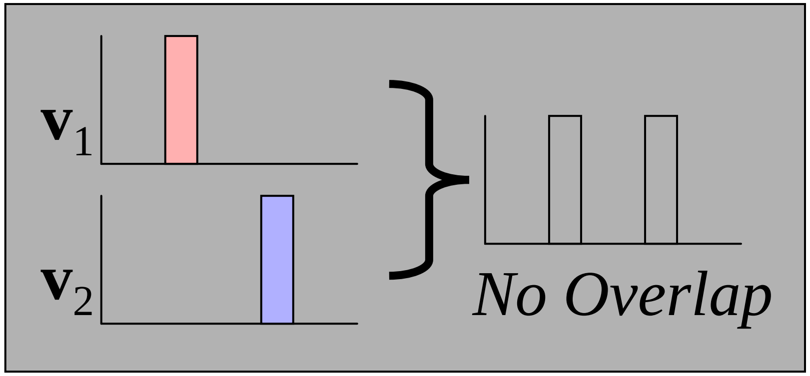


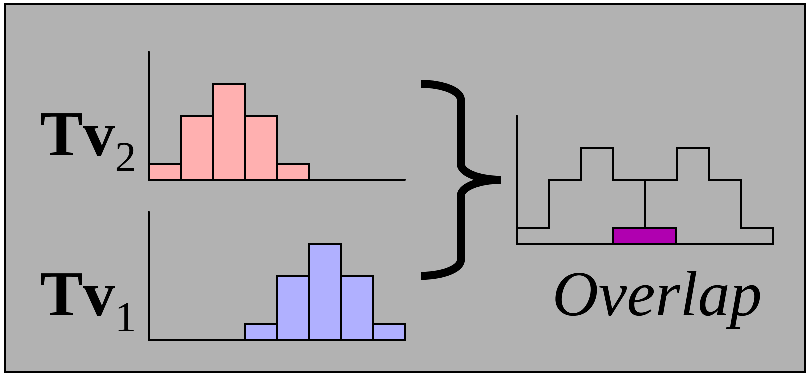
- Smaller matrices are determined by the similarity of the feature values.
  - e.g., ***Blue*-*Green*** vs. ***Blue* -*Orange*.**

Alternate View of **S** Matrix

- Cholesky factorization of **S**: $\mathbf{S} = \mathbf{T}^{\mathrm{T}}\mathbf{T}$
- Cosine metric of modified vectors:

$$D(\mathbf{v_1}, \mathbf{v_2}) = \cos^{-1}\left(\frac{(\mathbf{Tv}_1)^{\mathrm{T}}(\mathbf{Tv}_2)}{\left((\mathbf{Tv}_1)^{\mathrm{T}}(\mathbf{Tv}_1)\right)\left((\mathbf{Tv}_2)^{\mathrm{T}}(\mathbf{Tv}_2)\right)}\right)$$





Optimizations

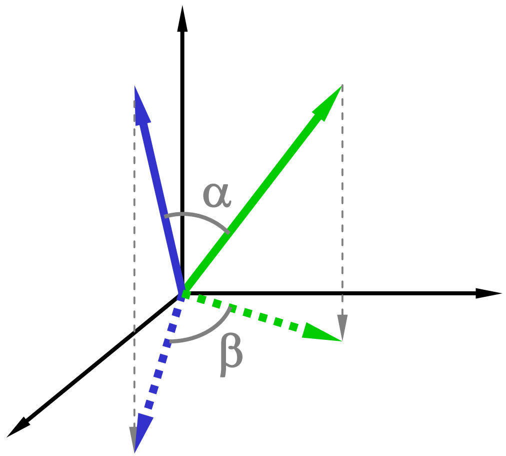
- Similarity computation is linear in sparse vector **v**.

$$D(\mathbf{v_1},\mathbf{v_2}) = \cos^{-1}\left(\frac{\overbrace{\mathbf{v_1^T}}^{\text{Computed once per query}}\mathbf{S}\mathbf{v_2}}{\sqrt{\underbrace{(\mathbf{v_1^T}\mathbf{S}\mathbf{v}_1)}\underbrace{(\mathbf{v_2^T}\mathbf{S}\mathbf{v_2})}}}\right)$$

Precompute & cache

Search Pruning

- Nearest neighbor search can be pruned by projection onto lower-dimensional spaces.



- β is lower bound on α.
- Images with β greater than some cutoff need not be considered.

Dividing the Color Space

- Color seeds are dispersed evenly in HSV color cone.
- Divided into Voronoi regions.
- Ensures perceptual uniformity.