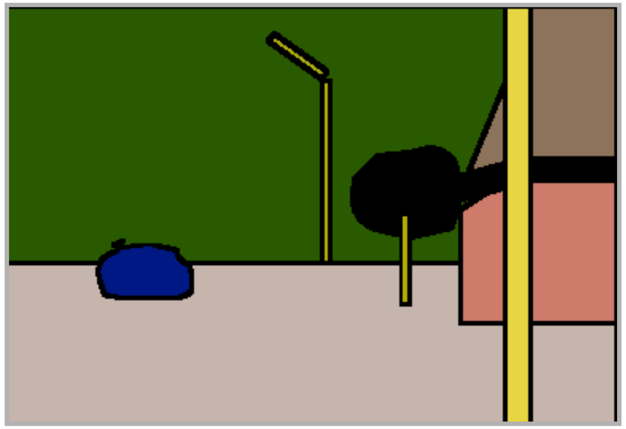
REGIONS

- Defining the problem
- Automated algorithms
- Interactive methods
- Introducing semantics

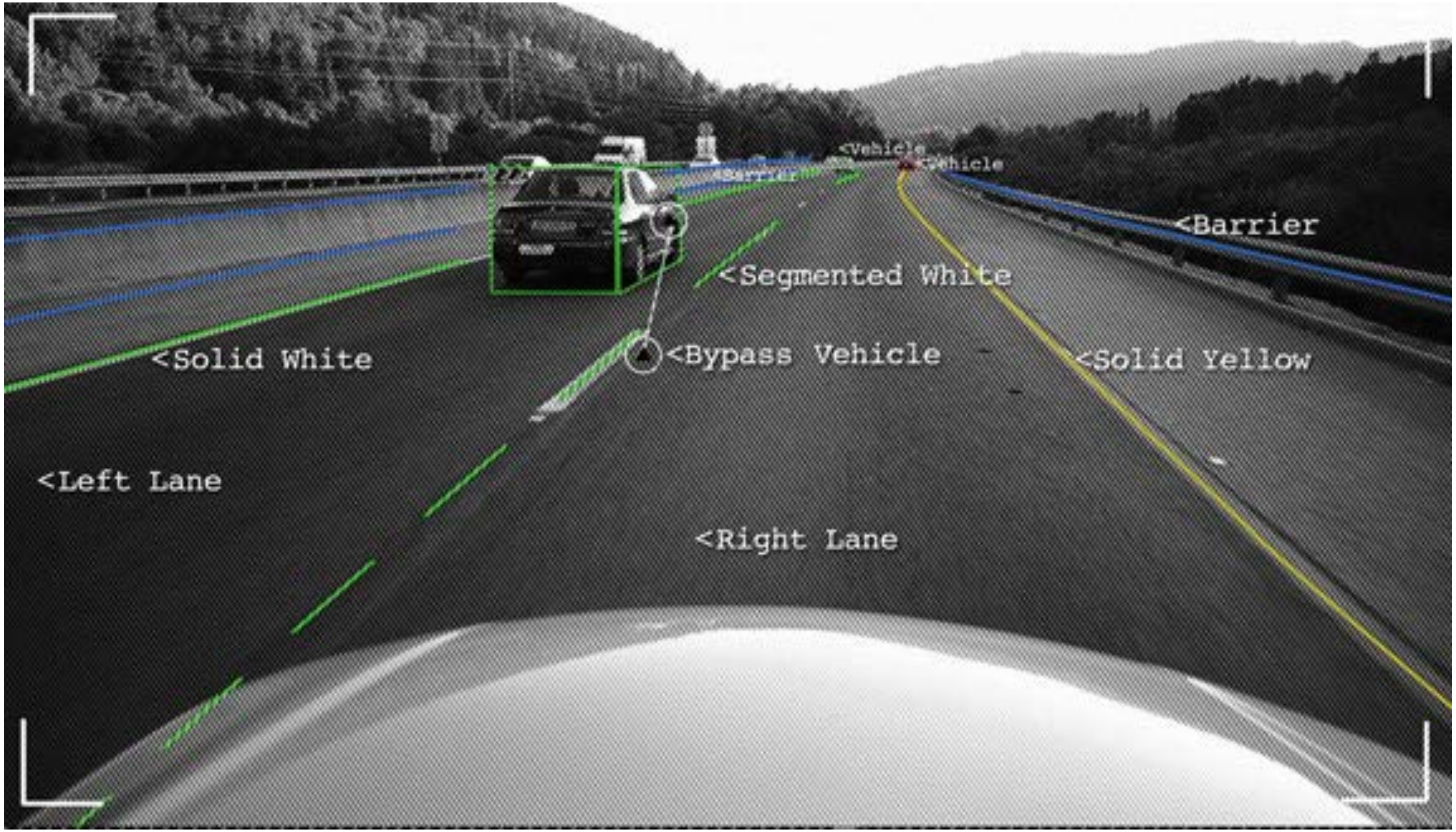
REGION SEGMENTATION



Ideal region: Set of pixels with the same statistical properties and corresponding to the same object.

Purpose: Should help with recognition, tracking, image database retrieval, and image compression among other high-level vision tasks.

AUTOMATED DRIVING



http://www.mobileye.com/

IN THEORY

Look for an image partition such that:

$$I = \bigcup_{i=1}^{m} S_i$$

$$S_i \cap S_j = 0, \forall i \neq j$$

$$H(S_i) = True, \forall i$$

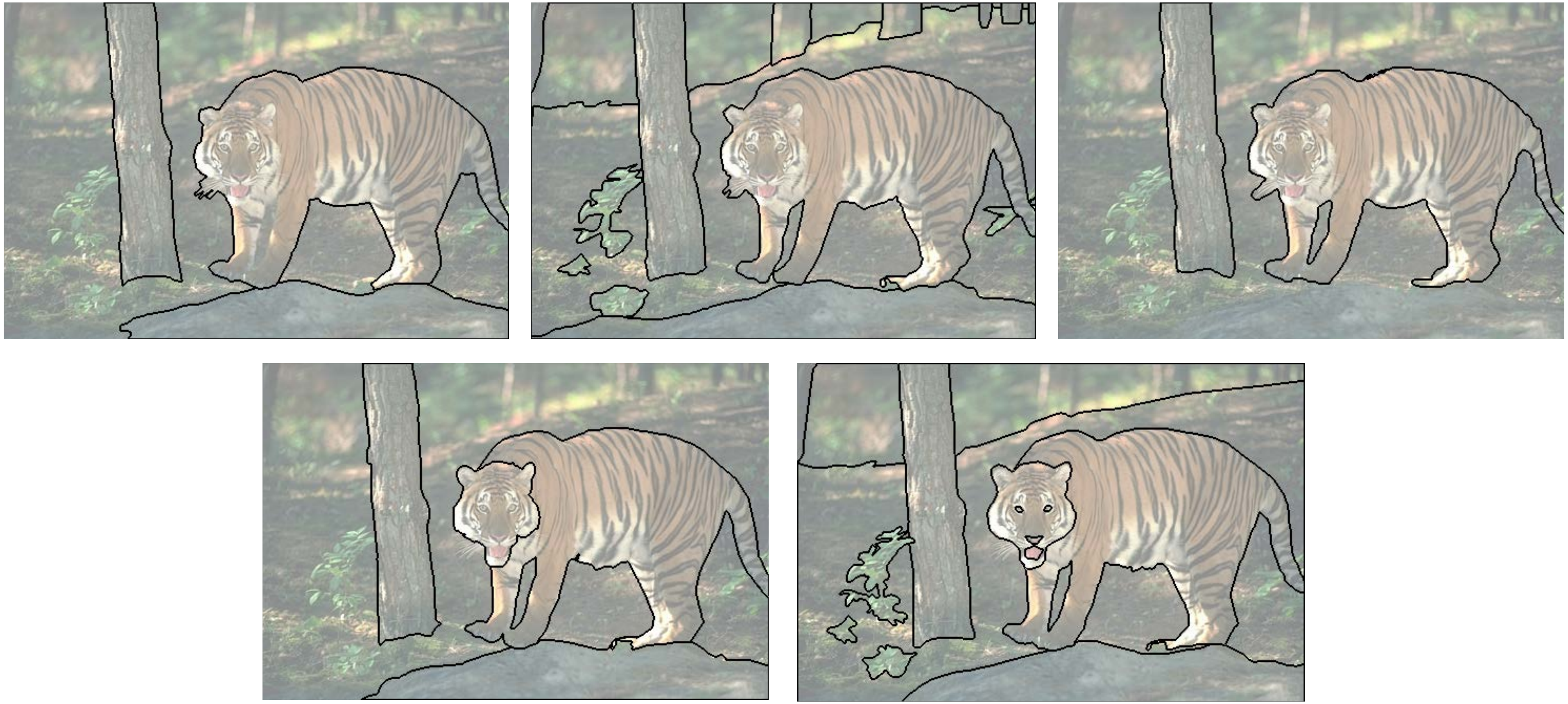
$$H(S_i \cup S_j) = False, \text{ if } S_i \text{ and } S_j \text{ are adjacent.}$$

where H measures homogeneity.

IN PRACTICE

CONTEXT IS ESSENTIAL

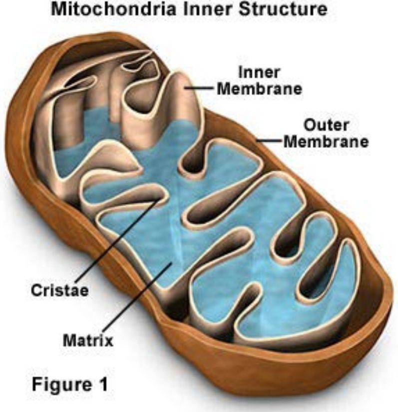
MULTIPLE ANSWERS

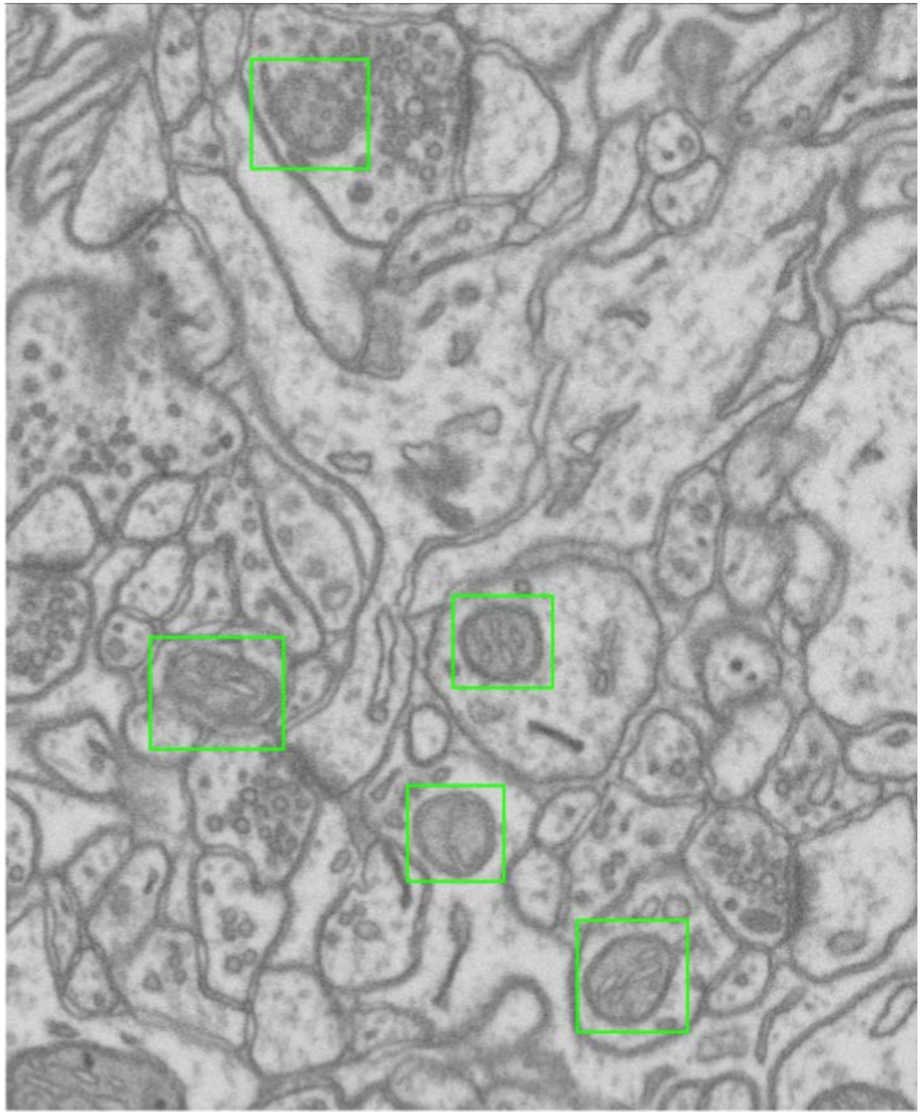


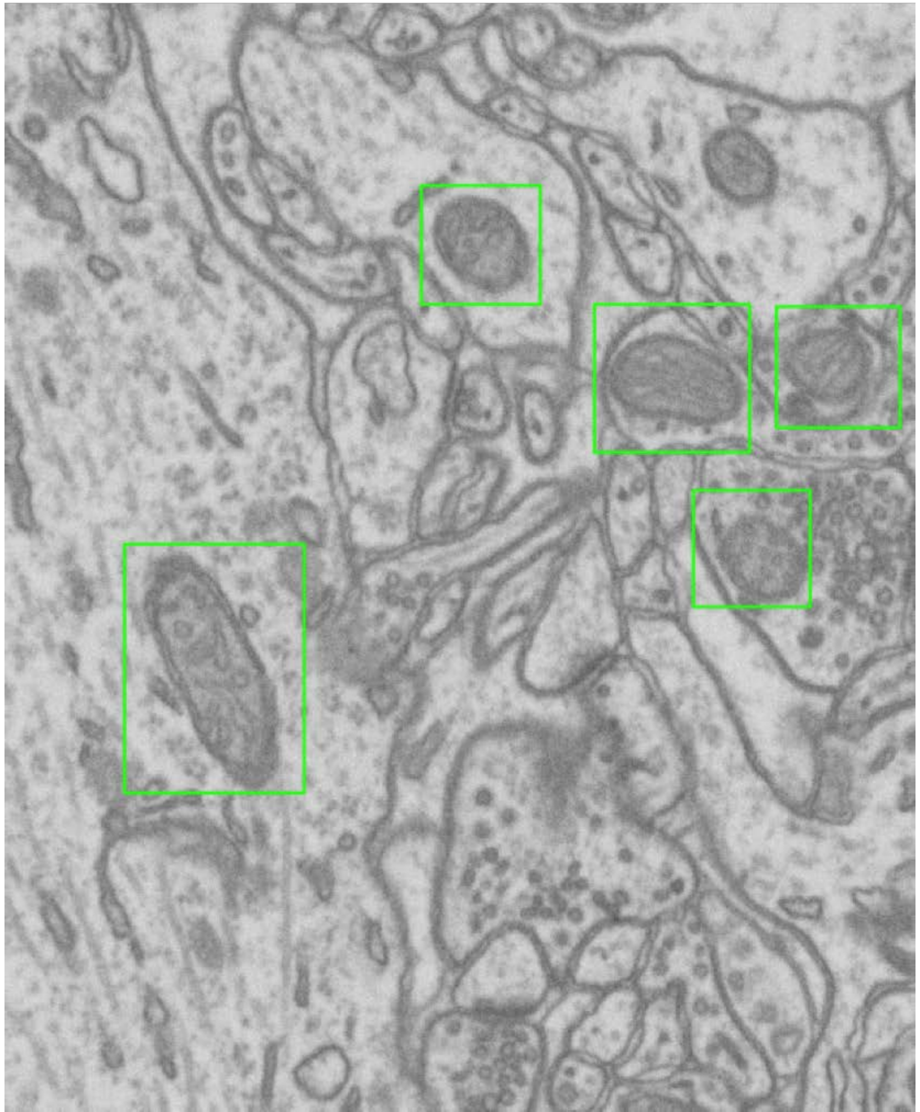
Segmentations hand-drawn by 5 different people.

HOMOGENEOUS OR NOT?

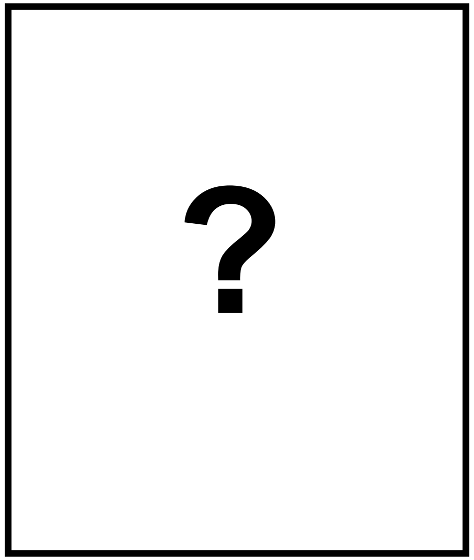
MITOCHONDRIA







DERIVED IMAGES



Homogeneity can be evaluated in the original image data or in 'derived' images:

- Gray level images
- Color images from R, G, B
- Textural images
- Displacement images from motion analysis
- 3D depth images

IN THEORY

Split:
- Start with a partition that satisfies Eq. 4.
- Split regions until they all satisfy Eq. 3.

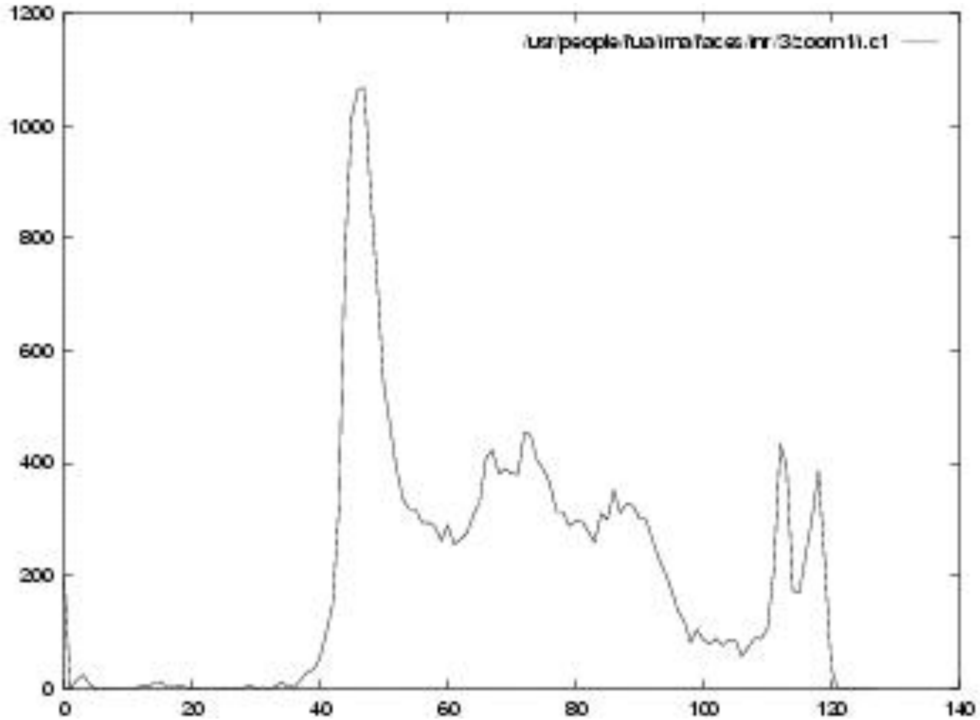
Merge:
- Start with a partition that satisfies Eq. 3.
- Satisfy Eq. 4 by merging regions.

Homogeneity:
- Uniform gray-level or color statistics.
- Regions to which a parametric surface can be fitted.

IN PRACTICE

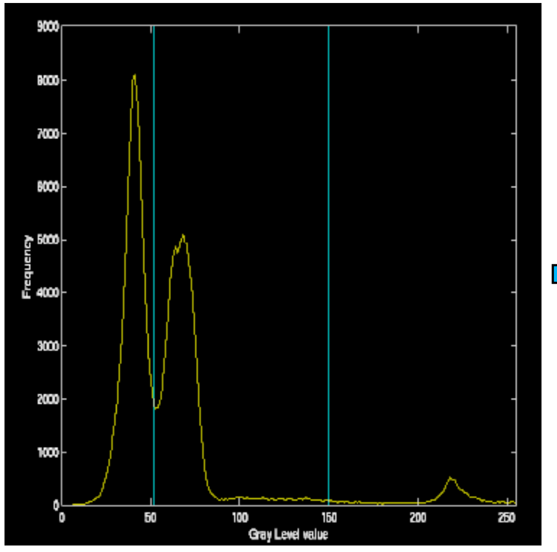
- Histogram splitting.
- K-Means.
- Mean Shift.
- Graph theoretic methods.
- Convolutional Neural Nets

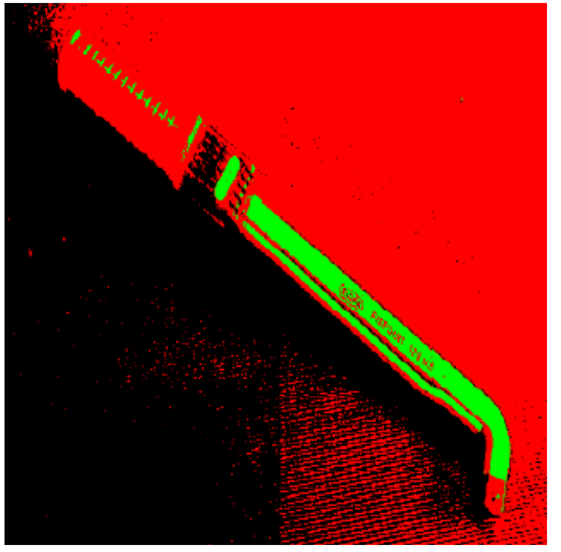
IMAGE HISTOGRAM



Number of pixels that have a given gray level.

HISTOGRAM SPLITTING



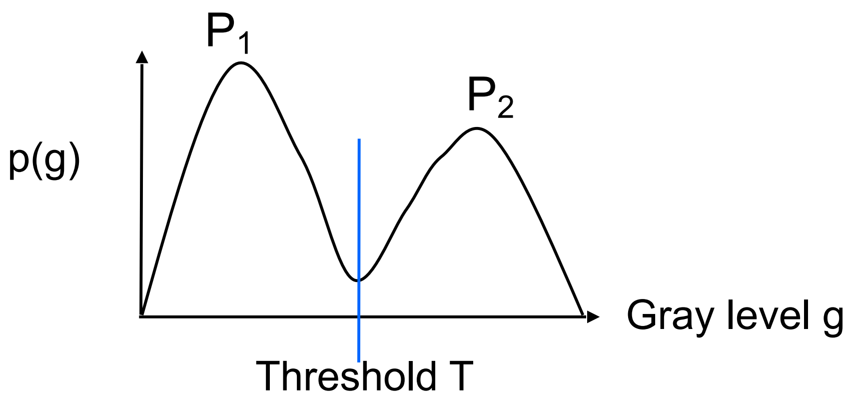


might want to do morphology (shrink and grow) to remove outliers

- Groups of similar pixels appear as bumps in the brightness histogram
- Split the histogram at local-minima
- Label pixels according to which bump they belong to

RECURSIVE SPLITTING

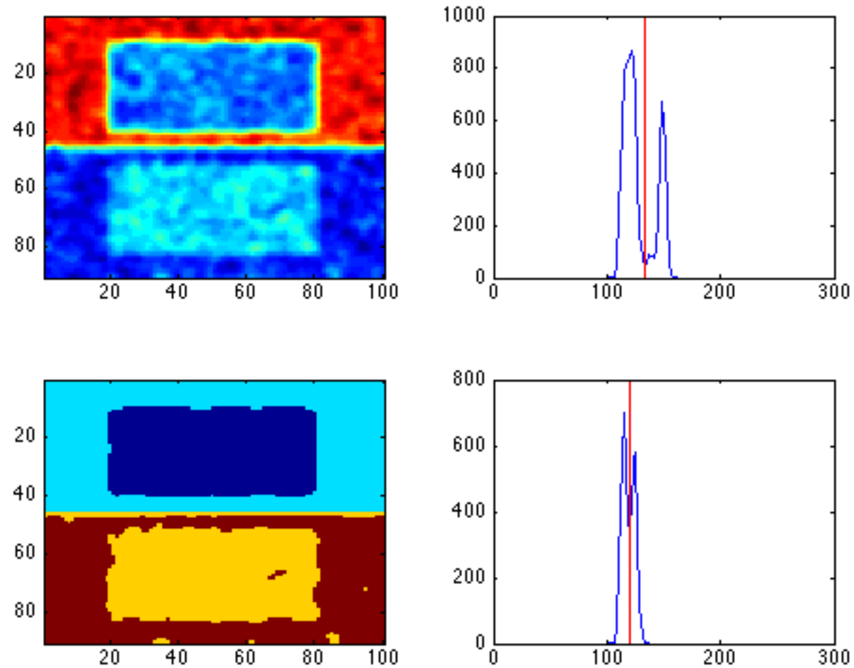
the histograms of P1 and P2 will probably have another valley. Use finer bins. Repeat until cannot split



- Compute image histogram.
- Smooth histogram.
- Look for peaks separated by deep valleys.
- Group pixels into connected regions.
- Smooth these regions.
- Iterate.

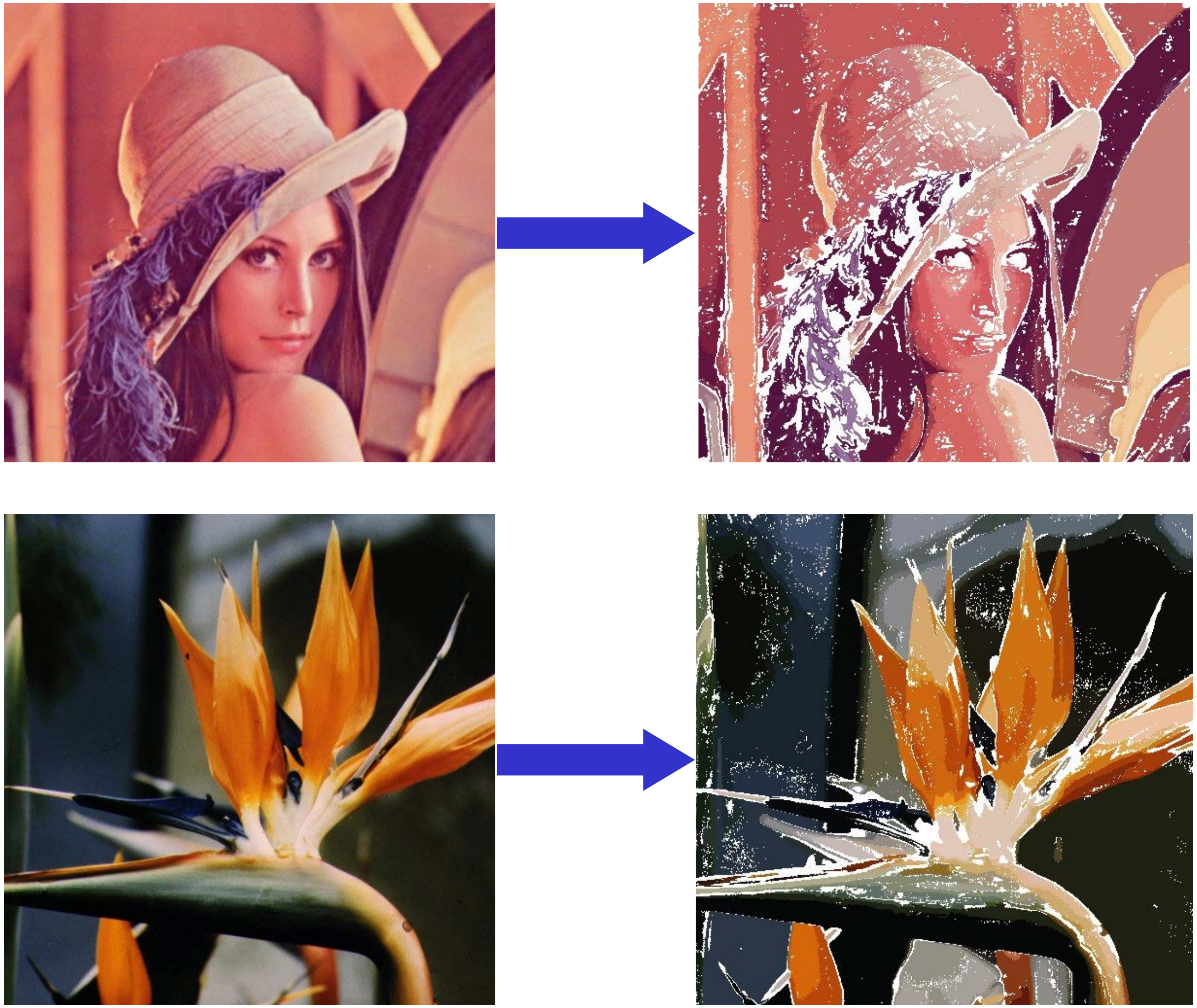
RECURSIVE SPLITTING

- A first threshold is used to segment the dark pixels.
- This yields two regions, the bottom half of the picture and the dark rectangle at the top.
- The bottom half of the picture can now be more easily segmented into two regions.



-> Decisions can be deferred until enough information becomes available.

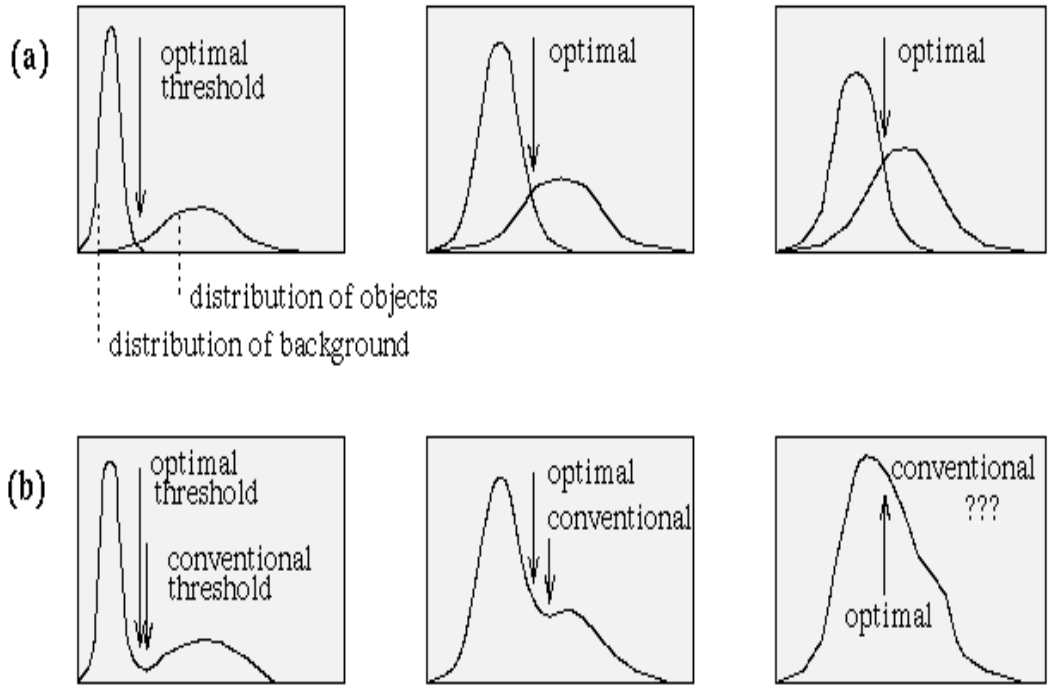
OVERSEGMENTATION



without cleaning

THRESHOLDS

regions in image are created by 2 different process

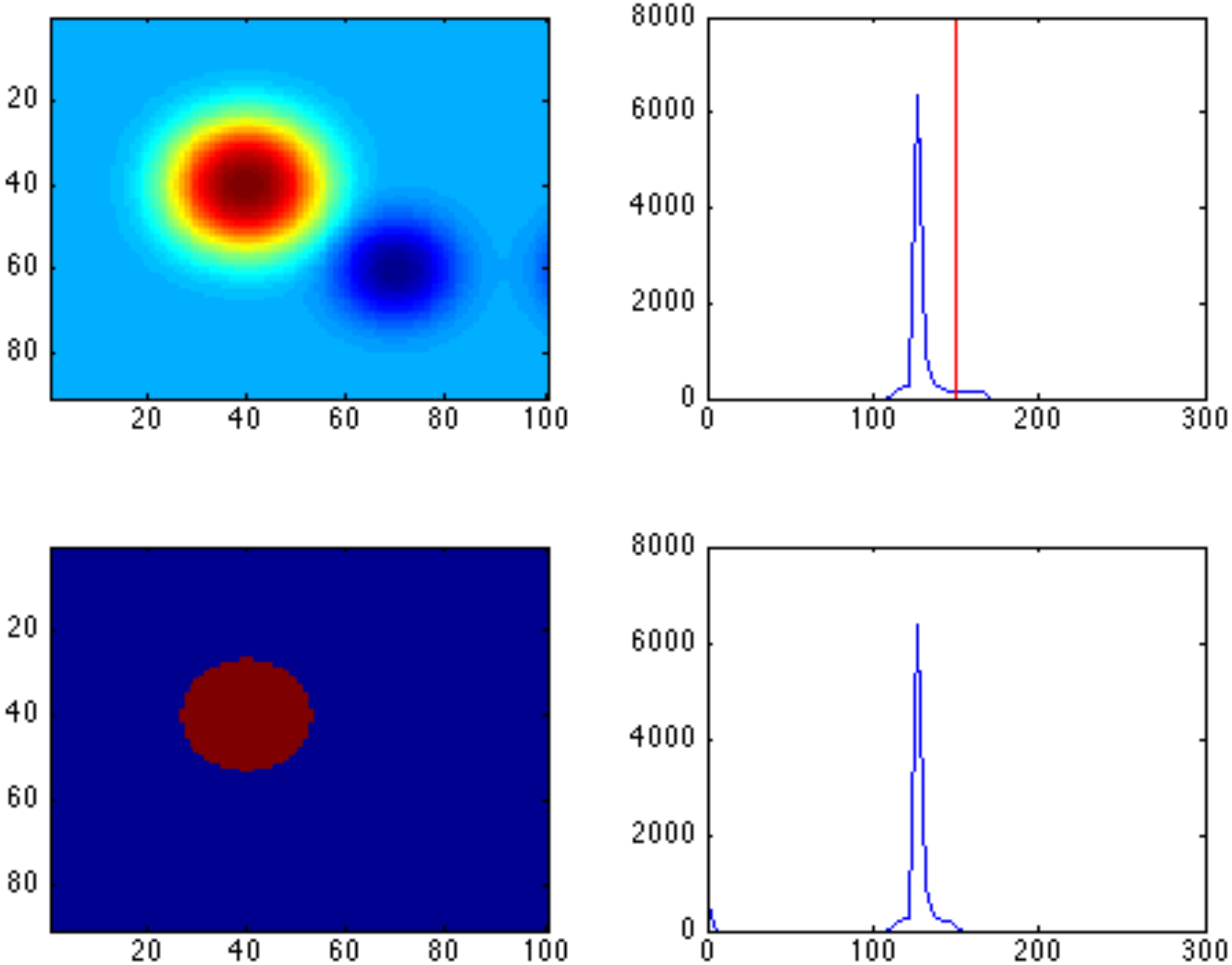


Probability distributions

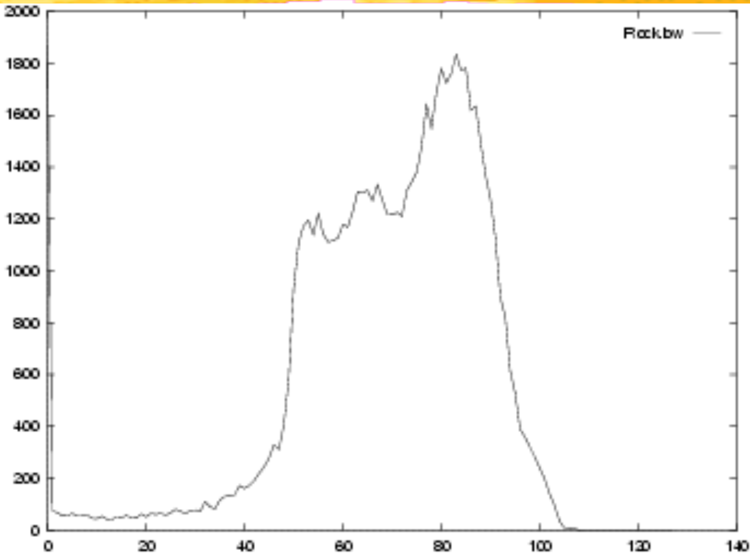
Corresponding histograms

Choosing optimal thresholds is a difficult optimization problem.

NO VALID THRESHOLD



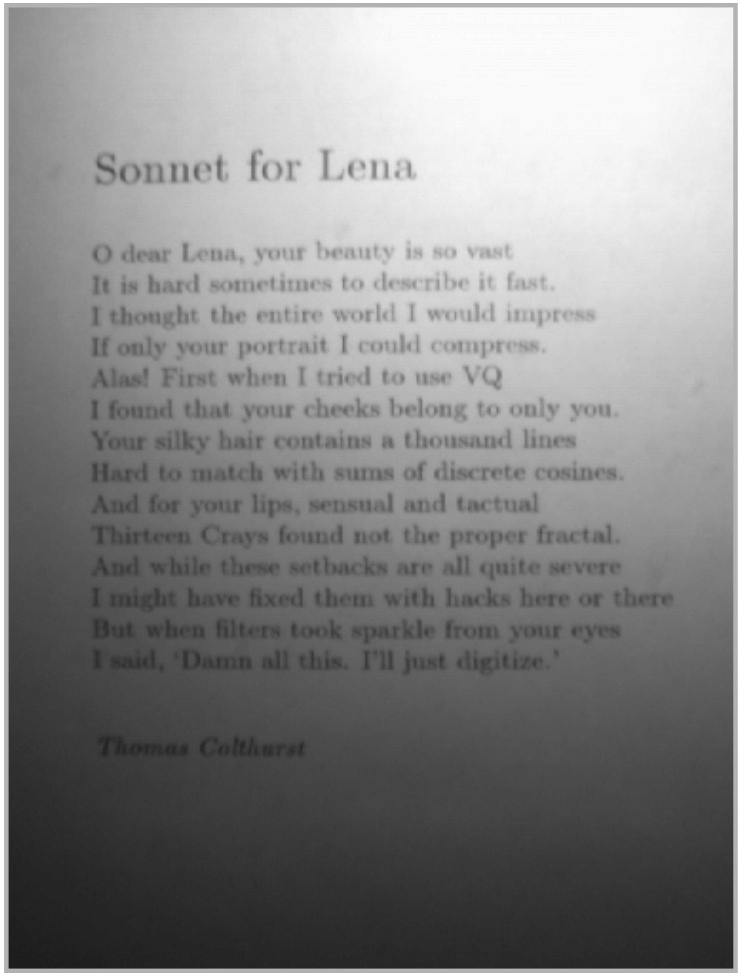
NO VALID THRESHOLD



WEAKNESSES

- Histograms do not account for neighborhood relationships.
- Thresholds are hard to find.
- Some edges can have gray levels on both sides that belong to the same histogram peak.

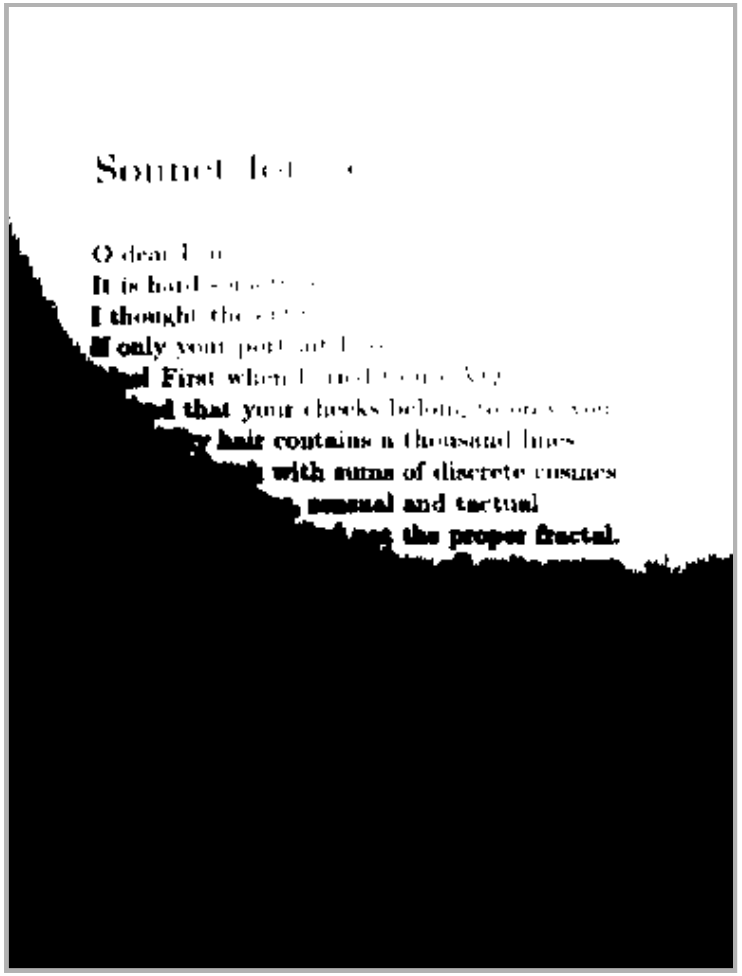
ILLUMINATION PROBLEMS



Sonnet for Lena

O dear Lena, your beauty is so vast
It is hard sometimes to describe it fast.
I thought the entire world I would impress
If only your portrait I could compress.
Alas! First when I tried to use VQ
I found that your cheeks belong to only you.
Your silky hair contains a thousand lines
Hard to match with sums of discrete cosines.
And for your lips, sensual and tactual
Thirteen Crays found not the proper fractal.
And while these setbacks are all quite severe
I might have fixed them with hacks here or there
But when filters took sparkle from your eyes
I said, 'Damn all this. I'll just digitize.'

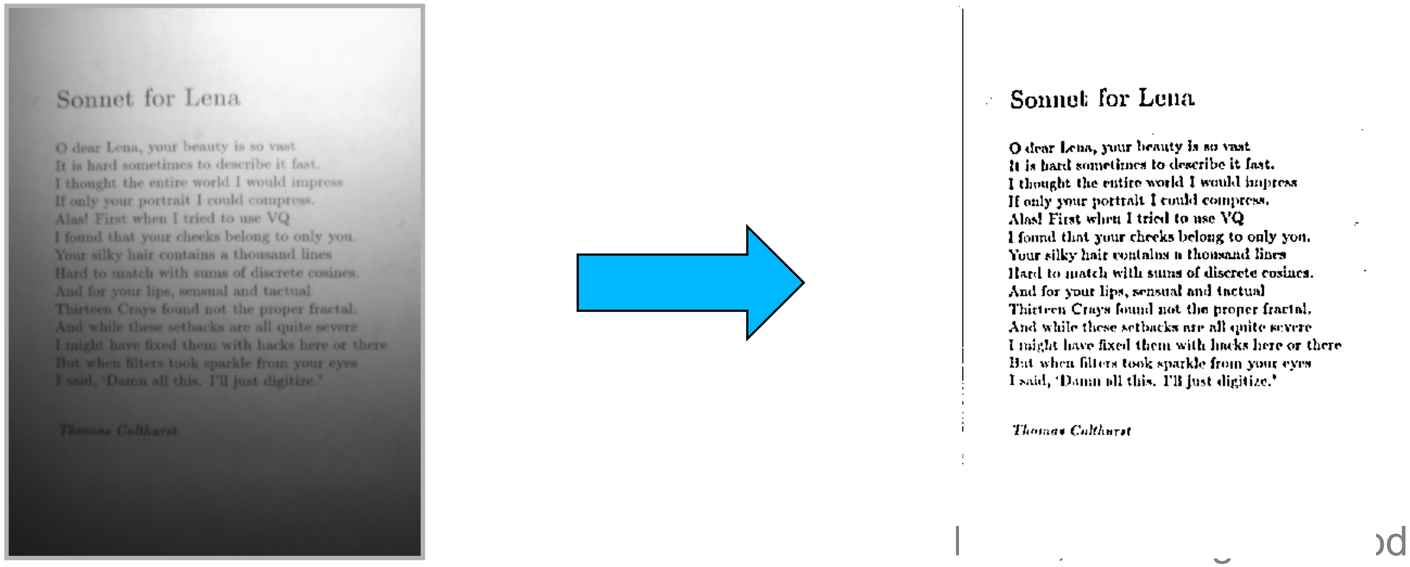
Thomas Colthurst



No global threshold -> Local ones are required.

Find threshold for local imaged

LOCAL THRESHOLDING

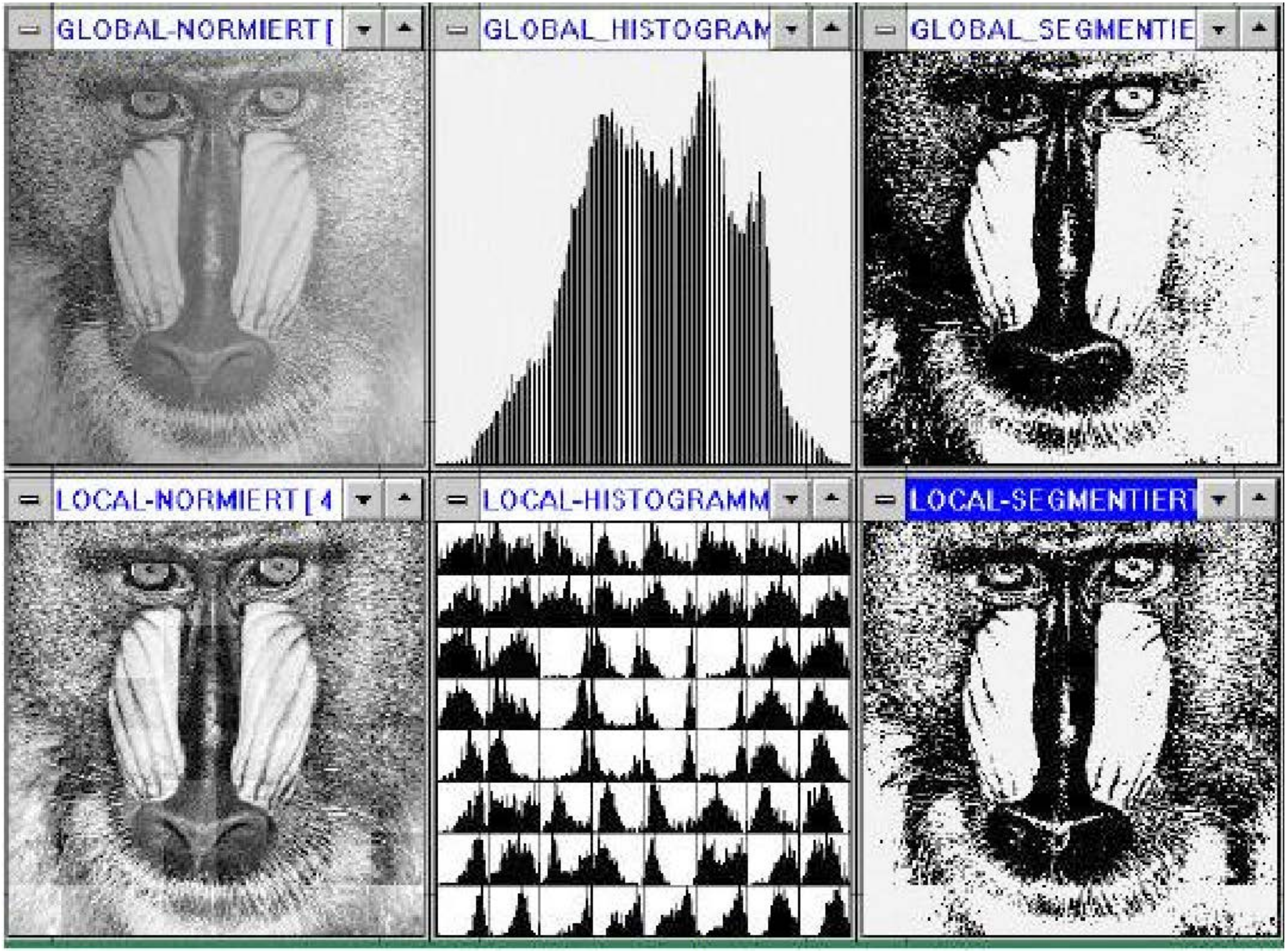

Sonnet for Lena

O dear Lena, your beauty is so vast
It is hard sometimes to describe it fast.
I thought the entire world I would impress
If only your portrait I could compress.
Alas! First when I tried to use VQ
I found that your cheeks belong to only you.
Your silky hair contains a thousand lines
Hard to match with sums of discrete cosines.
And for your lips, sensual and tactual
Thirteen Crays found not the proper fractal.
And while these setbacks are all quite severe
I might have fixed them with hacks here or there
But when filters took sparkle from your eyes
I said, 'Damn all this. I'll just digitize.'

Thomas Colthurst

- Examine statistically pixel values in local neighborhood around pixel to be thresholded.
- Use local statistic as threshold.
- Possibilities include mean, median, or mean of max and min value.

IMPROVED RESULTS

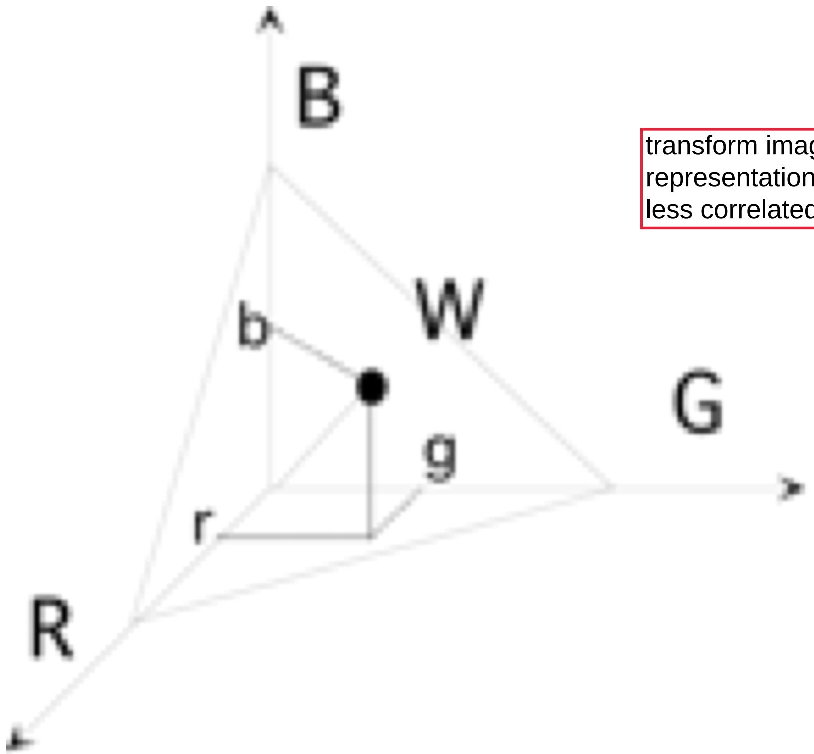


USING COLOR

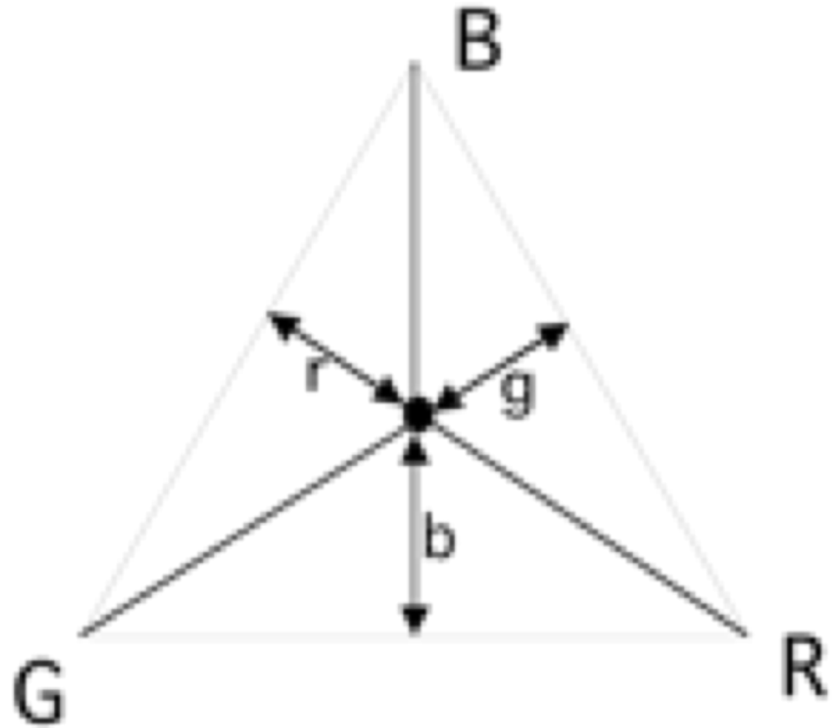
colors are highly correlated. so its still is not easy even with color information

RGB CHROMATICITY DIAGRAM

the colors are independent of the intensity



transform images to a representation that is less correlated

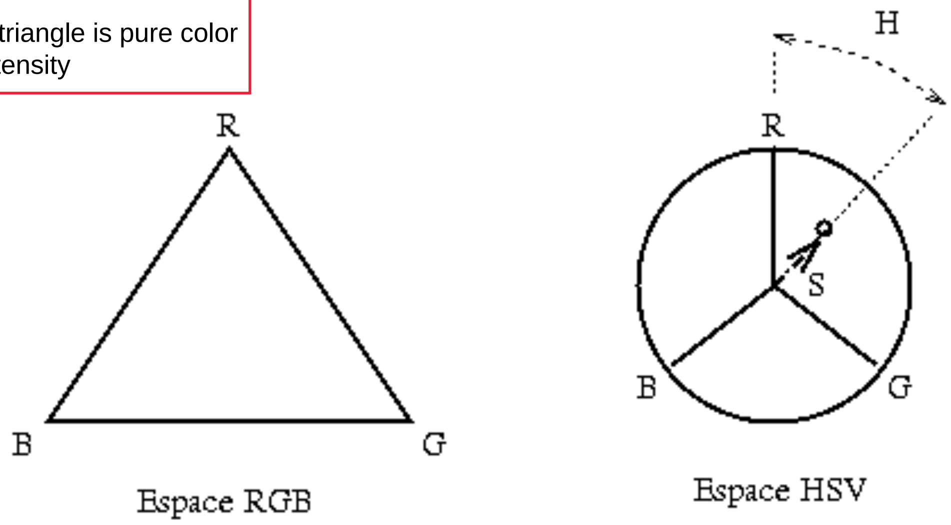


The Maxwell triangle involves projecting the colors in RGB space onto R+G+B=1 plane.

→ Chromaticity independently of luminance.

COLOR SPACE

Hue: Around the circle
Saturation: Center of triangle is white. Edge of triangle is pure color
Value: Not represented in triangle. pixel intensity



Hue 

Saturation 

Value 

HSV SPACE

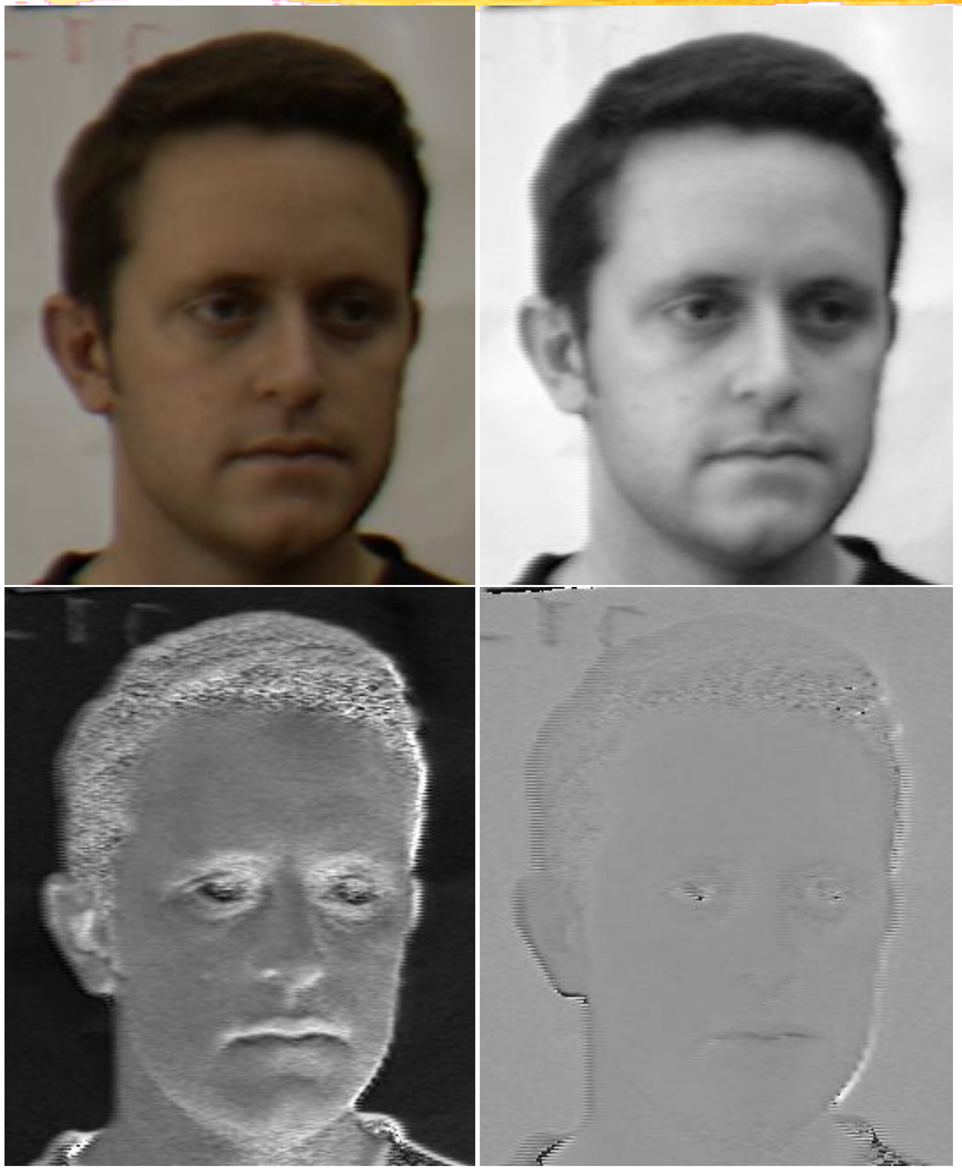
Normalized colors:

$$
\begin{aligned}
r &= \frac{R}{R+G+B} \\
g &= \frac{G}{R+G+B} \\
b &= \frac{B}{R+G+B}
\end{aligned}
$$

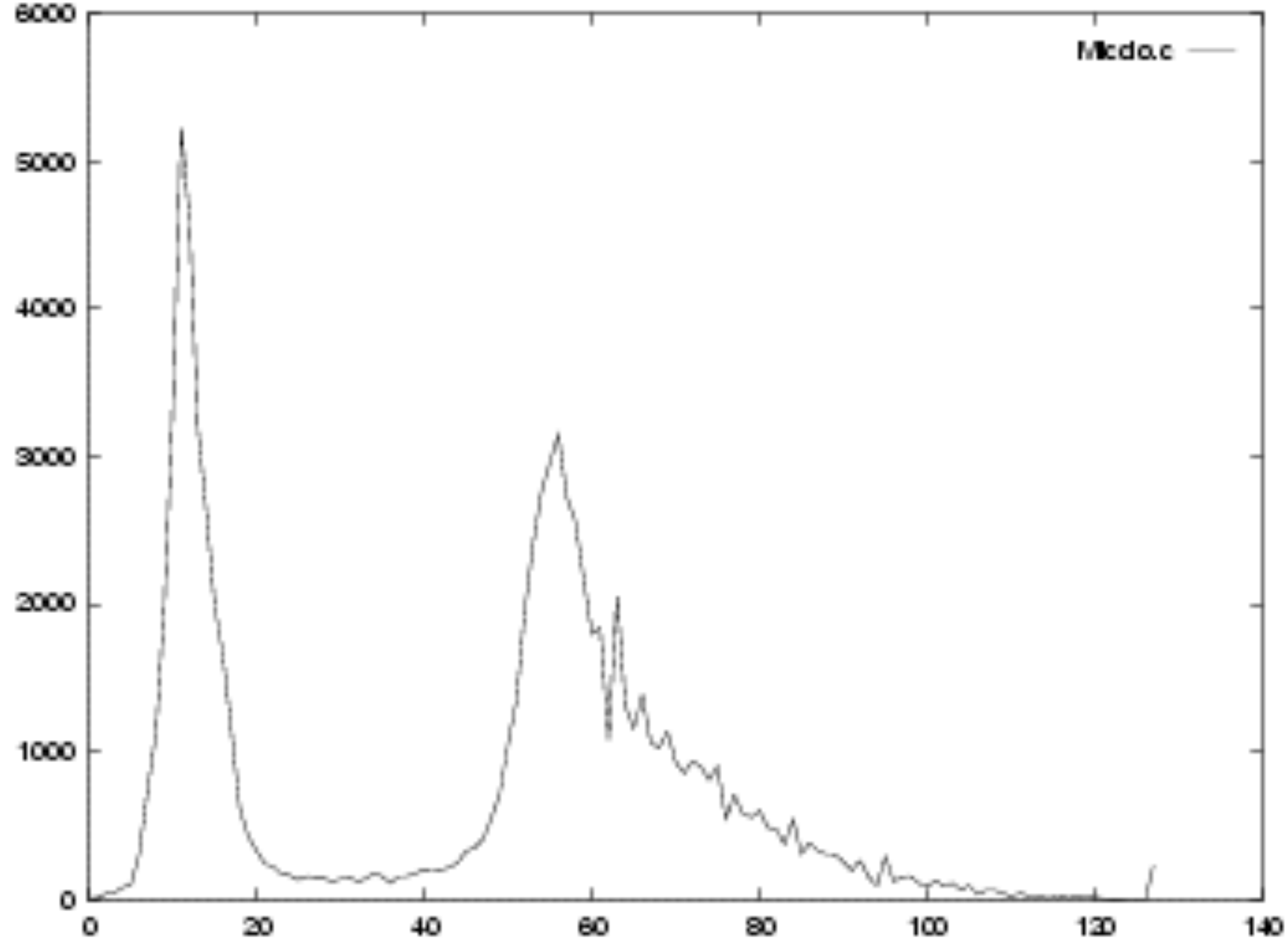
Hue/Saturation/Value

$$
\begin{aligned}
V &= R+G+B \\
S &= 1 - \frac{3\min(r,g,b)}{I} \\
H &= \arccos\left(\frac{0.5(2r-g-b)}{\sqrt{(r-g)^2+(r-g)(g-b)}}\right) \text{ if } b < g
\end{aligned}
$$

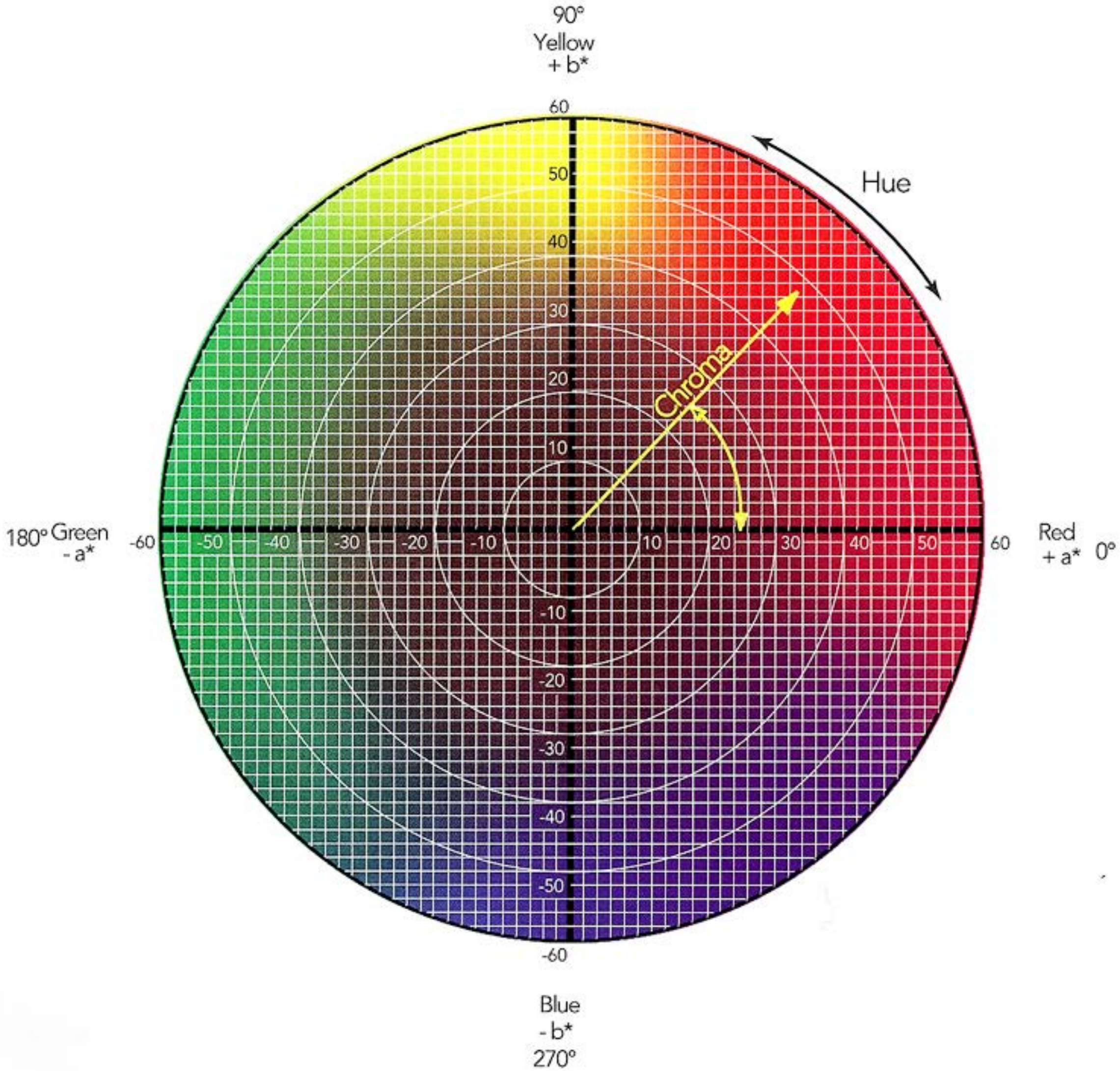
HSV IMAGES



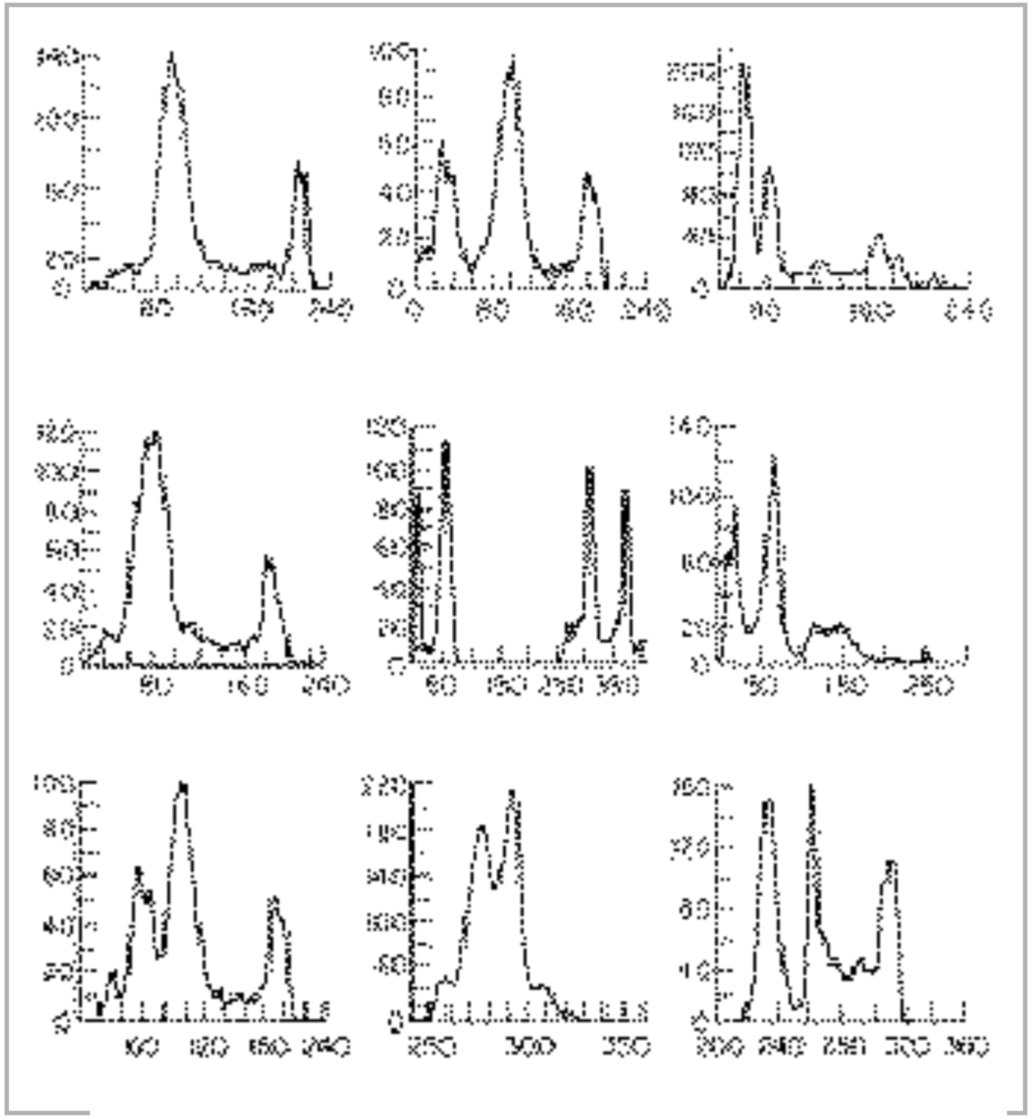
IMPROVED HISTOGRAM



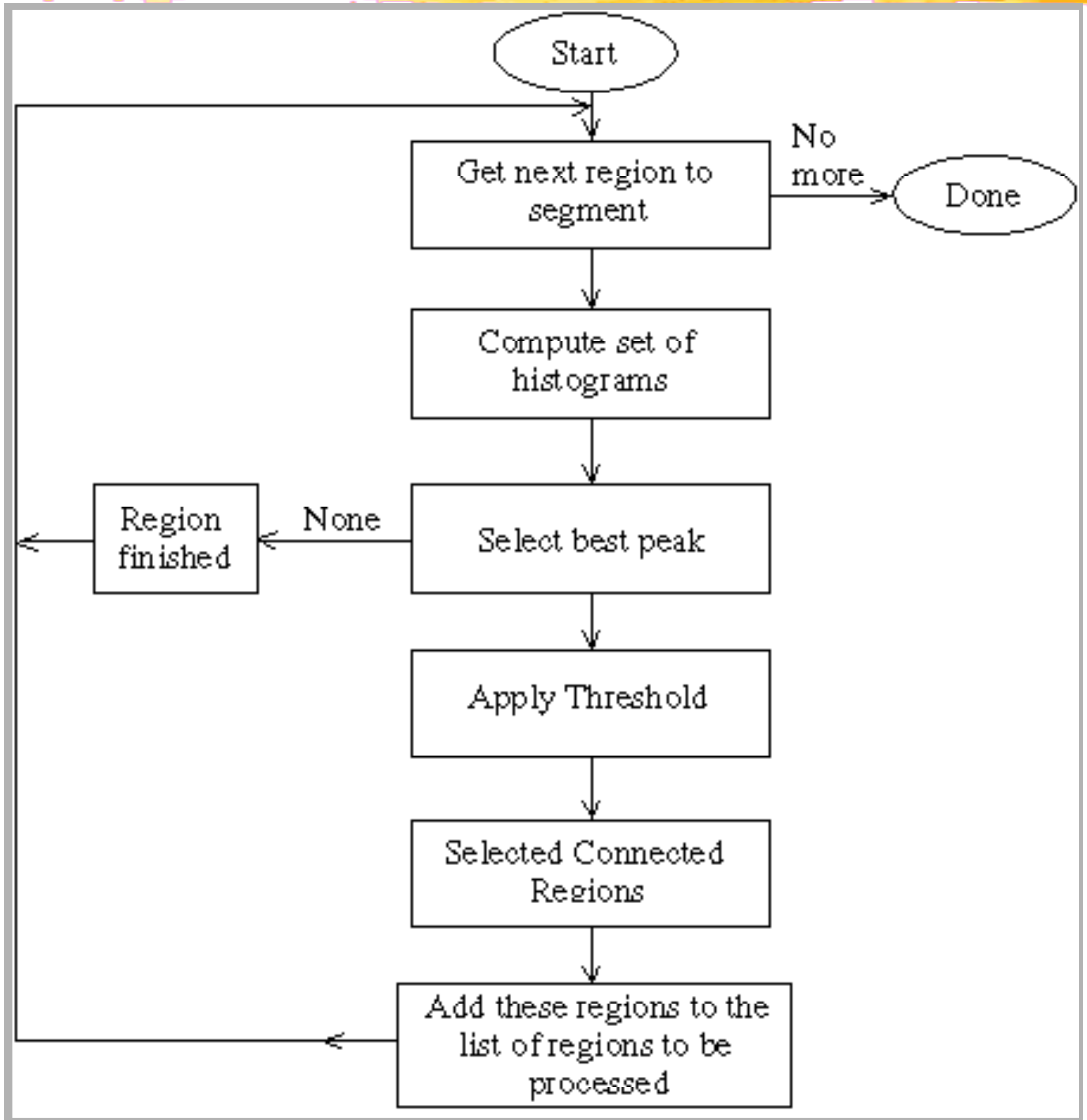
CIE LAB COLOR SPACE



MULTIPLE HISTOGRAMS

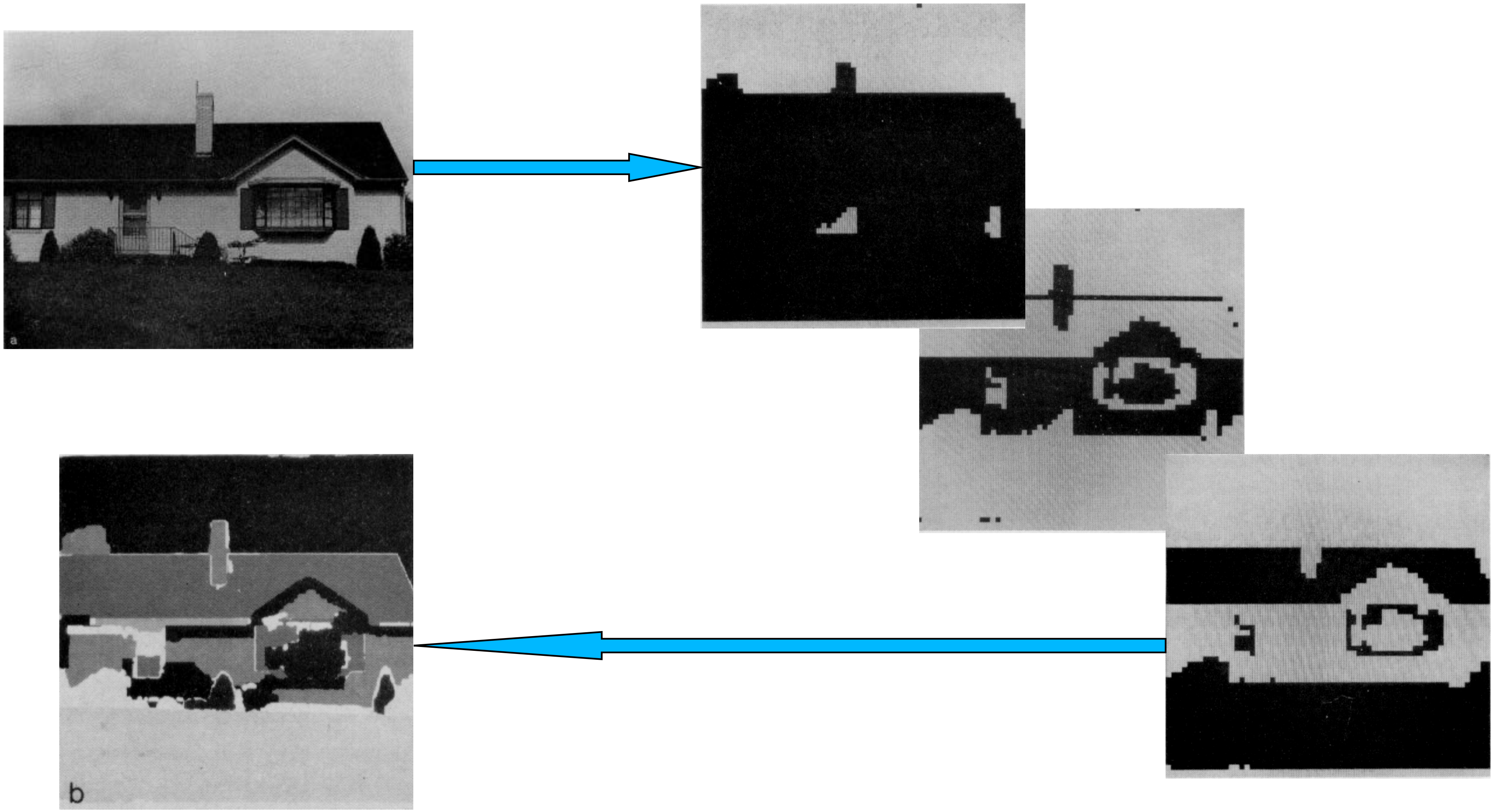


ALGORITHM



Start
Get next region to segment
No more
Done
Compute set of histograms
Select best peak
None
Region finished
Apply Threshold
Selected Connected Regions
Add these regions to the list of regions to be processed

HIEARCHICAL SEGMENTATION



COLOR SEGMENTATION

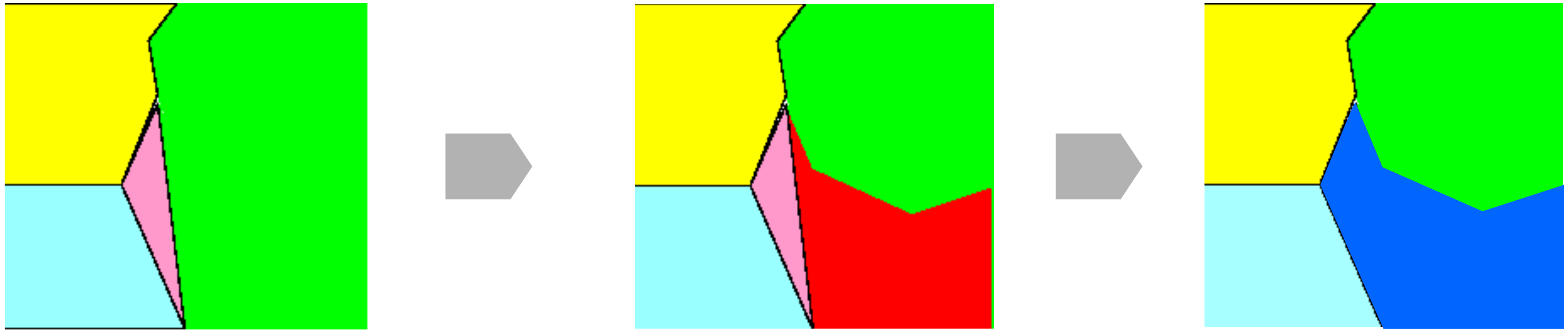
Ohlander & Price,
CGIP'78

MERGING REGIONS

Oversegmentation: Too many small regions.

Merging: Each region is merged with the most similar one until no regions smaller than a threshold remain.

SPLIT AND MERGE



Split, merge, and split again on the basis of a homogeneity criterion.

RECURSIVE MERGING

- Create an image partition.
- Compute an adjacency graph.
- For each image region:
  - Test its similarity with it neighbors.
  - Group the most similar ones.
- Iterate until no more regions can be grouped.

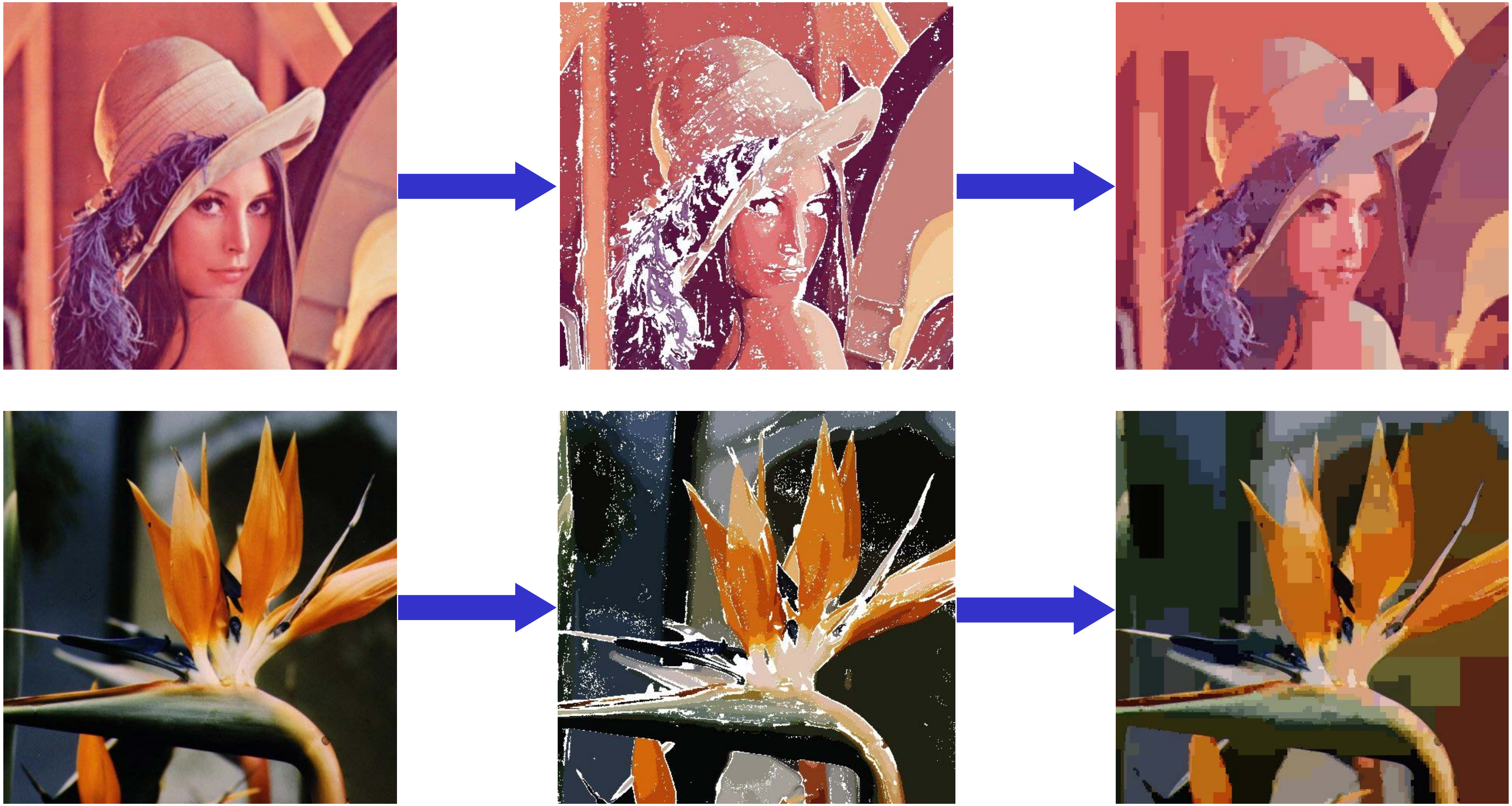
FISHER'S CRITERION

Discrimination between regions of different means and standard deviations can be done using

$$\frac{\mu_1 - \mu_2}{\sqrt{\sigma_1^2 + \sigma_2^2}} > \lambda$$

where λ is a threshold. If two regions have good separation in the means and low variance, then we can separate them.

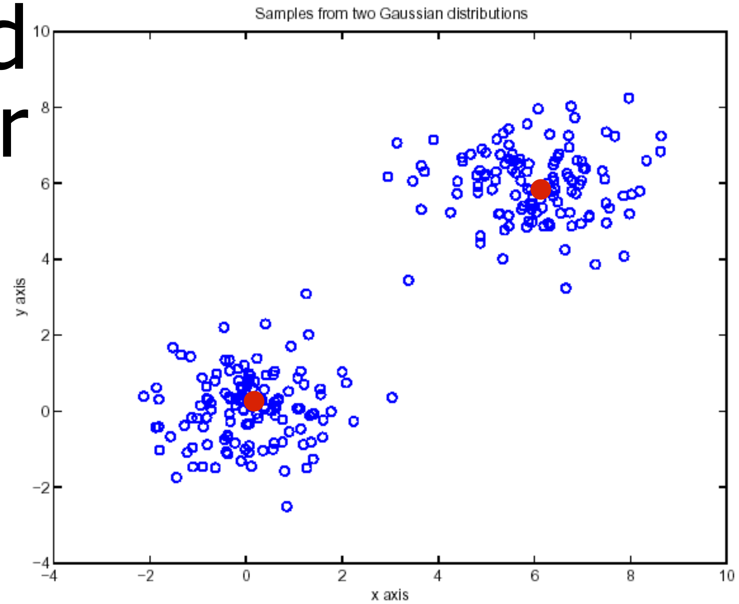
RESULTS



K-MEANS CLUSTERING

Assuming there are k regions and each one is described by a vector x_j, define:

- An objective function that measures the compactness of these regions.
- An optimization method that finds the most compact ones.



For a set of points in space, x_j is a coordinate vector. For black and white images, there are 1 or 3. For color images there are 3 or 5.

OBJECTIVE FUNCTION

$$\Phi(\text{clusters}, \text{data}) = \sum_{i \in \text{clusters}} \left\{ \sum_{j \in i\text{'th cluster}} (\boldsymbol{x}_j - \boldsymbol{c}_i)^T (\boldsymbol{x}_j - \boldsymbol{c}_i) \right\}$$

If the allocation of points to clusters were known, we could compute the best centers easily.

But there are far too many combinations for an exhaustive search.

→ Define an algorithm that alternates

- Assume centers are known, allocate points
- Assume allocation is known, compute centers

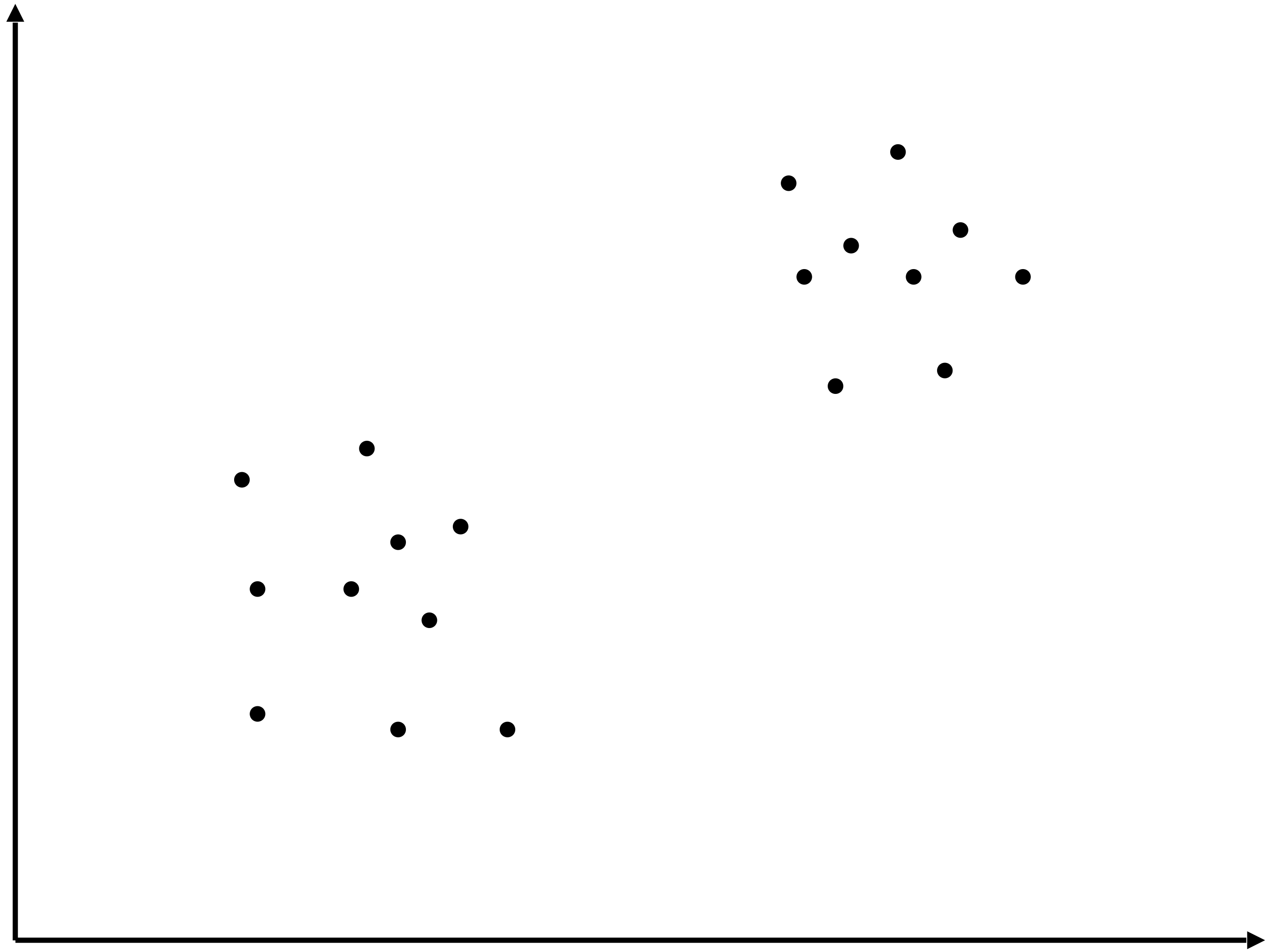
ALGORITHM

- Choose, for example randomly, k points that will serve as cluster centers.

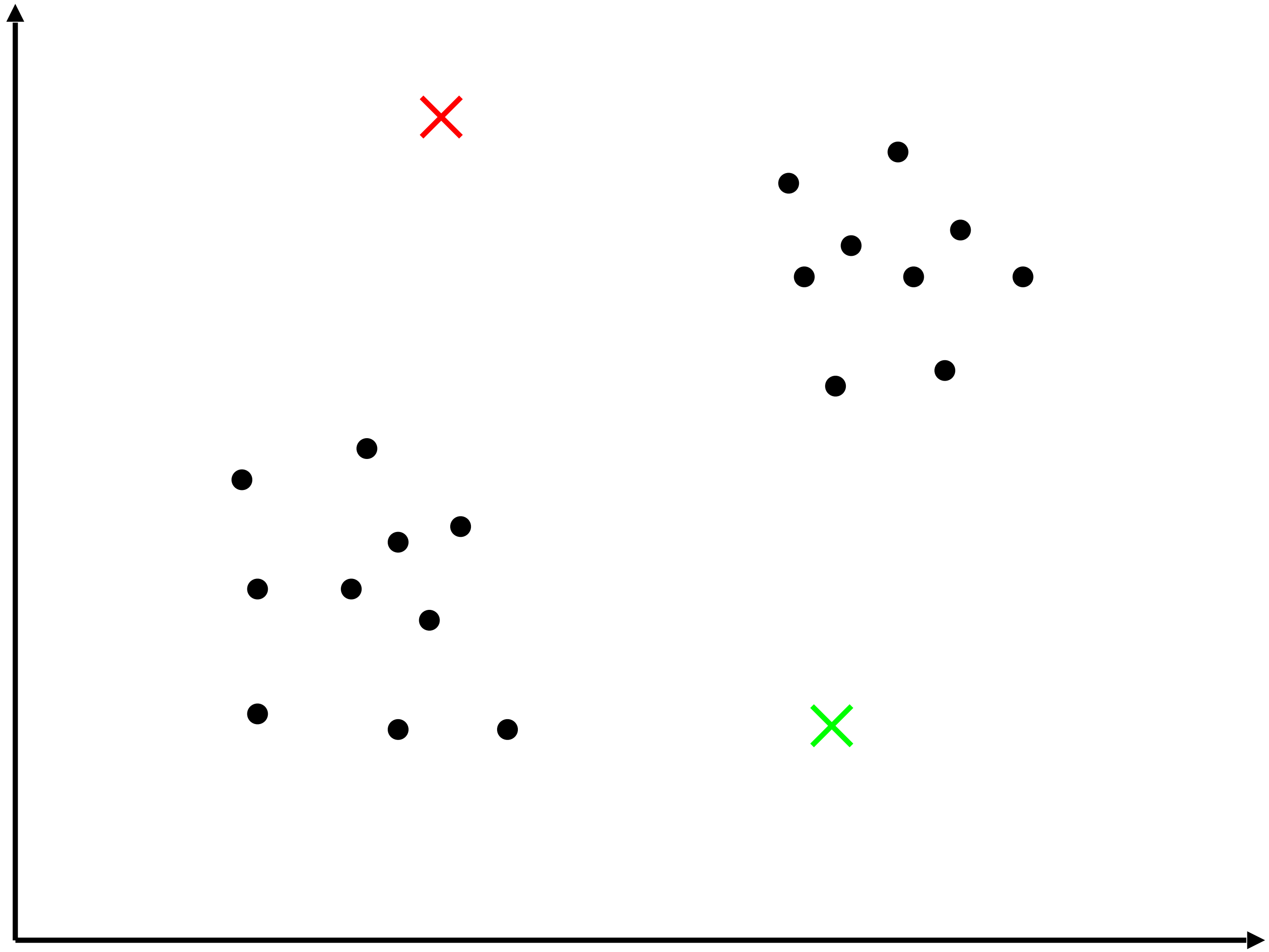
- Until their positions stabilize:
  1. Associate each pixel to the cluster whose center is closest;
  2. Recompute the centers by averaging the elements of each cluster.

- Extract connected components.

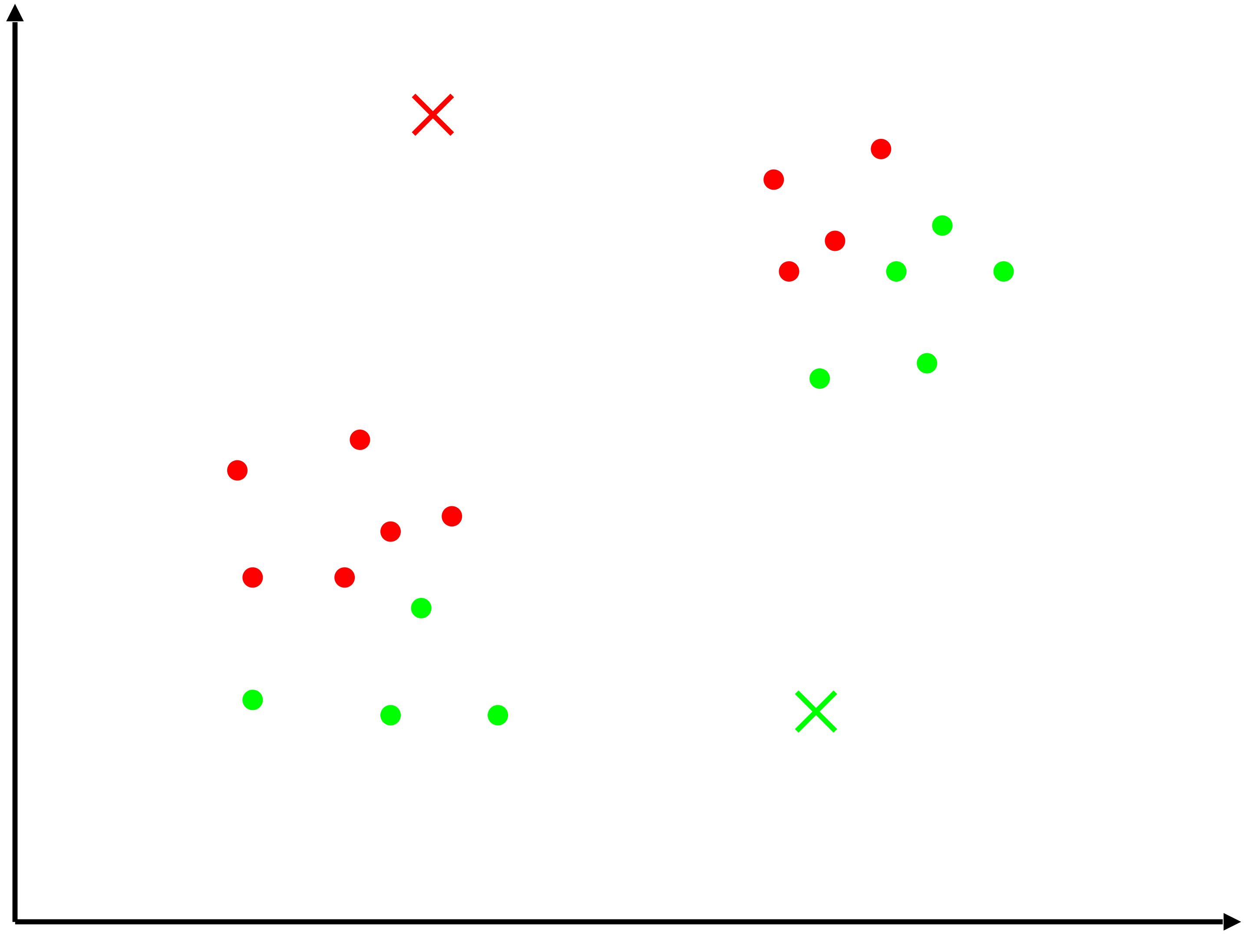
K-MEANS CLUSTERING



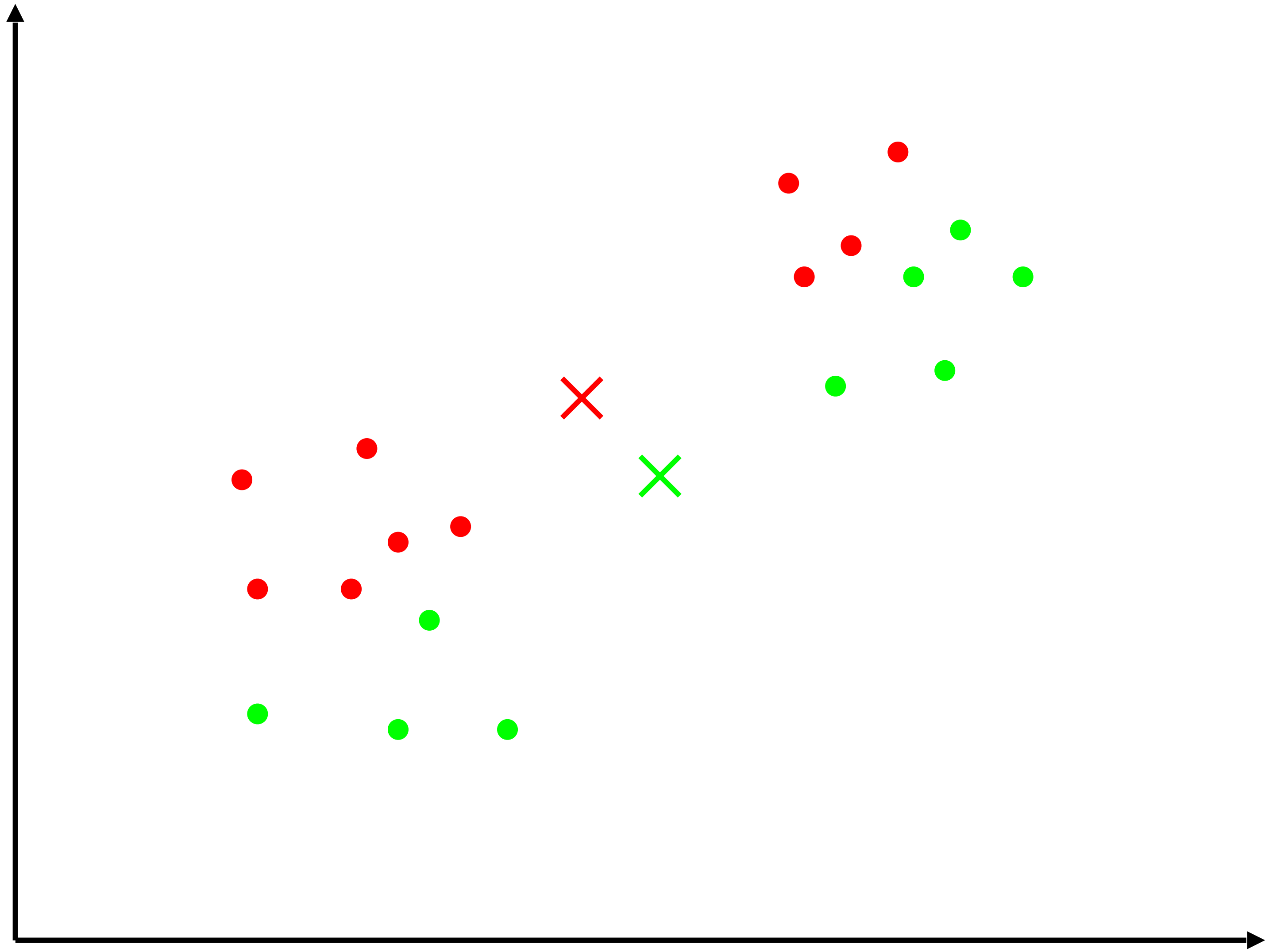
K-MEANS CLUSTERING



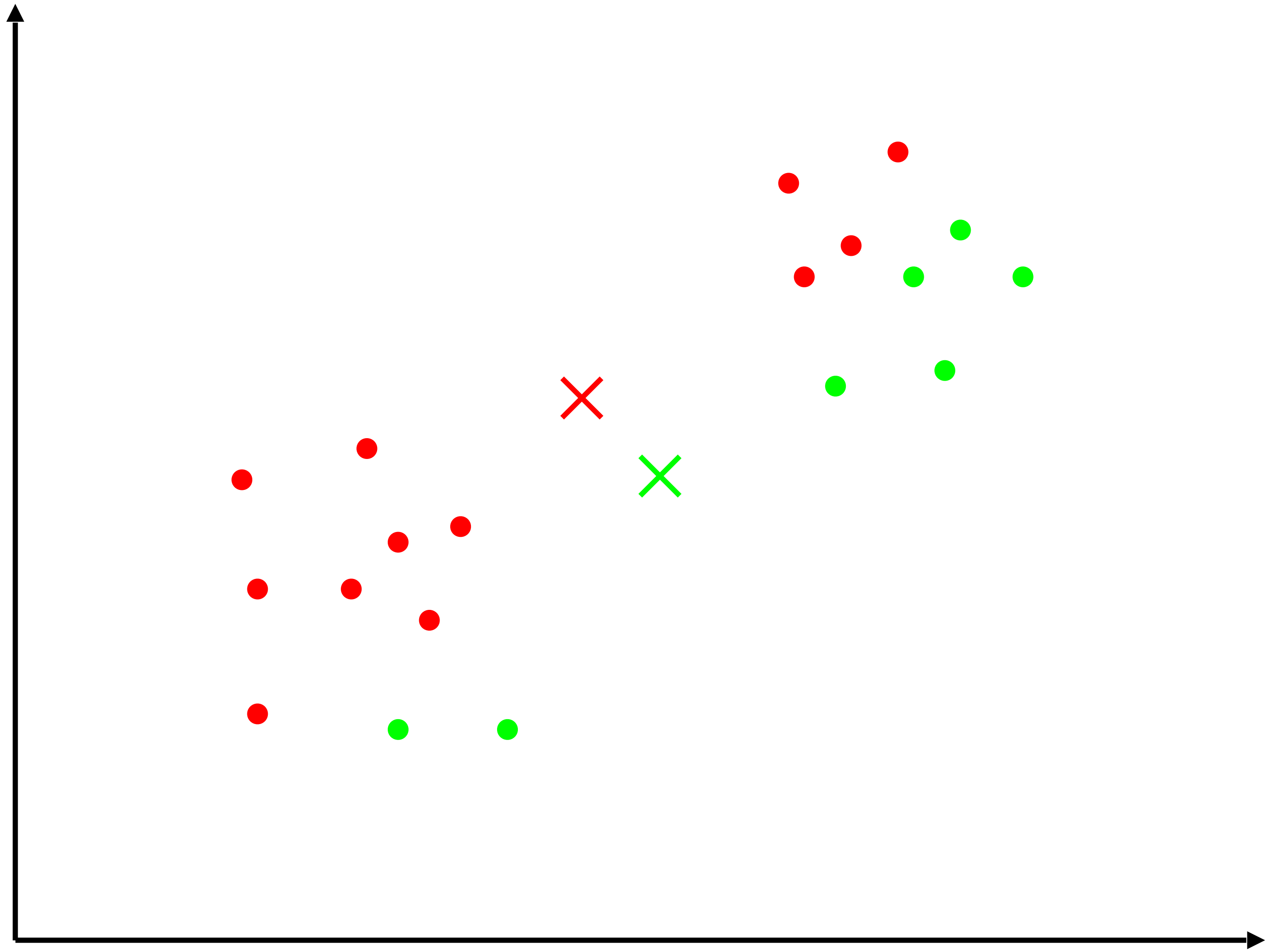
K-MEANS CLUSTERING



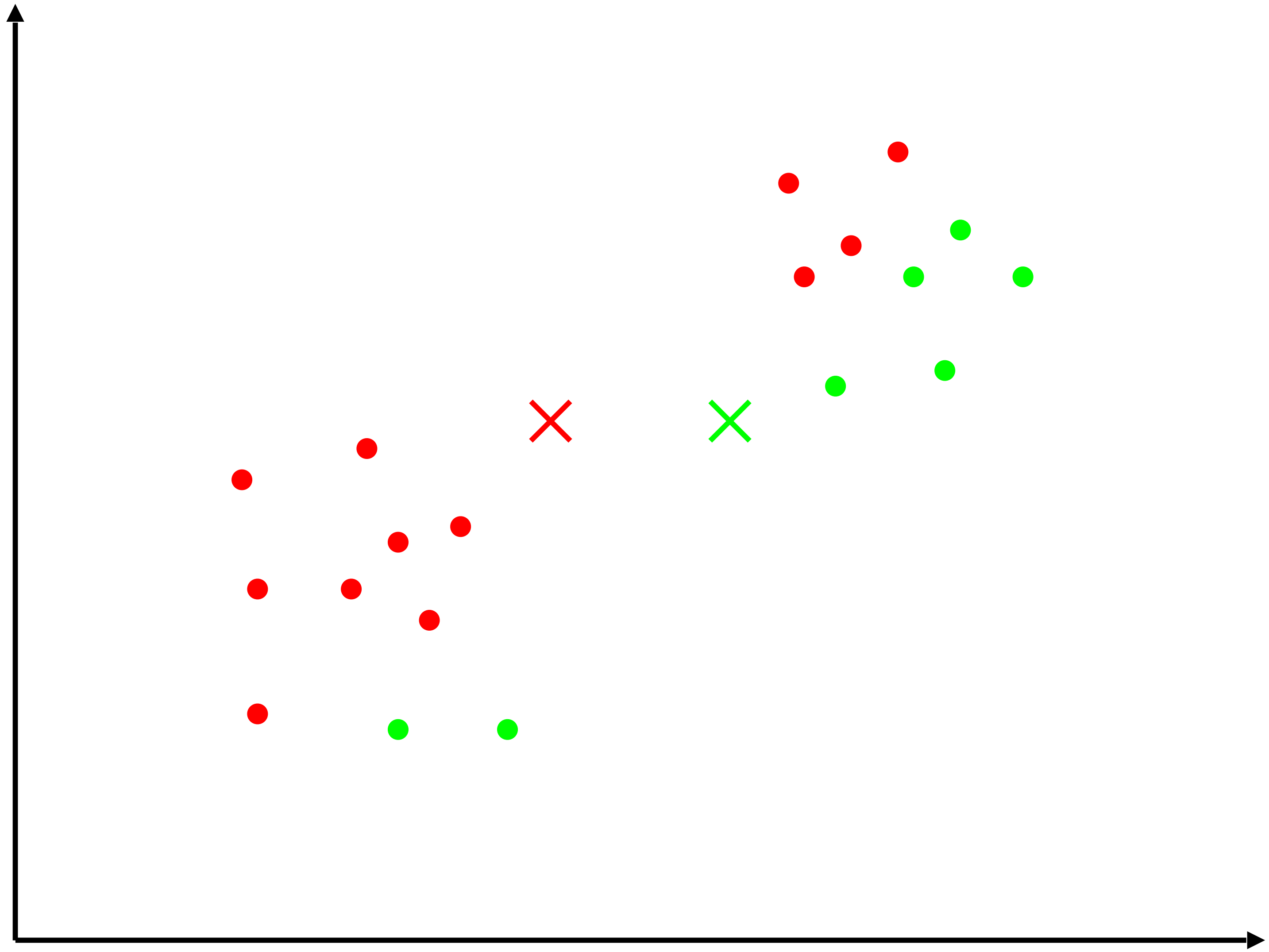
K-MEANS CLUSTERING



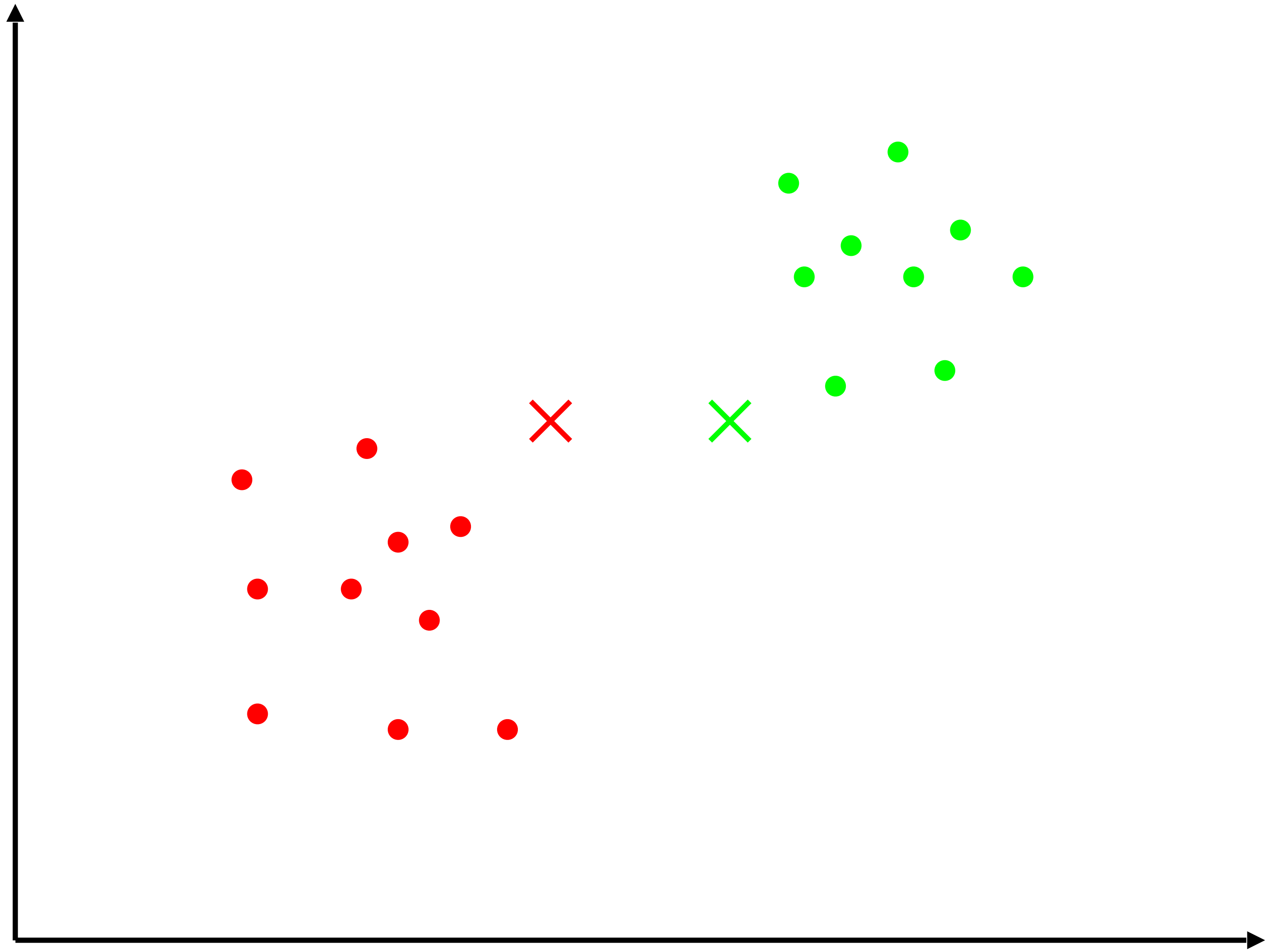
K-MEANS CLUSTERING



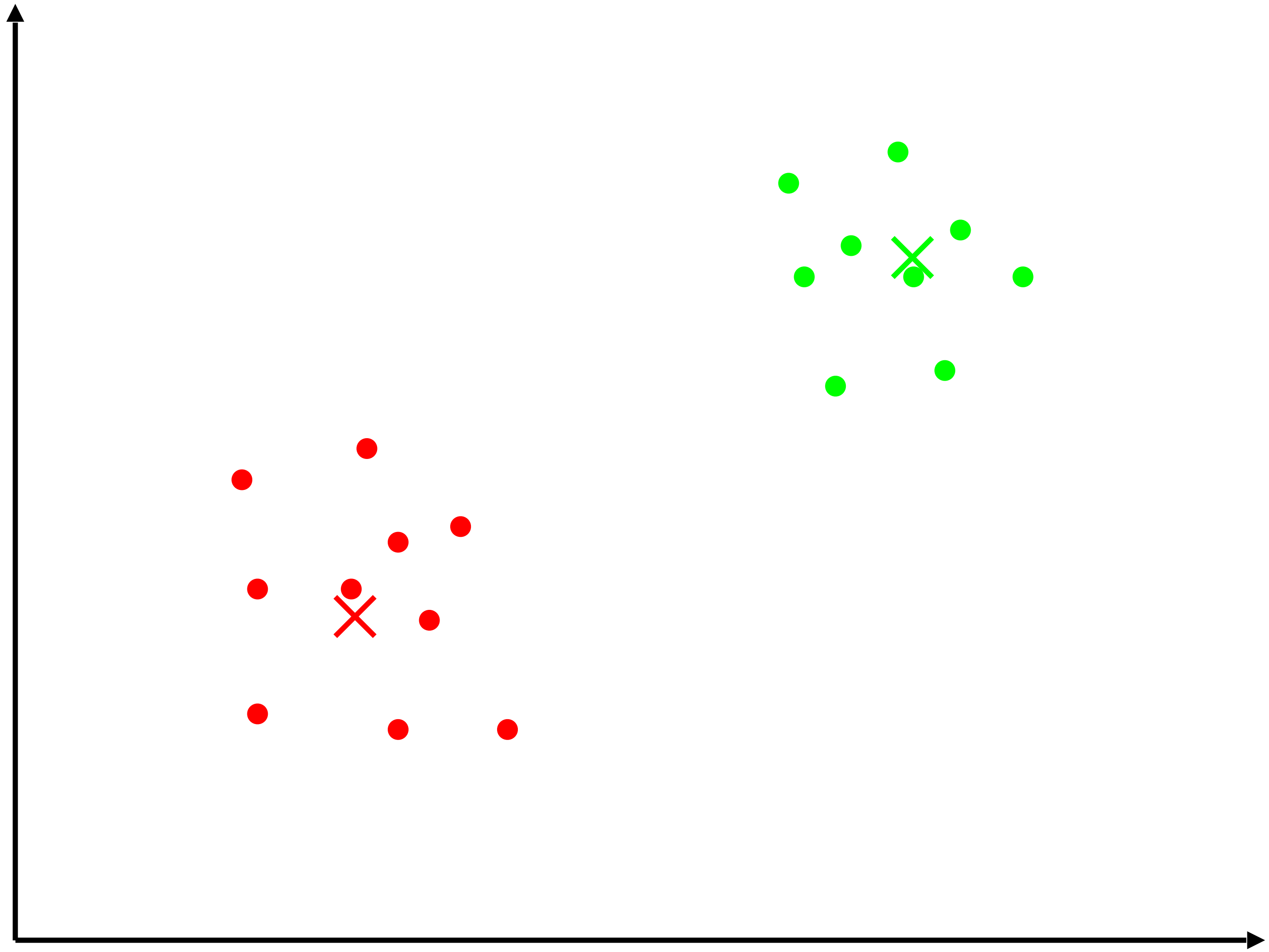
K-MEANS CLUSTERING



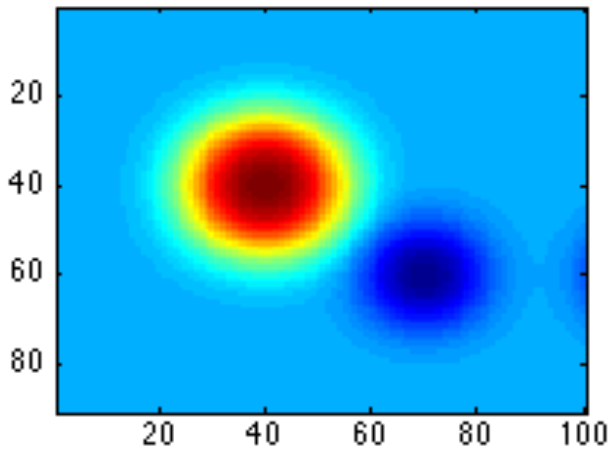
K-MEANS CLUSTERING

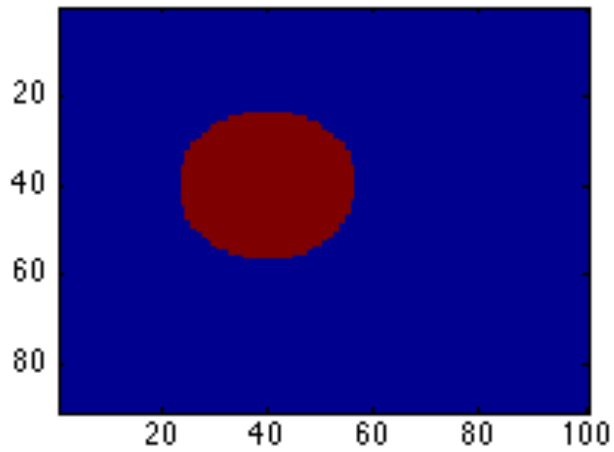


K-MEANS CLUSTERING

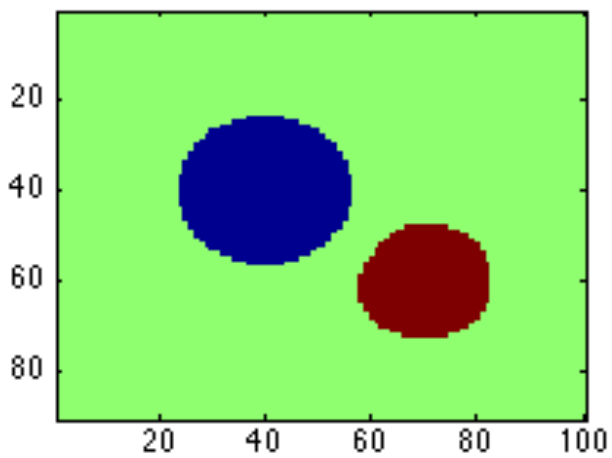


SYNTHETIC CASE

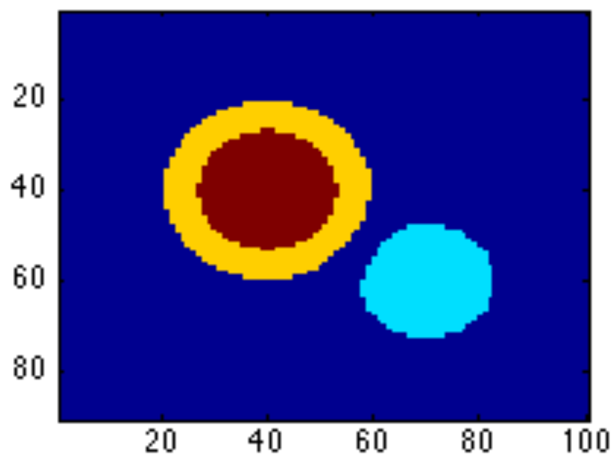




K=2

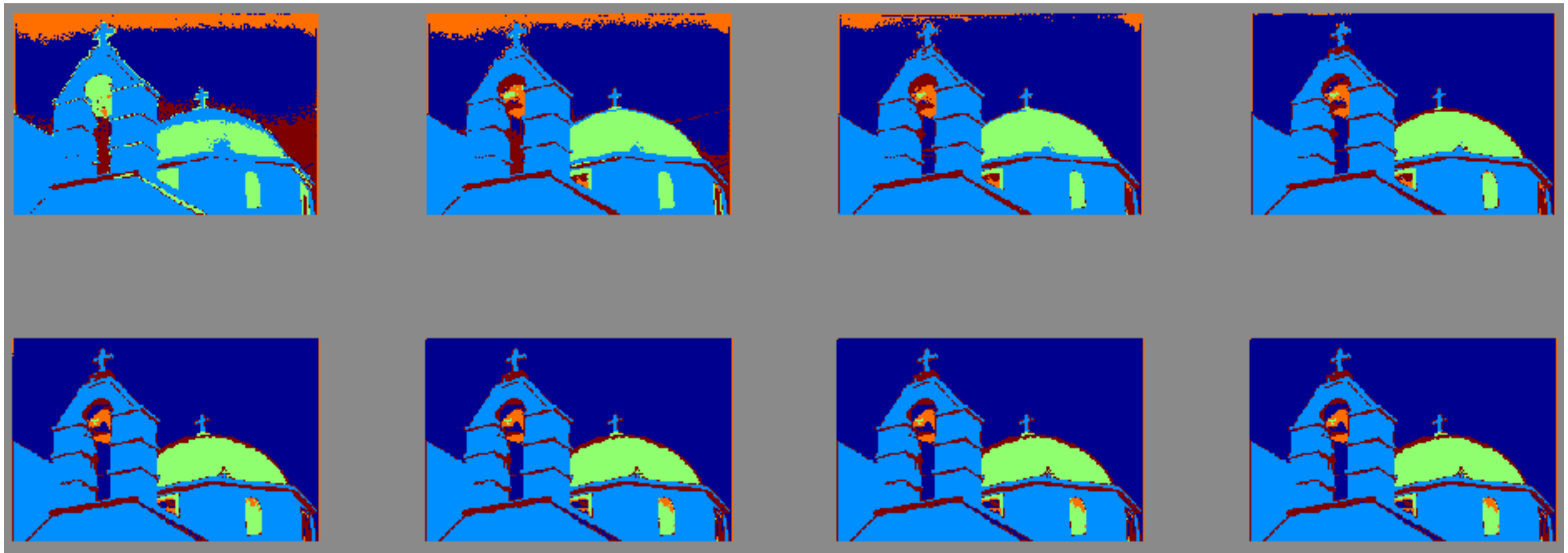


K=3

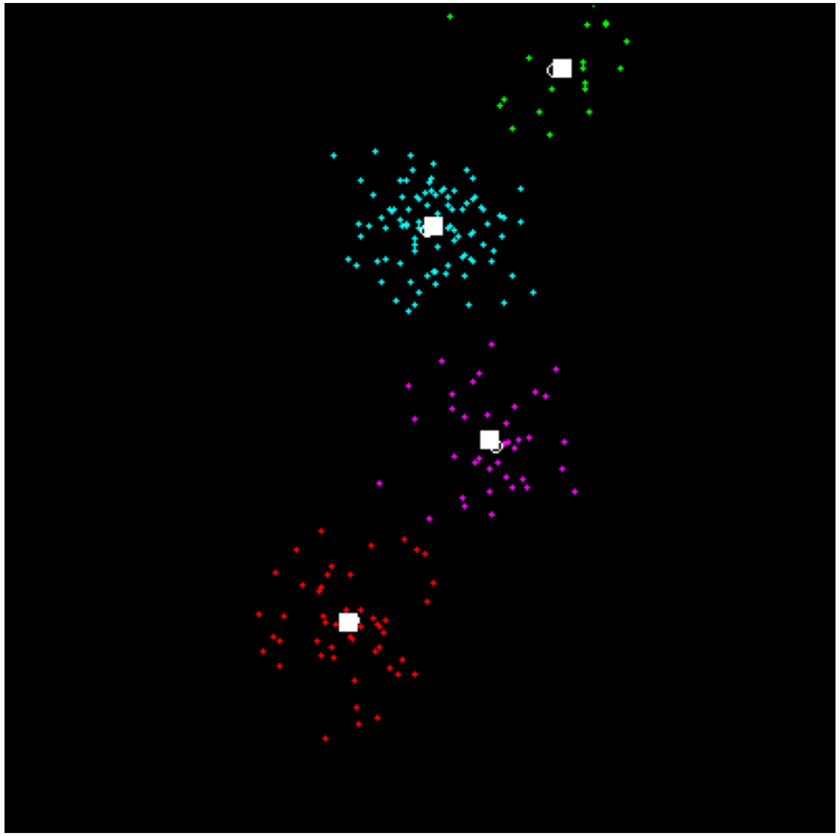


K=4

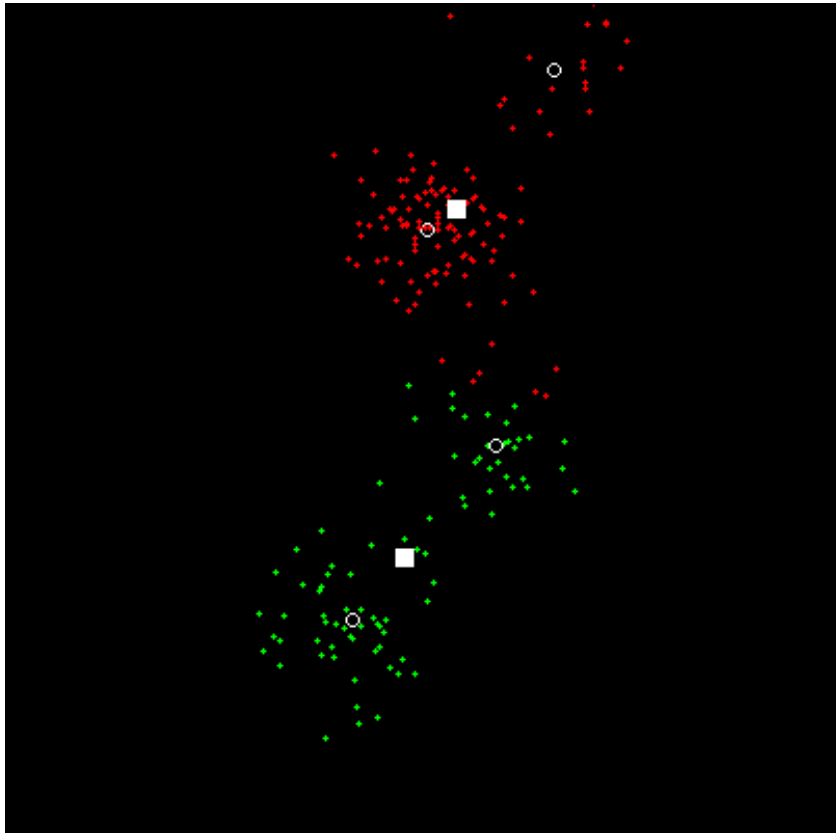
8 ITERATIONS FOR K=5



INITIAL CONDITIONS MATTER

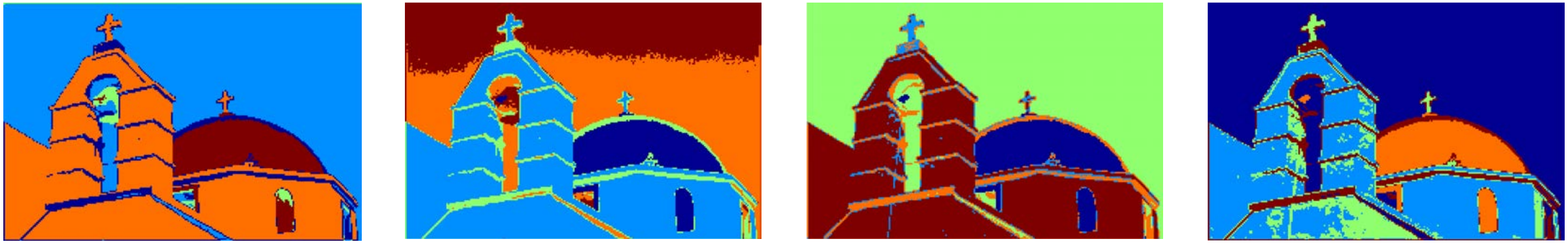


Initially, the points are assigned randomly to each one of the clusters.

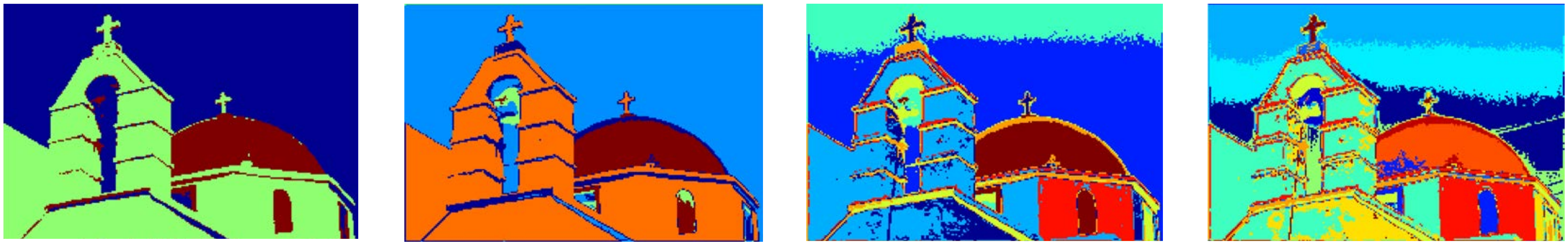


Initially, the points are assigned to the closest cluster.

K-MEANS RESULTS

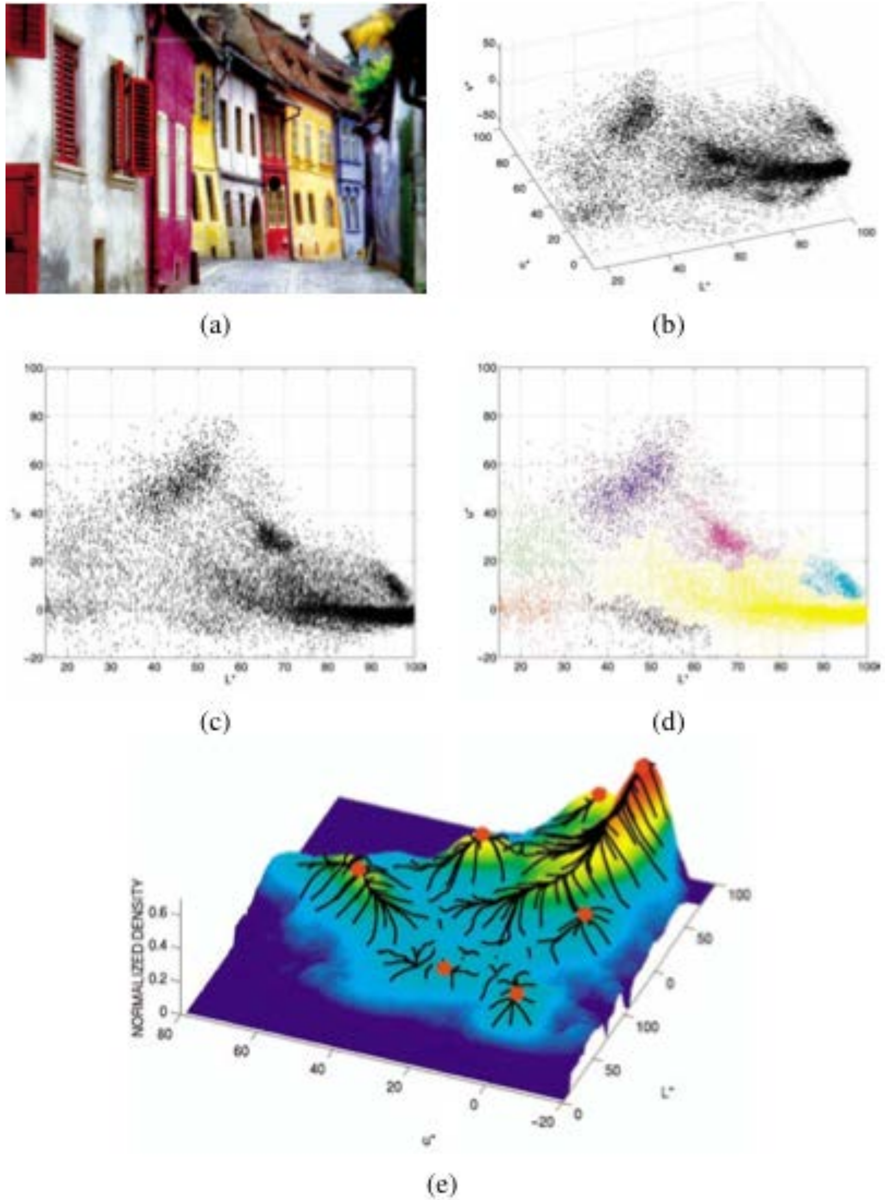


Random Initialization



K=3 K=5 K=8 K=15

FROM IMAGES TO PROBABILITY DENSITY FUNCTION



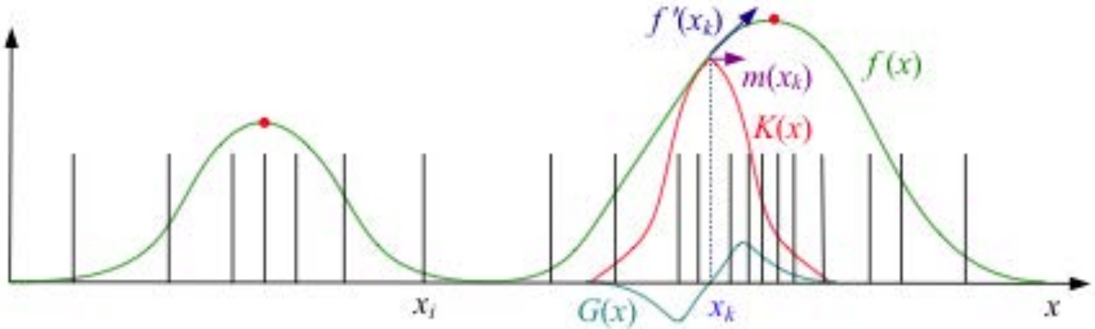
- Image pixels can be thought of samples of a probability distribution function.
- Regions then become major peaks in that probability density function.
- A way to estimate this probability density function is needed.

PROBABILITY DENSITY FUNCTION AND ITS GRADIENT

original formulation:
1/n * sum(x) - x_o

new formulation:
[1/sum(px)] * x * sum(px) - x_o

$$f(\mathbf{x}) = \sum_i K\left(\frac{\left\|\mathbf{x}-\mathbf{x}_i\right\|^2}{h^2}\right)$$

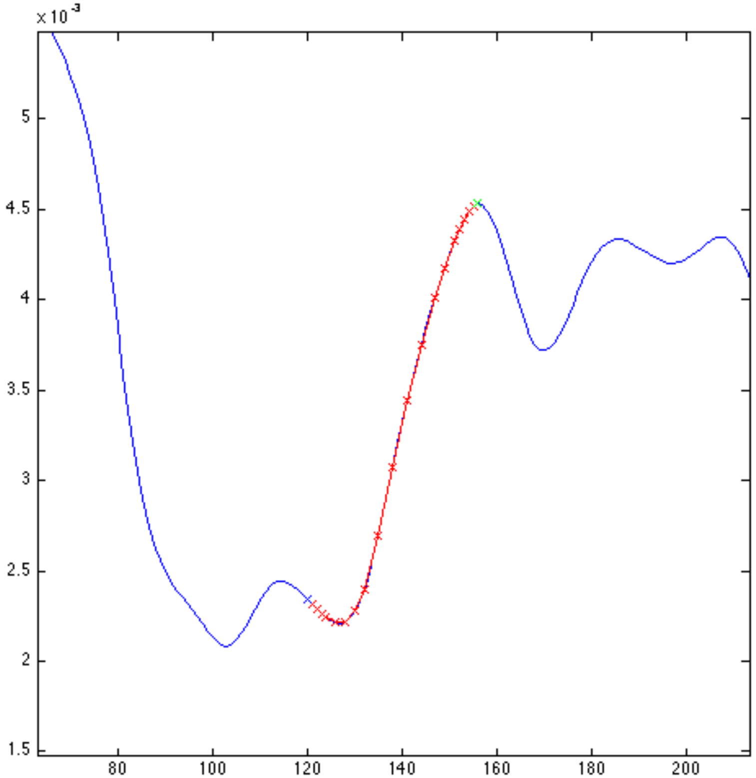


$$\nabla f(\mathbf{x}) = \sum_i (\mathbf{x}_i - \mathbf{x}) G(\mathbf{x}-\mathbf{x}_i) \text{ with } \mathrm{G}(\mathbf{x}-\mathbf{x}_i) = -K'\left(\frac{\left\|\mathbf{x}-\mathbf{x}_i\right\|^2}{h^2}\right)$$

$$= \left[\sum_i G(\mathbf{x}-\mathbf{x}_i)\right]\mathbf{m}(\mathbf{x}) \text{ with } \mathbf{m}(\mathbf{x}) = \frac{\sum_i \mathbf{x}_i G(\mathbf{x}-\mathbf{x}_i)}{\sum_i G(\mathbf{x}-\mathbf{x}_i)} - \mathbf{x}$$

- **m(x)** is known as the mean shift because it is the difference between the weighted mean of the values of the neighbors of **x** and that of **x** itself.
- **m(x)** is the direction of steepest ascent.

1D MEAN-SHIFT PROCEDURE



$$\mathbf{y}_{k+1} = \mathbf{y}_k + \mathbf{m}(\mathbf{y}_k) = \frac{\sum_i \mathbf{x}_i G(\mathbf{y} - \mathbf{x}_i)}{\sum_i G(\mathbf{y} - \mathbf{x}_i)}$$

$$k_e(r) = \max(0, 1 - r) \Rightarrow g(r) = \begin{cases} 0 \text{ if } r < 1 \\ 1 \text{ otherwise} \end{cases}$$

$$k_n(r) = \exp(-r/2) \Rightarrow g(r) = \frac{1}{2}\exp(-r/2)$$

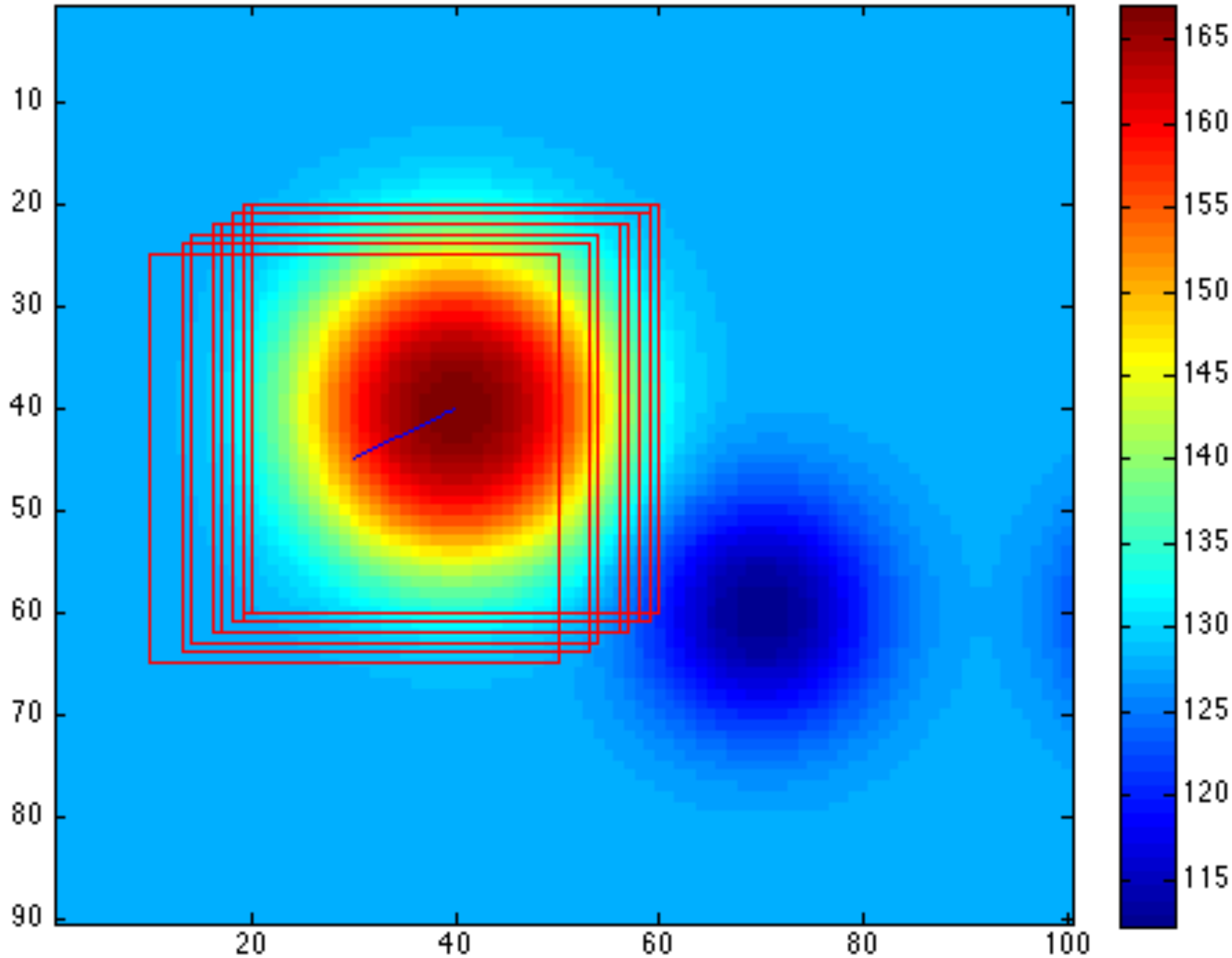
3D MEAN-SHIFT PROCEDURE

$$\mathbf{y}_{k+1} = \mathbf{y}_k + \mathbf{m}(\mathbf{y}_k) = \frac{\sum_i \mathbf{x}_i G(\mathbf{y} - \mathbf{x}_i)}{\sum_i G(\mathbf{y} - \mathbf{x}_i)}$$

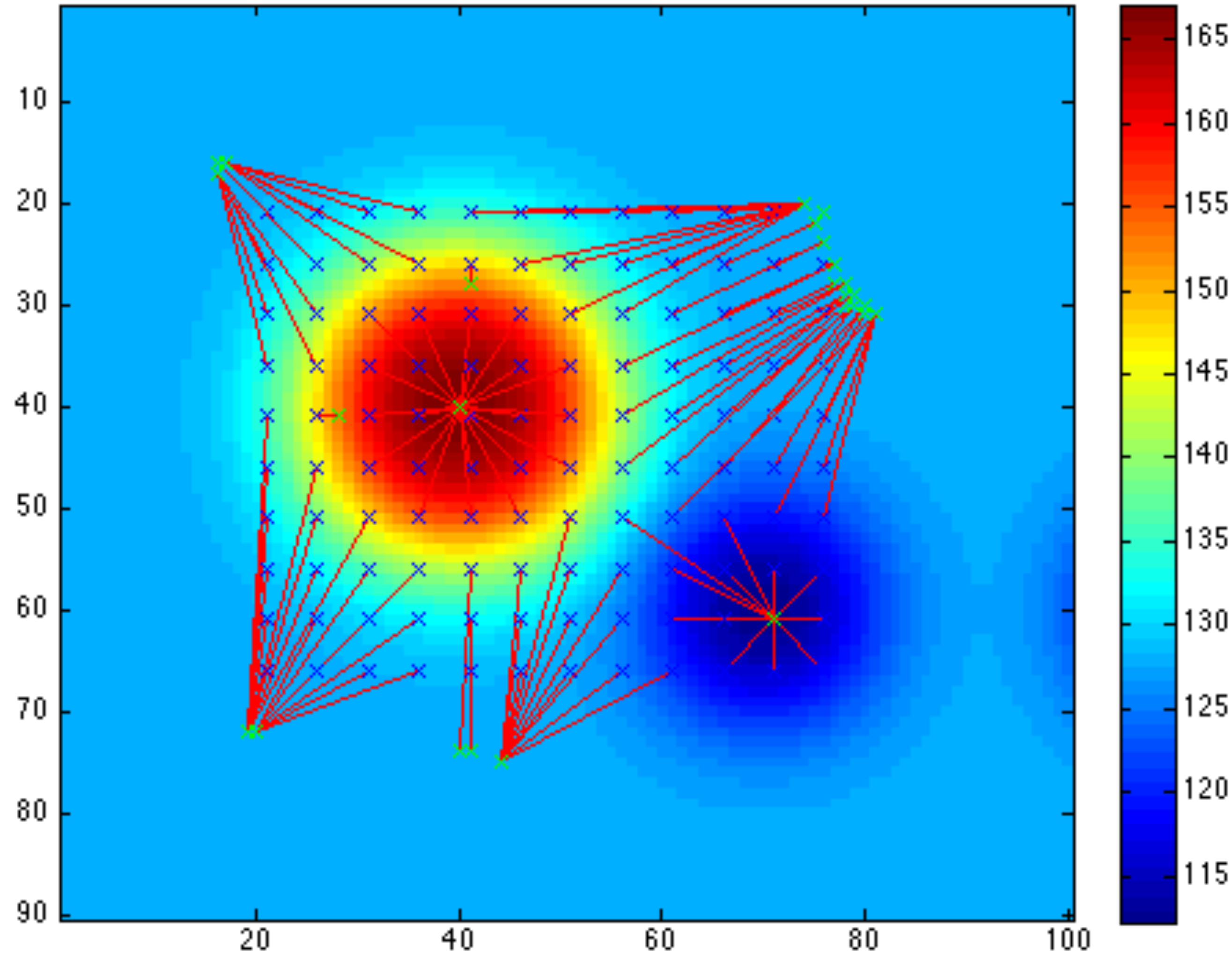
$$\mathbf{x} = \begin{bmatrix} u \\ v \\ g \end{bmatrix}$$

$$K(\mathbf{x}) = k_n\left(\frac{u^2 + v^2}{h_s^2}\right) k\left(\frac{g^2}{h_r^2}\right)$$

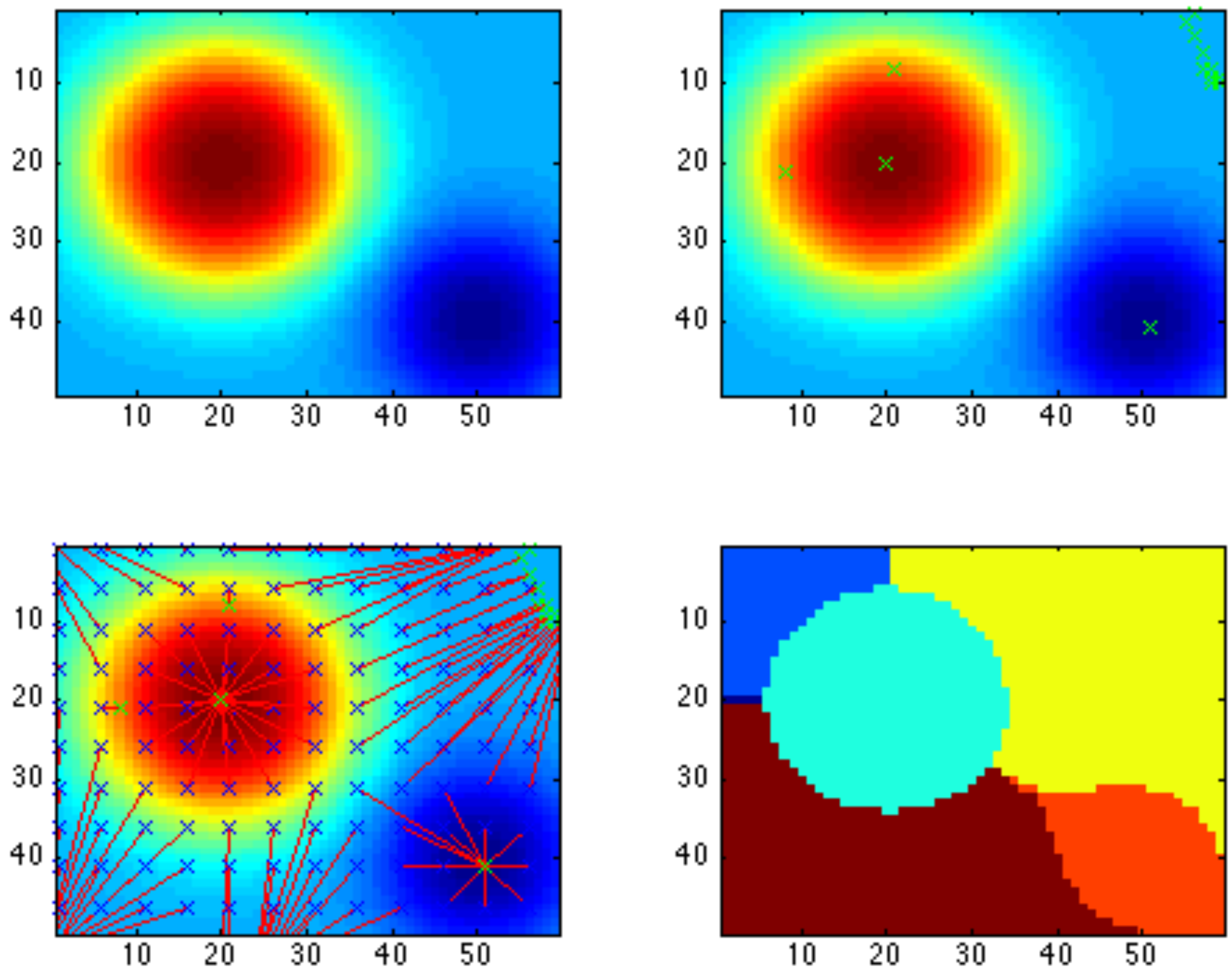
$$G(\mathbf{x}) = \exp\left(-\left(\frac{u^2 + v^2}{h_s^2} + \frac{g}{h_r^2}\right)\right)$$



MEAN SHIFT MODES



MEAN SHIFT CLUSTERING



5D MEAN-SHIFT

$$\mathbf{y}_{k+1} = \mathbf{y}_k + \mathbf{m}(\mathbf{y}_k) = \frac{\sum_i \mathbf{x}_i G(\mathbf{y} - \mathbf{x}_i)}{\sum_i G(\mathbf{y} - \mathbf{x}_i)}$$

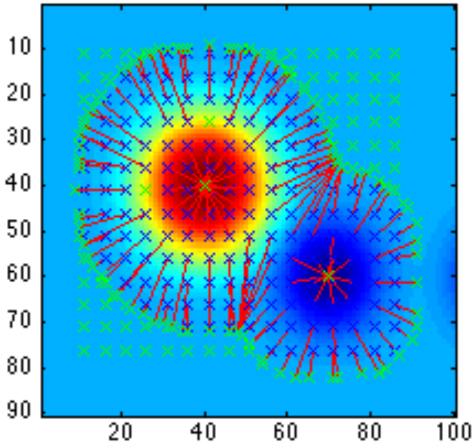
$$\mathbf{x} = \begin{bmatrix} u \\ v \\ R \\ G \\ V \end{bmatrix} \text{ or } \begin{bmatrix} u \\ v \\ L \\ a \\ b \end{bmatrix}$$

$$G(\mathbf{x}) = \exp(-(\frac{u^2 + v^2}{h_s^2} + \frac{R^2 + G^2 + B^2}{h_r^2}))$$

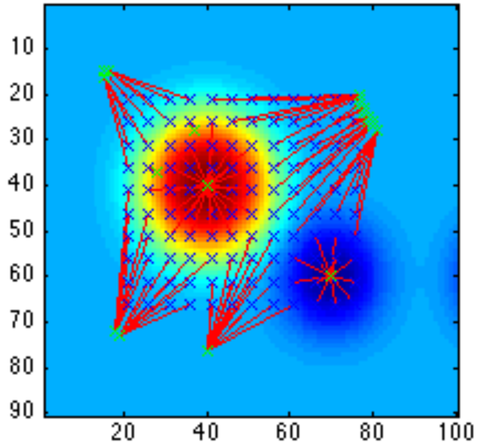
Ohlander & Price, CGIP'78

Comaniciu & Meers, PAMI'02

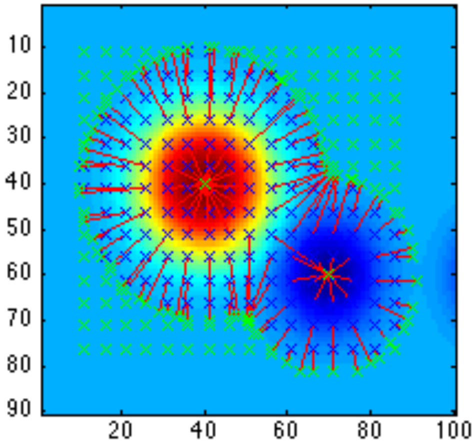
PARAMETERS



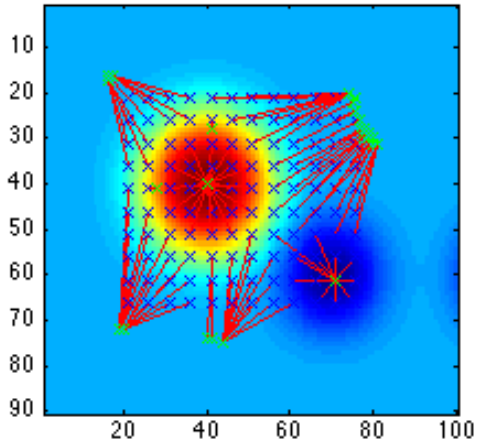
$h_s = 5,\ h_r = 5$



$h_s = 10,\ h_r = 5$

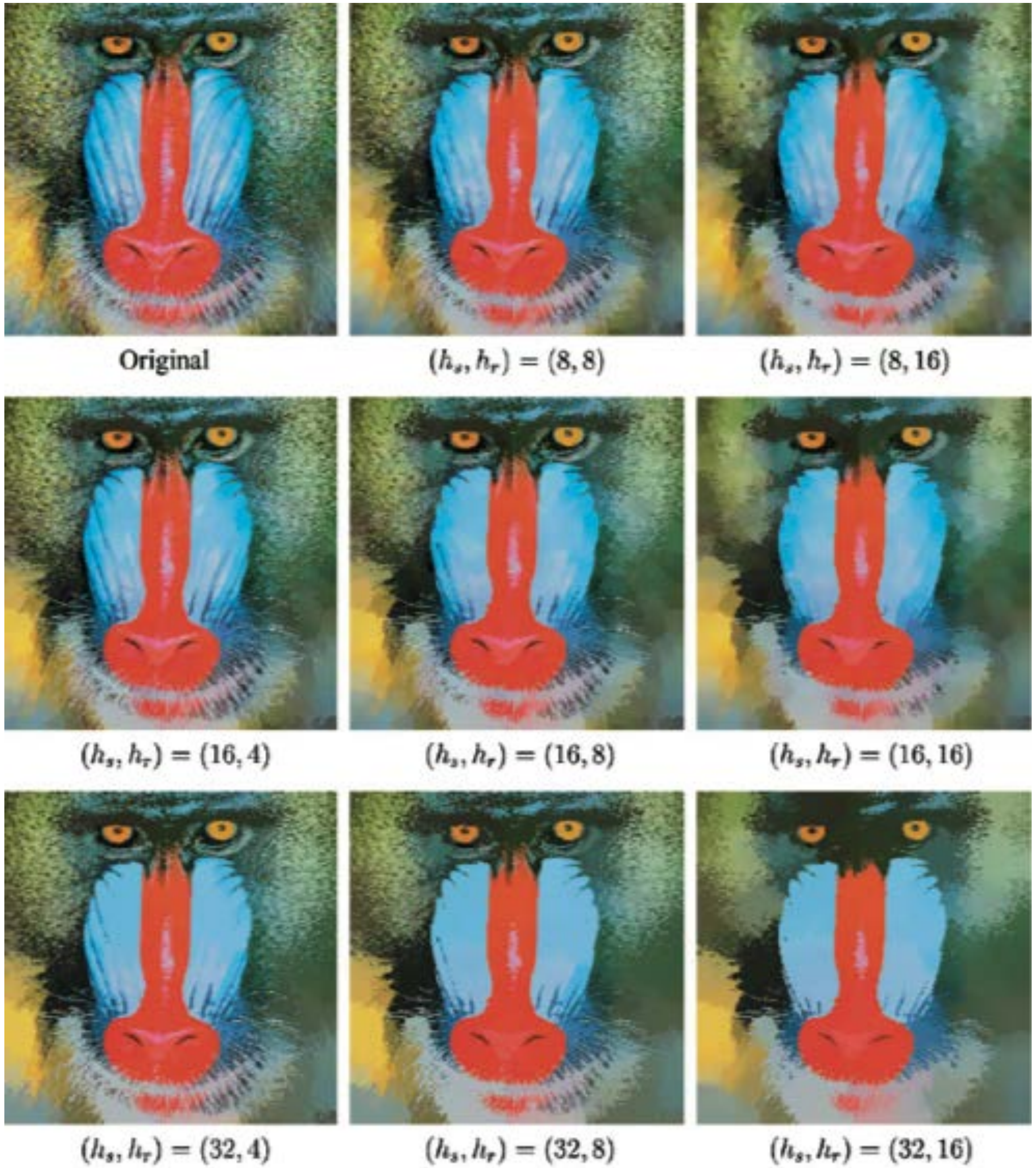


$h_s = 10,\ h_r = 10$



$h_s = 10,\ h_r = 10$

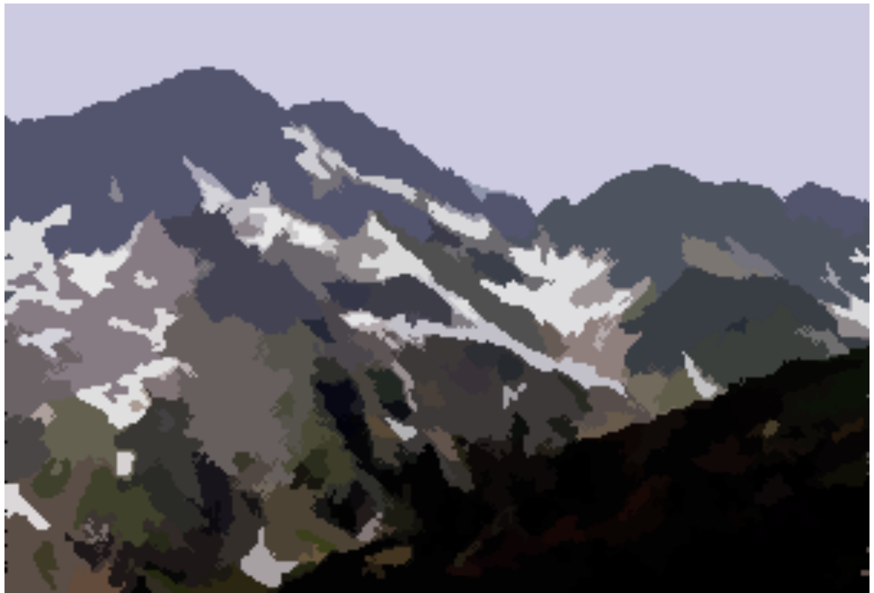
PARAMETERS



- h_s and h_r control the amount of smoothing in the spatial and color domains.
- They can be taken to be the ones that give the most stable response to small perturbations.

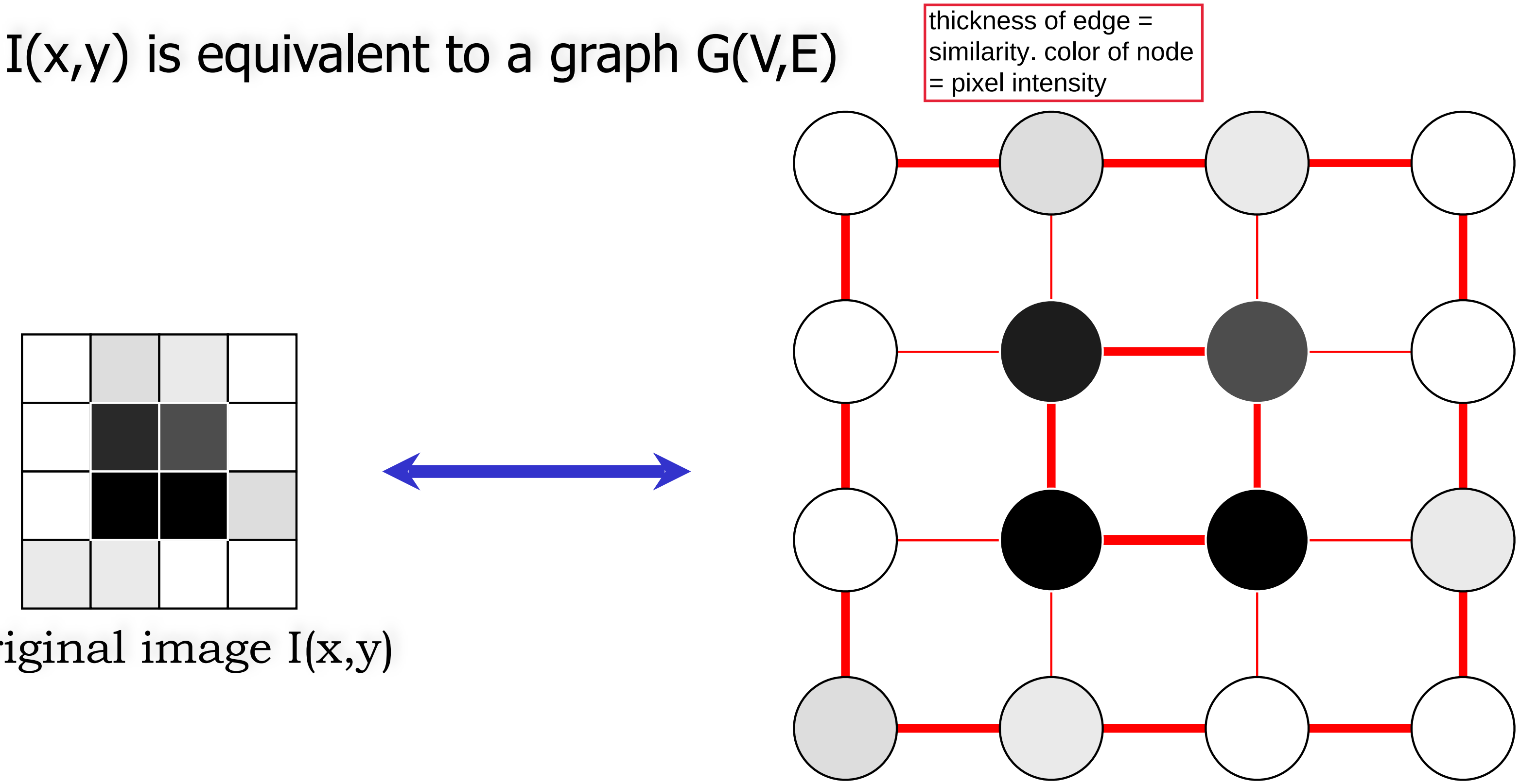
MORE RESULTS



IMAGES AS GRAPHS

An image I(x,y) is equivalent to a graph G(V,E)

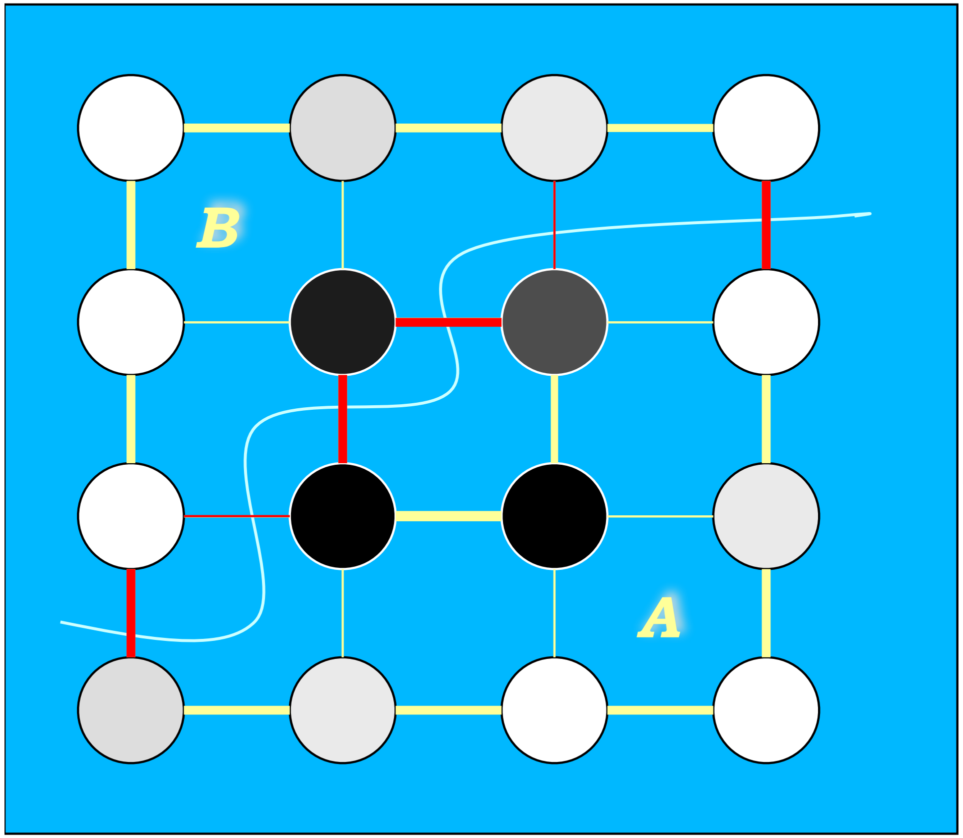


Original image I(x,y)

Graph G(V,E)

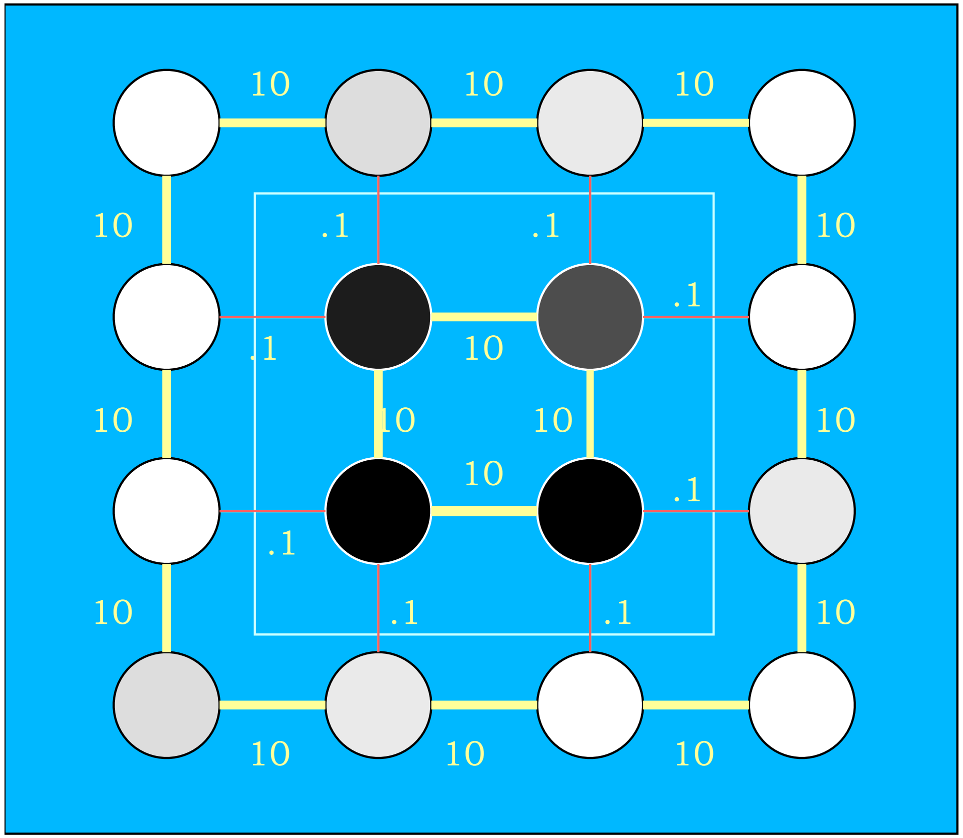
- V is a set of vertices or nodes that represent individual pixels.
- E is a set of edges linking neighboring nodes together. The weight or strength of the edge is proportional to the similarity between the vertices it joins together.

GRAPH-CUT



A cut through a graph is defined as the total weight of the links that must be removed to divide it into two separate components.

MIN-CUT

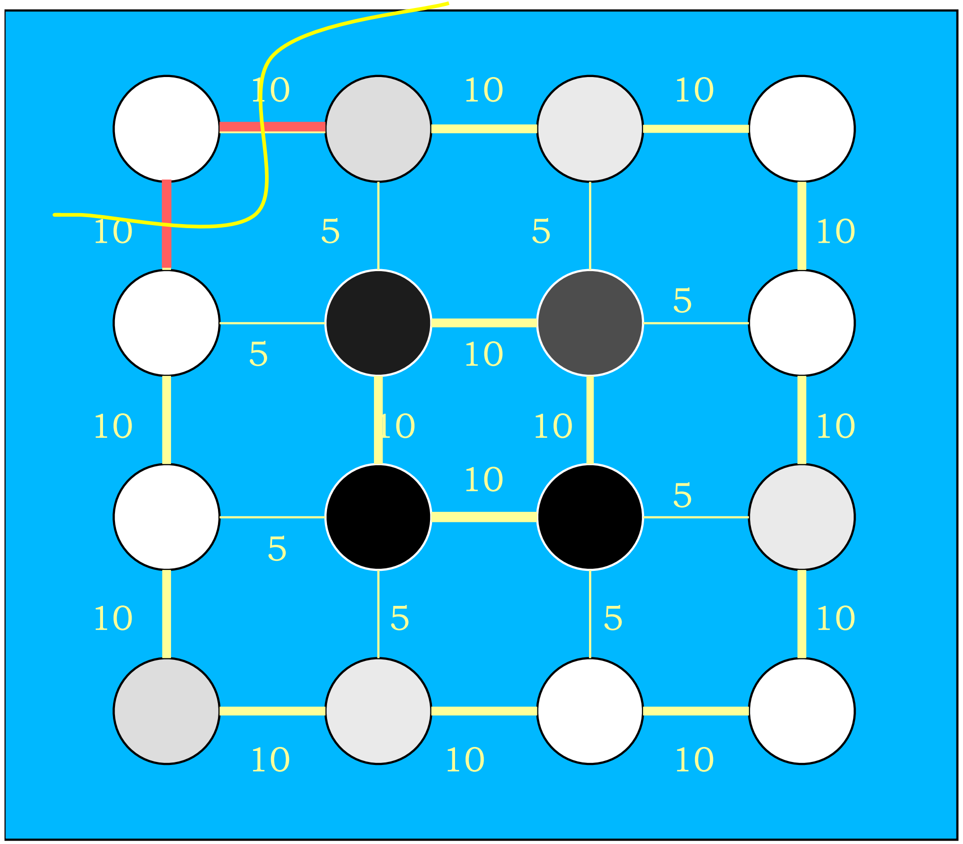


- Find the cut through the graph that has the overall minimum weight, which can be done effectively.

- Should correspond to the subset of edges of least weight that can be removed to partition the graph

- Since weight encodes similarity, this should be equivalent to partitioning the graph along the boundary of least similarity

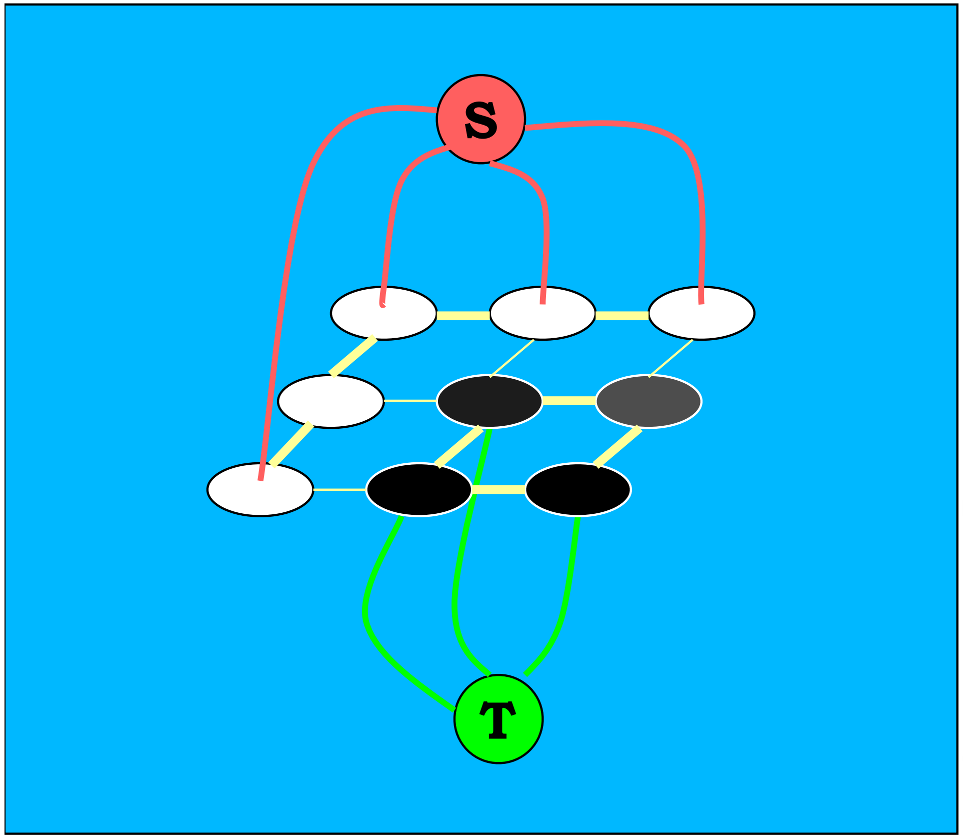
TRIVIAL CUT



- Has a preference for short cuts, which may sometime result in trivial solutions.

- Must be constrained to avoid them.

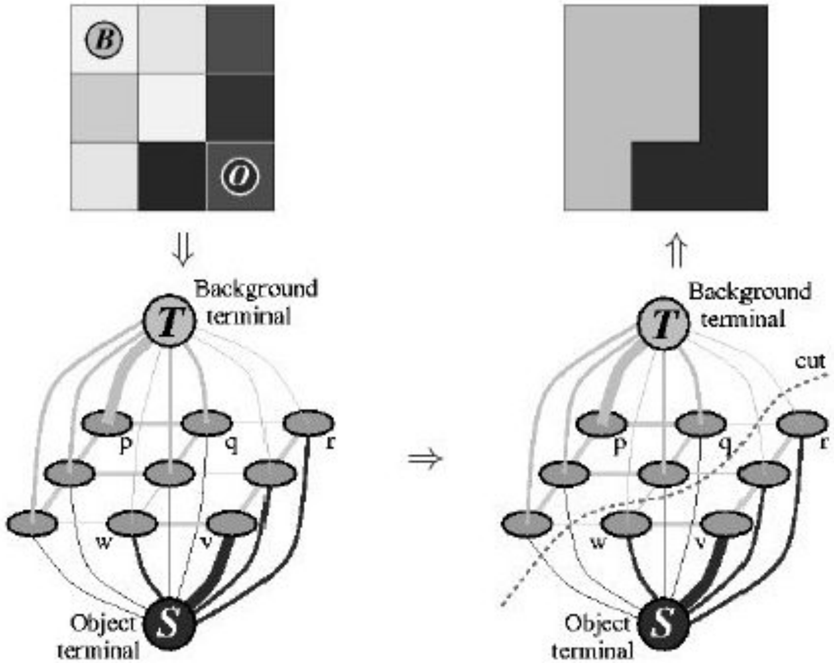
ST MIN-CUT



- Introduce two special nodes called source (S) and sink (T)
- S and T are linked to some image nodes by links of very large weight that will never be selected in a cut.
- Find the minimum cut that separates the source from the sink

--> The problem becomes deciding how to connect S and T to the image nodes.

GRAPH CUT

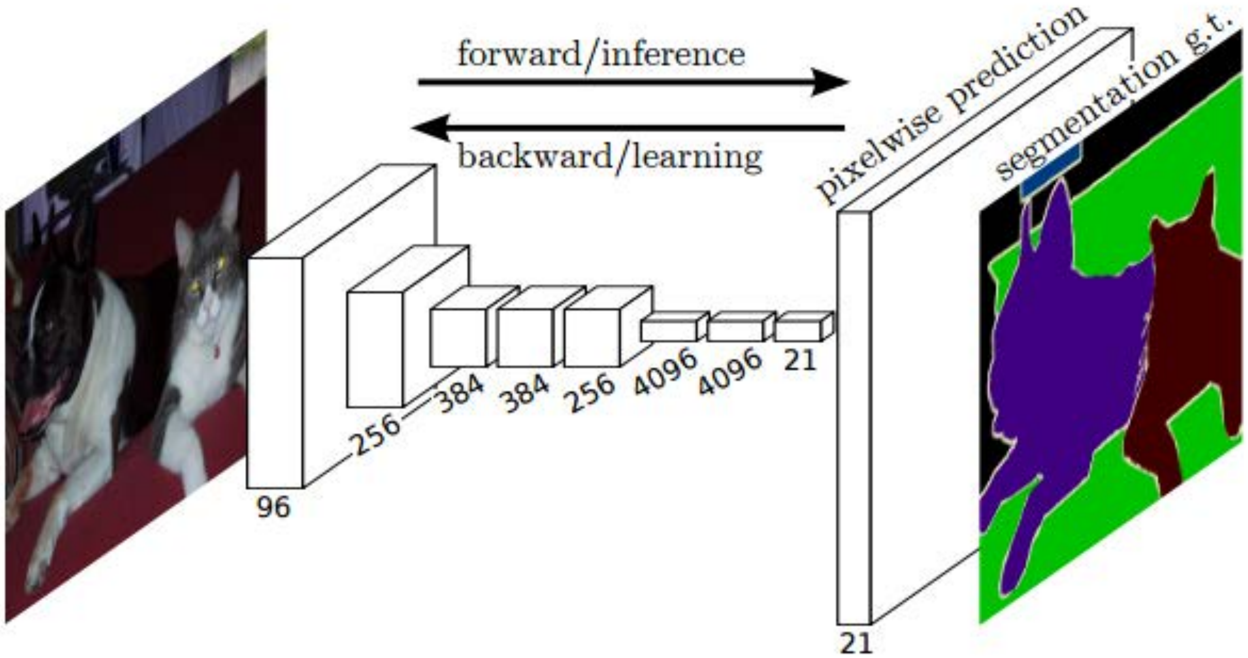


Minimize

$$E(y|x,\lambda) = \sum_i \underbrace{\psi(y_i|x_i)}_{\text{unary term}} + \lambda \sum_{(i,j)\in\mathcal{E}} \underbrace{\phi(y_i,y_j|x_i,x_j)}_{\text{pairwise term}} \;,$$

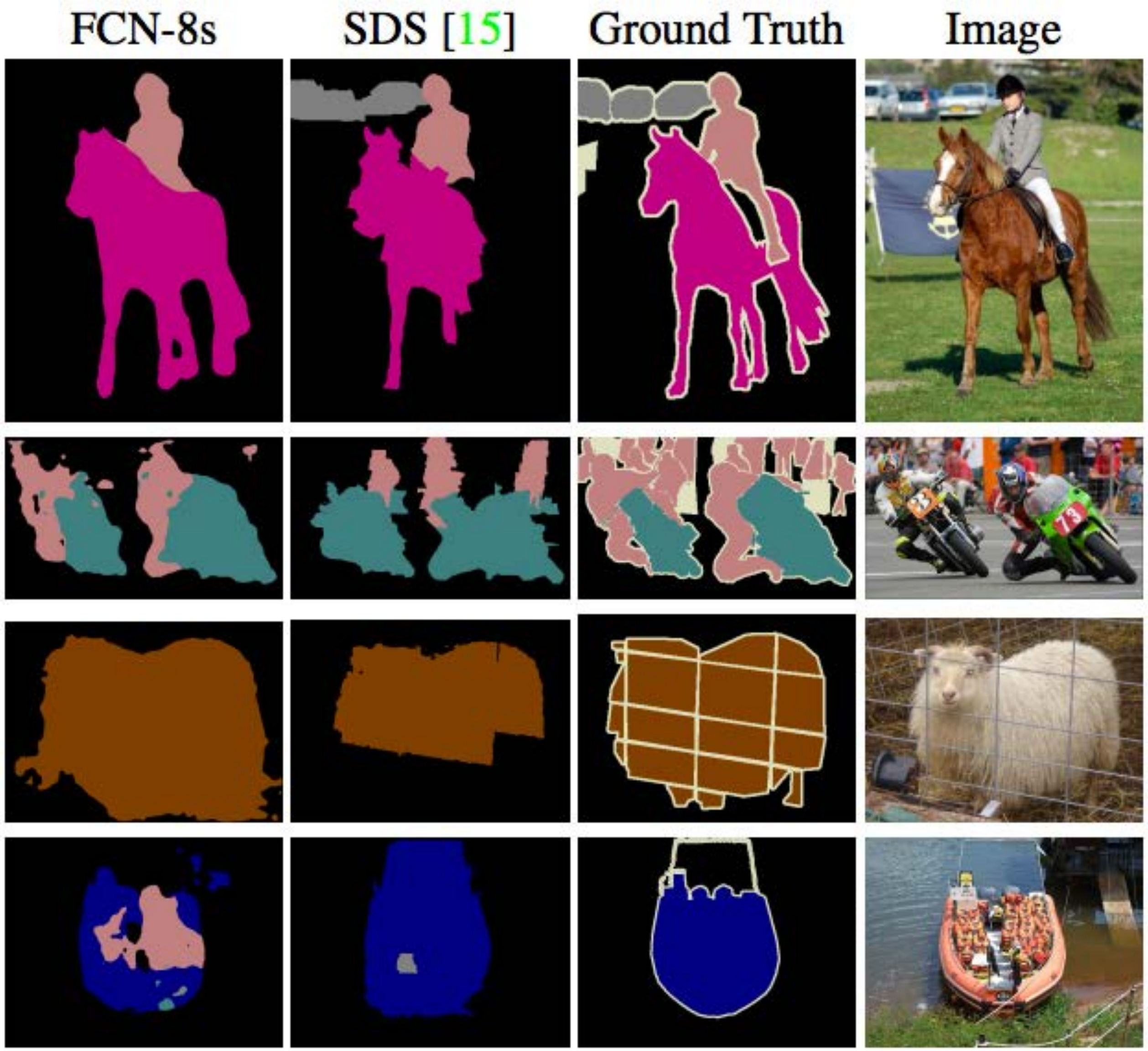
with respect to y.

CONVOLUTIONAL NETS

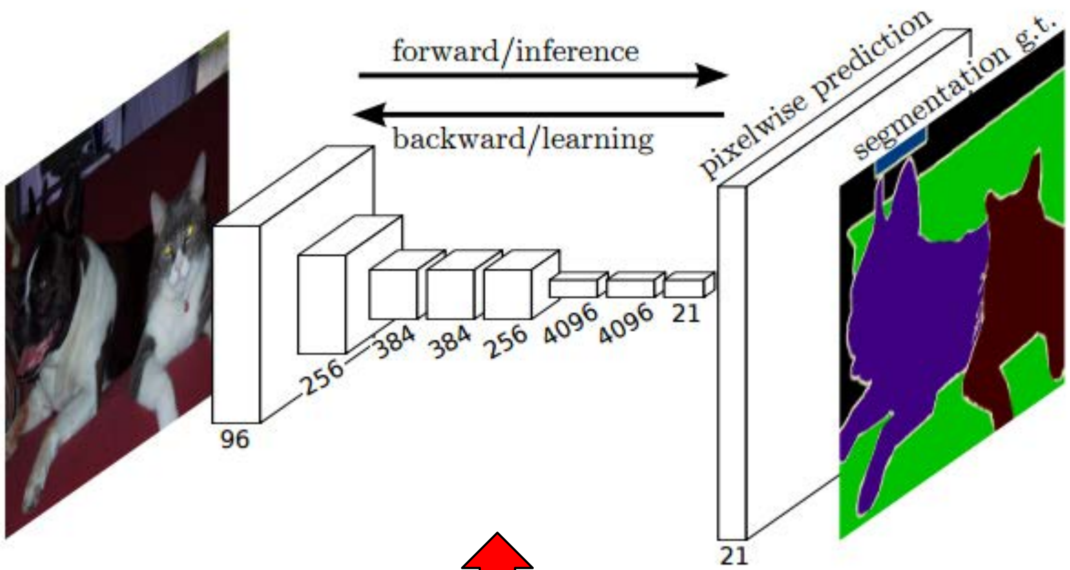


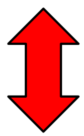
- Connect input layer to output one made of segmentation labels.
- Need layers that both downscale and upscale.
- Connect the lower layers directly to the upper ones.

STATE OF THE ART RESULTS



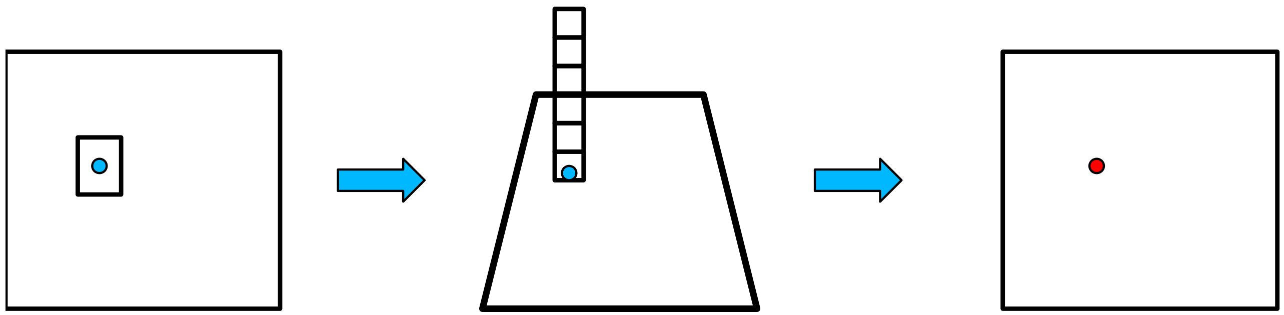
A PARTIAL EXPLANATION?





- Can be understood as generating for every output pixel a feature vector containing the output of all the intermediate layers.

- Preliminary experiments show that this might be the case.

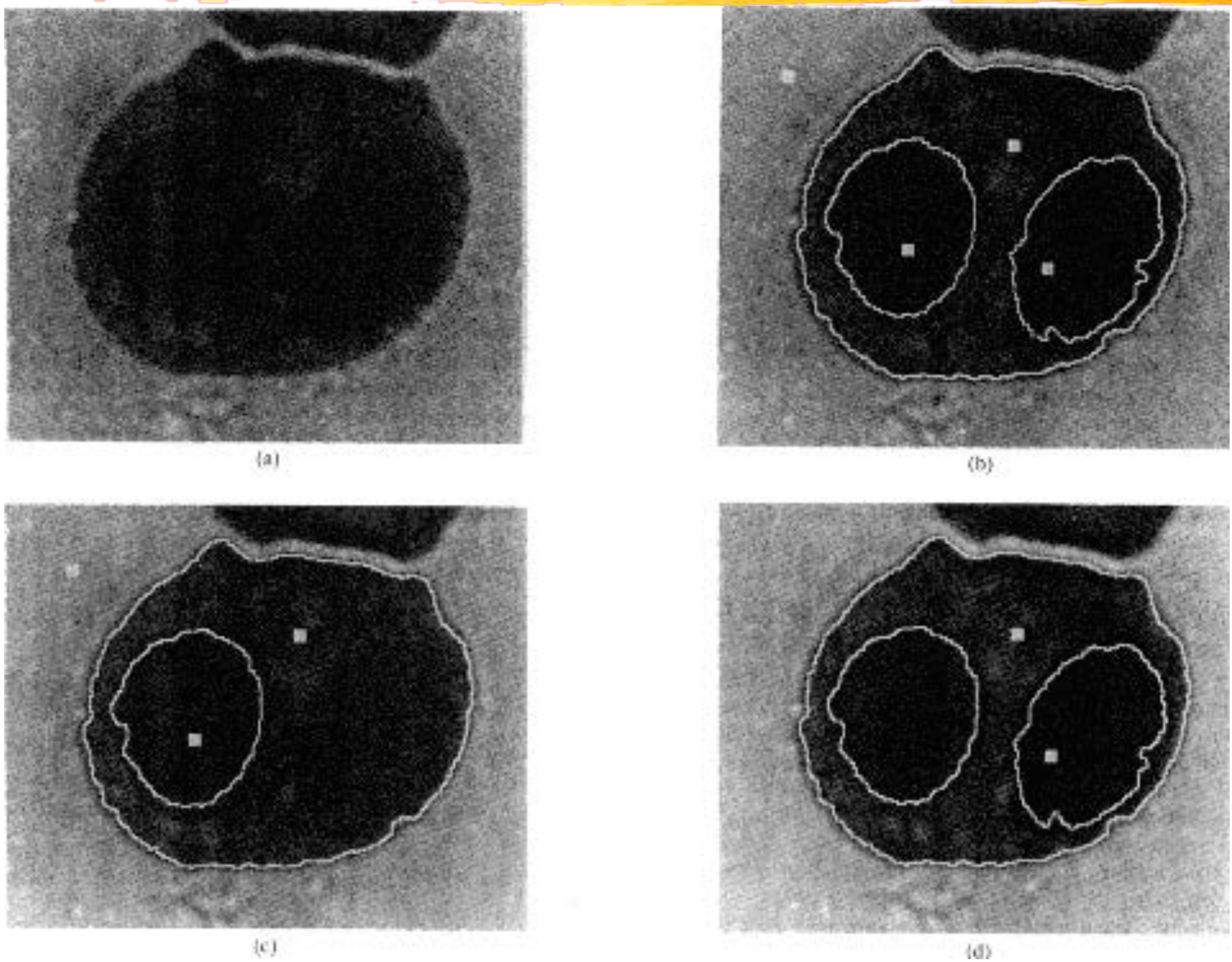


GENERIC TECHNIQUES

Low-level methods can extract useful information but are inherently limited. One must also take into account:

- Region outlines.
- Region shape.
- Context.
- Domain knowledge.

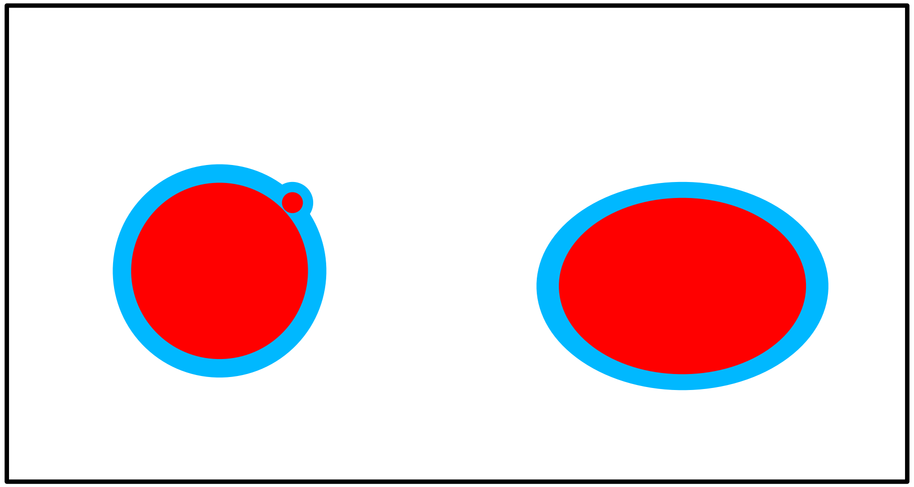
PROVIDING SEEDS



Interactive Segmentation of a Cell

Adams and Bischof, PAMI'94

REGION GROWING



Given a set of regions $A_1, \ldots, A_N$, consider

$$T = \left\{ x \notin \bigcup_{i=1}^{N} A_i \text{ such that } N(x) \cap \left(\bigcup_{i=1}^{N} A_i \right) \neq \varnothing \right\},$$

the set of unlabeled pixels that are neighbors of already labeled ones.

- Define a metric, e.g. $\delta(x) = \left| g(x) - \underset{y \in A_{i(x)}}{\text{mean}} \left[g(y) \right] \right|.$

Difference between gray level of pixel and mean of gray level of region

- Represent T as a sorted list according to this metric, the SSL.

REGION GROWING

<u>While</u> SSL is not empty <u>do</u>

Remove first pixel y from SSL.

<u>If</u> all already labeled neighbors of y, other than boundary pixels, have the same label

<u>then</u>

Set y to this label.

Update running mean of corresponding region.

Add neighbors of y that are neither already set nor already in the SSL to the SSL according to their value of delta.

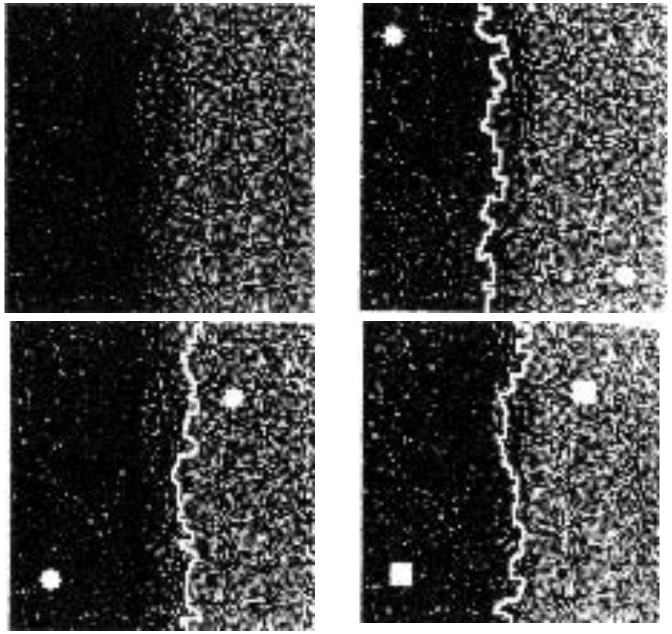
<u>else</u>

Flag y as a boundary pixel.

<u>fi</u>

<u>od</u>

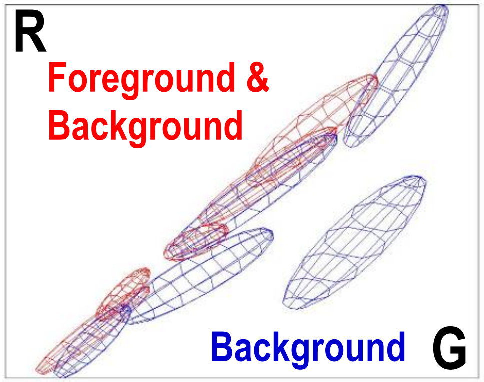
LIMITATIONS



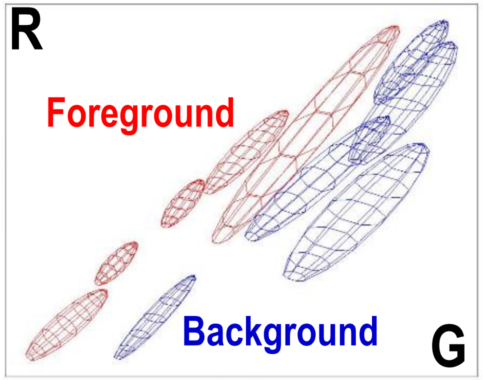
different boundaries
with different seeds

- In general, the result depends on the order in which the pixels are taken into consideration.
- The homogeneity measure is noise sensitive.

INTERACTIVE FOREGROUND EXTRACTION



Iterated
graph cut



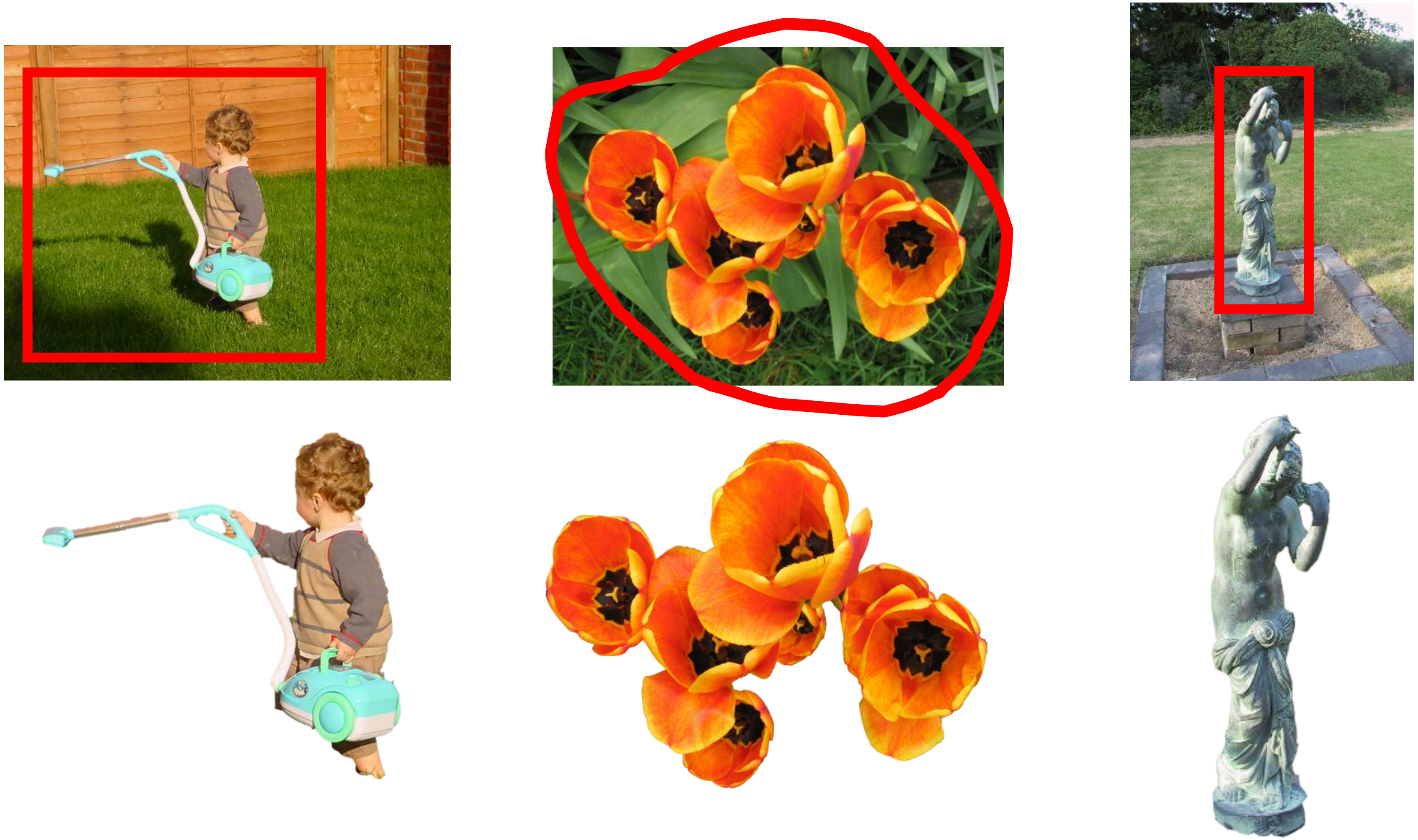
- Draw rectangle.
inside fg outside bg
- build 5 cluster of colors in fg and bg
- compute how far each pixel in the foreground is from the clusters and assign to either source or sink
- graph cut
- iterate because we would now have a better segmentation than the initial rectangle

• [K-m]eans to learn color distributions

• Graph cuts to infer the segmentation

GrabCut
Rother & al. SIGGRAPH 04

RELATIVELY EASY EXAMPLES



GrabCut
Rother & al. SIGGRAPH 04

MORE DIFFICULT EXAMPLES

Camouflage & Low Contrast

Fine structure

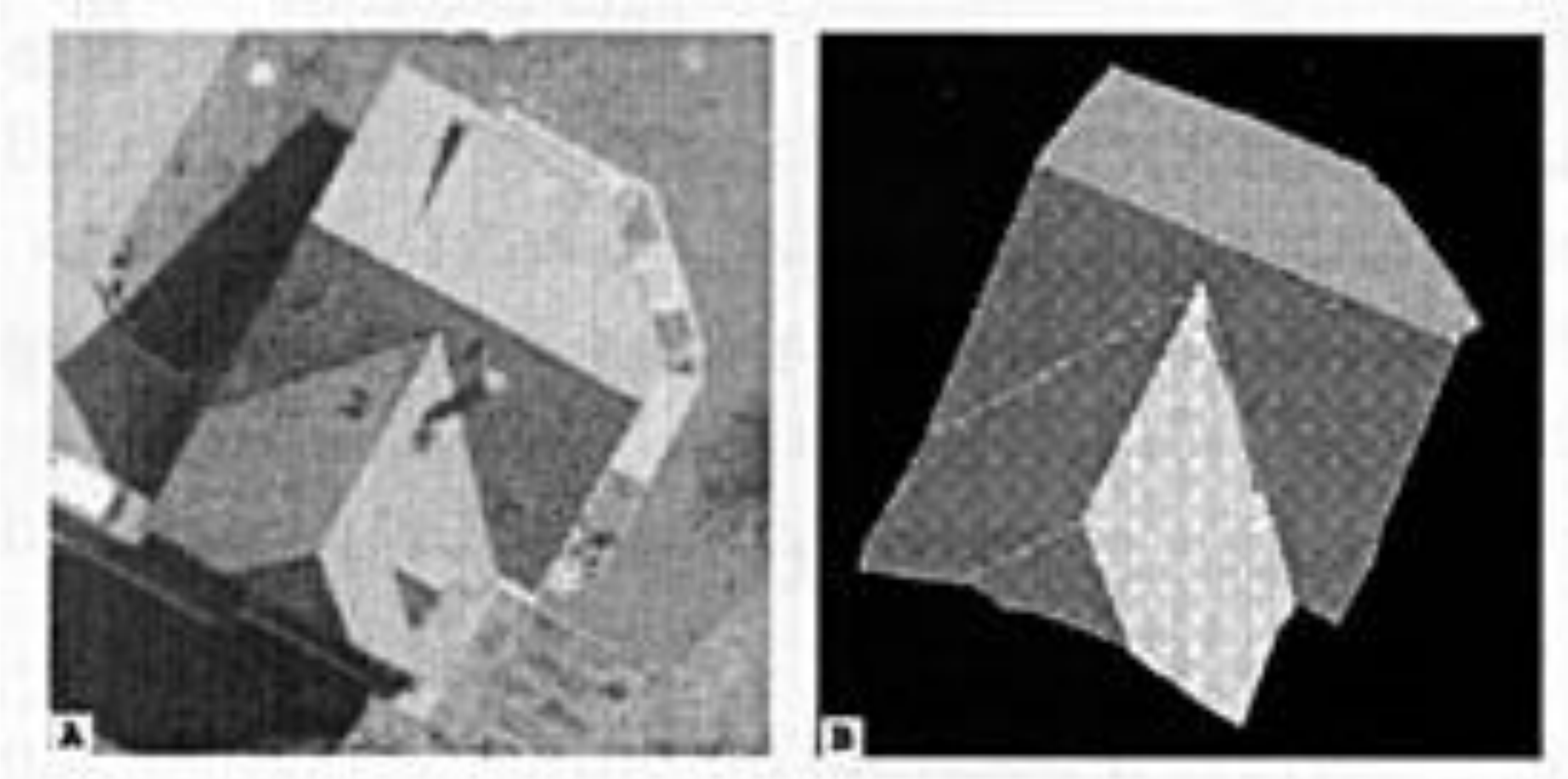
No telepathy

Initial Rectangle

Initial Result

GrabCut
Rother & al. SIGGRAPH 04

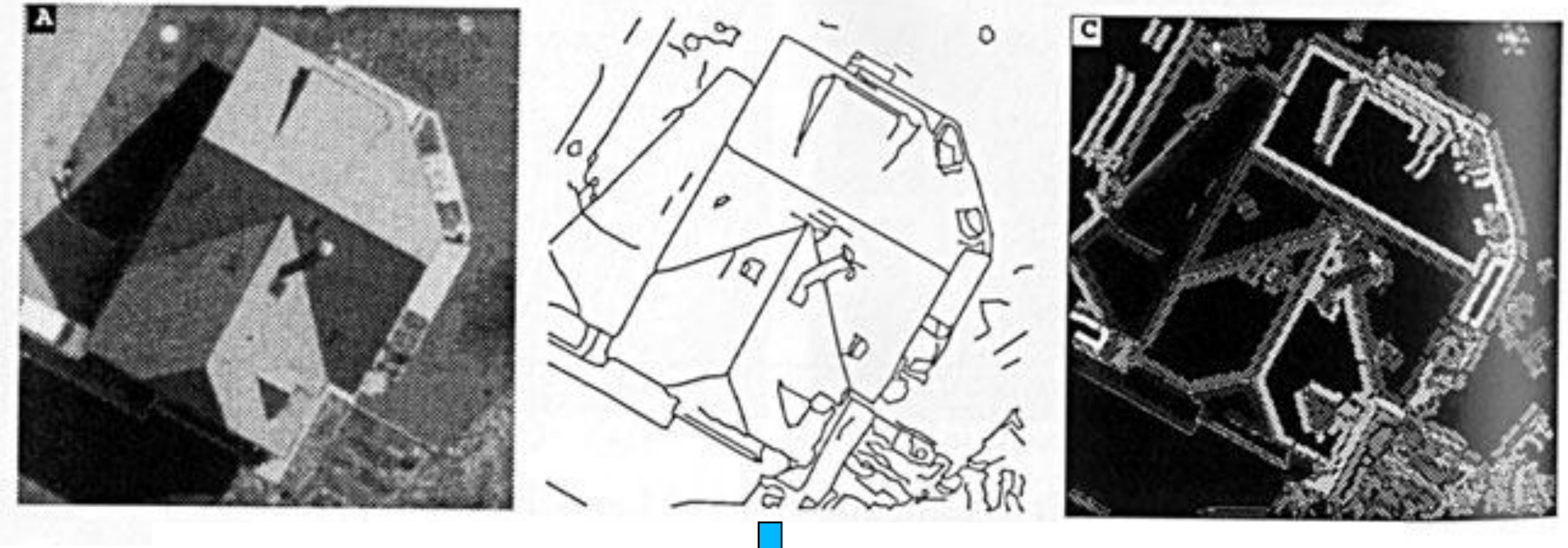
INTRODUCING SEMANTICS

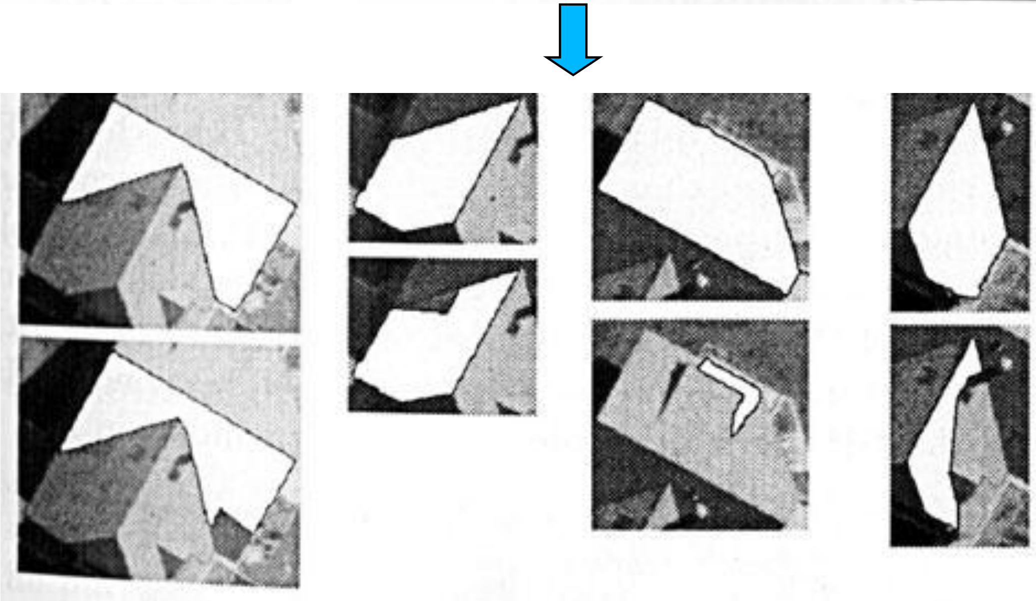


Segmentation of a complex roof with many sides

Bignone et al., ECCV96

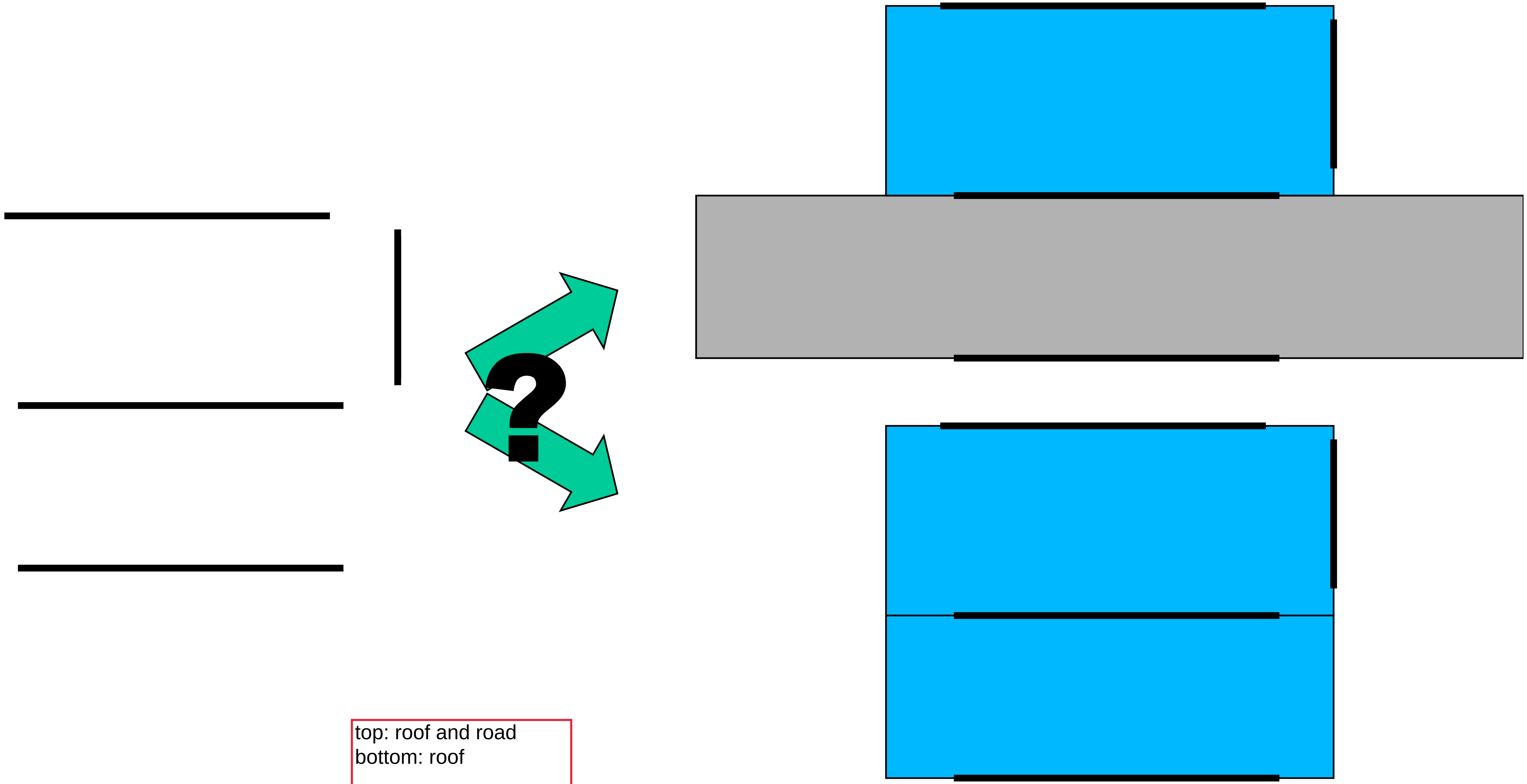
FROM ROOF EDGES TO ROOF PARTS





regions near a boundary from different edge segments should have similar color if they belong to roof

AMBIGUOUS INTERPRETATIONS

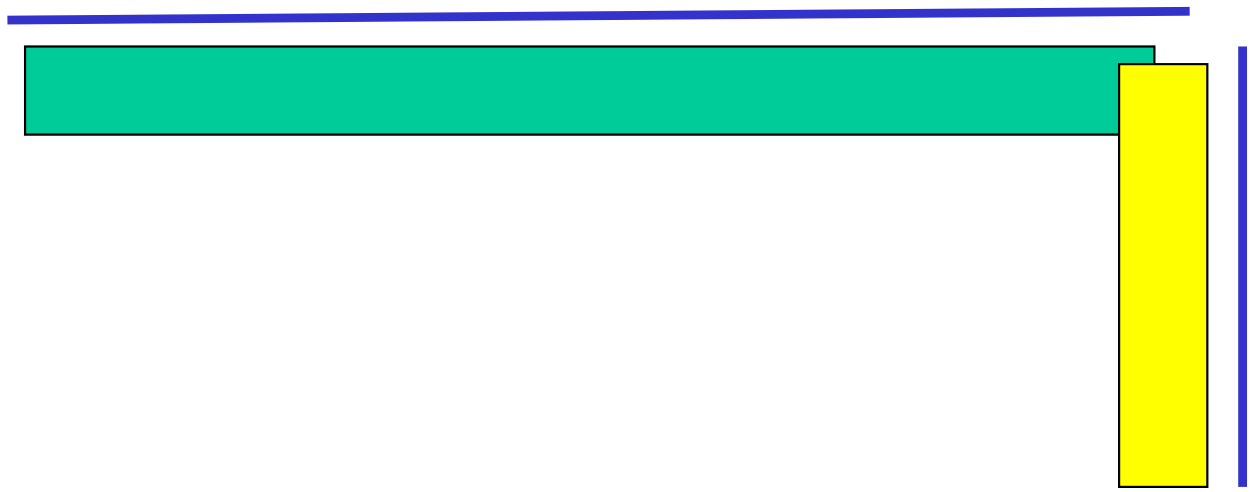


top: roof and road
bottom: roof

reason why cannot
only look at edges

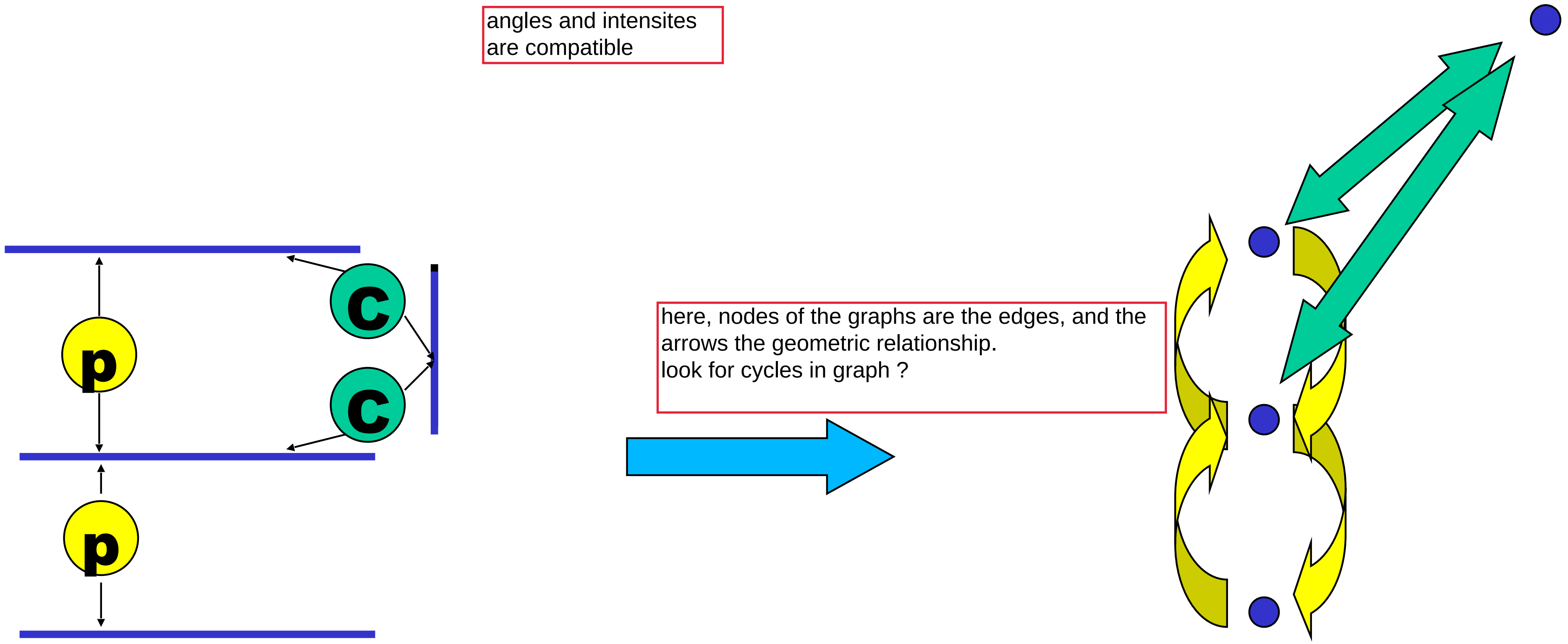
COMBINING EDGE AND REGION INFORMATION

can compute fisher discriminant between green and yellow



Check photometric consistency before associating edges to prune the graph.

SEGMENTATION AS A GRAPH SEARCH PROBLEM



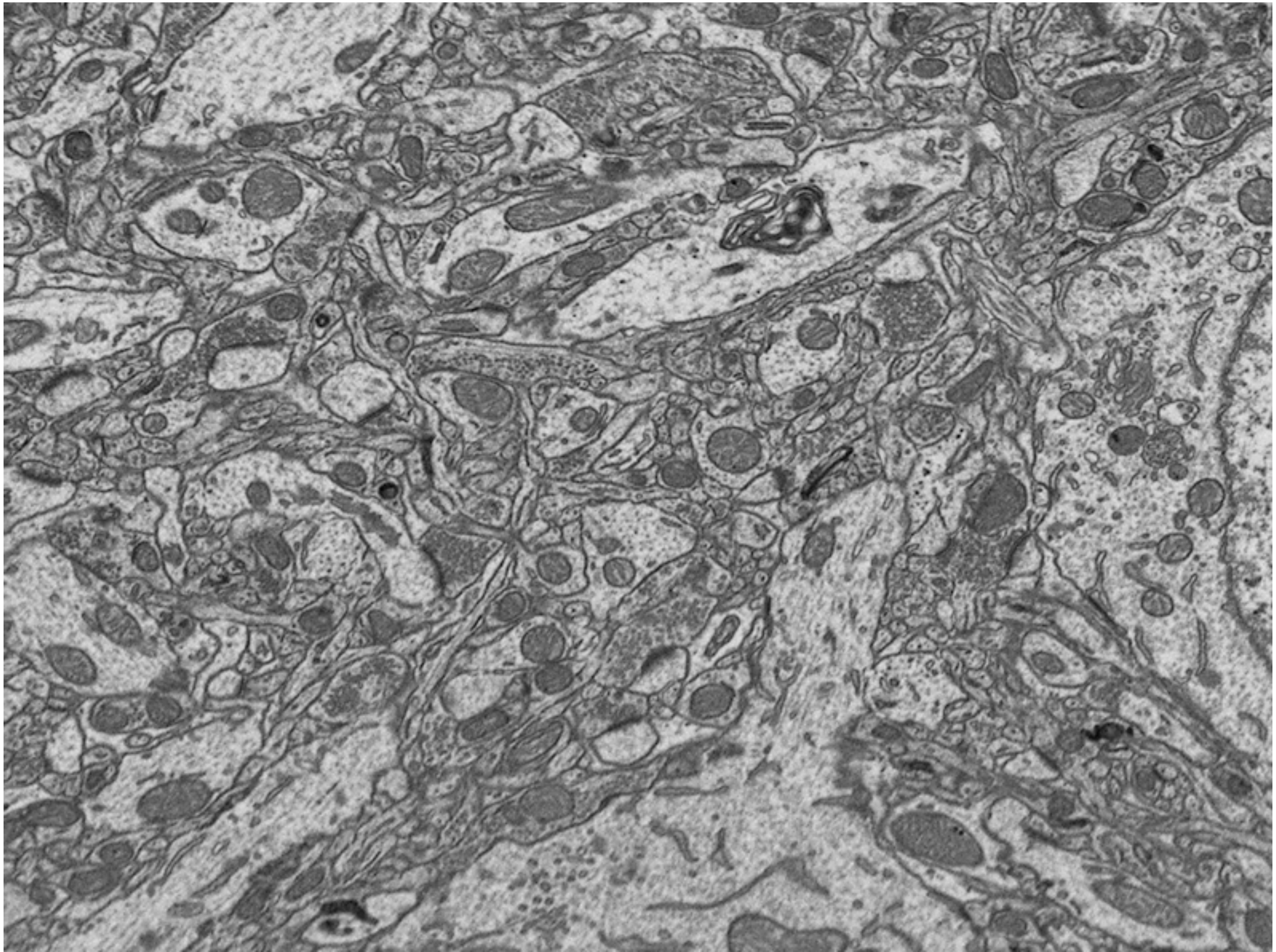
Finding candidate regions amounts to finding cycles in the graph → Can use graph-search techniques to handle the combinatorics.

ALGORITHM

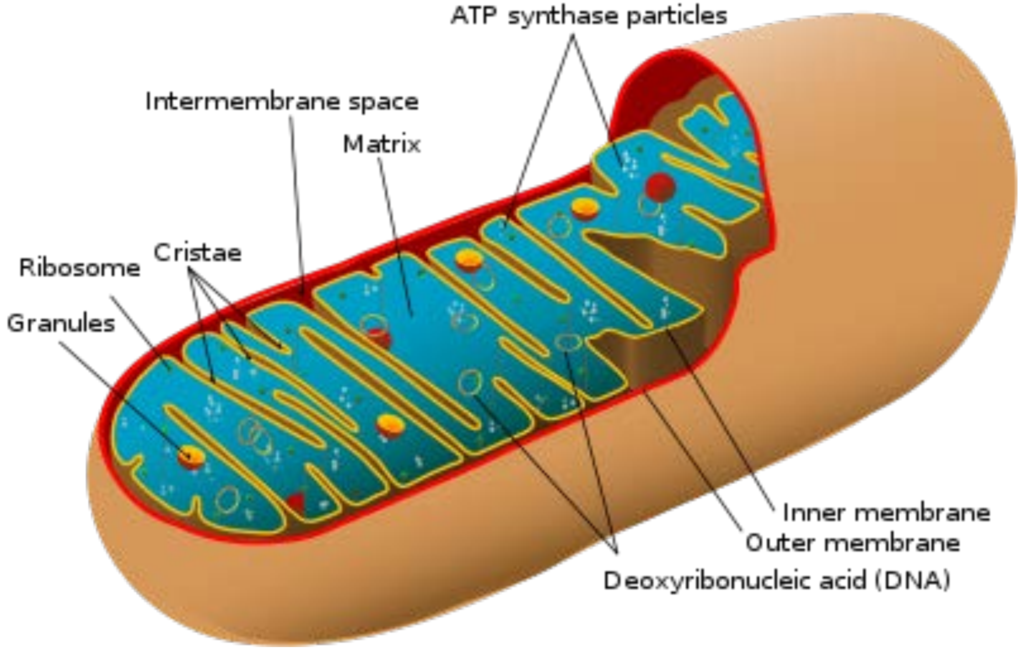
- Edge extraction
- Stereo and coplanar grouping
- Photometric and chromatic analysis
- Grouping on an homogeneity basis
- Selection of compatible hypothesis

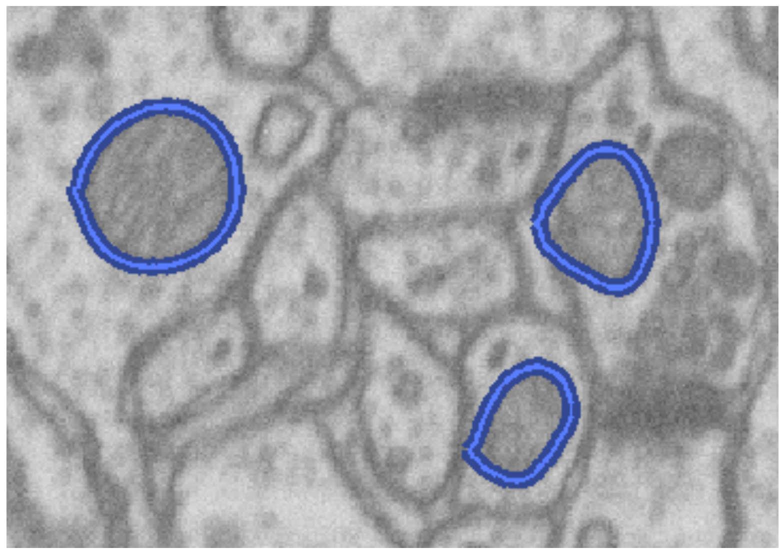
→ Ok for single houses but combinatorial explosion in a dense urban environment.

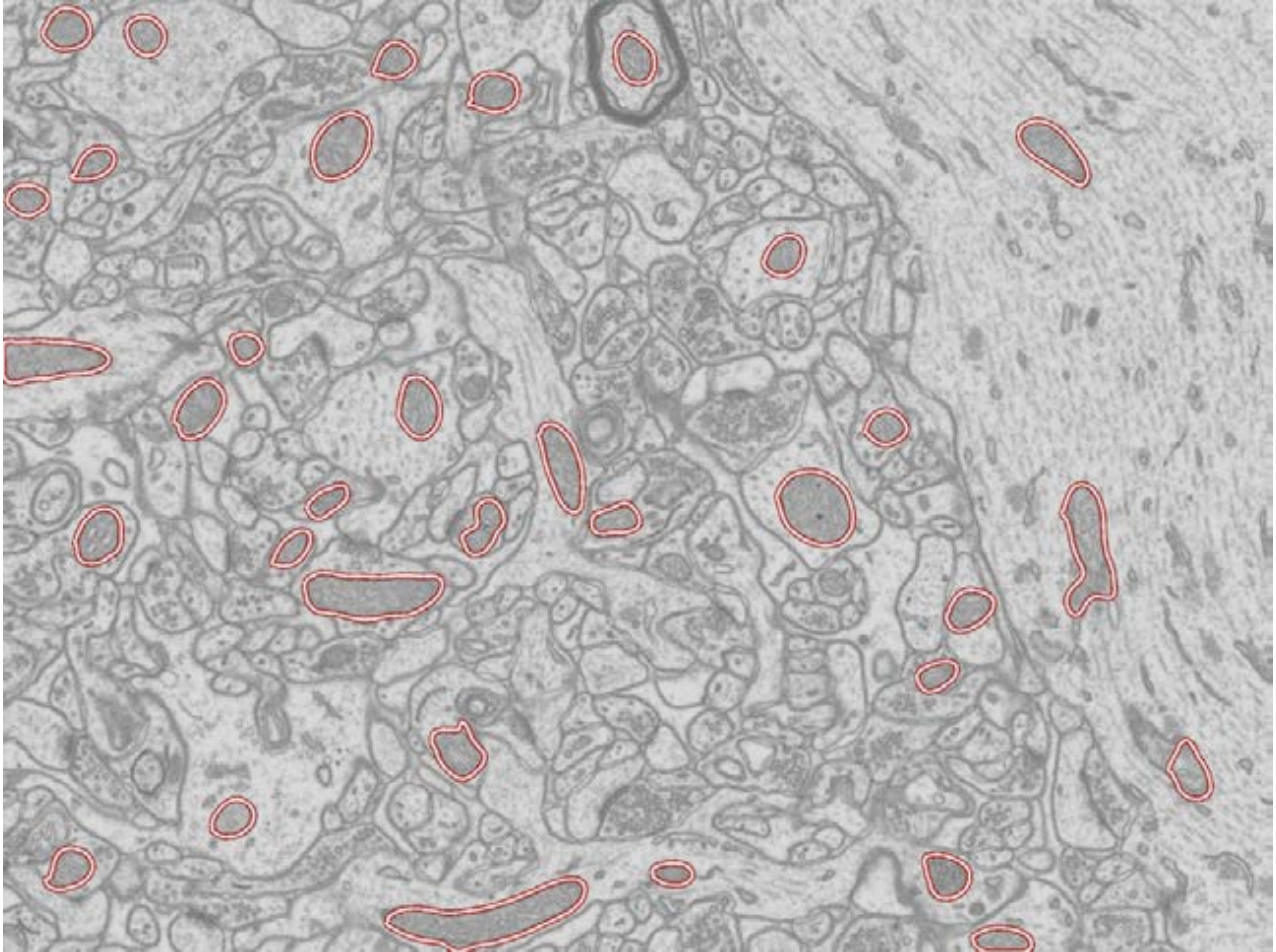
ELECTRON MICROSCOPY



MITOCHONDRIA



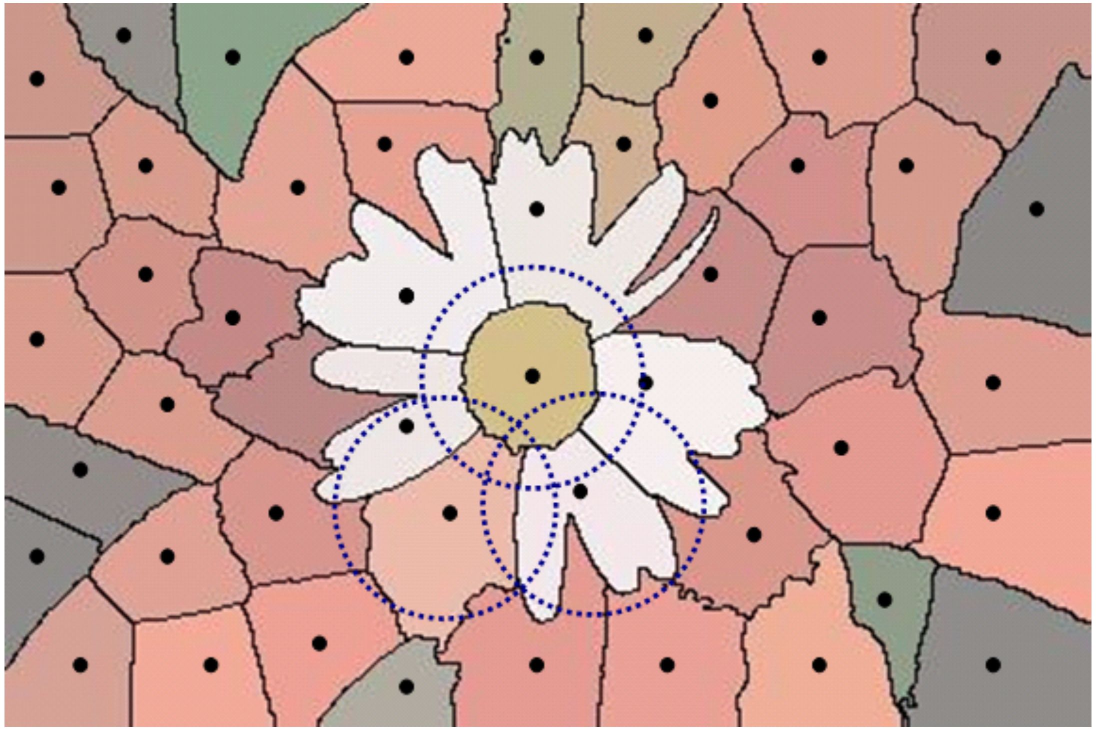




ALGORITHM

1. Superpixels oversegmentation
2. Feature extraction
   - Ray features
   - Gray level histograms
3. SVM classification
4. Graph cuts segmentation

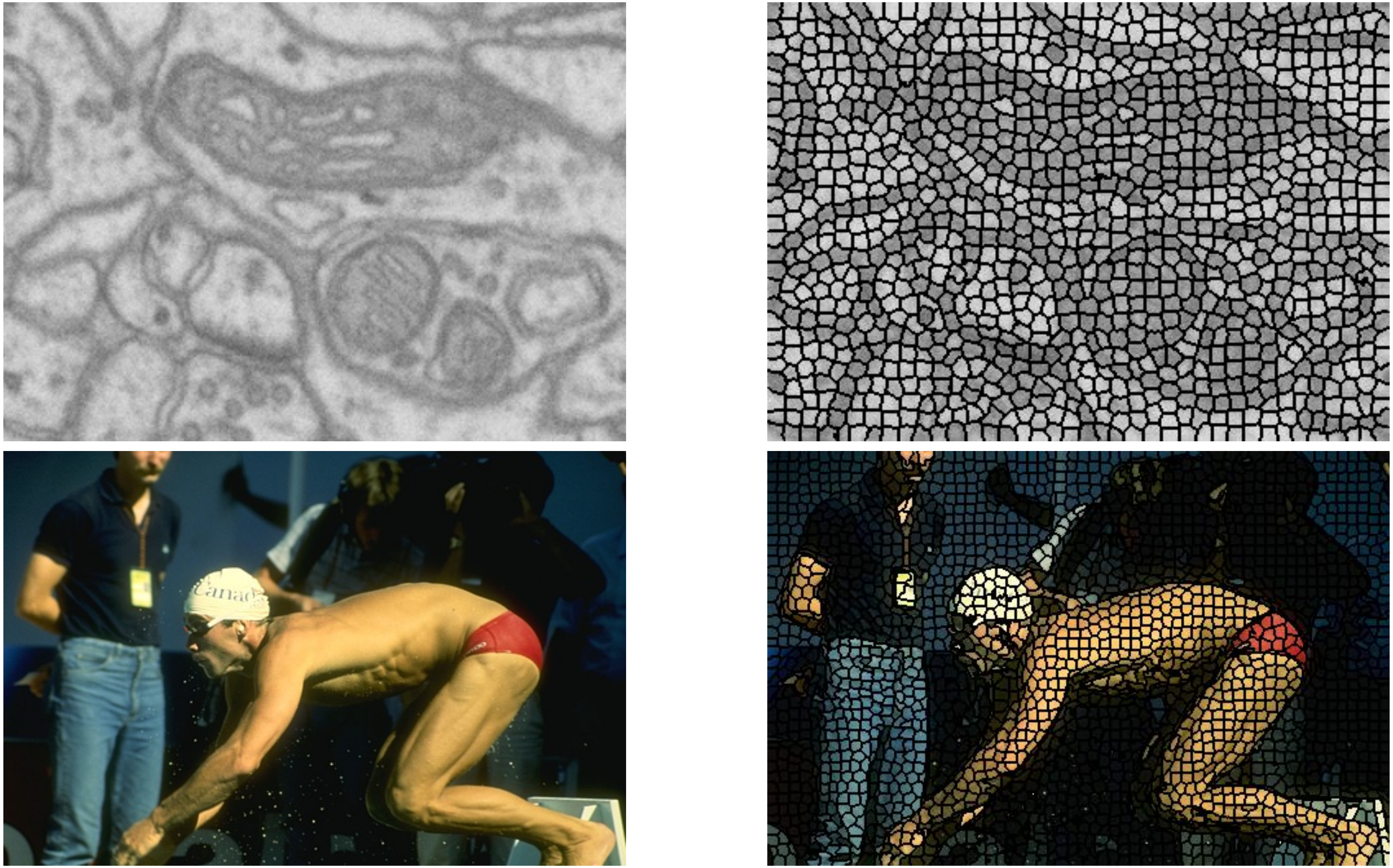
SUPERPIXELS



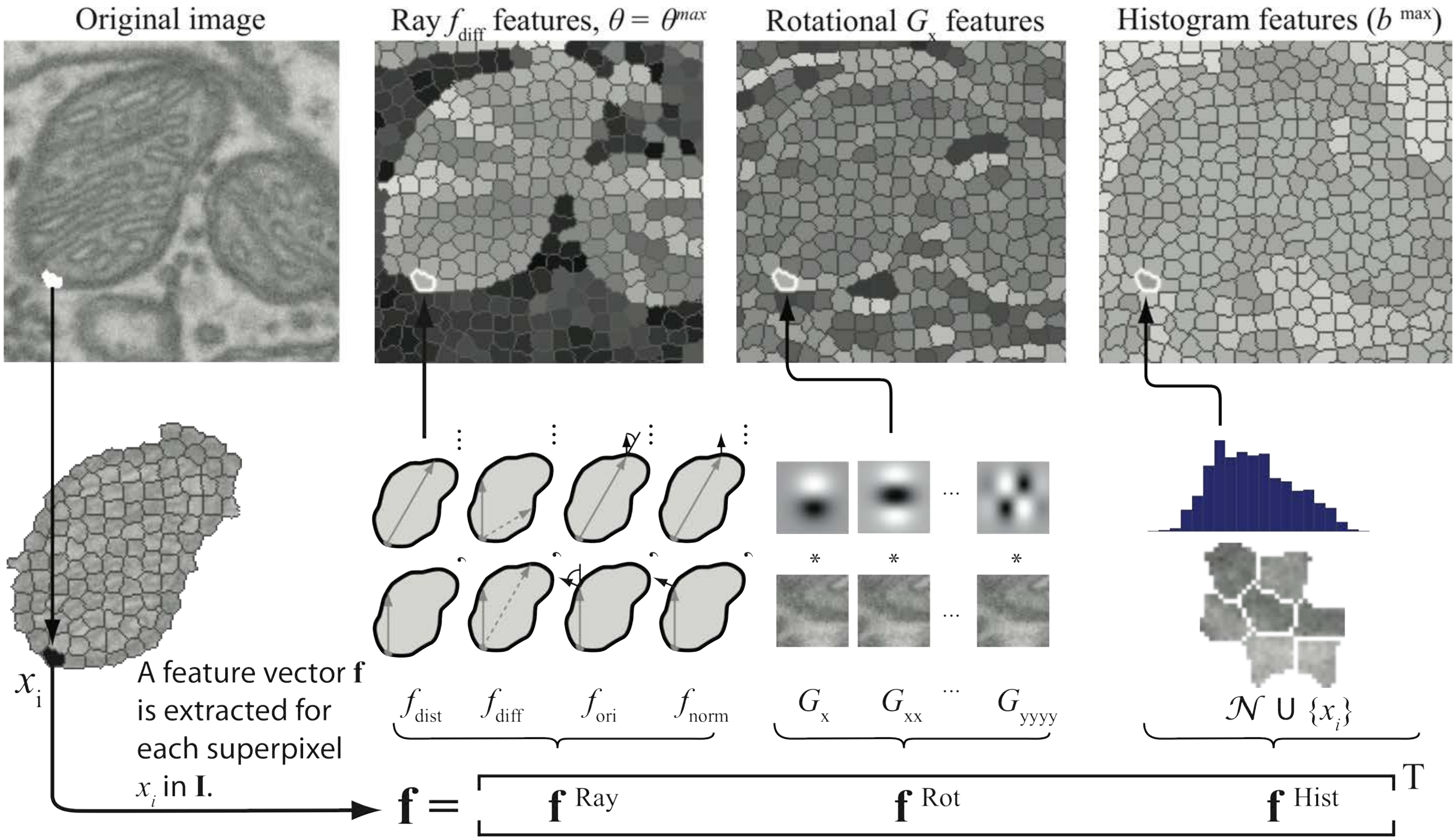
Run K-Means algorithm with regularly spaced seeds on a grid and using a distance that is a weighted sum of distances in image space and in gray level/color space.

Achanta et al. PAMI'12

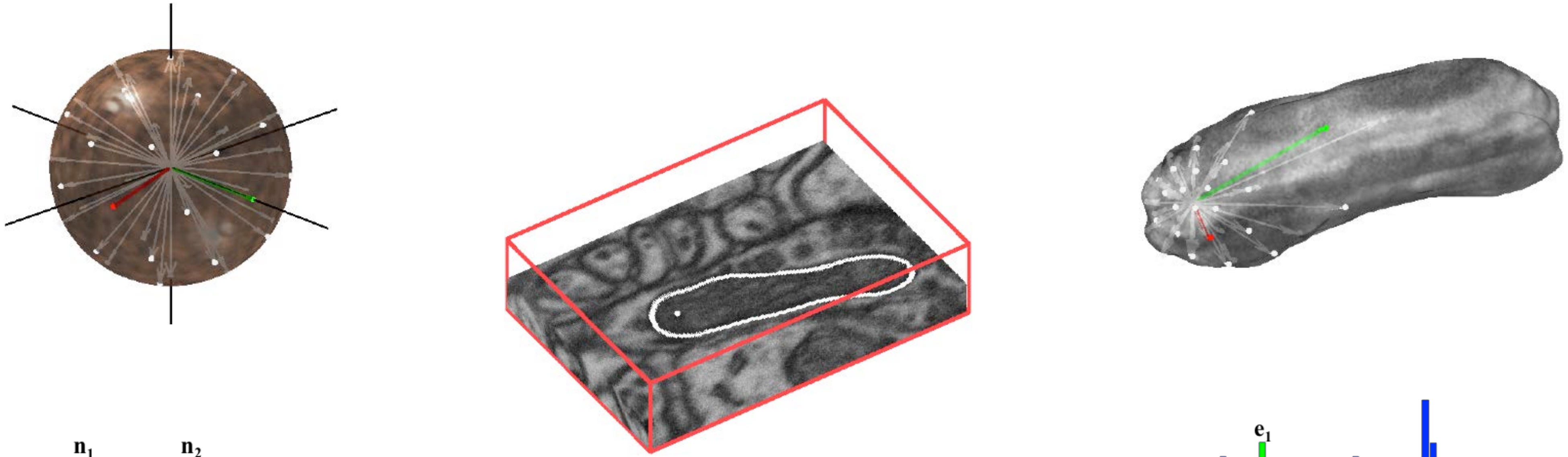
GRAY LEVELS OR COLORS



MITOCHONDRIA

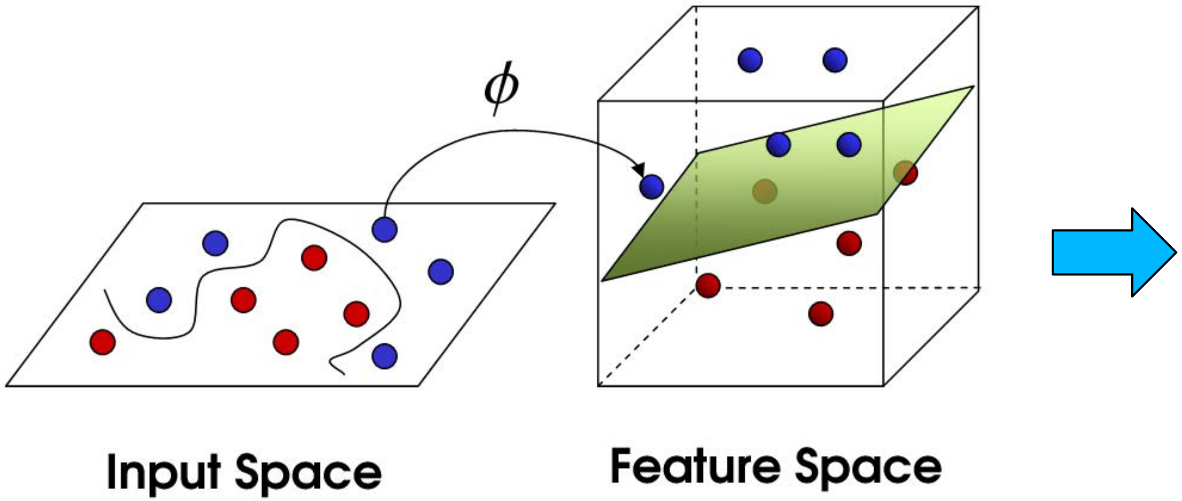


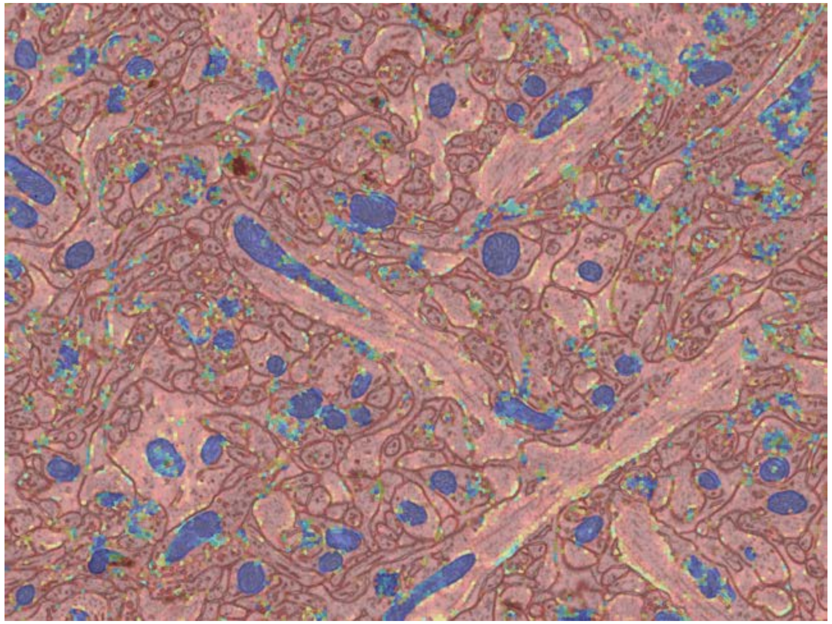
RAY FEATURES



- Compute statistics of distances to nearest boundary.
- Adds global information to a purely local measure.

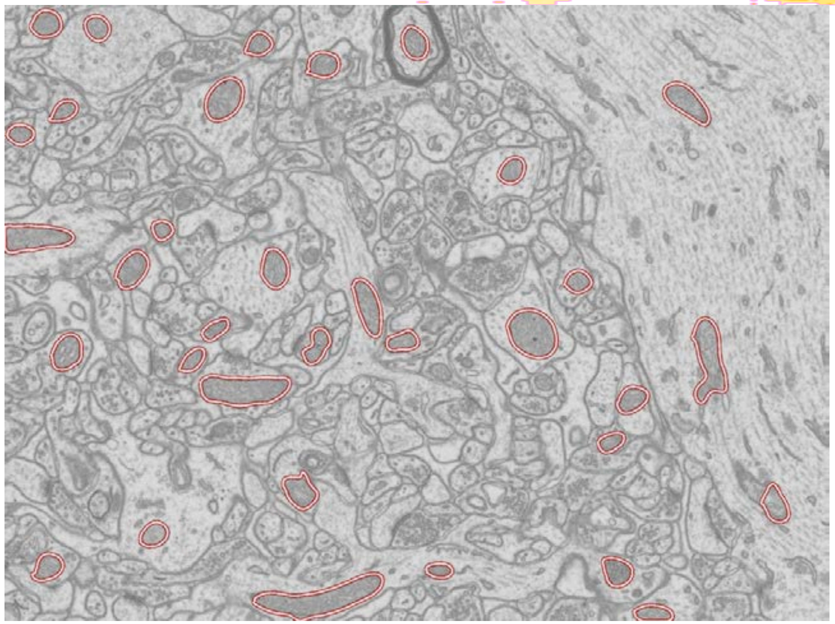
SVM CLASSIFICATION

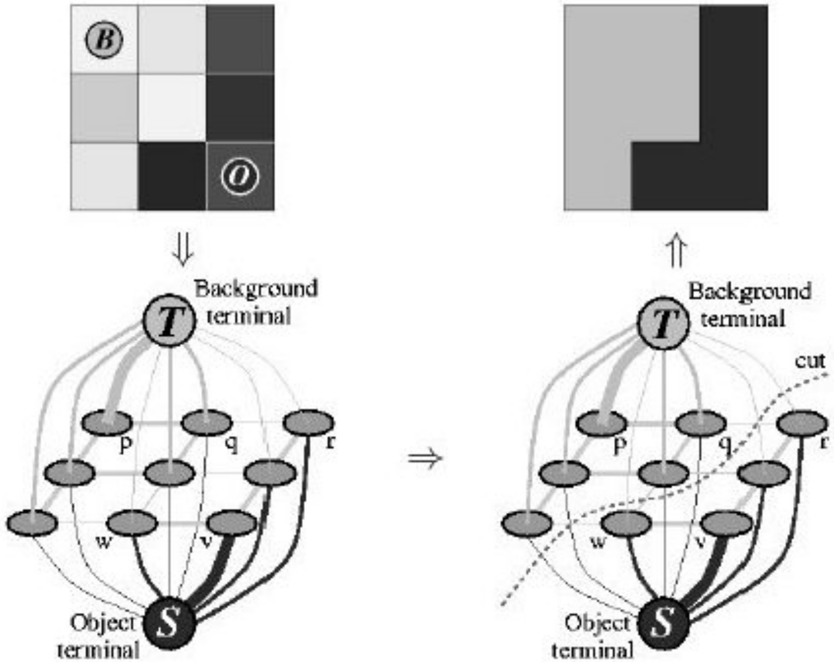




- The features incorporate the filter responses among other things.
- The probability of a superpixel belonging to a mitochondria is estimated from the SVM output.

GRAPH CUT



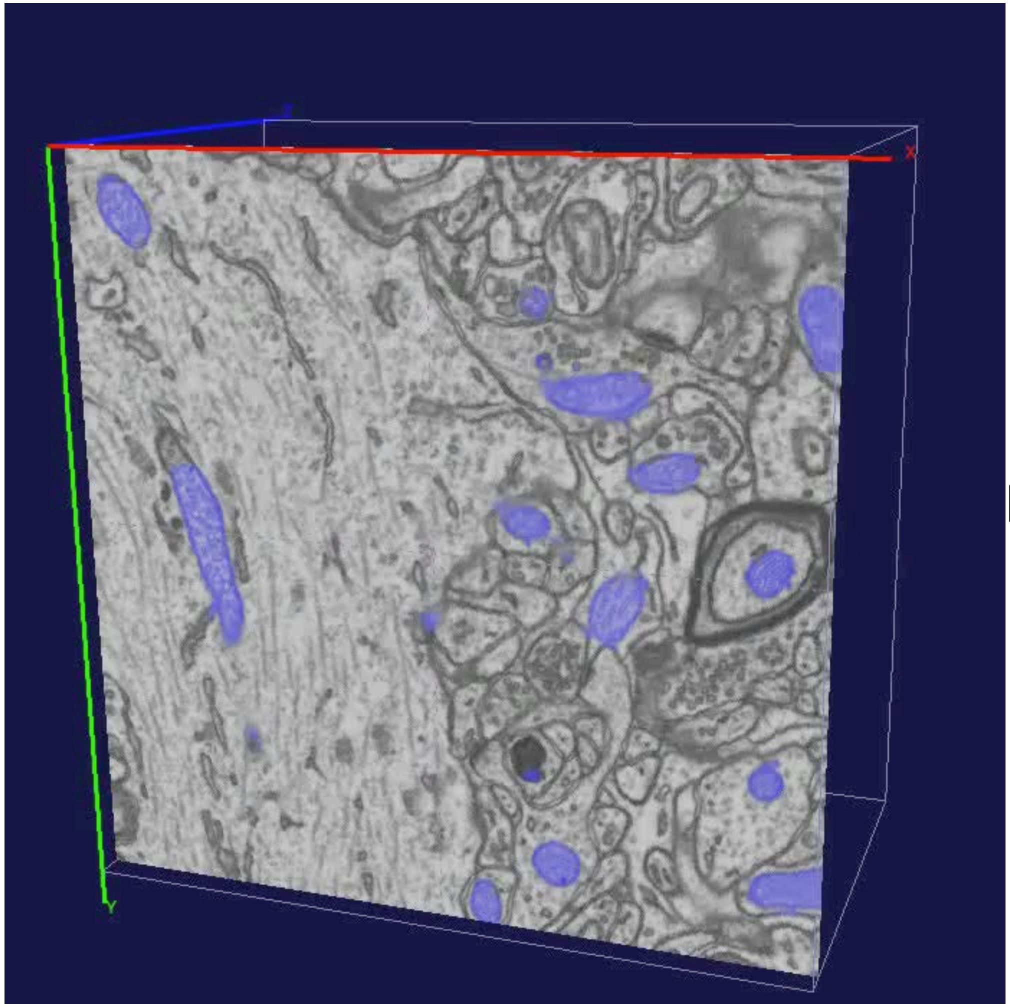


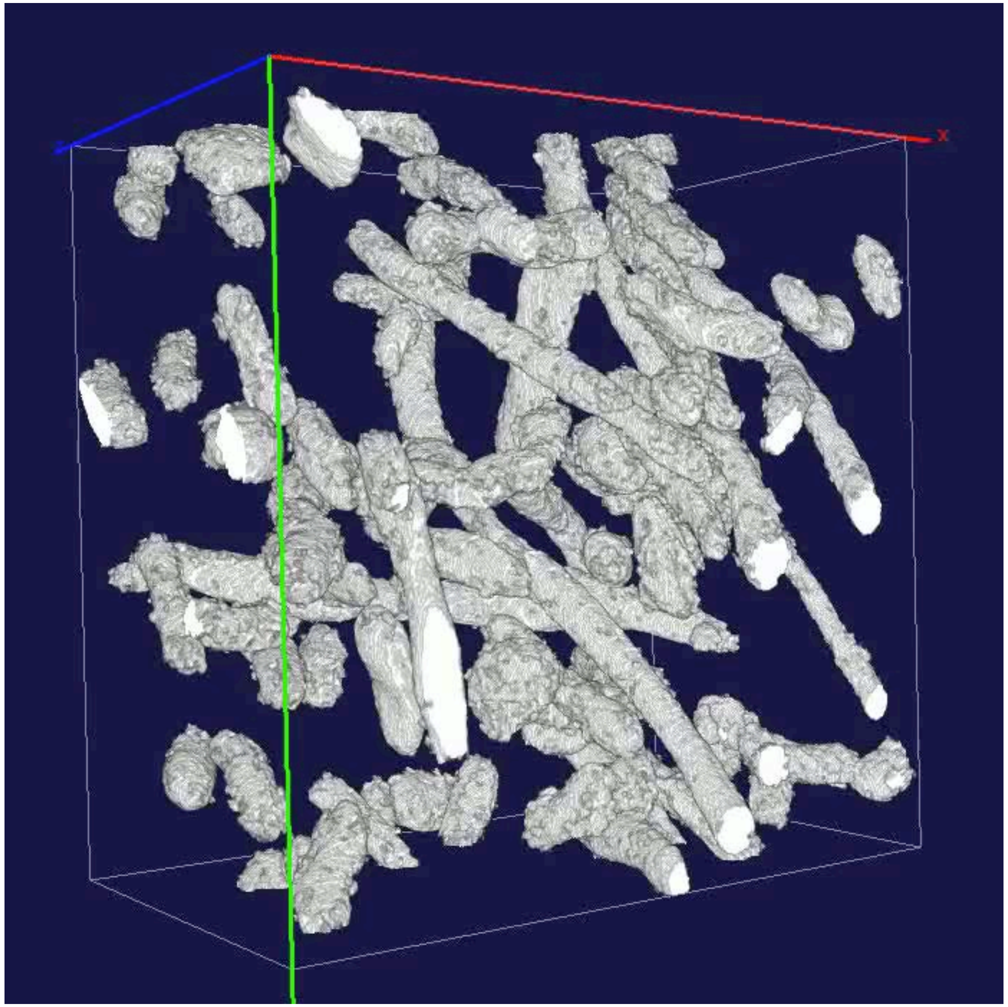
Minimize

$$E(y|x,\lambda) = \sum_i \underbrace{\psi(y_i|x_i)}_{\text{unary term}} + \lambda \sum_{(i,j)\in\mathcal{E}} \underbrace{\phi(y_i, y_j|x_i, x_j)}_{\text{pairwise term}} \ ,$$

with respect to y.

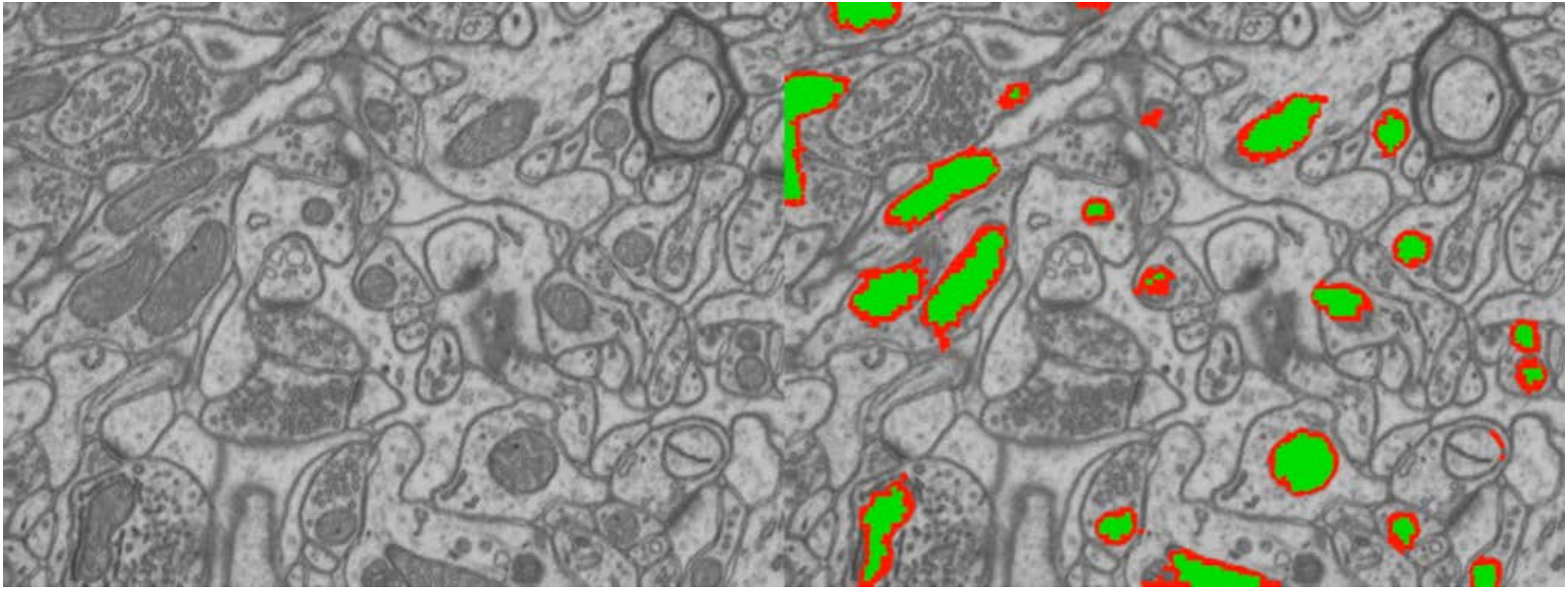
3D MITOCHONDRIA





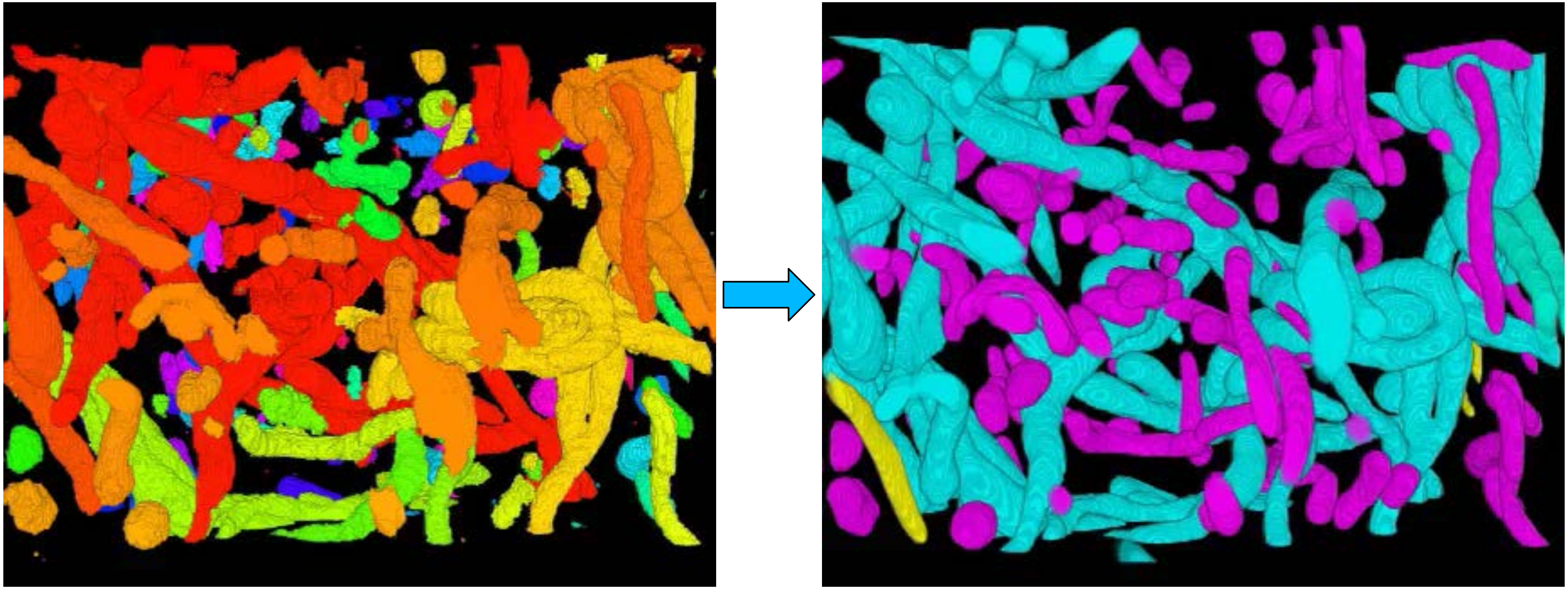
Lucchi et al. TMI’11

MODELING MEMBRANES



Explicitly model membranes as separate regions and exploit the fact that the inside is enclosed within them to retain the graph cut formulation.

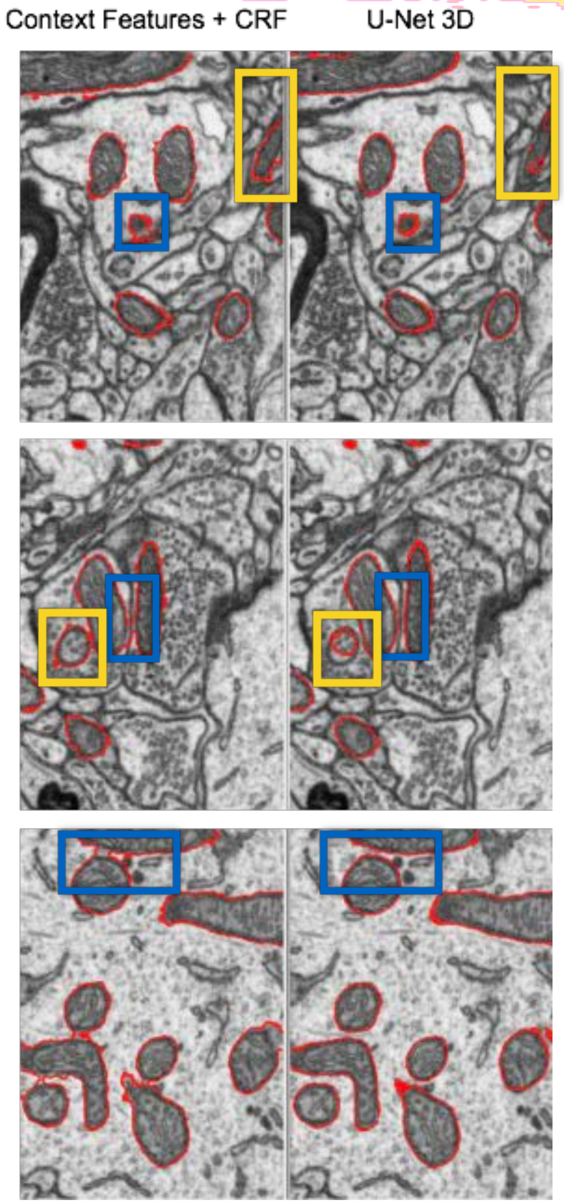
SAVING TIME



3.21 μm × 3.21 μm × 1.08 μm: 53 mitochondria

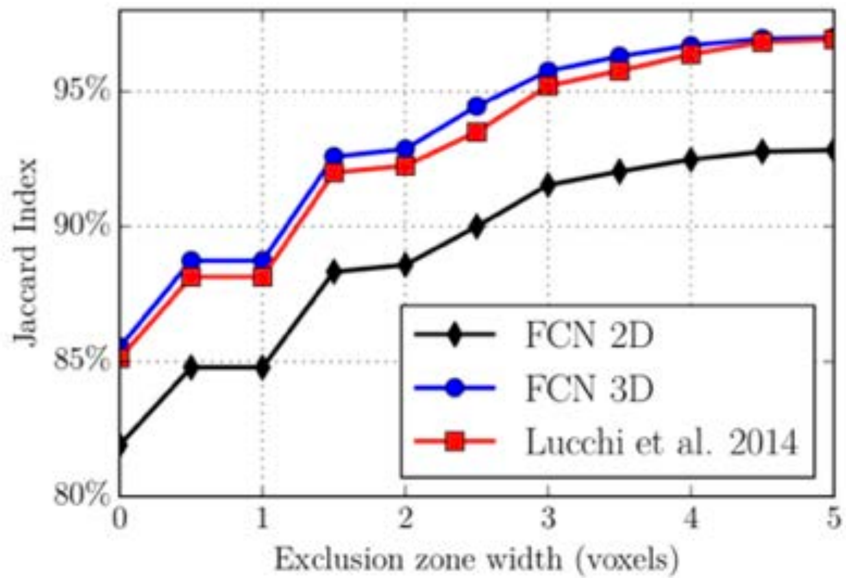
By hand: 6 hours. Semi-automatically: 1.5 hours

INTRODUCING DEEP NETS



Striatum Mitochondria

Method	Jaccard Index
Context F. + CRF	84.6%
U-Net 2D	82.4%
U-Net 3D	86.1%



IN SHORT

- Low-level methods can provide valuable data but are inherently limited.
- Domain knowledge, user interaction, and training data can be used to turn this data into usable results.
- Same philosophy as for delineation.

WHAT ABOUT THE DOG?