Image Segmentation

Jayanta Mukhopadhyay
Dept. of Computer Science and Engg.

Segmentation

- Process by which compact representation of interesting image data is derived.
- The interesting property depends upon the final objectives of processing.
  - Summarizing videos
  - Finding machined parts.
  - Finding people.
  - Finding roads in a satellite image, etc.

Problem formulation: Mathematical modelling

- Typical examples :
  - Segments defined by roughly coherent texture, color, geometric primitives, etc.
  - Fitting lines, curves, etc., to edge points.
  - Fitting a fundamental matrix to a set of feature points.
  - Clustering over a feature space.
  - Component labelling /clique finding in a graph.
  - Modelling class probability distributions.
  - Hypothesis testing.
  - Optimization of energy functions related to formation of shape, partitioning of a feature space, etc.

Active contours

- Methods for locating boundary curves in images.
  - Snakes (Kass, Witkin & Terzopoulos (1988))
    - Energy minimizing, 2-D spline curve that evolves or moves toward image features such as strong edges.
  - Intelligent Scissors (Mortensen and Barret (1995))
    - Allows the users to sketch a curve in real time that clings to object boundaries.
  - Level set techniques
    - Evolves the curve as zero-set characteristic function.

Snakes

Curve: $\overrightarrow{f(s)} = (x(s), y(s))$

To minimize the spline internal energy

$$E_{int} = \int_s \left(\alpha(s) \left\| \vec{f}_s(s) \right\|^2 + \beta(s) \left\| \vec{f}_{ss}(s) \right\|^2 \right) ds$$

Discrete form of spline internal energy

$$E_{int} = \sum_i \left(\alpha(i) \frac{\left\| \vec{f}(i+1) - \vec{f}(i) \right\|^2}{h^2} + \beta(i) \frac{\left\| \vec{f}(i+1) - 2\vec{f}(i) + \vec{f}(i-1) \right\|^2}{h^4} \right)$$

Step size

Snakes : Computation

- Start with initial sampling
- Resample after every iteration.
- In addition to internal spline energy (E_{int}), it minimizes external image based and constrained based potential.
  - $E_{image} = w_{line}.E_{line} + w_{edge}.E_{edge} + w_{term}.E_{term}$
  - Attracts to: Dark ridges | Strong gradients | Line terminators

- Opposite to snake dynamics is ballooning.

Additional consideration

- Use of distance map on extracted edges.
- Attractive forces toward anchor points (say, d), e.g. spring model

$$E_{spring} = k_i \left\| \overrightarrow{f(i)} - \vec{d} \right\|^2$$

- Repulsive force (inversely proportional to distance)

Energy terms and minimization

$$v_i \equiv (x_i, y_i) \equiv \big(x(ih), y(ih)\big)$$

$$E_{int}(i) = \alpha(i)\frac{\|v_i - v_{i-1}\|^2}{2h^2} + \beta(i)\frac{\|v_{i+1} - 2v_i + v_{i-1}\|^2}{2h^4}$$

$$E_{snake} = \sum_i \left(E_{int}(i) + E_{ext}(i)\right)$$

For minimization of the snake energy we need to solve Euler equations:

$$\alpha x_{ss} + \beta x_{ssss} + \frac{\partial E_{\text{ext}}}{\partial x} = 0$$

$$\alpha y_{ss} + \beta y_{ssss} + \frac{\partial E_{\text{ext}}}{\partial y} = 0$$

Euler equations in the discrete space

Let $v(0) = v(n)$ $f_x(i) = \dfrac{\partial E_{ext}}{\partial x_i}$ $f_y(i) = \dfrac{\partial E_{ext}}{\partial y_i}$

Corresponding Euler equation:

$$\begin{aligned}
&\alpha_i(\mathbf{v}_i - \mathbf{v}_{i-1}) - \alpha_{i+1}(\mathbf{v}_{i+1} - \mathbf{v}_i) \\
&\quad + \beta_{i-1}[\mathbf{v}_{i-2} - 2\mathbf{v}_{i-1} + \mathbf{v}_i] \\
&\quad - 2\beta_i[\mathbf{v}_{i-1} - 2\mathbf{v}_i + \mathbf{v}_{i+1}] \\
&\quad + \beta_{i+1}[\mathbf{v}_i - 2\mathbf{v}_{i+1} + \mathbf{v}_{i+2}] \\
&\quad + (f_x(i), f_y(i)) = 0
\end{aligned}$$

In matrix form

$$\mathbf{A}\mathbf{x} + \mathbf{f}_{\mathbf{x}}(\mathbf{x}, \mathbf{y}) = 0$$

$$\mathbf{A}\mathbf{y} + \mathbf{f}_{\mathbf{y}}(\mathbf{x}, \mathbf{y}) = 0$$

Where A is a pentadiagonal matrix.

Euler equations in the discrete space

Euler equation:

$$\alpha_i(\mathbf{v}_i - \mathbf{v}_{i-1}) - \alpha_{i+1}(\mathbf{v}_{i+1} - \mathbf{v}_i) + \beta_{i-1}[\mathbf{v}_{i-2} - 2\mathbf{v}_{i-1} + \mathbf{v}_i] - 2\beta_i[\mathbf{v}_{i-1} - 2\mathbf{v}_i + \mathbf{v}_{i+1}] + \beta_{i+1}[\mathbf{v}_i - 2\mathbf{v}_{i+1} + \mathbf{v}_{i+2}] + (f_x(i), f_y(i)) = 0$$

In matrix form

$$\mathbf{A}\mathbf{x} + \mathbf{f}_\mathbf{x}(\mathbf{x}, \mathbf{y}) = 0$$

$$\mathbf{A}\mathbf{y} + \mathbf{f}_\mathbf{y}(\mathbf{x}, \mathbf{y}) = 0$$

Where A is a pentadiagonal matrix.

Assume constants at every point

Constant between two iterative stages

Iterative formulation

Step size

$$\mathbf{A}\mathbf{x}_t + \mathbf{f}_\mathbf{x}(\mathbf{x}_{t-1}, \mathbf{y}_{t-1}) = -\gamma(\mathbf{x}_t - \mathbf{x}_{t-1})$$

$$\mathbf{A}\mathbf{y}_t + \mathbf{f}_\mathbf{y}(\mathbf{x}_{t-1}, \mathbf{y}_{t-1}) = -\gamma(\mathbf{y}_t - \mathbf{y}_{t-1})$$

Euler equations in the discrete space

Iterative formulation

Step size

$$\mathbf{A}\mathbf{x}_t + \mathbf{f}_\mathbf{x}(\mathbf{x}_{t-1}, \mathbf{y}_{t-1}) = -\gamma(\mathbf{x}_t - \mathbf{x}_{t-1})$$

$$\mathbf{A}\mathbf{y}_t + \mathbf{f}_\mathbf{y}(\mathbf{x}_{t-1}, \mathbf{y}_{t-1}) = -\gamma(\mathbf{y}_t - \mathbf{y}_{t-1})$$

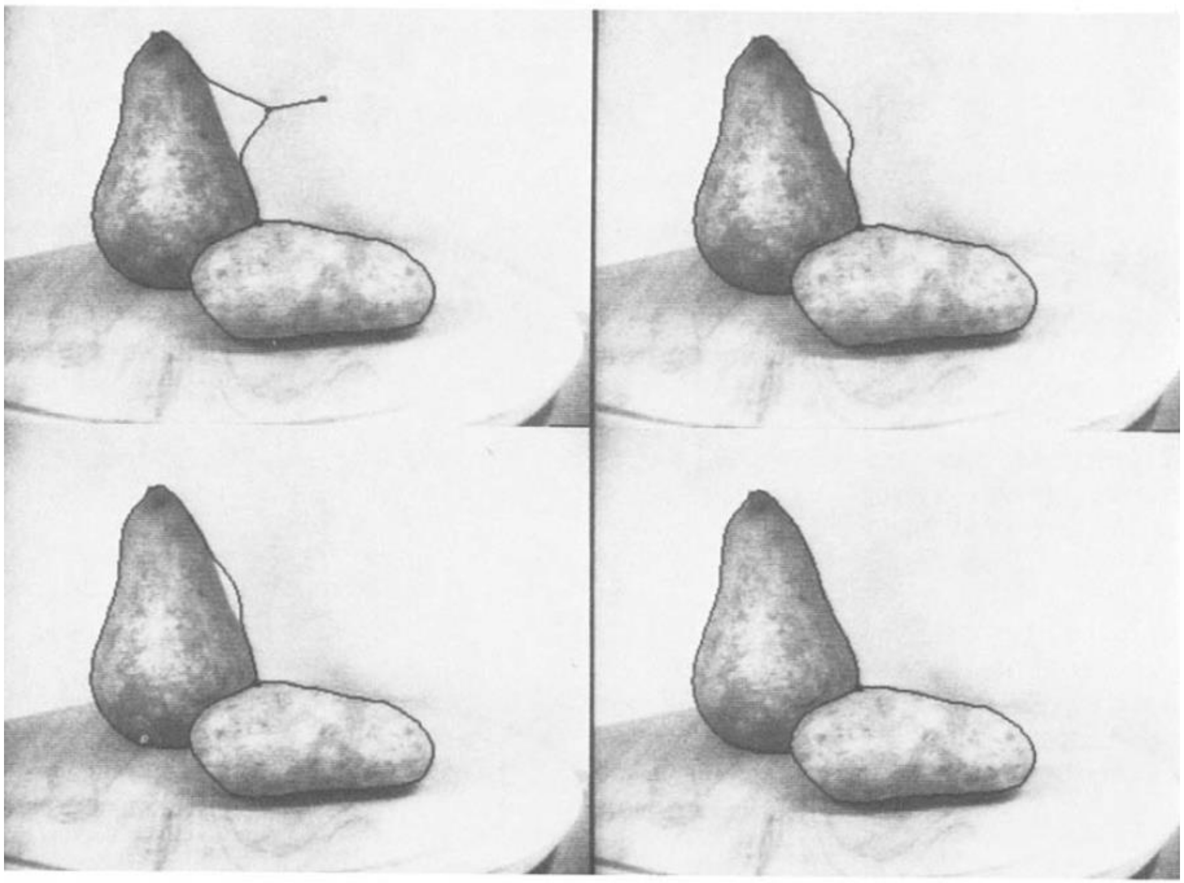
As it is pentadiagonal banded, inverse can be computed through LU decomposition in O(N).

Iterative updates:

$$\mathbf{x}_t = (\mathbf{A} + \gamma\mathbf{I})^{-1}(\mathbf{x}_{t-1} - \mathbf{f}_\mathbf{x}(x_{t-1}, y_{t-1}))$$

$$\mathbf{y}_t = (\mathbf{A} + \gamma\mathbf{I})^{-1}(\mathbf{y}_{t-1} - \mathbf{f}_\mathbf{y}(x_{t-1}, y_{t-1}))$$

A typical example



From “Snakes: Active Contours: Kass, Witkin, and Tzopouloser, IJCV, (1988)”

Mode finding techniques

- Represent each pixel by a feature descriptor.
- Mode finding techniques compute the modes of pdf in the feature space.

  - K-means clustering
    - No explicit estimation of pdf
  - Mixture of Gaussians
    - Parametric estimation of pdf
  - Mean shift technique (Comaniciu and Meer, PAMI, 2003)
    - Non-parametric estimation of pdf.

K-means clustering

- Implicitly models the prob. density as a superposition of spherically symmetric distributions.
- Does not require any probabilistic reasoning or modeling.
- Given k, computes k clusters.
- Assume k initial cluster centers and iteratively converge to them.

K-means clustering (contd.)

$$E = \sum_{k} \sum_{\forall x \in c_k} \|x - c_k\|^2$$

- Given k initial centers, assign a point to the cluster represented by its center, if it is the closest among them.
- Re-compute the centers.
- Iterate above two steps, till the centers do not change their positions.
- Trying to minimize the energy function defined by the sum of divergences of each cluster from its center.
- May get stuck at local minima.

Mixture of Gaussians

- Each cluster center is augmented by a covariance matrix, whose values are re-estimated from corresponding samples.
- Mahalanabis distance function:

$$d(x, \mu_k; \Sigma_k) = (x - \mu_k)^T \Sigma_k^{-1} (x - \mu_k)$$

Cluster center   Covariance matrix

Parametric PDF: $p(x|\{\pi_k, \mu_k, \Sigma_k\}) = \sum_k \pi_k N(x|\mu_k, \Sigma_k)$

$where,\ N(x|\mu_k, \Sigma_k) = \frac{1}{|\Sigma_k|} e^{-d(x,\mu_k;\Sigma_k)^2}$ Mixing coefficients

Expectation Maximization (EM) Algorithm

$$z_{ik} = \frac{1}{Z_i} \pi_k N(x|\mu_k, \Sigma_k)$$

Normalizing factor

$$Z_i = \sum_k z_{ik}$$

- Start with initial set :$\{\pi_k, \mu_k, \Sigma_k\}$.
- E-Step (Expectation stage)
  - Compute likelihood (z_{ik}) of x belonging to kth Gaussian cluster.
  - Assign x to the mth cluster whose *likelihood* is maximum.
- M-Step (Maximization Stage)
  - Re-estimate parameters ($\{\pi_k, \mu_k, \Sigma_k\}$) from class distribution
- Iterate above two steps till it converges.

Parameter re-estimation

$$z_{ik} = \frac{1}{Z_i} \pi_k N(x|\mu_k, \Sigma_k)$$

Normalizing factor

$$N_k = \sum_i Z_{ik}$$

Expected number of pixels in class k.

$$\mu_k = \frac{1}{N_k} \sum_i z_{ik}\, x_i$$

$$\Sigma_k = \frac{1}{N_k} \sum_i z_{ik}\, (x_i - \mu_k)(x_i - \mu_k)^T$$

$$\pi_k = \frac{N_k}{N}$$

Segmentation using Mean Shift Computation

- Efficiently finds peak of a high dimensional data distribution without computing the complete pdf.
- Steps for segmentation.
  - Find major peaks.
  - Trace path from a pixel to peak.
  - Associate pixels climbing to the same peak.
- Inverse of watershed algorithm, which climbs downhill to find basins of attraction.

Estimation of pdf

- Given a sparse set of samples $\{x_i\}$, smooth the data by convolving with a kernel of width h.

$k(r)$: Kernel function or Parzen window.

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

- For computing peak a brute force computation could have been employed.
- In mean shift, we compute gradient of pdf to move to peak.

Estimation of gradient in pdf

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

- Compute gradient to move along the direction of +ve gradient to climb the peak.

$$\nabla f(x) = \sum_i (x_i - x)\, G(x - x_i) = \sum_i (x_i - x)\; g\left(\frac{\|x - x_i\|^2}{h^2}\right)$$

$$g(r) = k'(r)$$

$$\Rightarrow \nabla f(x) = \left[\sum_i G(x - x_i)\right] m(x)$$

$$m(x) = \frac{\sum_i x_i G(x - x_i)}{\sum_i G(x - x_i)} - x$$

Mean shift

Iterative updates of modes

- Current estimate of the mode at y_t at iteration t is replaced by its locally weighted mean.

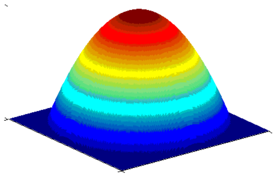
$$y_{t+1} = y_t + m(y_t) = \frac{\sum_i x_i G(y_t - x_i)}{\sum_i G(y_t - x_i)}$$

- Convergence is guaranteed if $k(r)$ is a monotonically decreasing function.

Various kernel functions

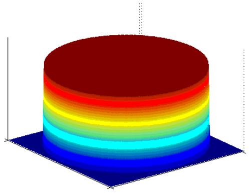
- Epanechnikov Kernel

$$K_E(\mathbf{x}) = \begin{cases} c\left(1-\|\mathbf{x}\|^2\right) & \|\mathbf{x}\| \le 1 \\ 0 & \text{otherwise} \end{cases}$$



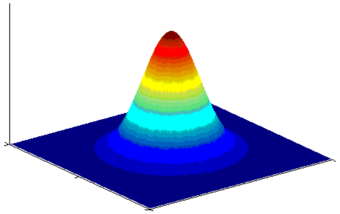
- Uniform Kernel

$$K_U(\mathbf{x}) = \begin{cases} c & \|\mathbf{x}\| \le 1 \\ 0 & \text{otherwise} \end{cases}$$

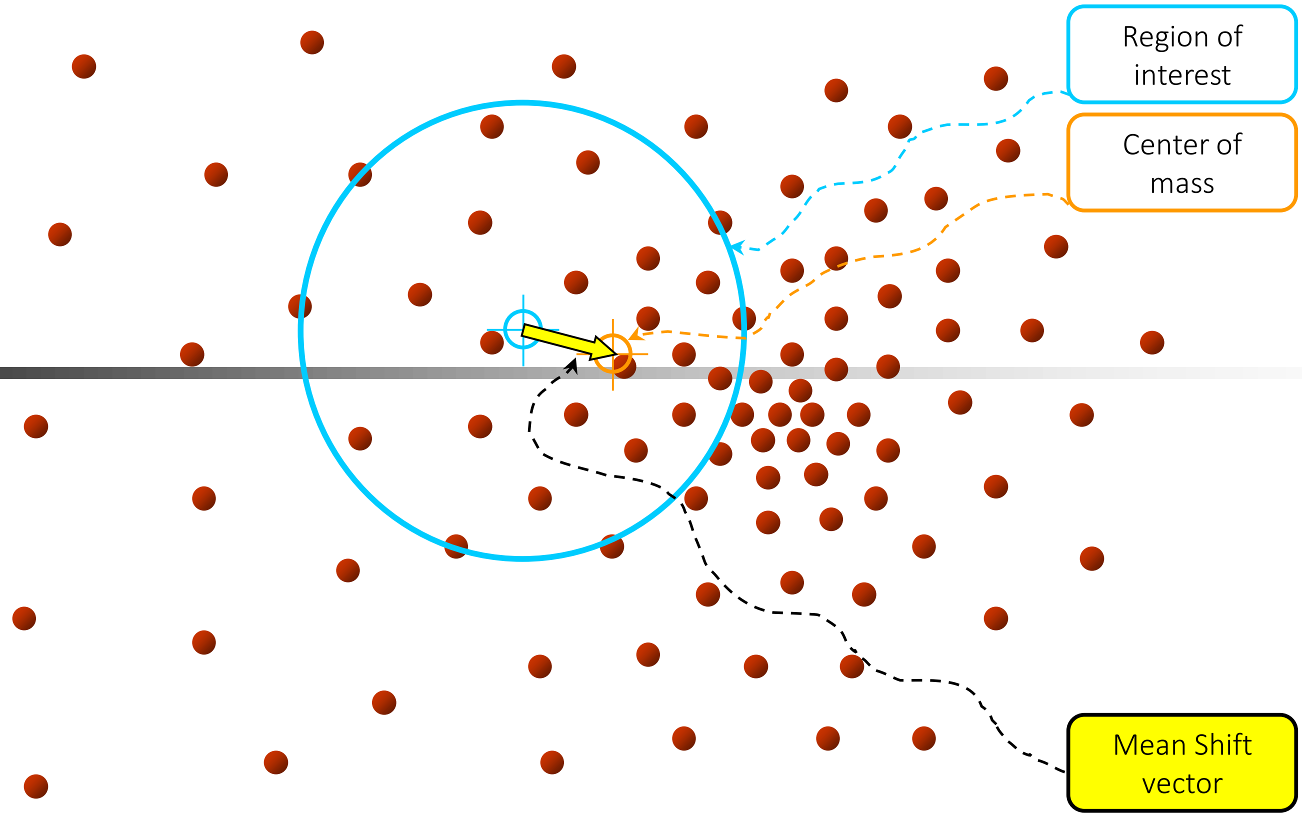


- Normal Kernel

$$K_N(\mathbf{x}) = c \cdot \exp\left(-\frac{1}{2}\|\mathbf{x}\|^2\right)$$



Intuitive Description

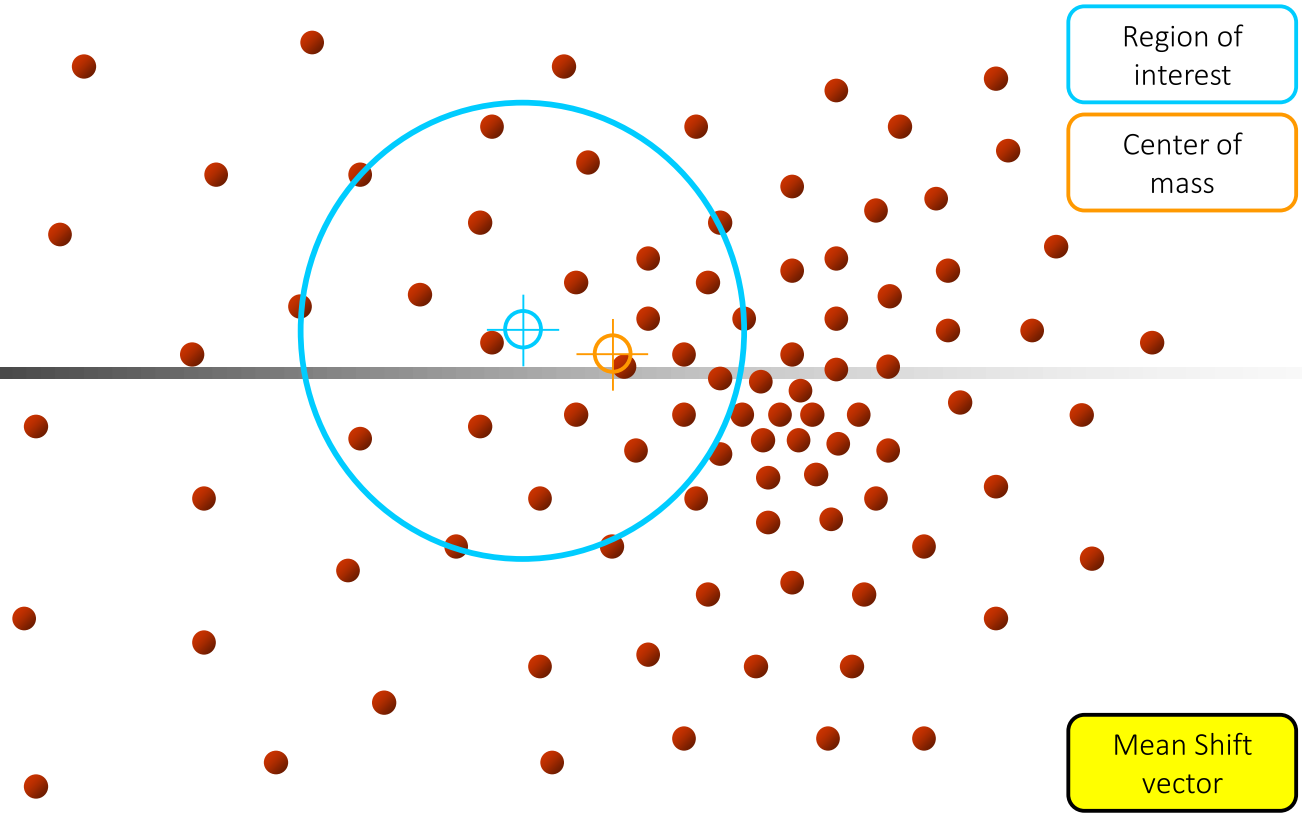


Objective : Find the densest region

Distribution of identical billiard balls

h: size of window

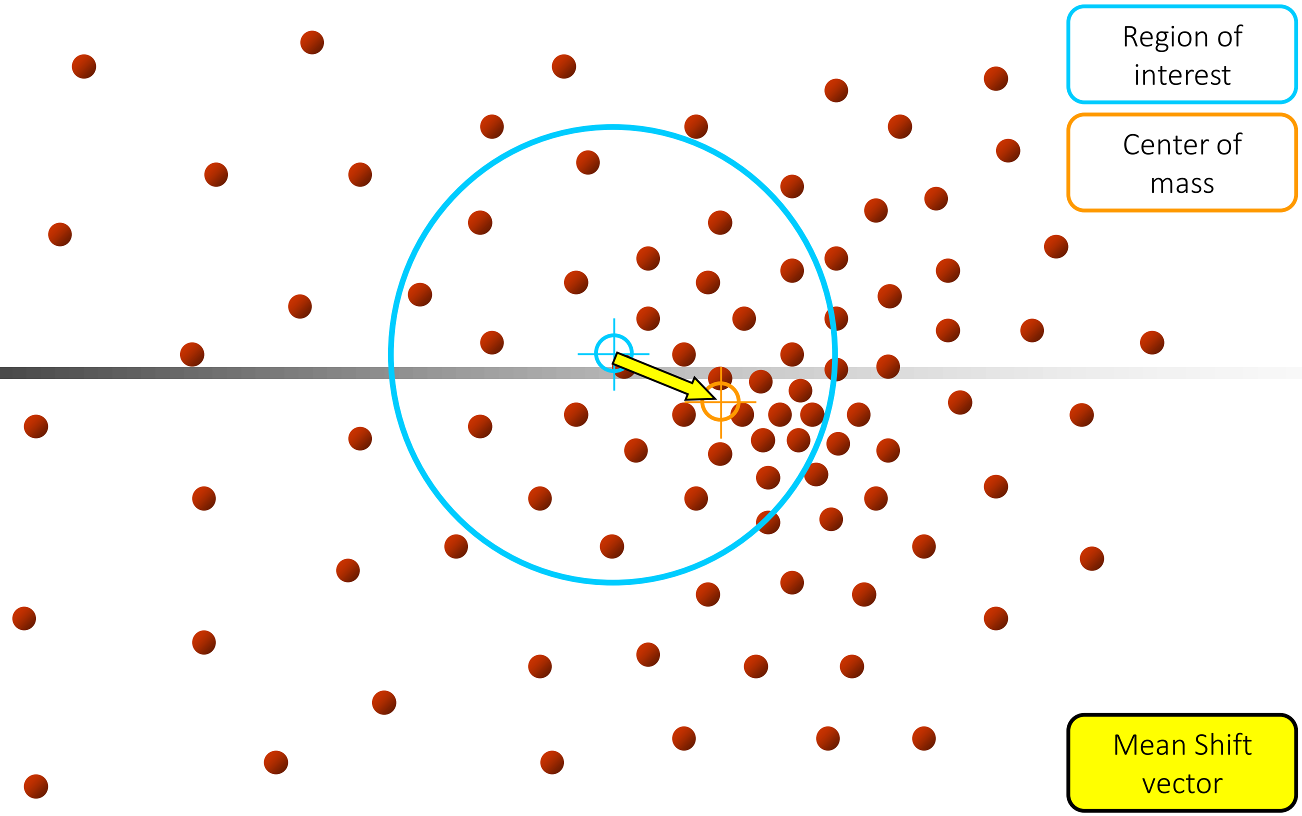
Intuitive Description



Objective : Find the densest region

Distribution of identical billiard balls

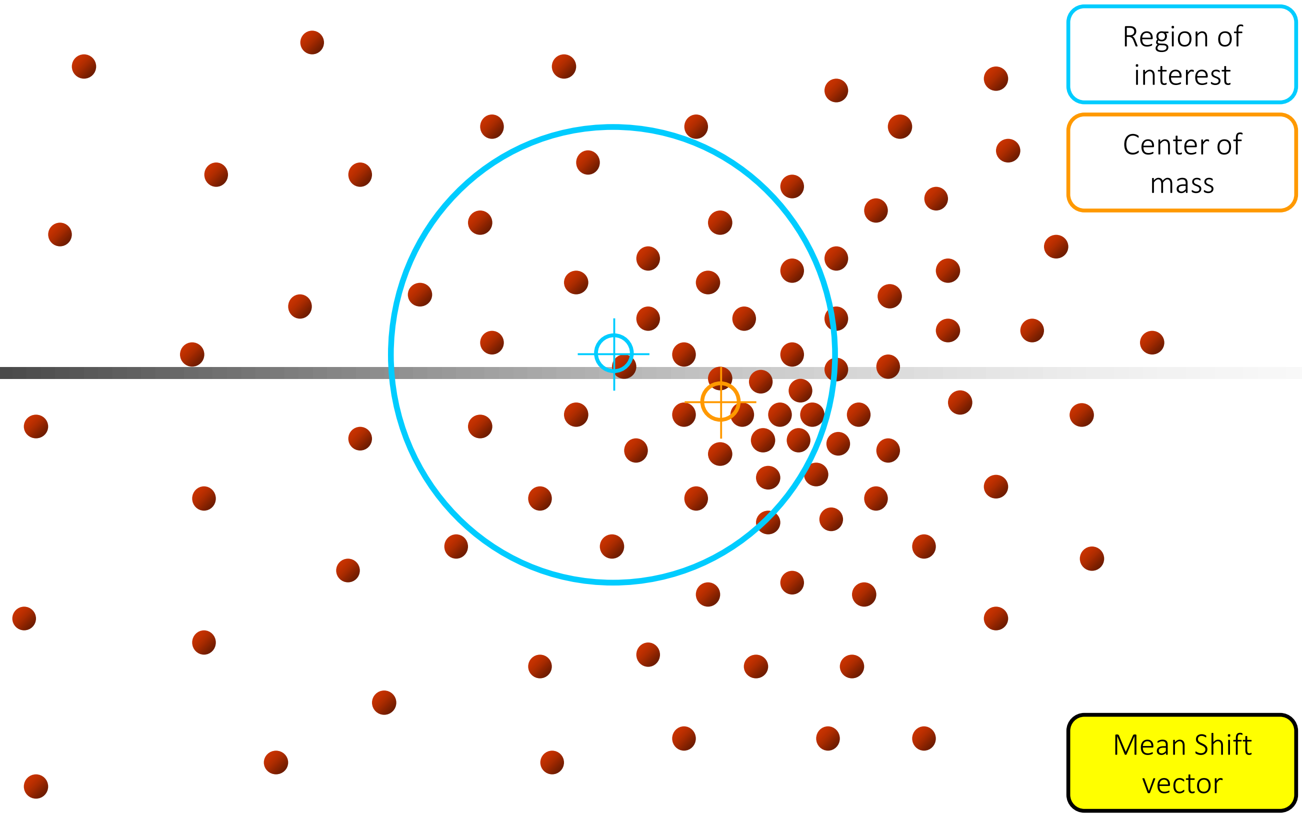
Intuitive Description



Objective : Find the densest region

Distribution of identical billiard balls

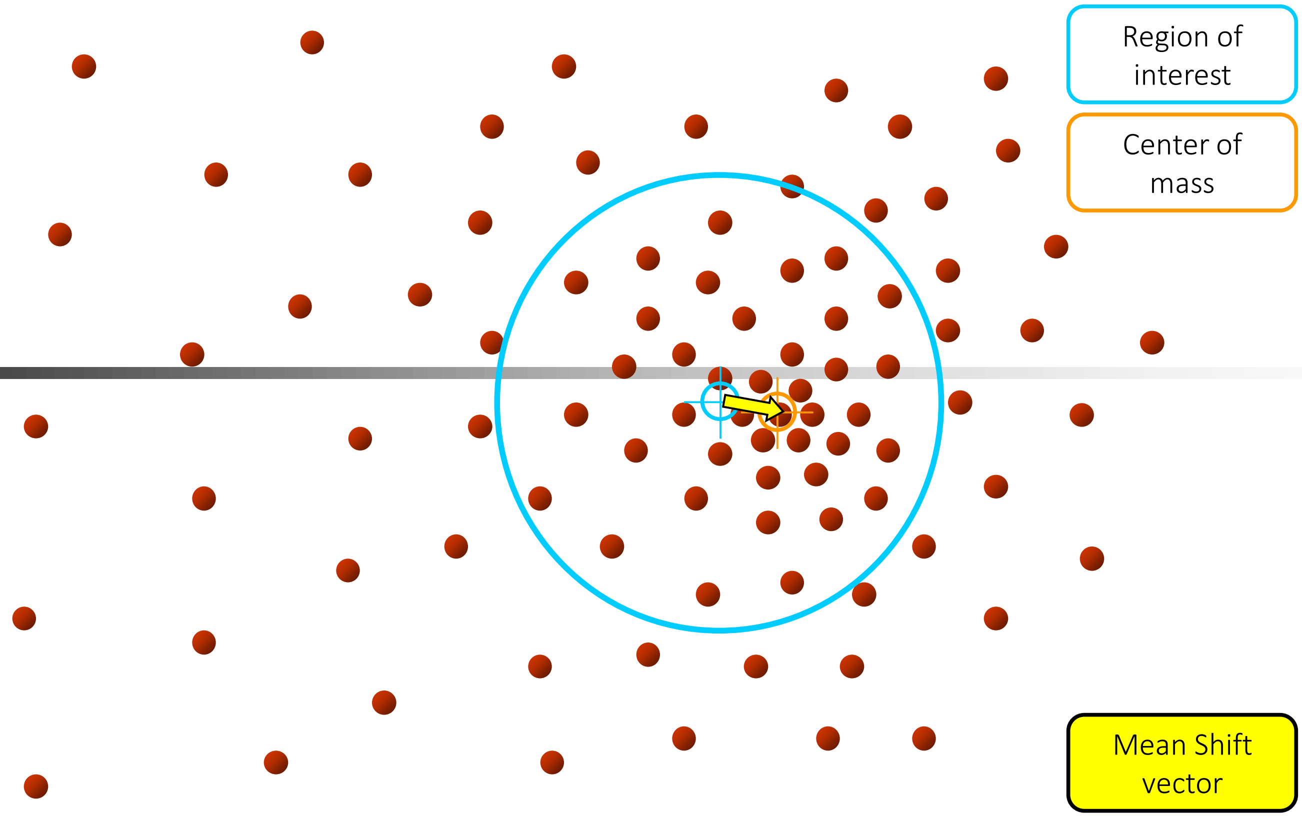
Intuitive Description



Objective : Find the densest region

Distribution of identical billiard balls

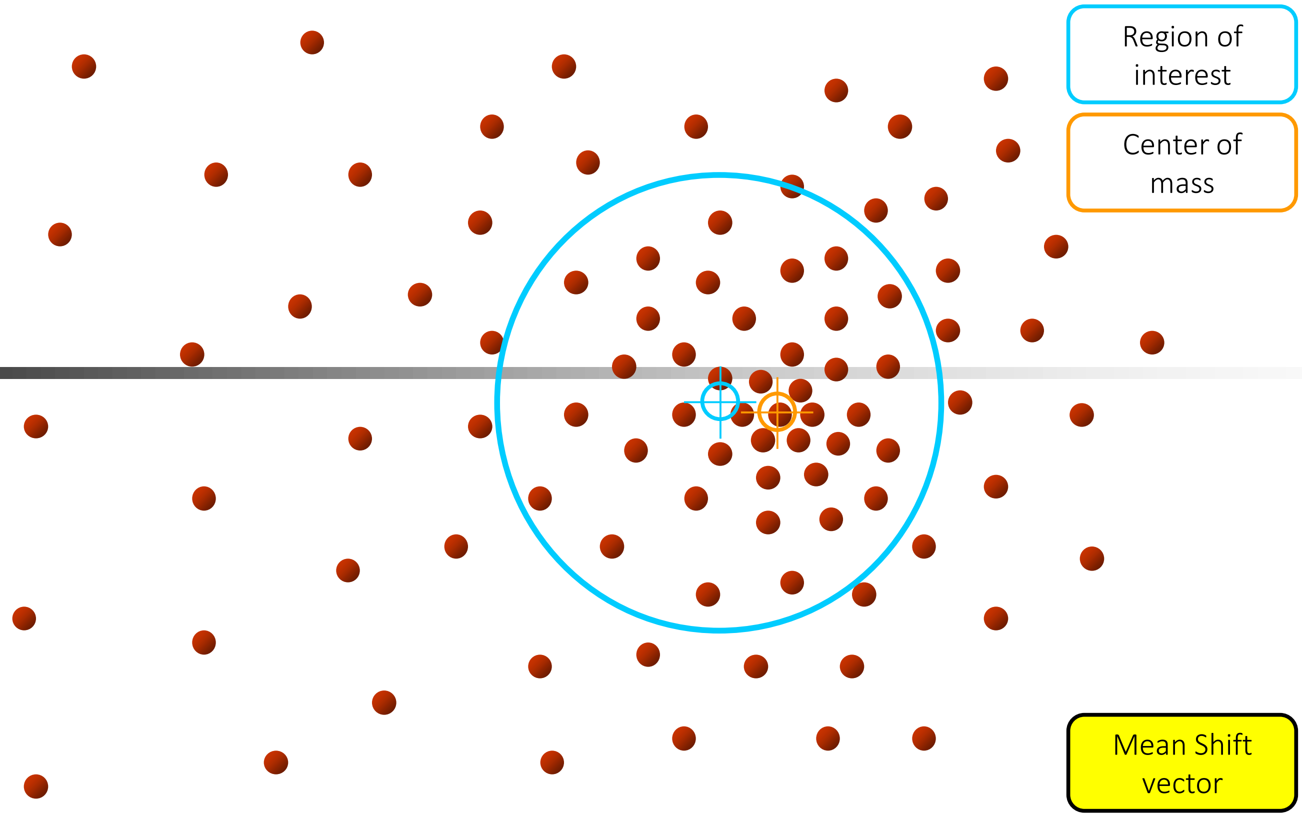
Intuitive Description



Objective : Find the densest region

Distribution of identical billiard balls

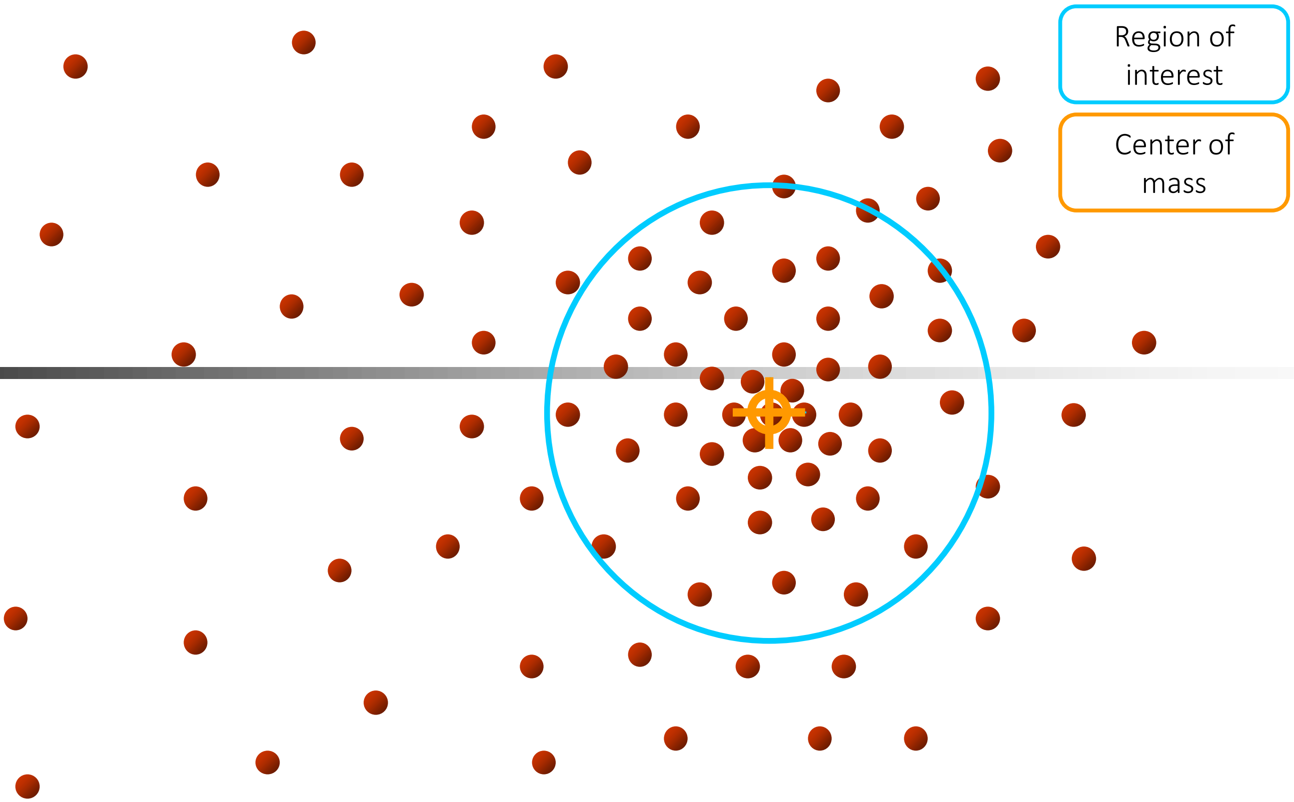
Intuitive Description



Objective : Find the densest region

Distribution of identical billiard balls

Intuitive Description



Objective : Find the densest region

Distribution of identical billiard balls

Segmentation Algorithm

- Cluster all local peaks which are closely spaced (into a few clusters).
- If from a pixel x, a local peak in ith cluster is obtained, assign segment i to x.

Color Segmentation:
Use joint domain of color(x_r) and location(x_s)

$$k(x_j) = k\left(\frac{\|x_r\|^2}{{h_r}^2}\right) k\left(\frac{\|x_s\|^2}{{h_s}^2}\right)$$

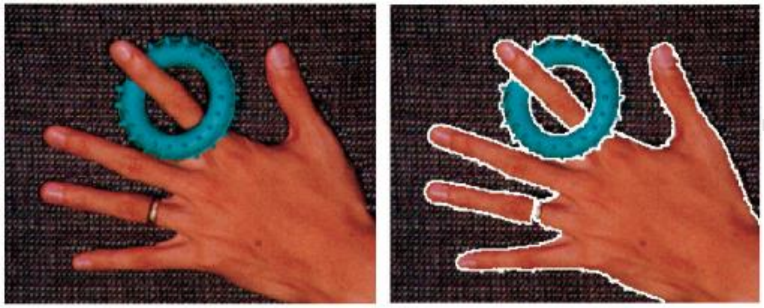
Segmentation Example

…when feature space is only gray levels…

Segmentation Example

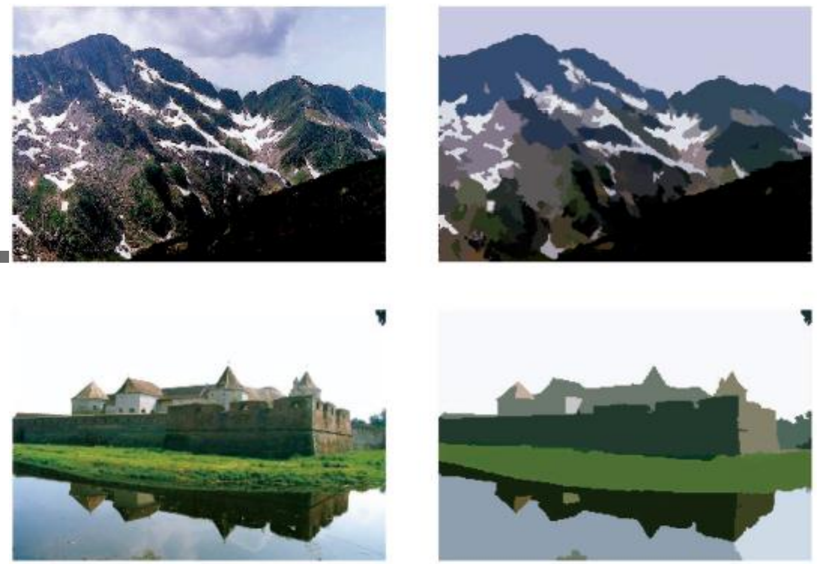
Segmentation Example

Segmentation Example



Segmentation Example

Segmentation Example



Segmentation Example

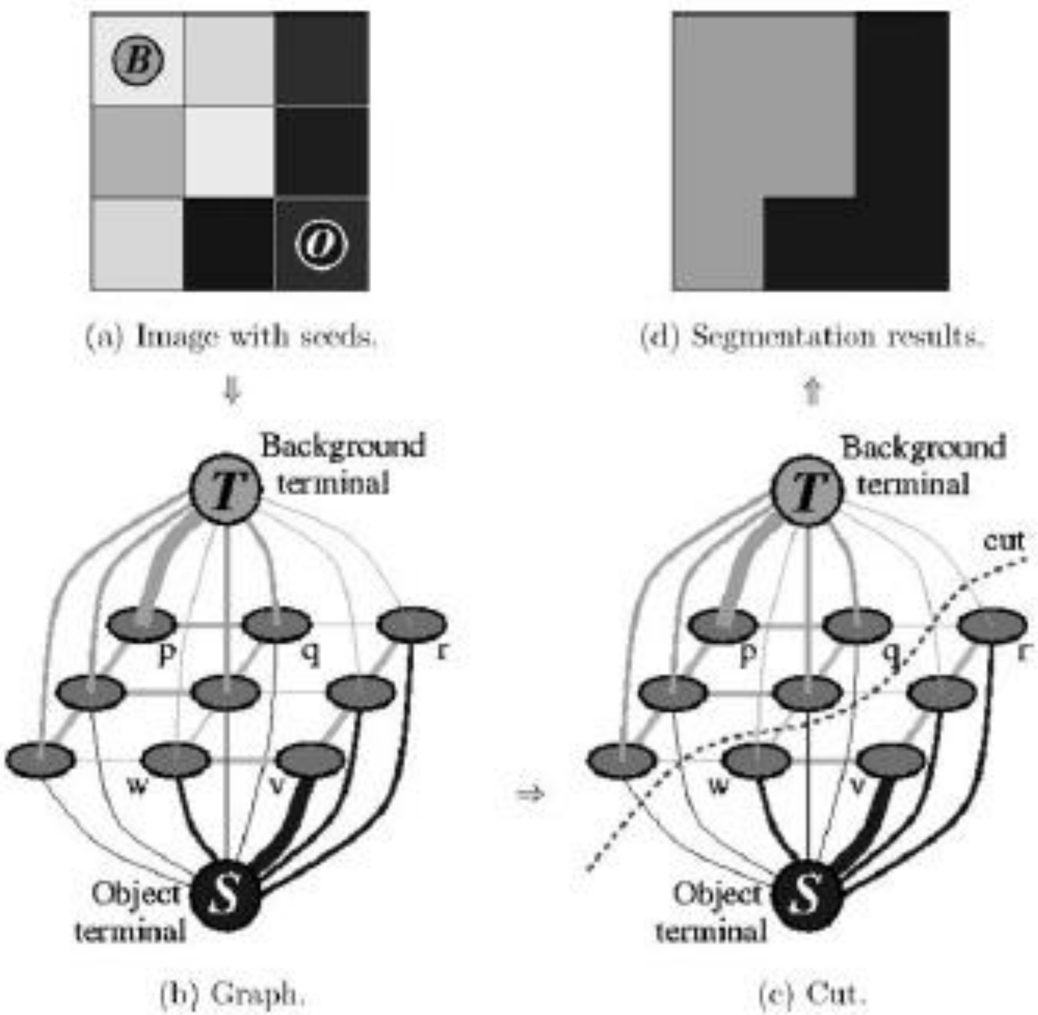
Graph cuts

- A global optimization technique.
- $G(V,E)$
  - V: Nodes as pixels and two special terminal nodes s (foreground), and t (background).
  - E: Two types of edges
    - n-links (edges between neighbors)
    - t-links (edges from s / t to a pixel).
    - w_e: weight of an edge indicating affinity of belongingness to the same segment.

s-t cut

- A subset C of edges E, such that the terminals s and t become completely separated on the induced graph, $G(C)=\{V,E\backslash C\}$
- To compute optimal cut such that $|C|$=*Sum of the weights of edges in C* is minimized.
  - Severed n-links located in the segment boundary.
  - Their total cost represents the cost of segment boundary.
  - Severed t-links represent regional properties of segments.
- Minimum cost corresponds to desirable balance between regional and boundary properties.

An illustration



Graph Cuts and Efficient N-D Image Segmentation, Boykov et al, IJCV (2006).

Energy function

- P: Set of all pixels.
- N: Neighborhood system: $\{(p,q) | (p,q) \text{ in } P\}$
- A: $\{A_1, A_2, \ldots, A_{|P|}\}$ ➔ A binary vector representing assignment of each pixel to background (0) or foreground (1).
- $E(A)$: Energy for the configuration in A.

$$E(A) = \lambda R(A) + B(A), \lambda > 0$$

$= 1, if\ A_p \neq A_q$
$= 0$, otherwise

Region term

$$R(A) = \sum_{p \in P} R_p(A)$$

Boundary term

$$B(A) = \sum_{(p,q) \in N} B_{p,q}(A).(\delta_{A_p} \neq \delta_{A_q})$$

Typical measures

- Regional measures
  - $R_p(\text{“}obj\text{”}) = -\ln Pr(I_p | \text{"}obj\text{”})$
  - $R_p(\text{“}bkg\text{”}) = -\ln Pr(I_p | \text{”}bkg\text{”})$
- Boundary measures

$$B_{p,q} \propto e^{-\frac{(I_p - I_q)^2}{2\sigma^2}} \frac{1}{dist(p,q)}$$

Optimal solution via graph cut

$$V = P \cup \{s, t\} \qquad E = N \cup \left(\bigcup_{p \in P} \{\{p, s\}, \{p, t\}\} \right)$$

edge	weight (cost)	for
$\{p, q\}$	$B_{p,q}$	$\{p, q\} \in \mathcal{N}$
	$\lambda \cdot R_p(\text{"bkg"})$	$p \in \mathcal{P},\ p \notin \mathcal{O} \cup \mathcal{B}$
$\{p, S\}$	K	$p \in \mathcal{O}$
	0	$p \in \mathcal{B}$
	$\lambda \cdot R_p(\text{"obj"})$	$p \in \mathcal{P},\ p \notin \mathcal{O} \cup \mathcal{B}$
$\{p, T\}$	0	$p \in \mathcal{O}$
	K	$p \in \mathcal{B}$

$$K = 1 + \max_{p \in \mathcal{P}} \sum_{q:\ \{p,q\} \in \mathcal{N}} B_{p,q}.$$

Use of minimum cut / max-flow algorithm (Low order Polynomial time)

Hard constraints

Graph Cuts and Efficient N-D Image Segmentation, Boykov et al, IJCV (2006).

Properties of minimum cut C

- Severs exactly one t-link at each p.
- (p,q) in C iff p and q are t-linked to different terminals.
- If p belongs to Obj, (p,t) belongs to C.
- If p belongs to Bkg, (p,s) belongs to C.

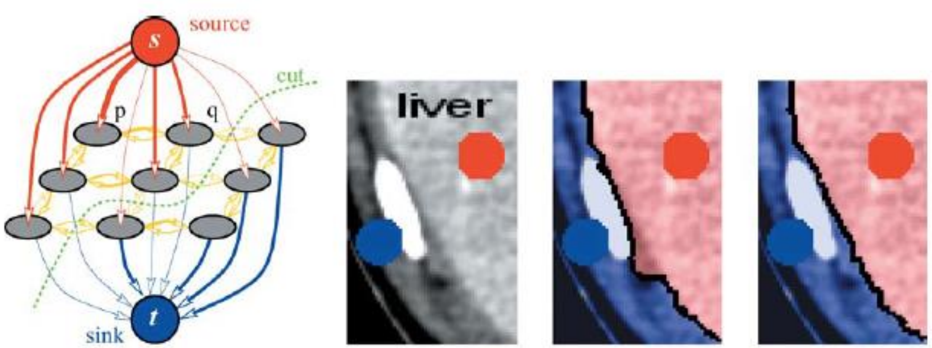
Examples of interactive segmentation

Hard constraints

Graph Cuts and Efficient N-D Image Segmentation, Boykov et al, IJCV (2006).

Use of directed graph

- The same method is applicable to directed graph.
- Sometimes produce better results (if the transitions between two segments are differentially preferred).

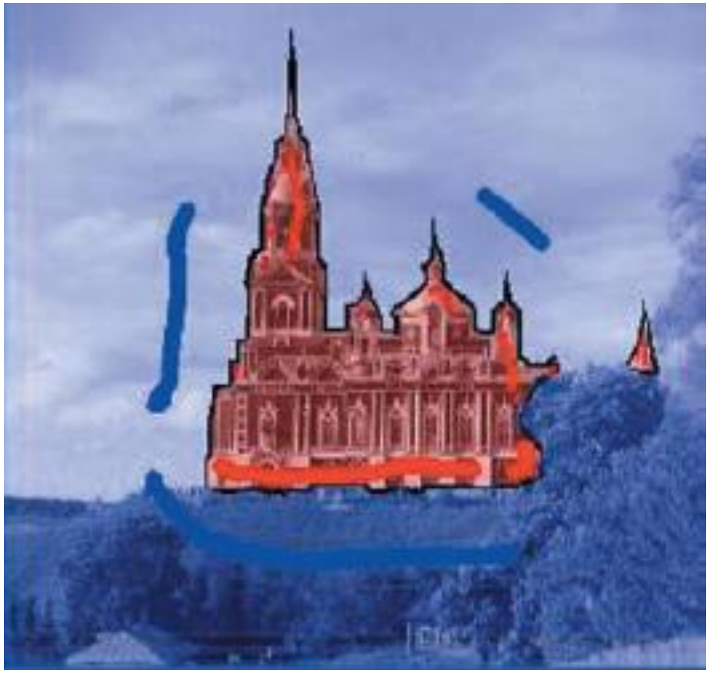


(a) directed graph (b) image (c) undir. result (d) dir. result

$$w_{(p,q)} = \begin{cases} 1 & \text{if } I_p \le I_q \\ \exp\left(-\frac{(I_p - I_q)^2}{2\sigma^2}\right) & \text{if } I_p > I_q \end{cases}$$

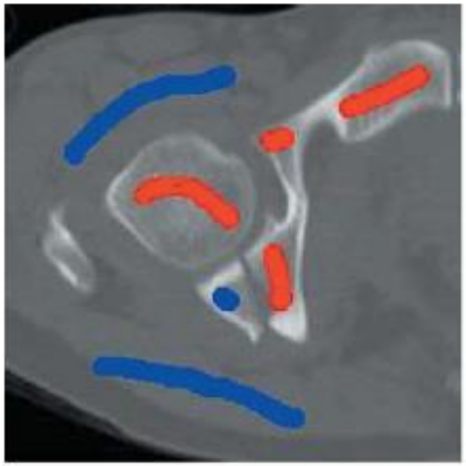
Graph Cuts and Efficient N-D Image Segmentation, Boykov et al, IJCV (2006).

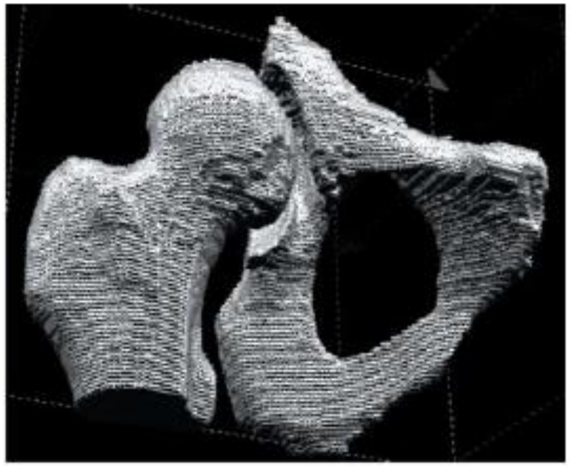
Application: Photo Editing

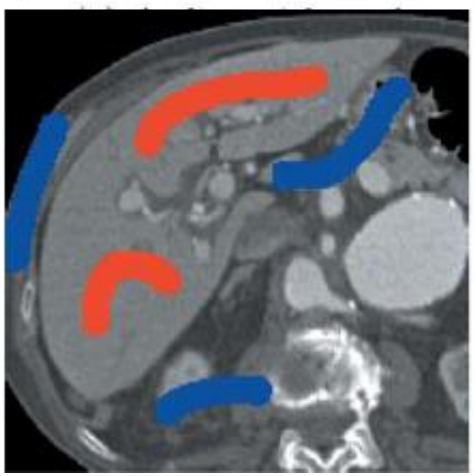


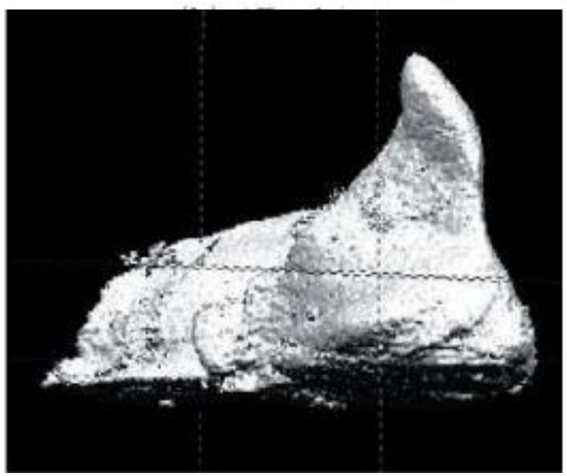
Graph Cuts and Efficient N-D Image Segmentation, Boykov et al, IJCV (2006).

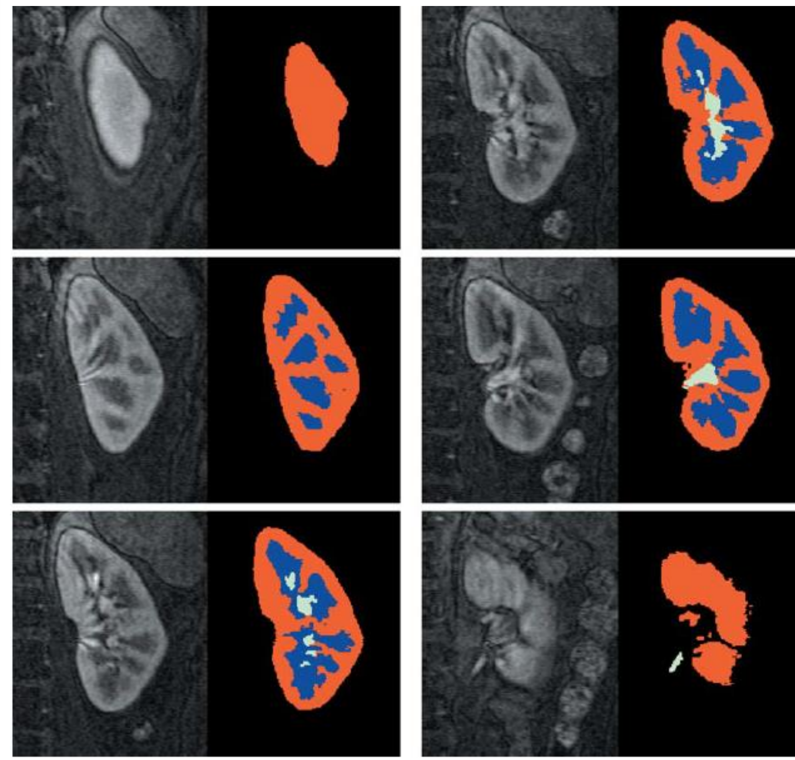
Segmentation of 3-D medical images











Summary

- Energy optimization approaches
  - Active contouring
  - Graph cuts
- Mode finding methods
  - K-means clustering
  - Mixture of Gaussian modeling
  - Mean shift algorithm

Thank you!