IMAGE SEGMENTATION

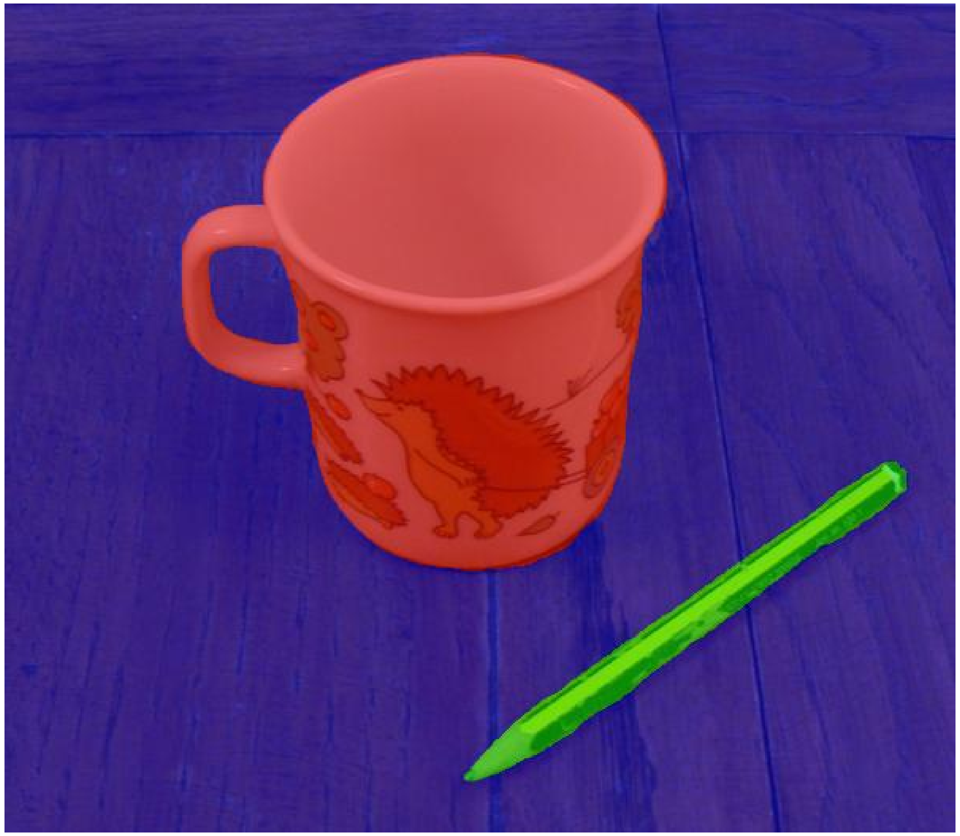
Václav Hlaváč

Czech Technical University in Prague
Faculty of Electrical Engineering, Department of Cybernetics
Center for Machine Perception
`http://cmp.felk.cvut.cz/~hlavac, hlavac@fel.cvut.cz`

Outline of the talk:

- What is segmentation? Segmentation is application dependent because it needs image interpretation.
- Taxonomy of segmentation methods.
- Thresholding-based segmentation.
- . . . *the rest comes in another presentation.*

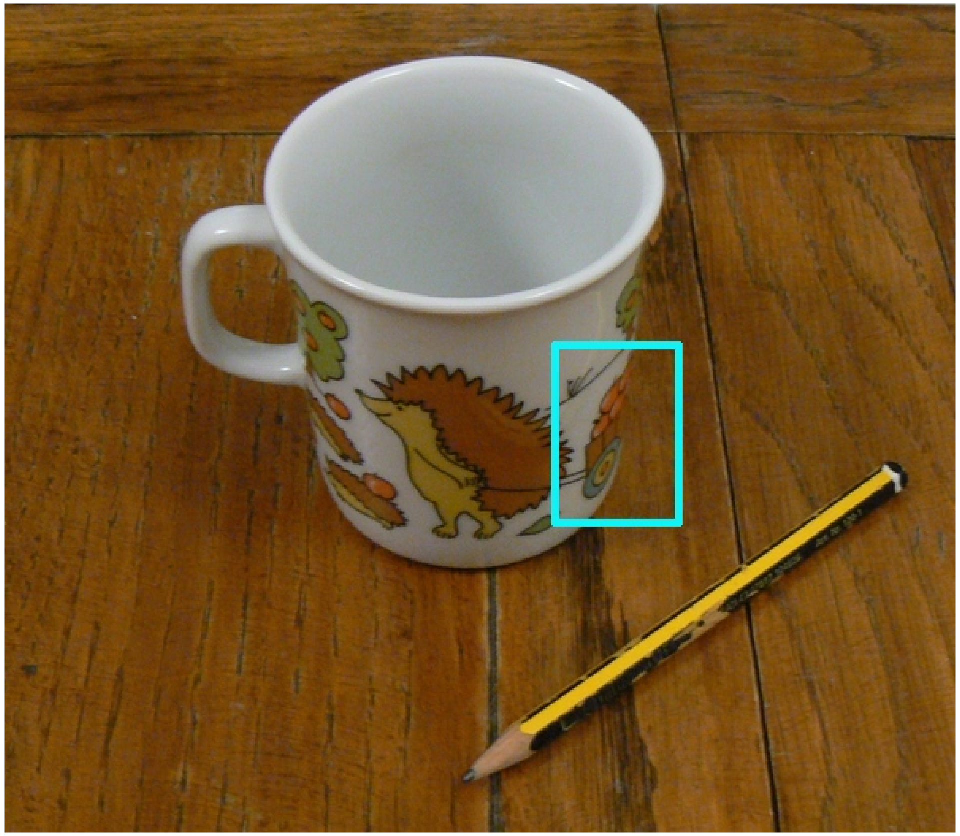
What is image segmentation?

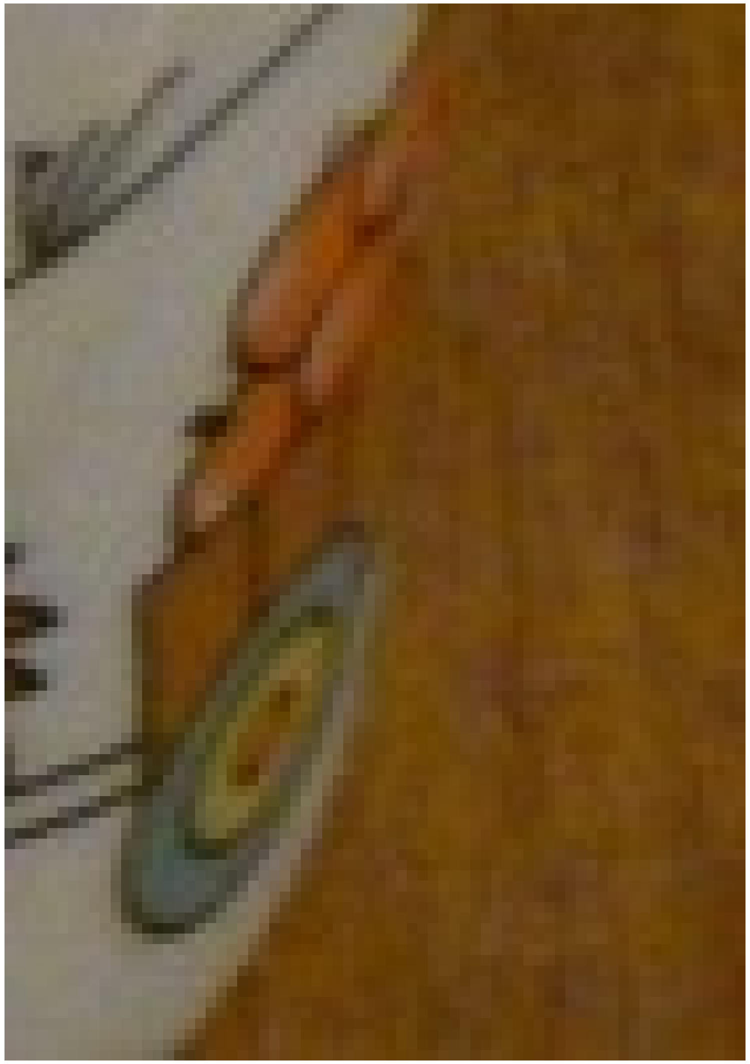


What is image segmentation?

- Segmentation is a collection of methods allowing to interpret spatially close parts of the image as objects.
- Regions (i.e., compact sets) represent spatial closeness naturally and thus are important building steps towards segmentation. Objects in a 2D image very often correspond to distinguishable regions.
- The object is everything what is of interest in the image (from the particular application point of view). The rest of the image is background.
- The approach is similar to that used in pattern recognition, i.e., division of the image into set of equivalence classes.

Easy or not?

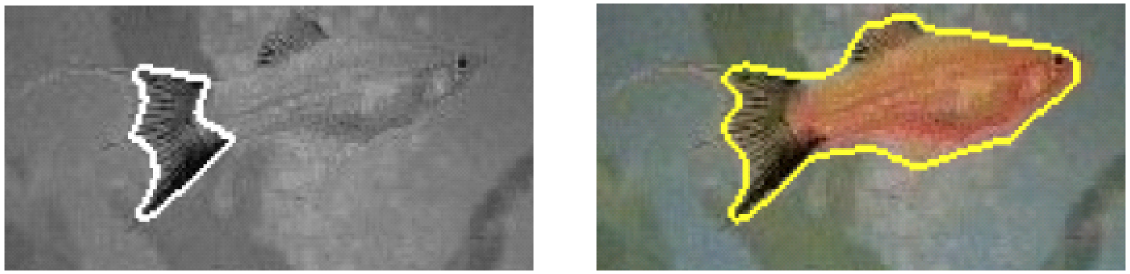




Where is the border between the cup and the background?

Image segmentation

- Easy for humans – we use top-down information (segmentation after recognition)
- Many segmentation methods settle for less. They put bits of image which are similar to the same segment.
- There is often no single answer how to segment.



Courtesy, images: Thomas Brox, TU Dresden, 2008

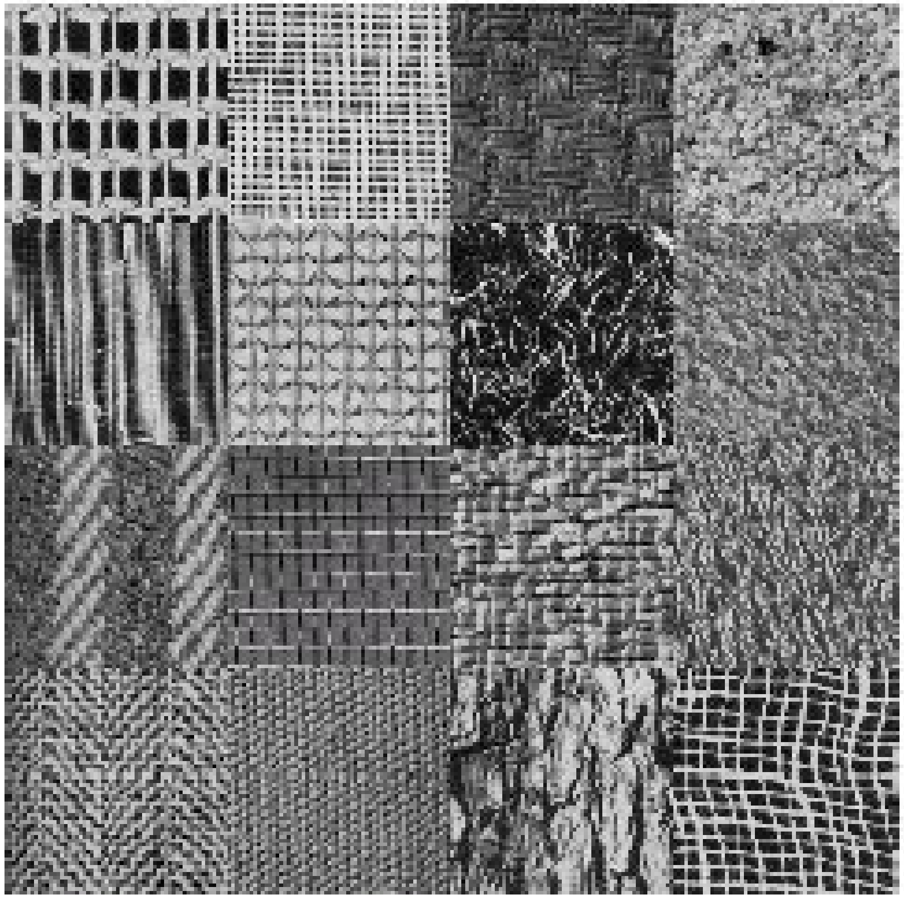
- Special cases when segmentation to e.g. foreground and background is easy.
- Segmentation usually makes sense in a scope of a particular application.

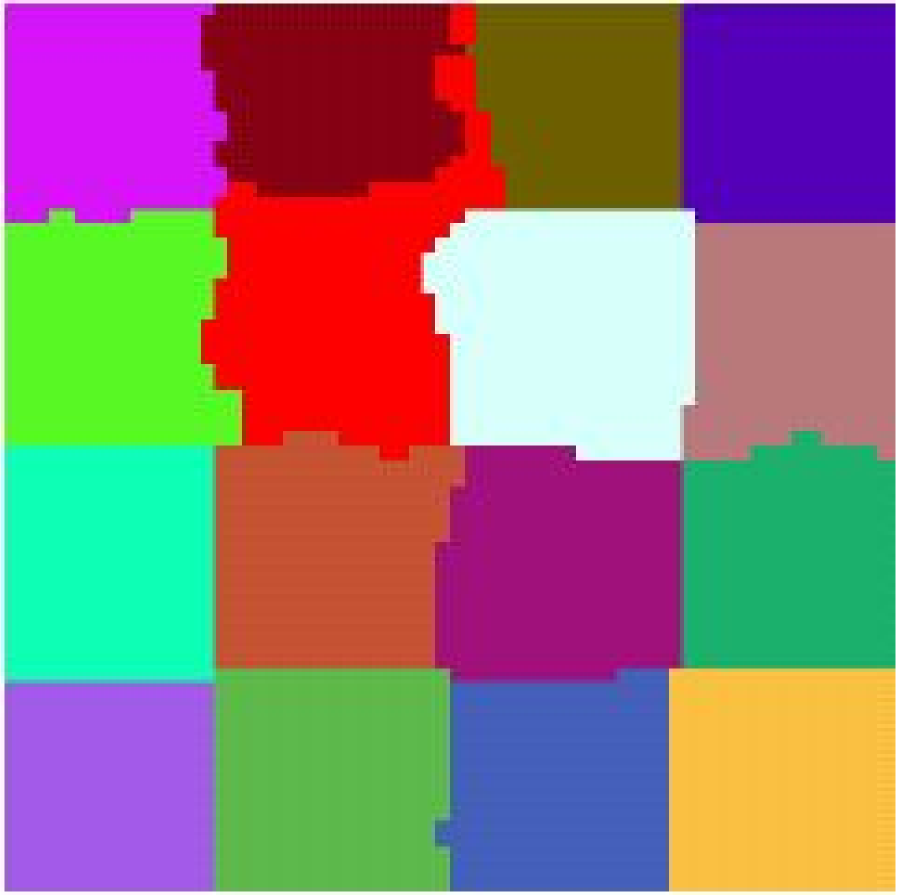
Segmentation is application dependent

- Methods are usually not universally applicable to all images.
- Direct segmentation of the input image takes pragmatically into account all available information about a particular application.
- A vital role of a priori information:
  - Low-level: e.g., brightness, spatial coherence, color, texture, motion, . . .
  - Mid-level: object symmetries, proximity on larger scale, . . .

Segmentation, Example 1

Segmentation by texture.





Segmentation, Example 2

Segmentation of an aerial image of the sea coast.

Complete vs. partial segmentation (1)

]

Complete segmentation – divides an image into non-overlapping regions that match to the real world objects.

Complete segmentation divides an image R into the finite number S of regions $R_1, \ldots, R_S$

$$R = \bigcup_{i=1}^{S} R_i \,, \qquad R_i \cap R_j = \emptyset \,, \qquad i \neq j \,.$$

Partial segmentation – it is possible to find only parts with semantic meaning in the image (e.g., regions, collection of edgels) which will lead to interpretation in later analysis.

Complete vs. partial segmentation (2)

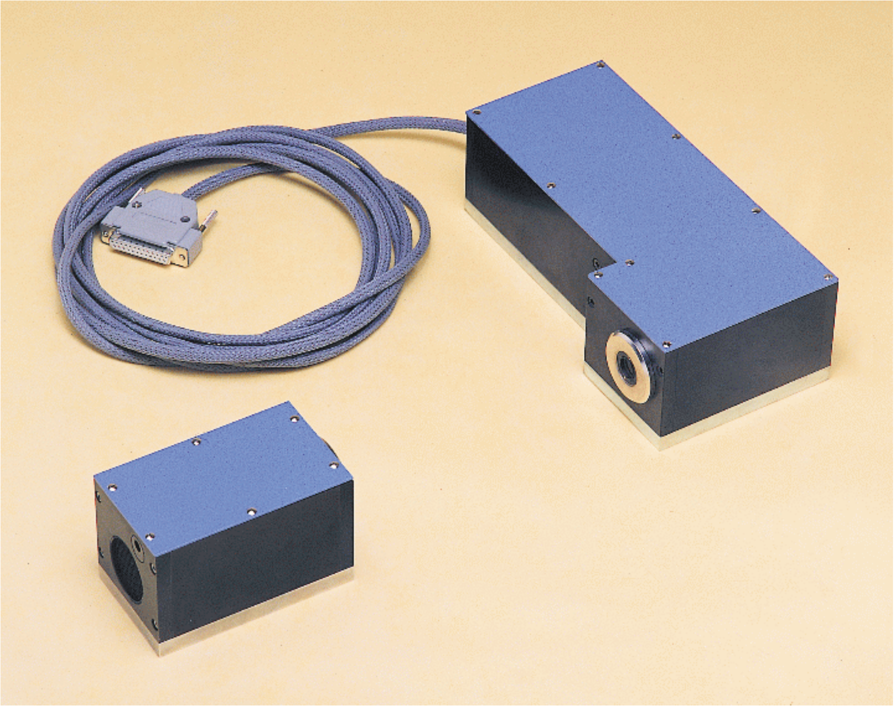
- To proceed from partial to complete segmentation it is needed to explore the higher-level of information processing.
- This can be performed iteratively in a feed-back loop.

Examples of a complete 2D segmentation

Seek contrast objects in a homogeneous background. Intensity thresholding provides a silhouette corresponding to objects, *e.g., printed characters, cell kernels, back-light illuminated details inspected in industry.*

Example – A digital profile projector

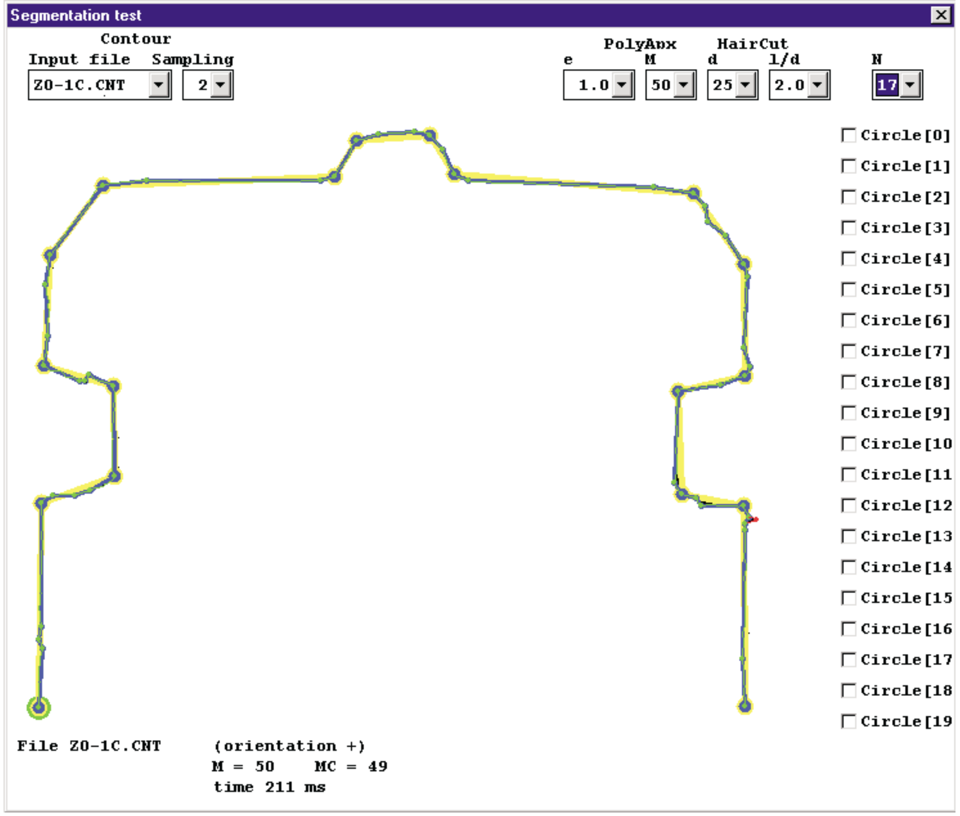
- Provides the back-light illumination where object shape appears as a silhouette.
- The principal is very often used in manufacturing for gauging and verifying correctness of a shape.
- The picture shows the illumination source (the smaller box containing LED diode and lens) and camera (a bigger box with cable, there is mirror inside). *Courtesy: Neovision s.r.o.*



Back-illuminated detail

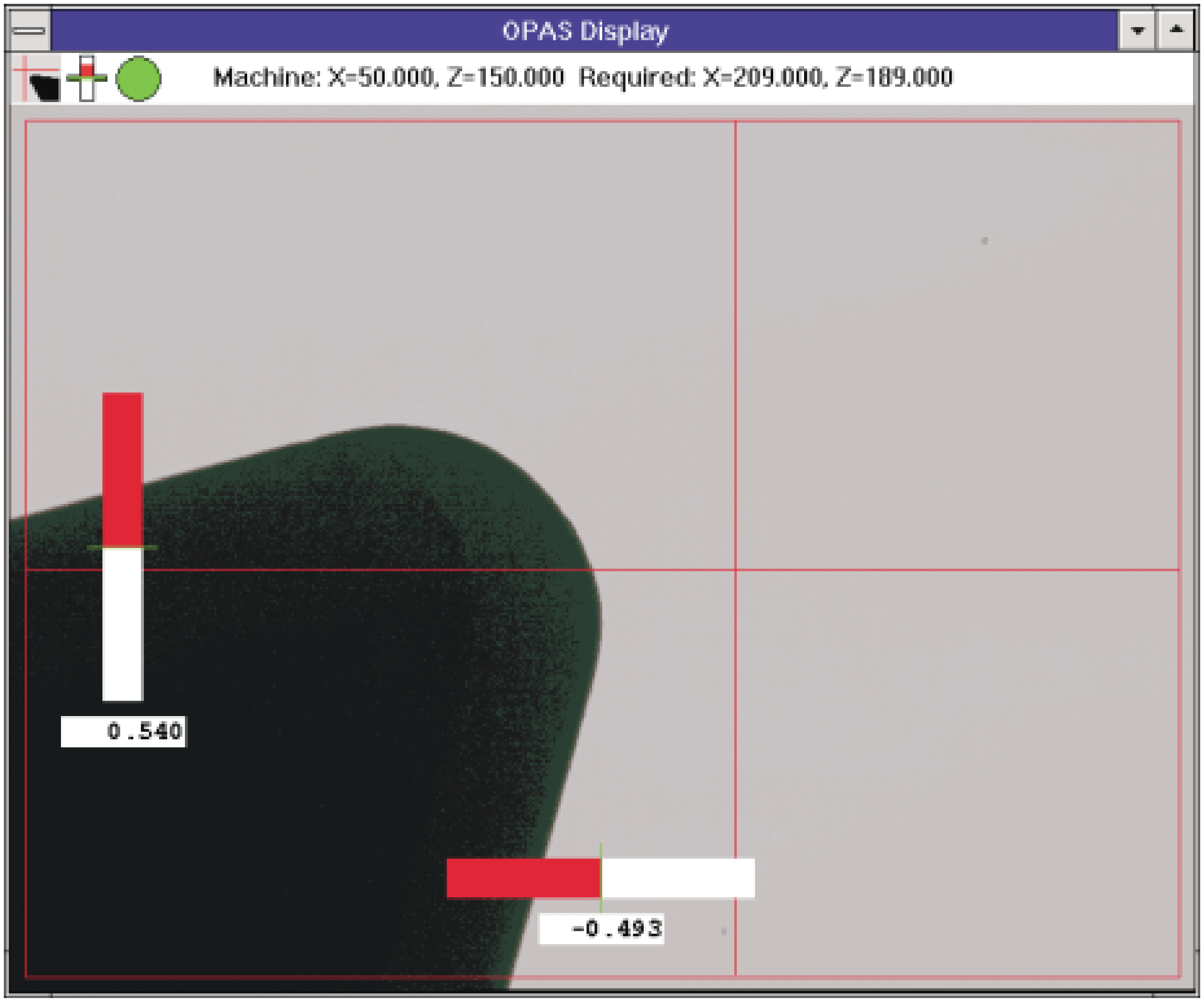
Example from the lathe-turning: *Courtesy: Neovision s.r.o.*

- Left image – a the back-illuminated detail showing imperfection and dust.
- Right image – automatically approximated shape allowing for automatic gauge check against a technical drawing.



Profile projector – User's screen

Another example: Automatic gauging of the lathe-turning knife from a sintered carbide.



Use of a priori information

The more a priori information the better.

Illustrative examples:

- Required shape of the region.
- Required position, orientation.
- Known initial and final point of the boundary *(e.g., in the application analyzing shape of a polymer drop, where the polymer sample comes out a tube of known position).*
- Relation of the region considered to other regions with required properties *(e.g., above, inside).*

Examples from two application areas:

Remote sensing: Look for ships in the water. Typical properties of railway lines, highways (minimal curvatures). Rivers do not cross.

Medical: Blood vessels ar roughly parallel. Relate to anatomic atlas (model-based approach).

Simple approaches for segmentation

- Threshold-based, according to a global property, usually intensity, where the global knowledge is represented by the intensity histogram.
- Spatial coherence-based ($\approx$ clustering of 'tokens').
  - Connecting, e.g., edgels because edges bear often an important information about objects (cf. human visual system).
  - Region merging/splitting. Regions come from aggregating pixels with similar properties (homogeneity criterion)
- Template matching – detection and fitting tokens in the image to a priori known template.
  - Parametric model detection, e.g., straight line, circle, ellipse, ...
- Unusual phenomena-based.
  - Camouflage detection based on an unusual texture.
  - Region segmentation for the image compression.

Thresholding

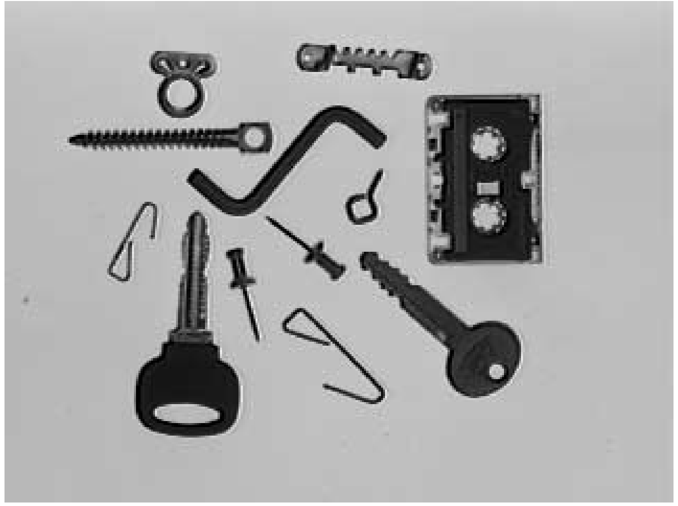
Input image $f(i,j)$, output image $g(i,j)$.

For each pixel (i,j)

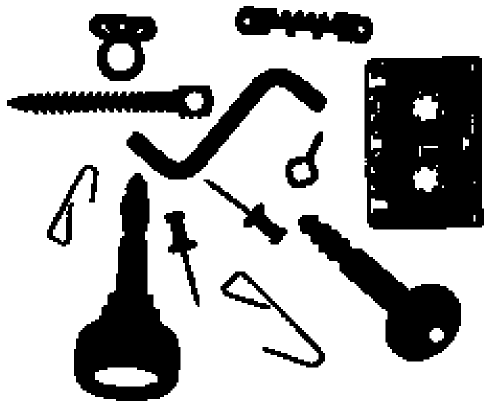
$$g(i,j) = \begin{cases} 1 & \text{for} \quad f(i,j) \geq \mathrm{Threshold}\,, \\ 0 & \text{for} \quad f(i,j) < \mathrm{Threshold}\,. \end{cases}$$

+ Simple technique, long time and more often used.
+ Easy in hardware, intrinsically parallel.
- The threshold is a parameter which is difficult to adjust automatically in general.
- Works only for subclass of images in which objects are distinct from background in intensity.

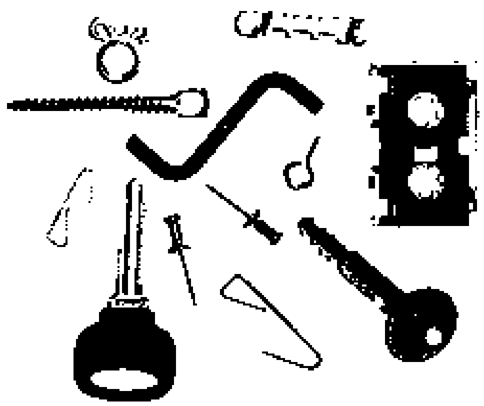
Example, threshold influence



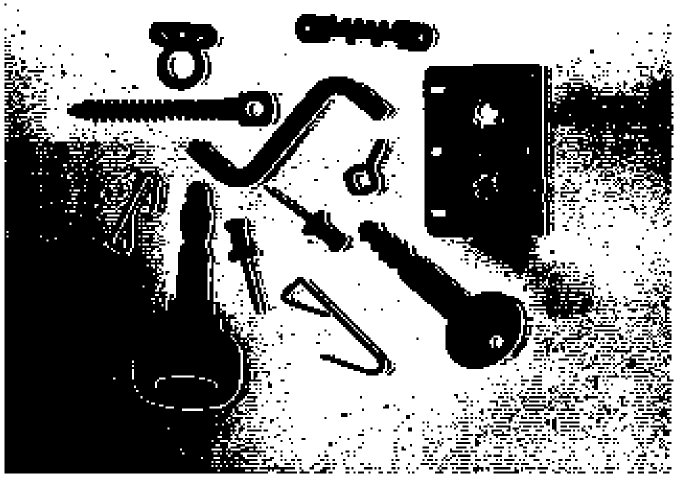
Original image.



Threshold segmentation.



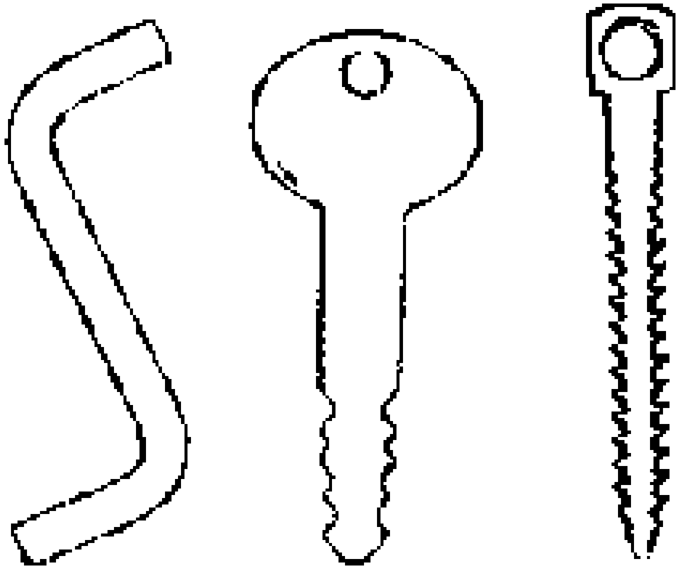
Threshold too low.



Threshold too high.

Example
Border regions by the band thresholding

Original image.



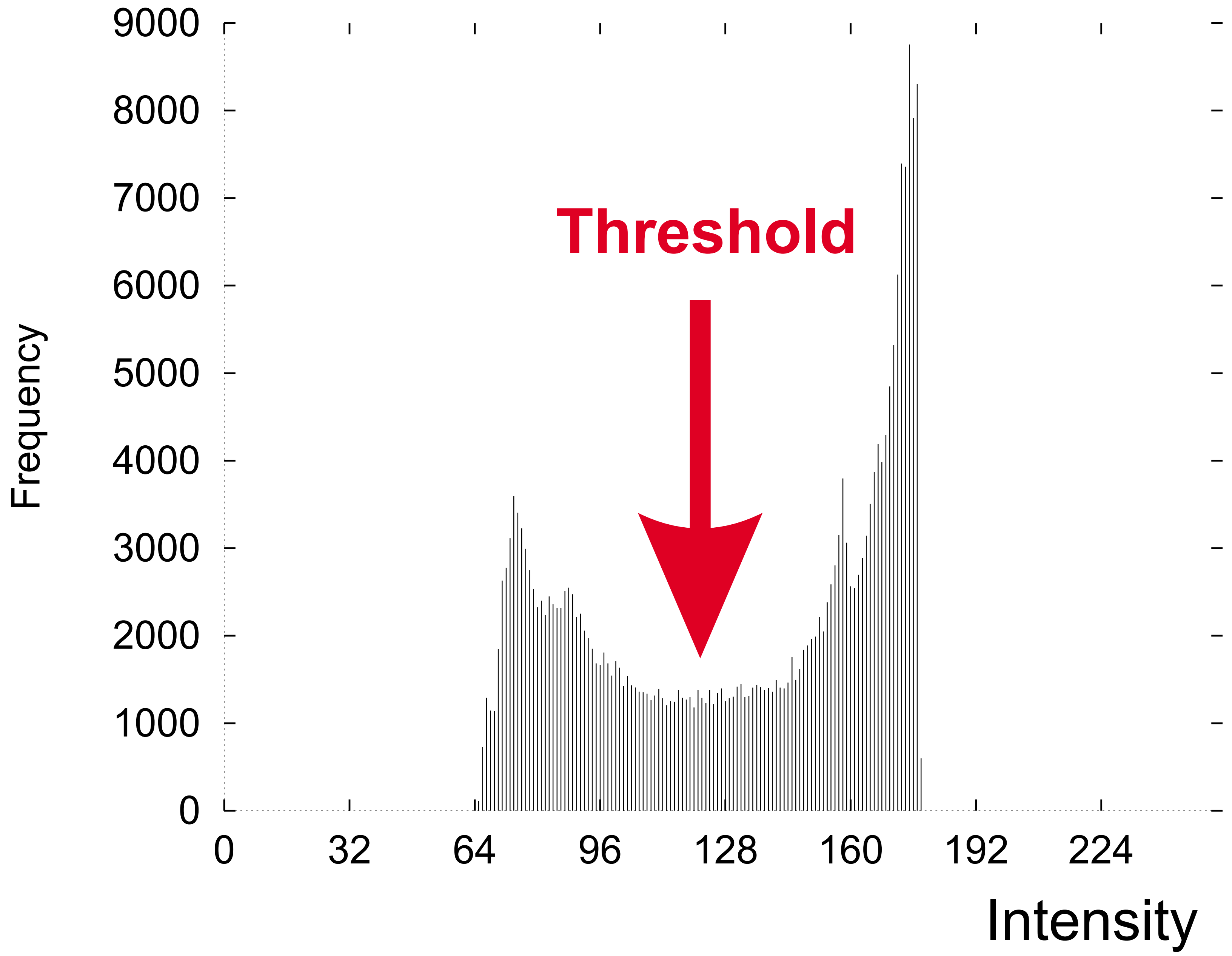
Border regions detected.

Threshold detection, use of the histogram

p-tile thresholding, if we know that the objects cover $1/p$ of the image, e.g. printed characters on a sheet $\Longrightarrow 1/p$ area of a histogram.

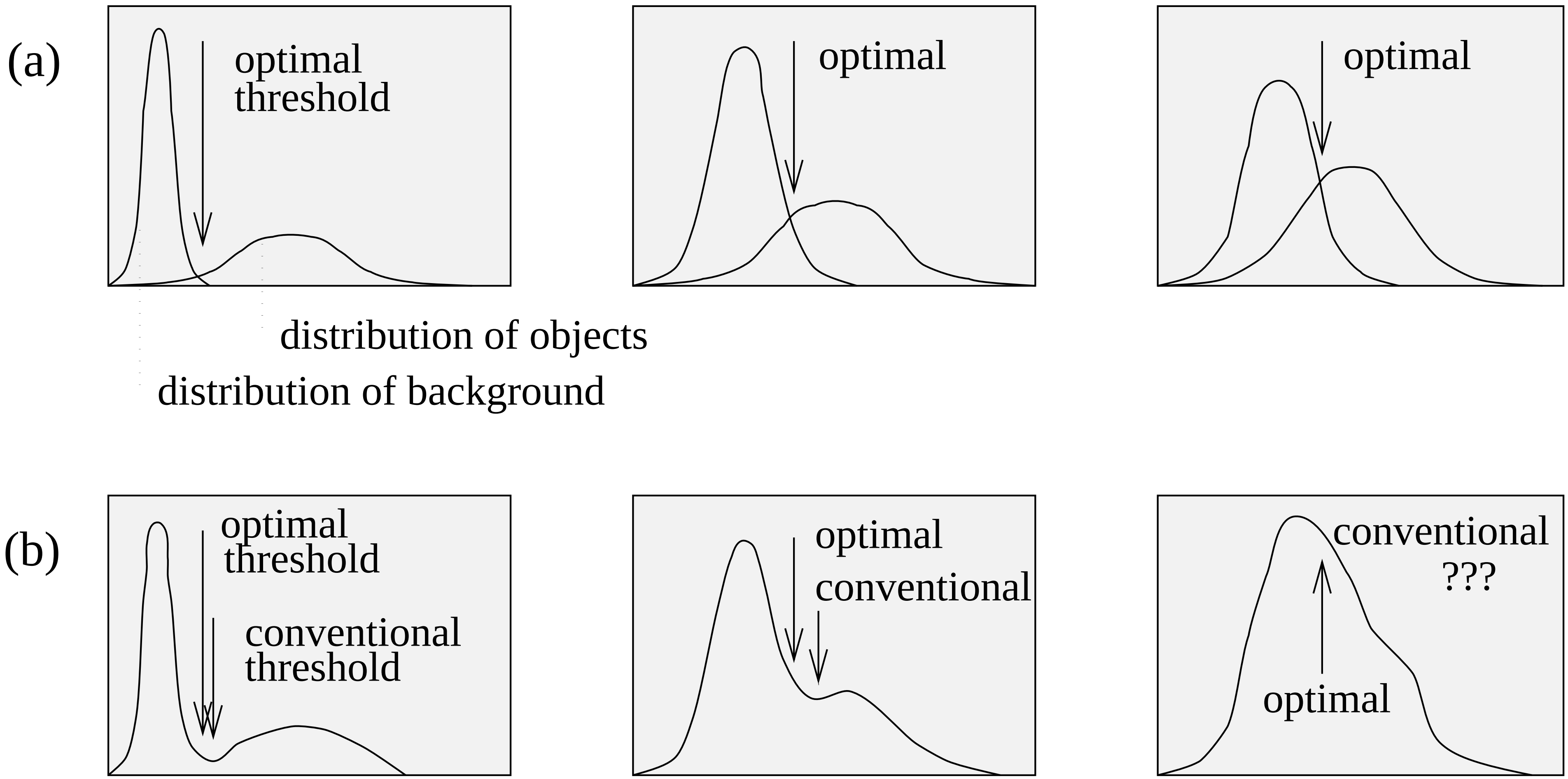
Histogram shape analysis, distinct objects on background correspond to a bi-modal histogram. Find middle between the modes.

Automatically found threshold according to a bi-modal histogram



Optimal thresholding by a mixture of Gaussians

Motivation:



Local thresholds by mixture of Gaussians

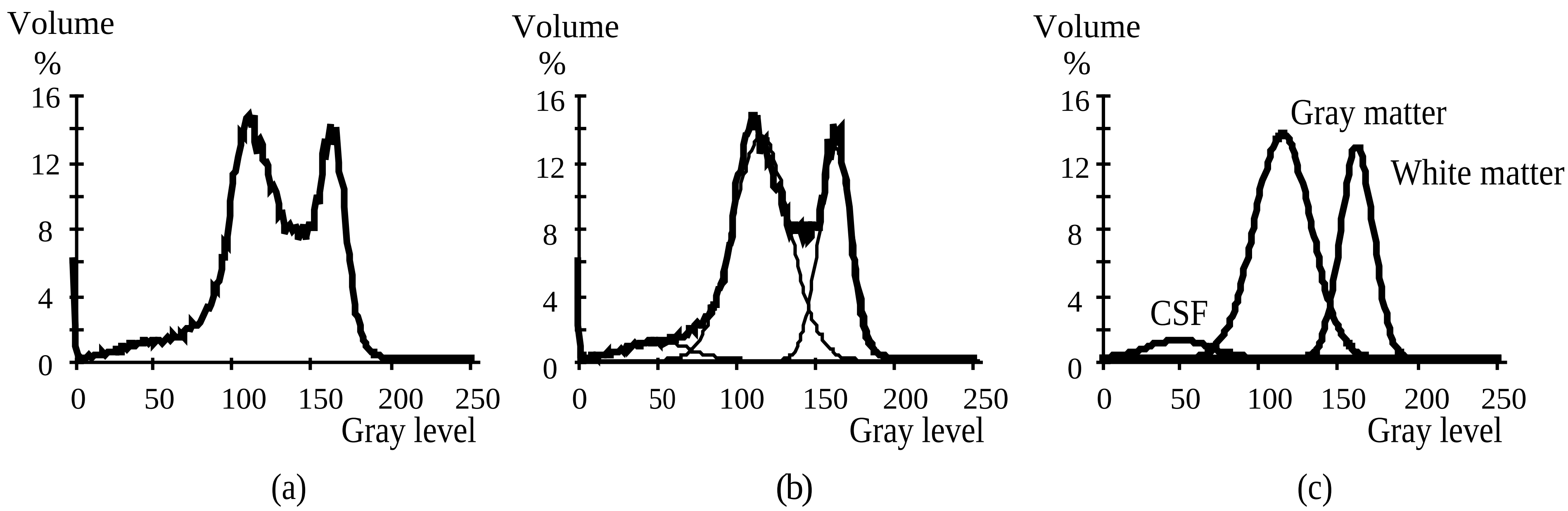
- h_{region} – local histogram.
- h_{model} – approximation of a histogram by n Gaussian distributions,

$$
\begin{aligned}
h_{\mathrm{model}}(x) &= \sum_{i=1}^{n} \pi_i \frac{1}{\sqrt{2\pi\sigma_i^2}} e^{\left(-\frac{(x-\mu_i)^2}{2\sigma_i^2}\right)} = \sum_{i=1}^{n} \pi_i N(x, \mu_i, \sigma_i) = \\
&= \sum_{i=1}^{n} \pi_i p(x|i) \,.
\end{aligned}
$$

- find the parameters of the approximation, using e.g. EM algorithm
- compute posterior probabilities $p(i|x)$ (soft assignments of individual data to models i)

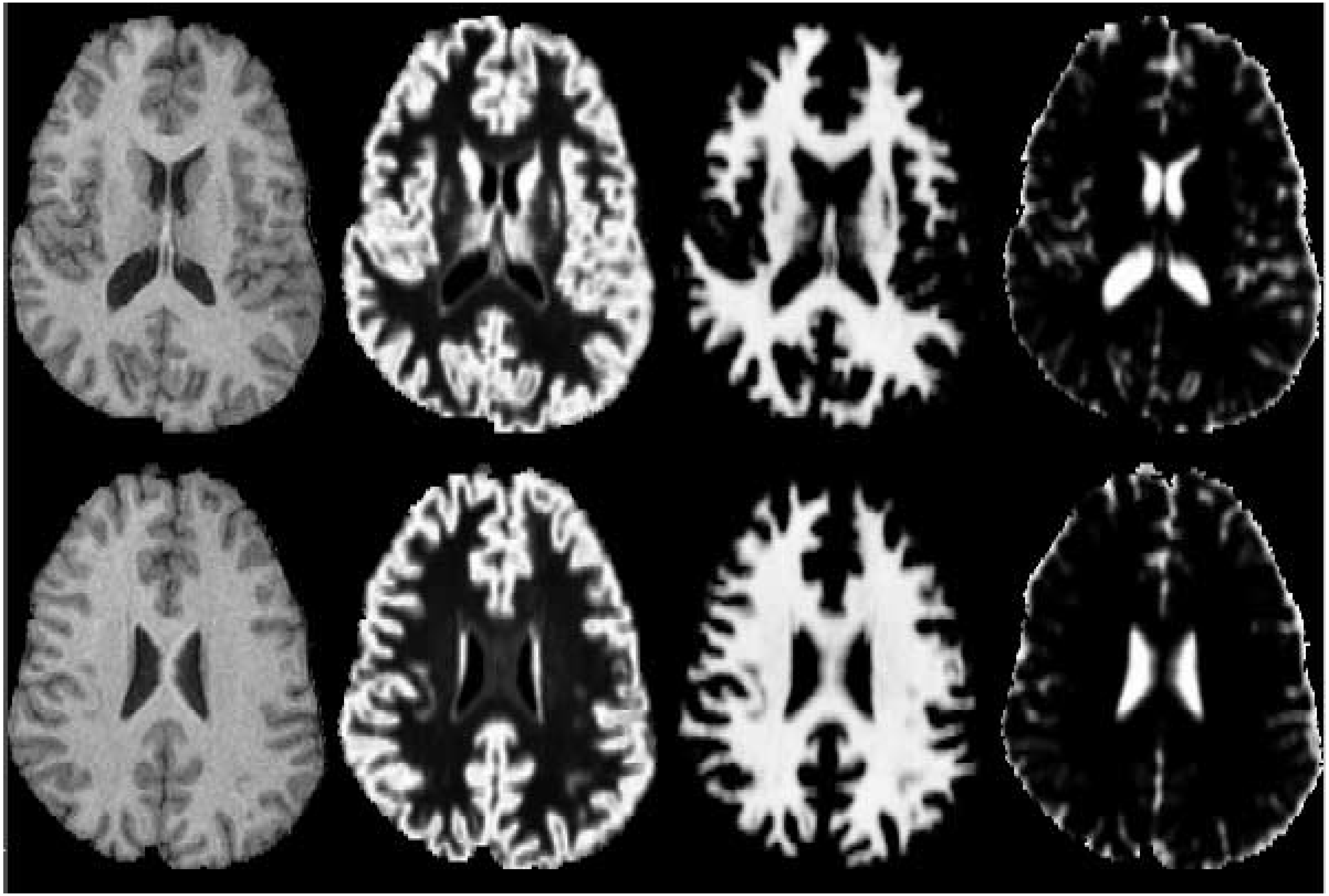
Example, Segmentation of the brain MR

- Input: T1-weighted NMR images.
- Desired classes: white matter, grey matter, celebro-spinal fluid (CSF)



Courtesy: Milan Šonka, University of Iowa.

Brain MR, Segmentation result



original gray matter white matter CSF

Segmentation as clustering

- Image segmentation can be formulated as clustering which has been studied in statistics, pattern recognition or machine learning.
- Problem formulation: Assign a pixel to a cluster which is the most spatially adjacent and the most most homogeneous with respect to some criterion (e.g., brightness variance).
- There are several clustering methods. Seek unsupervised learning algorithms in pattern recognition. E.g., K-means, EM.

K-means clustering

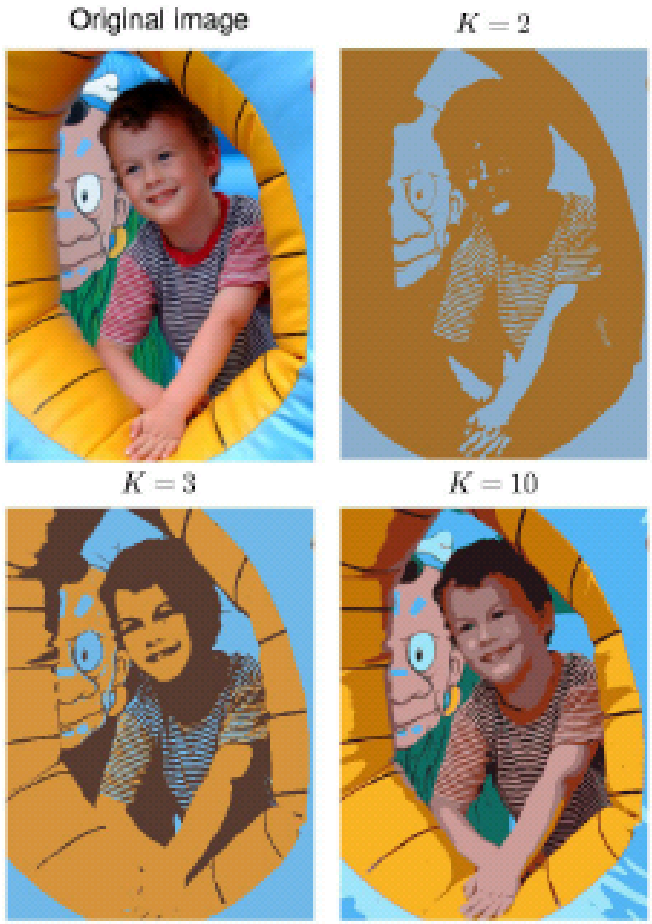
Lloyd's Algorithm

- Initialize the pixels to belong to a random region.
- Compute the mean feature vector in each region.
- Move a pixel to another region if this decreases the total distance J.

$$J = \sum_{n=1}^{N} \sum_{k=1}^{K} \|x_n - \mu_k\|^2 \,,$$

where n points to individual pixels, N is number of pixels, K is an a priori given number of clusters, $K < N$, x_n is a pixel value, μ_k is the cluster representative (mean point of a cluster).

- Iterate until pixels do not move any longer.



Author: Christopher Bishop

S. Lloyd, Last square quantization in PCM's. Bell Telephone Laboratories Paper (1957). In the journal much later: S. P. Lloyd. Least squares quantization in PCM. Special issue on quantization, IEEE Trans. Inform. Theory, 28:129–137, 1982.

K-means clustering

- can be thought of as a simplified EM algorithm with Gaussian mixtures
- all σ_i equal, all π_i equal $\Rightarrow$ only distance to μ_i plays a role
- there is no posterior probabilities, the assignments $p(i|x)$ are hard (binary)

$$\sum_{i=1}^{n} \pi_i N(x, \mu_i, \sigma_i) = \sum_{i=1}^{n} \pi_i p(x|i)$$

K-means clustering, Example

- feature: absolute value of partial derivatives, $(|\frac{\partial I}{\partial x}|, |\frac{\partial I}{\partial y}|)$
- $K = 2$

K-means clustering (2)

- There is a danger of ending up in a local minimum of J.
- Performance depends much on the initial choice of the clusters.
- There are point sets on which K-means takes superpolynomial time $\mathcal{O}(2^{\sqrt{n}})$ to converge.

  D. Arthur, S. Vassilvitskii (2006). How Slow is the k-means Method?. Proceedings of the 2006 Symposium on Computational Geometry.
- With $K = 2$ the K-means algorithm can be regarded as a thresholding with an automatically determined threshold (1D case).

Mean shift

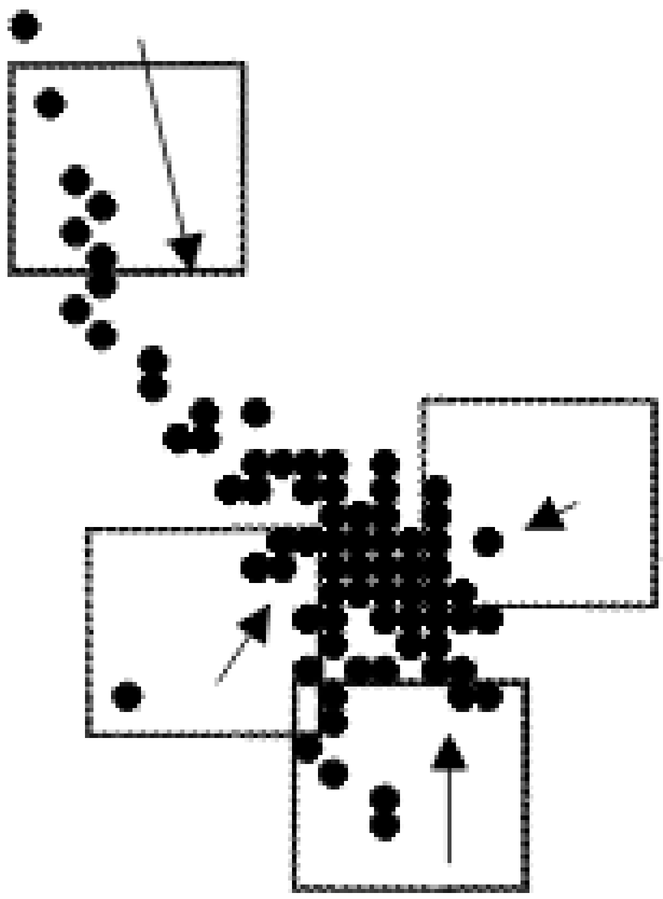
- Estimation of the density gradient - Fukunaga K.: Introduction to Statistical Pattern Recognition, Academic Press, New York, 1972.
- Sample mean of local samples points in the direction of higher density. It provides the estimate of the gradient.
- Mean shift vector m of each point p

$$m = \sum_{i \in \mathrm{window}} w_i \left(p_i - p\right) , \quad w_i = \mathrm{dist}(p, p_i)$$

- Based on the assumption that points are more and more dense as we are getting near the cluster "central mass".

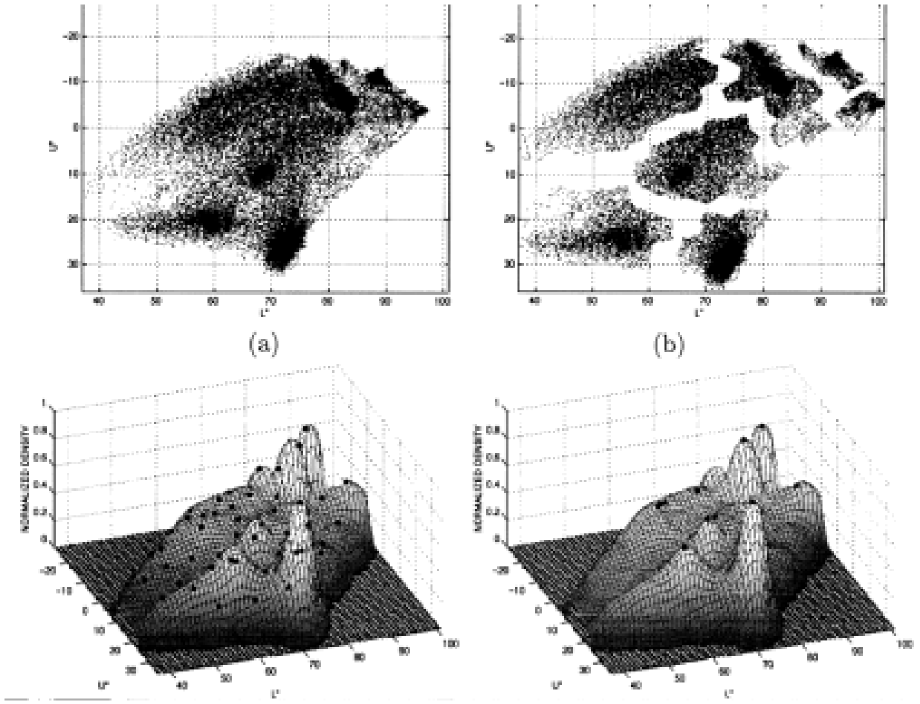
Mean shift algorithm

- Input: points in the Euclidean (feature) space.
- Determine the kernel size (usually small).
- Choose the initial location of the search window.
- Compute the mean location (centroid of the data) in the search window.
- Center the search window at the mean location computed in the previous step.
- Repeat until convergence.



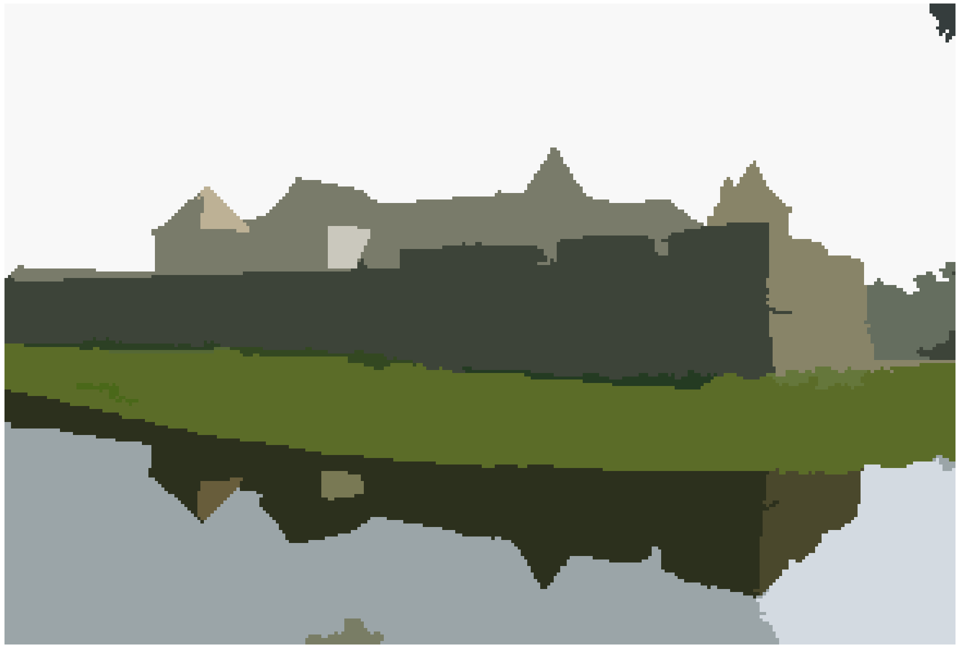
Mean shift segmentation algorithm

1. Convert the image into tokens (via color, gradients, texture measures etc).
2. Choose initial search window locations uniformly in the data.
3. Compute the mean shift window location for each initial position.
4. Merge windows that end up on the same 'peak' or mode.
5. The data these merged windows traversed are clustered together.



(a) (b)

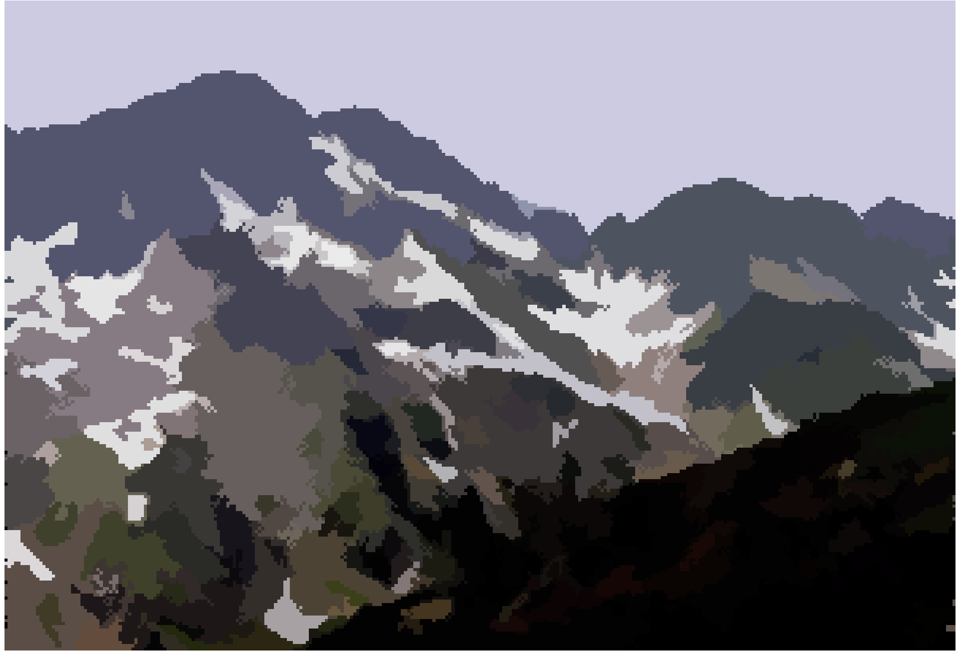
Mean shift segmentation, Example 1



http://www.caip.rutgers.edu/∼comanici/MSPAMI/msPamiResults.html

Mean shift segmentation, Example 2

Mean shift segmentation, Example 3



http://www.caip.rutgers.edu/∼comanici/MSPAMI/msPamiResults.html