Computer Vision (ZDO)

Image segmentation - advanced techniques

Ing. Zdeněk Krňoul, Ph.D.

Department of Cybernetics
FACULTY OF APPLIED SCIENCES
University of West Bohemia

Outline:

- Segmentation based on edge detection:
  - Boundary searching using the position
  - Sequential division
  - Active contours
- Segmentation by area analysis:
  - Algorithm Split & Merge
  - Markov random field
- Template matching

Outline:

- **Segmentation based on edge detection:**
  - Boundary searching using the position
  - Sequential division
  - Active contours
- Segmentation by area analysis:
  - Algorithm Split & Merge
  - Markov random field
- Template matching

Segmentation - Edge detection

Edges places of the image where there is a certain discontinuity, mostly in brightness, but also in color, texture, depth, etc.

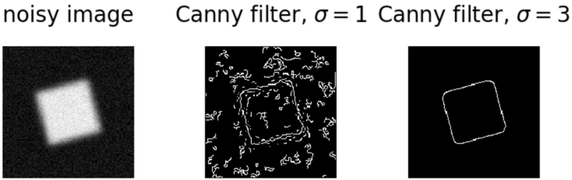
Image of edges is created by applying an edge operator;

Boundary is a description of the edge of the segmented object

- **Algorithm**: the task of segmentation is to join edges into strings that match the boundaries;
- We usually use information about "*Where the edges are*?"and "*How they relate to other parts of the image?*";
- The method must take into account **local properties** together with **general knowledge** specific to the application area.

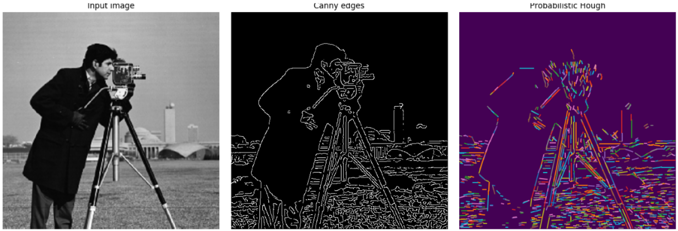
Gradient magnitude thresholding

- Usually very few places in the image have zero edge value. The reason is the presence of noise;
- The thresholding preserves only significant edges (the meaning of the words "small", and "significant"is related to the size of the threshold)
- The threshold value can be determined, for example, by percentage threshold methods;
- Post-processing of the result is sometimes applied - eg remove **edges** shorter than a certain value



Segmentation - Using knowledge of the position

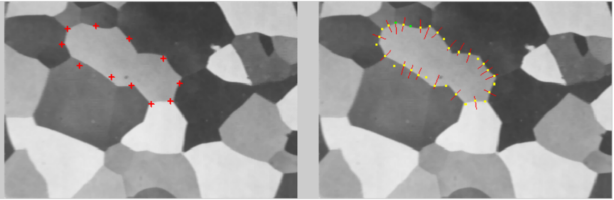
- We assume information about the probable **position and shape**;
- **Boundary segment** that are close to the expected location of the boundary and that have a direction close to the expected direction at that location;
- If a sufficient number of pixels meeting these conditions can be found, a suitable approximation function is defined by these points - *refined boundary* - Hough transform



DEPARTMENT OF CYBERNETICS

Segmentation - Division of the border

- If we know the endpoints of the boundary and assume small noise and curvature of the boundary;
- **Algorithm**: gradually dividing the already detected neighboring boundary elements and searching for another boundary element on the **perpendicular line** defined by the centre of the connecting line;
- The edge element that is closest to the existing detected boundary points and has a **above-threshold** edge size is considered the new boundary element and the iteration process is repeated.



DEPARTMENT OF CYBERNETICS

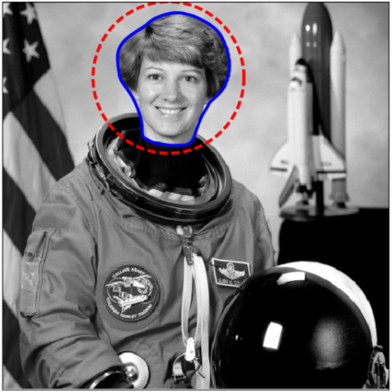
Segmentation - Active Contour Model

Model determines the boundary by a deformable open/closed spline function (snake) ... y. 1987

Loss function based on the **image energy** under the spline function and from the **shape energy** of spline function (length, smoothness, ...)

Application - for noisy data, iteration with the user (initialization of position or energy) or with other segmentation/detection techniques

Active Contour Model



For example segmentation of the human face from the background so that the surrounding (closed) spline function finds them at the places of the largest edges around the object, initialization with the first applied face detector.

Outline:

- Segmentation based on edge detection:
  - Boundary searching using the position
  - Sequential division
  - Active contours
- **Segmentation by area analysis**:
  - Algorithm Split & Merge
  - Markov random field
- Template matching

Region Growing

- Can be used in images with noise;
- We define the criterion as **HOMOGENEITY**
- Algorithm divides the image into maximum contiguous regions so that these areas are homogeneous.

Criterion:

- based on the brightness of the image in that area;
- we usually require the following conditions for the area:

1. $$H(R_i) = TRUE \quad pro \quad i = 1, 2, \ldots, I$$
2. $$H(R_i \cup R_j) = FALSE \; for \; i, j = 1, 2, \ldots, I \; i \neq j \; R_i \;\; R_j$$

Where: I .... number of regions R_i .... single region $H(R_i)$ .... binary expression of homogeneity criterion $\rightarrow$ region must be (1) homogeneous and (2) maximum

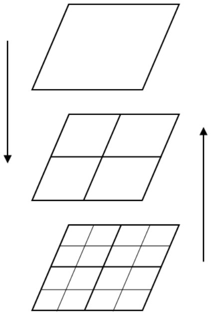
Region Growing - Algorithm

A simple method of joining regions is based on initial segmentation, where each initialization region (pixel) satisfies condition **(1)** but does not satisfy condition **(2)**. Furthermore, we always connect two adjacent areas, if the area is created by merging these two areas.

- The result of the join depends on the order in which the areas are presented for merging.
- Initialization of the image into 2x2, 4x4 or 8x8 regions;
- Criterion mostly based on brightness statistics (eg histogram of the region);
- The region description is compared with statistical tests with the description of the neighboring regions:
  - If the criterion is met - the two regions merge and a new region is created;
  - When no two regions can be merged, the process ends.

Split and Merge

- Uses a pyramidal image representation;
- Regions are square and correspond to an element of a given level of a pyramidal data structure.



1. We define the initial segmentation of the image;
2. If for the region R k-th level of the pyramidal structure $H(R) = FALSE$ (the region is not homogeneous), we divide R into 4 regions $(k+1)$ level;
3. If there are neighboring regions R_i and R_j such that $H(R_i \cup R_j) = TRUE$, we combine R_i and R_j to one region;
4. If no region can be merged or split, the algorithm ends.

Markov Random Fields (MRF)

- **context** = relation of pixels, ie the meaning of a pixel depends on neighboring pixels ... Markov property
- context definition is based on **conditional probability**

Segmentation task = Labeling

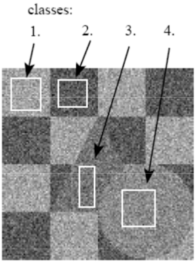
- Each pixel p is defined by the feature vector $\overrightarrow{f}_p$ (basically its brightness) and the set of all feature vectors $\ldots f = \{\overrightarrow{f}_p : p \in \mathcal{I}\}$
- The label set $\mathcal{L}$ specifies the segmentation eg $\mathcal{L} = \{$object, background, ...$\}$
- in terms of Markov models represents the label „Hidden variable“
- Each p pixel is assigned one label ω_p
- **Field configuration** $\ldots \omega = \{\omega_p : p \in \mathcal{I}\}$

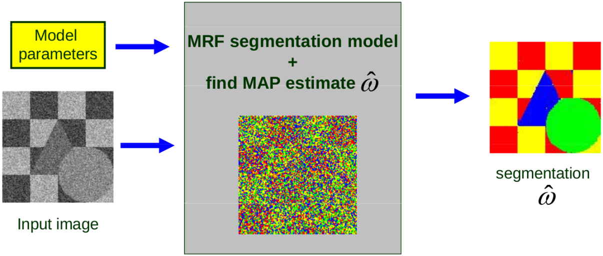
DEPARTMENT OF CYBERNETICS

- image with dimensions NxM $\rightarrow |\mathcal{L}|^{NM} = |\Omega|$ possible results
- how to choose the right one?

Maximum a posteriori estimation (MAP):

- **Objective:** Segmentation Probability
- The configuration probability of ω is determined as $P(\omega|f)$
- we want to find ω^* maximizing $P(\omega|f)$
- ${\omega^*}^{\text{MAP}} = \arg\max_{\omega \in \Omega} P(\omega|f)$
- Bayes rule: $P(\omega|f) = \frac{P(f|\omega)P(\omega)}{P(f)}$
- we can consider f is $P(f)$ constant $\Rightarrow P(\omega|f) \propto P(f|\omega)P(\omega)$
- specifying $P(\omega)$ a $P(f|\omega) \rightarrow$ is the task for **MRF**
- MRF converts to minimum energy optimization task (data energy $+$ region energy)
- random field can be defined as a graph (usually Graph-Cuts method)

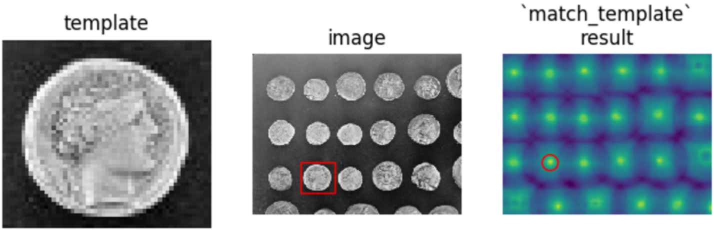

classes:
1.
2.
3.
4.


Model parameters
MRF segmentation model
+
find MAP estimate $\hat{\omega}$
Input image
segmentation
$\hat{\omega}$

Outline:

- Segmentation based on edge detection:
  - Boundary searching using position
  - Sequential division
  - Active contours
- Segmentation by area analysis:
  - Algorithm Split & Merge
  - Markov random field
- **Template matching**

Segmentation - Template Matching



- The task is to find known objects (patterns) in the image;
- Objects (patterns) usually have the character of an image;
- No noise and distortion - the task would be very easy, because we would find an exact copy of the searched pattern in the image;
- Similarity criterion - cross-correlation:

$$C(u,v) = \frac{1}{\sum_{(i,j)\in V} (f(i+u, j+v) - h(i,j))^2} \qquad (1)$$

Algorithm:

- We test the matching of the image f with the pattern h located in the position (u, v);
- For each position of the pattern h in the image f we get the value of the criterion C;
- Local maxima that are greater than the specified threshold represent the pattern found;
- **Problem**: the pattern in the image occurs rotated, with a different size or other geometric distortion;
- In this case, we need to test the criterion for all these possible geometric distortions;
- *Note. This problem can be partially solved when the pattern is composed of several parts flexibly connected. Then we first test these individual parts and then we determine the flexible connections*

Template Matching - performance

- The method can be accelerated by performing the comparison in lower resolution and then in the place of the local maximum by the exact position $(u, v)^*$, for which the largest value of the criterion occurs;
- Frequency spectrum correlation calculation (see color blue demo doc matlab)

Template matching - Key-Point Matching

- **Principle**: search for matching key points in the image and in the pattern;
- Key point detector (corner detector: Harris, Moravec, etc. + SIFT, SURF, KAZE descriptors);
- **Algorithm**: search for a valid transformation between the pattern and the subarea of the input image (eg shift, rotation, affine transformations, homography), for the search we will use eg the RANSAC method.