Advanced Informatics and Control

Ewaryst Rafajłowicz

Introduction to Computer Vision in Quality Control

Choose yourself and new technologies

HUMAN CAPITAL
HUMAN – BEST INVESTMENT!

Wrocław University of Technology

EUROPEAN SOCIAL FUND

Project co-financed from the EU European Social Fund

Introduction to Computer Vision in Quality Control

Lecture 8 – Finding and describing objects

Ewaryst Rafajłowicz

Summary of Lecture 7

Methods of finding objects III

Summary of Lecture 8

Industrial Image Processing 8

Describing objects

After labeling objects on a binary image our starting point is a binary image + additional information on addresses of pixels belonging to each object (or blob). It can be stored as:

- **as a matrix L_{ij} of the same size as the original binary image + integers, which are labels attached to each pixel (L_{ij} = "k" means that pixel (i,j) belongs to object "k", if "k" = "0" we treat it as the background,**
- **as lists of pixels, which constitute objects (as in Matlab).**

Industrial Image Processing 8
Describing objects 2

Our aim is to describe these objects in a way dependent on application. We shall use the convention $L_{ij} = 1$ for a current object and $L_{ij} = 0$ for its background (note that in a programming practice this is not the way of storing objects). Even simple notions have to be redefined in pixel terms.

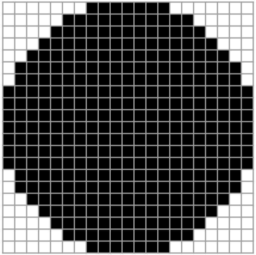
- Area A is the number of pixels: $L_{ij} = 1$.
- It can be used to estimate the area of a real object as $c_A \cdot A$, where c_A, $[\mathrm{mm}^2/\mathrm{pixel}]$ is the area of one pixel in reality.

Industrial Image Processing 8
Describing objects 3

- Constant c_A can be obtained by the camera calibration.
- The simplest way is to take an image of a square with known area and to calculate the number of pixels in it. Note that c_A depends on the distance between the object and a camera and on its lens also.
- Note that for small objects the area estimate can not be too exact. Why ?

Industrial Image Processing 8
Describing objects 3

The reason is visible from the figure:



Disk of the radius = 10 pixels. One can improve the accuracy by decreasing the distance between the object and the camera or using a higher resolution.

C.a. 50 pixels can be erroneously classified to the disc area. If $c_A = 0.1\,\mathrm{mm}^2/\mathrm{pixel}$, then we may commit an error of c.a. $5\,\mathrm{mm}^2$. Why the accuracy of estimating areas may be of importance in some applications ?

Industrial Image Processing 8
Describing objects 4

Answer: if we sample some items for inspection and then multiply their areas by thousands of items from a production day, then even 5 mm^2 error can highly influence the final result.

- **It is also clear from the previous figure that we shall have errors in gauging the perimeter of this disc.**
- **By perimeter P of an object we mean the count of the number of pixel sides traversed around the boundary of the object.**

Industrial Image Processing 8
Geometric attributes

Remark: Calculating P we can start at an arbitrary initial boundary pixel and finally return to it.

- The circularity of an object:

$$C = \frac{4\,\pi\,\text{Area}}{(\text{Perimeter})^2}$$

.

- $C = 1$ for discs.
- for elongated (thin) objects $C < 1$ (e.g., for the regular triangle $C \approx 0.6$).

Industrial Image Processing 8
Geometric attributes 2

- For the square $C = \pi/4$, but for an elongated rectangle with the edges $c = 10 \cdot b$, we obtain $C = 0.26$.
- Note that C is scale independent.
- There are many other (Haralick's, Ferret's, solidity =(Area/Convex area) etc.) shape attributes. They serve for quick distinction between objects of different shapes. E.g., if grains are of ellipsoidal shapes, then it is possible to distinguish them quickly from unwanted ingredients.

Industrial Image Processing 8

Geometric attributes 3

There are many other geometric attributes.
Matlab provides the following possibilities:

Area, Centroid, BoundingBox, SubarrayIdx, MajorAxisLength, MinorAxisLength, Eccentricity, Orientation, ConvexHull, ConvexImage, ConvexArea, FilledArea, EulerNumber, Extrema, EquivDiameter, Solidity, Extent, PixelIdxList, PixelList, Perimeter.

They are very useful, but their definitions are very dull. Glance at them when necessary.

Industrial Image Processing 8
Geometric moments

Recall from the course of statistics that for two random variables X, Y, their moments of the order $m \geq 0$, $n \geq 0$, if exist, are defined as:

- $q(m, k) = E\left(X^k Y^m\right)$ – noncentral,
- $Q(m, k) = E\left((X - E(X))^m (Y - E(Y))^n\right)$ – central (or centered) moments.

Although binary images may not fulfill requirements of the probability theory, the empirical versions of q, Q (differently normalized) are used as shape descriptors.

Industrial Image Processing 8
Geometric moments 2

Consider object labeled as "1".

- Define $I_{ij} = 1$ if $L_{ij} =$ "1" and $I_{ij} = 0$, otherwise.
- Let $\mathcal{L}$ be the total number of pixels labeled as "1". Clearly, $\mathcal{L} = \sum_{i=1}^{N} \sum_{j=1}^{M} I_{ij}$.
- Define $\beta(m, n) = 1 + (m + n)/2$ and

$$v(m, n) = \frac{1}{\mathcal{L}^{\beta(m,n)}} \cdot \sum_{i=1}^{N} \sum_{j=1}^{M} I_{ij} \, (x_i - \hat{x})^m \, (y_j - \hat{y})^n,$$

where $\hat{x}_i$'s and $\hat{y}_j$'s are the empirical means:

Industrial Image Processing 8

Geometric moments 3

- $\hat{x} = \mathcal{L}^{-1} \sum_{i=1}^{N} \sum_{j=1}^{M} x_i I_{ij}$
- $\hat{y} = \mathcal{L}^{-1} \sum_{i=1}^{N} \sum_{j=1}^{M} y_j I_{ij}$
- Using combinations of v(m, n) Hu derived 7 moments, which (in continuous version) are invariant to translations, rotations and scaling. Hence they are good candidates for features, which are fed to a pattern recognition system. In 1996 Liao and Pawlak indicated that these moments are sensitive to roundoff errors and they proposed moments based on Zernike polynomials, which are orthogonal on the unit disc and rotation invariant.

Industrial Image Processing 8
Geometric moments 4

The first four Hu moments, μ say, are defined as follows:

- $\mu_1 = \mathsf{v}(2,0) + \mathsf{v}(0,2)$,
- $\mu_2 = (\mathsf{v}(2,0) - \mathsf{v}(0,2))^2 + 4\,\mathsf{v}^2(1,1)$,
- $\mu_3 = (\mathsf{v}(3,0) - 3\,\mathsf{v}(1,2))^2 + (\mathsf{v}(0,3) - 3\,\mathsf{v}(2,1))^2$,
- $\mu_4 = (\mathsf{v}(3,0) + \mathsf{v}(1,2))^2 + (\mathsf{v}(0,3) - \mathsf{v}(2,1))^2$.

The rest is too complicated to be reproduced here (see Pratt page 609). We shall return to descriptors later (Fourier descriptors for contours).

Industrial Image Processing 8

Finding objects – again

Segmentation can also be done by:

- **region growing**
- **watershed methods:**
  - **rainfall approach,**
  - **flooding approach.**

They have nice interpretations, but they are rarely used in on-line image processing – we skip their detailed descriptions.

Industrial Image Processing 8
Finding objects by detecting edges

Finding objects by detecting edges is a large class of methods, which try to describe (and then – recognize or gauge) objects by their boundaries. They include boundary detection by:

- the first order approximation of the image intensity gradient,
- by the approximation of the Laplace operator (2-nd order),
- by morphological operations (later).

Industrial Image Processing 8
Image convolution with masks

Before entering into details, we need a general and powerful concept of convolving images with masks.

- A mask is a $(2K+1) \times (2K+1)$ array with elements $h(k, l)$, say, which defines weights applied locally to an image f_{ij}.
- Output image y_{ij} is formed by convolving f with h as follows:

$$y_{ij} = \sum_{k=-K}^{K} \sum_{l=-K}^{K} f_{(i+k)(j+l)} \cdot h(k, l) \qquad (*)$$

Industrial Image Processing 8
Image convolution with masks 2

Mask 3×3, $(K = 1)$:

$h(-1,-1)$	$h(-1,0)$	$h(-1,1)$
$h(0,-1)$	$h(0,0)$	$h(0,1)$
$h(1,-1)$	$h(1,0)$	$h(1,1)$

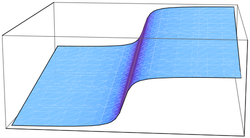
f_{11}	f_{12}	f_{13}						
f_{21}	$\bullet$	f_{23}						
f_{31}	f_{32}	f_{33}						

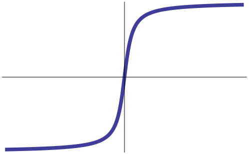
Industrial Image Processing 8
Image convolution with masks 3 – Remarks:

- It is customary and convenient to use masks $(2\,K+1) \times (2\,K+1)$, $K = 1, 2 \ldots,$ but one can also use masks of even size.
- 2D Fast Fourier Transform (FFT) can be used for calculating the convolution (*) efficiently (in $O(\log(M \cdot N))$ time. This approach is advisable when a mask is large.
- As usual, we meet the boundary problem.
- We shall use masks for detecting edges, but their applications are much wider.

Industrial Image Processing 8
Edge – what is it ?

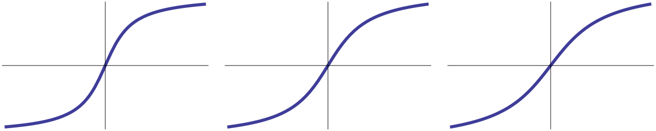
Assume that only one edge is present:





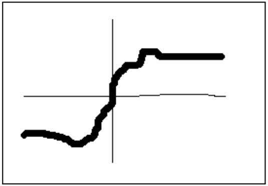
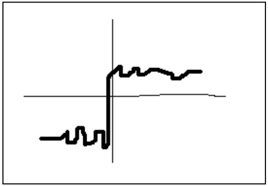
Edge is an abrupt change in an image intensity, but when a change is abrupt ?

Out[35]=



Industrial Image Processing 8

These are also edges:

Working definition: an edge is what is detected by edge detectors.

A nightmare from calculus – gradient operator:

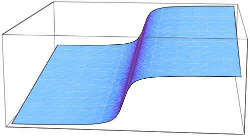
$$\mathrm{grad}\, f(x, y) = [f_x(x, y), f_y(x, y)]^T,$$

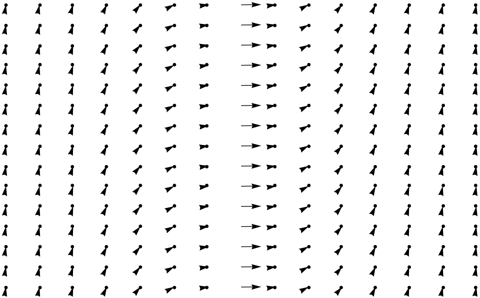
$$f_x(x, y) = \frac{\partial f(x, y)}{\partial x}, \quad f_y(x, y) = \frac{\partial f(x, y)}{\partial y}.$$

Industrial Image Processing 8

Gradient field:

The magnitude of the gradient vector is much larger on edges.





Idea: find places where the intensity gradient magnitude G is sufficiently large (above a threshold) to be treated as an edge of an object.

Industrial Image Processing 8

Gradient magnitude as the edge indicator

Usually, the squared length of $\operatorname{grad} f(x, y)$ is taken:

$$G(x, y) = f_x^2(x, y) + f_y^2(x, y)$$

(or $G = |f_x| + |f_y|$, $G = \max[|f_x|, |f_y|]$) as the indicator of edges.

Partial derivatives can be approximated as:

$$f_x(x, y) \approx \Delta^{-1}\, [f(x + \Delta, y) - f(x, y)]$$

$$f_x(x, y) \approx (2\,\Delta)^{-1}\, [f(x + \Delta, y) - f(x - \Delta, y)]\,.$$

Industrial Image Processing 8

Gradient approximating masks:

These approximations lead to the following masks:

-1	1

$\equiv \quad f_{(i+1)j} - f_{ij}$

-1	0	1

$\equiv \quad f_{(i+1)j} - f_{(i-1)j}$

The second one is preferred. Δ is "hidden" in a threshold for G.

The presence of intensity irregularities and noise forces us to use averages of the gradient approximates.

Industrial Image Processing 8
Gradient approximating masks 2

Masks in x and y directions of Prewitt's 3×3 (smoothed) gradient operator are (left):

-1	0	1
-1	0	1
-1	0	1

1	1	1
0	0	0
-1	-1	-1

-1	0	1
-2	0	2
-1	0	1

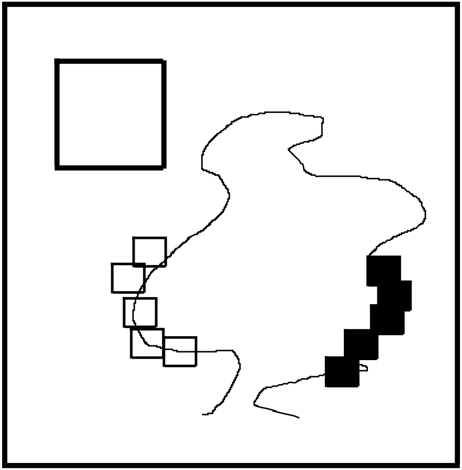
1	2	1
0	0	0
-1	-2	-1

but the Sobel version, putting the emphasis on the central pixel is frequently used (right).

An edge detector must have zero response in constant intensity areas $\Rightarrow$ the sum of all the coefficients must be 0.

Industrial Image Processing 8

Moving window with a mask:



What is it ?

Industrial Image Processing 8

The result of applying the Sobel masks to our slab:

x (left) and y (right) direction. Not thresholded yet and inverted.

Algorithm – edge detection:

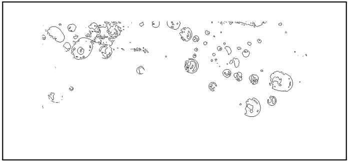
1) Select threshold T.

2) Apply to each (i, j) pixel (non-boundary) the Sobel (or the Prewitt) masks in both directions and obtain approximations $g_x(i, j)$ and $g_y(i, j)$ of f_x and f_y.

3) If $G_{ij} = |g_x(i, j)| + |g_y(i, j)| > T$, declare this pixel as edge.

Industrial Image Processing 8

The result of applying the Sobel masks and thresholding to our slab. Threshold $T = 0.1$.:



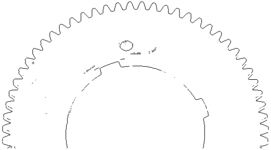
How to choose T ? The simplest: $T = \delta f \cdot C_{norm}$,

$\delta f > 0$ – the minimal jump of gray levels that we want to treat as jump,
C_{norm} – the normalization constant.

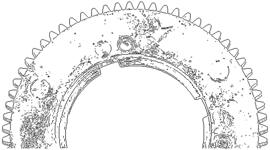
C_{norm} depends on the type of masks, on the definition of G and the image representation [0, 255] or [0, 1].

Industrial Image Processing 8

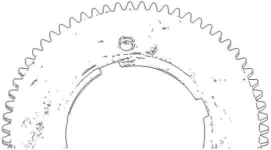
Example of edge detection:



T = 0.15



T = 0.03



T = 0.08

Image was represented in [0,1] gray scale.

Further reading

- . X. Liao and M. Pawlak, On Image Analysis by Moments,IEEE Trans. Pattern Analysis and Machine Intelligence, PAMI-18, 3, March 1996, 254266.
- Otsu, N., "A Threshold Selection Method from Gray-Level Histograms," IEEE Transactions on Systems, Man, and Cybernetics, Vol. 9, No. 1, 1979, pp. 62-66.
- Davies, E.R. (2005) Machine Vision: Theory, Algorithms, Practicalities, Morgan Kaufmann (3rd edition).
- Davies, E.R. (2000) Image Processing for the Food Industry, World Scientific, Singapore.
- Gonzales R. C., Woods R. E., *Digital Image Processing*, 2-nd ed.,Prentice Hall 2002.
- Pratt, W.K., *Digital image processing*, New York, Wiley, 1991.