Zurich University
of Applied Sciences

School of
Engineering
ISC Institute of Signal Processing
and Wireless Communications

Digital Image Processing

M. Weisenhorn/J. Rosset

15. November 2022

Laboratory – Principal Components and the Co-Occurrence Matrix

1 Introduction

Some descriptors require that the maximum diameter of an object is known. Thus the direction of the principal axes of the object must be found, which then is used to rotate the object such that the principal axis is in parallel with the x- or y-axis. With this the bounding box of the object can be constructed (approximated) whose length corresponds to the maximum diameter and the width to the minimum diameter of the object. In the second part, the co-occurrence matrix method is used to compare 3 different textures.

2 Learning goals

- You know how to compute the direction of the principal axes of an object.
- You are able to rotate an image based on the direction of the principal axes, by constructing a rotation matrix and applying it for image rotation.
- You can compare different textures of objects, based on the co-occurrence matrix method.

3 Assignments

3.1 Principal axis and approximation of minimum & maximum diameter

To retrieve an approximation of the maximum diameter and associated bounding box, compute the direction of the object by using the principal components method based on the covariance matrix. The object can be found in file `aspectRatio/bone.tif`.

- The pixel coordinates of the bone are already collected into a matrix $\boldsymbol{A}$. These pixel coordinates are a so called sparse representation of the image.
- Subtract the centroid from the pixel coordinates in $\boldsymbol{A}$ so that their new centroid is $(0, 0)$.
- Compute the covariance matrix of the centered point cloud. For the covariance matrix compute the eigenvector corresponding to the largest eigenvalue. This one points into the object's principal direction.

- Create a rotation matrix $\boldsymbol{R}$ that describes how the object is rotated relative to the desired object's orientation. The first column of $\boldsymbol{R}$ is the principal direction vector. The second column of this matrix is just the first column vector rotated by $90°$.
- To rotate the point cloud coordinates to the desired rotation you could now apply the inverse direction $\boldsymbol{R}^T$ of $\boldsymbol{R}$. Instead of working with the sparse point cloud coordinates, we decide to rotate the original dense image by applying the OpenCV method `cv2.warpAffine()`. This method accepts a 2×3 matrix $\boldsymbol{M}$ that specifies a so called affine transform. It is composed as

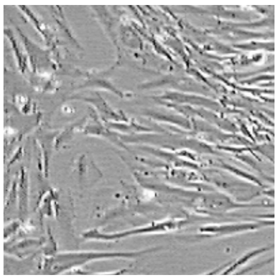
  $$\boldsymbol{M} = \begin{bmatrix} \boldsymbol{R} & \boldsymbol{v} \end{bmatrix},$$

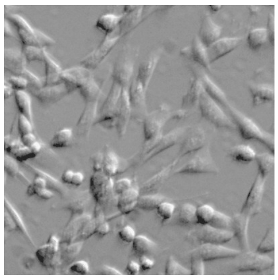
  where R is a 2×2 rotation matrix and $\boldsymbol{v}$ is a 2×1 offset vector. The offset will be needed to shift the rotated object such that the centroid will stay in place. A pretty defintion of the affine transform is given in the documentation of `cv2.getAffineTransform()`.
- After image rotation with the prepared call of the function `cv2.warpAffine()`, please identify the parameters of the bounding box and plot it into the image.
- Finally, compute the *aspect ratio* A, the *roundness* R as well as the *extent* E.

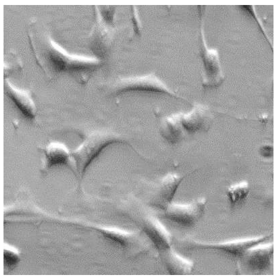
Python hints Related functions for this or alternative solutions are: `np.mean()` to compute the centroid of a pountcloude when given as a matrix, `np.matmul()`. The method `cv2.getRotationMatrix2D()` crates an affine transformation matrix given an offset vector and rotation angle in units of degrees.

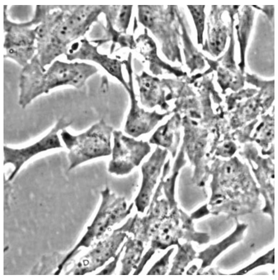
3.2 Co-Occurrence Matrix

A biologist wants to automatically classify different human muscle cells based on their structure. The following four images (here only sections) of such cells:


mc1.tif


mc2.tif

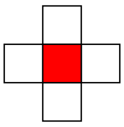

mc3.tif


mc4.tif

Can the four cell types be identified using the 4 features *energy*, *contrast*,*entropy*, and *homogeneity*?

To answer this question, proceed as follows:

a) Determine the co-occurrence matrix of the three images for 256 gray levels with a *4-neighbor* position operator:



b) Normalize the Co-Occurrence Matrix to 1.0 and determine for each image the features *energy*, *contrast*, *entropy* and *homogeneity*.

c) Based on the above results, guess whether the four cell types can be distinguished on the basis of these characteristics.

The biologist would also like to know if it matters at what magnification the cells are photographed. To do this, enlarge all images with factor 2 and again determine the 4 features. Use for this the function `cv2.resize()`. What answer can you give the biologist?