BVA2 – **Exercise 04: *Hough Transformation***

[18 points]

Hough Transformation for Detection of Circles (*iris of human eye*)

The goal of this exercise is to utilize Hough Transformation of circles to detect the iris of human eye. The iris part of human eye has an almost circular shape and thus can be perfectly detected utilizing Hough Transform approaches, even in cases where parts of the eye are occluded, e.g. by the eyelid.

To accomplish this exercise, the following aspects need to be handled:

- (a) Conversion from RGB to grayscale
- (b) in case of noise ➔ apply a smoothing filter
- (c) edge detection
- (d) utilizing a rectangular ROI to derive parameter range for search
- (e) calculation of the Hough Parameter Space
- (f) detect highest fitness in Parameter Space
- (g) Result presentation: (I) best fitting circle as 2D overlay in the input image, (II) 2D Hough-Image for best radius and (III) a MIP (*maximum intensity projection*) calculated in direction of the radius.

Details regarding the mentioned tasks:

For implementation, the interface definitions provided in `HoughTransformIrisDetection_` must be utilized:

- `public static double[][] ` **`cropImage`**`(double[][]inImg, int width, int height, Rectangle roi)` – crop the input image according to provided ROI (region of interest)
- `HoughSpace ` **`genHoughSpace`**`(double[][] edgeImage, int width, int height)` – calculate Hough Parameter Space for edge Image cropped according to user-provided ROI. Search granularity to be defined by the user
- `Vector<Point> ` **`getPointsOnCircle`**`(int x, int y, int radius)` returns coordinates on circle – step size / precision of at least 0.5 pixels useful
- `void ` **`plotBestRadiusSpace`**`(HoughSpace houghSpace)` plot the parameter space in x/y direction for the best radius parameter
- `public void ` **`plotRadiusMIPSpace`**`(HoughSpace houghSpace)` plot the MIP (maximum intensity projection) calculated in direction of the radius

(a) utilize strategies from exercise 01 for conversion RGB ➔ Grayscale

(b) regarding smoothing: evaluate the filter kernel and radius that fits best. Can and should the radius be kept constant at varying image sizes?

(c) as first approach utilize gradient magnitude derived by horizontal and vertical Sobel kernel run as fundament for calculation of circle-fitness. Later on evaluate if better results are achievable if edges are interpreted in a binary way (*edge [0;1] instead of gradient magnitude [0;510]*), similar to Canny edge detection strategy. Binary edges of constant width 1 can be approximated as follows:

- apply Sobel edge detection and calculate the gradient magnitude $|\vec{g}| = \sqrt{g_x^2 + g_y^2}$
- calculate all edge pixels with gradient magnitude > threshold T (chosen to conserve both, the mild and the significant edges)

- calculate the gradient direction ***α*** for all of those edge candidates and check the neighbour pixel in direction of the gradient (*8 neighbours in N8 to be targeted by [0°;360°]*). An edge candidate only remains an edge in the result image if the neighbour in the target direction shows a gradient magnitude lower or equal compared to the current value. $\alpha = \arccos\left(\frac{g_x}{|\vec{g}|}\right)$

(d) A rectangular ROI is provided by the user to restrict the search area. It is expected that the iris is fully located within the ROI. The ROI is utilized to crop the image and to derive the range per dimension in parameter space (*e.g. no need to check a radius > width/2 or > height/2*). The search space should be defined with reasonable limits to reduce runtime.

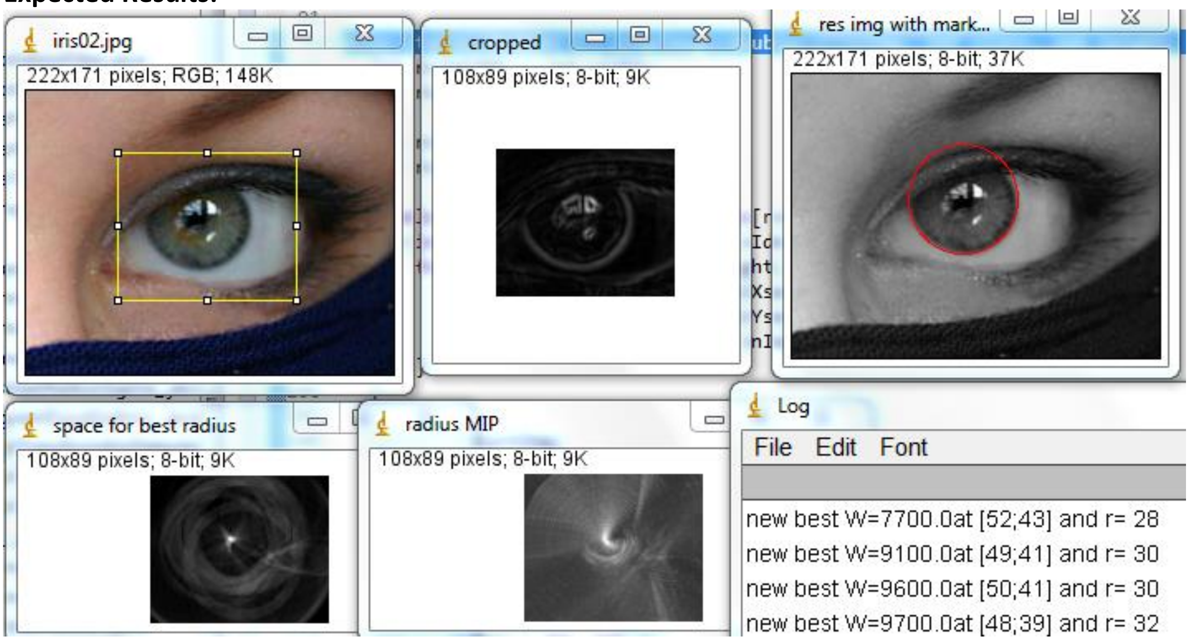
(e) Calculate the 3D parameter space. Fitness is thereby derived from circular shape evaluated according to gradient magnitude or the binary edge results (see (c)). **ATTENTION**: *Is the fitness influenced by the particular radius and why?* If that's the case evaluate and implement a strategy for normalization.

(f) Detect the maximum, i.e. best (x,y,radius) coordinate.

(g) Show the results ➔ the best circle in the original input image (*preferably as red-circle*). Furthermore plot the MIP (maximum intensity projection) in direction of the radius-dimension and the x/y 2D image for best radius. *Discuss these two plots – what kind of conclusions on result quality can be stated based on these evaluations?*

Relevant literature: article *"Probing the Uniqueness and Randomness of IrisCodes: Results From 200 Billion Iris Pair Comparisons"* by mathematician ***John Daugman***, utilizing results of iris detection for human-specific encoding provided on the elearning platform. Allows for identification of a person similar to fingerprints and other biometric features.

Expected Results:



ROI in yellow ➔ utilized for cropping (edge detection)

Best result, i.e. highest fitness is found iterating over entire 3D Parameter Space (see the Log). Best ***P****(x,y,radius)* is utilized to mark iris with red circle. 2D image for best radius and 2D MIP of radius are plotted in the lower row.