

Exercise 03 for MA-INF 2201 Computer Vision WS25/26

04.11.2025

Submission on 10.11.2025

Strict rules you have to follow for all your submissions for this course:

- You are required to write code compatible with Linux, python 3.12, opencv 4.11 and numpy 2.3.3.
- For each exercise, we will include all the packages you can use in the template code. You are not allowed to use any other packages.
- You must submit a code that runs and produces reasonable results.
- Do not cheat and copy the solution from anywhere. We need to verify that the code is yours and that you fully understand it.

Any violation of these rules will result in receiving zero points for the corresponding sheet.

1. **Distance Transform using Chamfer 5-7-11:** Implement a binary distance transform algorithm that computes the distance from each pixel to the nearest edge pixel in an image using the Chamfer 5-7-11 metric.

- Load the image `uni-bonn.jpg`, apply Canny edge detector (`cv2.Canny`) to obtain a binary edge image.
- Implement the Chamfer 5-7-11 distance transform using a two-pass algorithm.
- Visualize the original image, the edge image, and the distance transform result.
- Compare your result with OpenCV's distance transform (`cv2.distanceTransform`).

(4 Points)

2. **Hough Transform for Circle Detection:** Implement the Hough transform to detect circles in an image.

- Load the image `coins.jpg`, implement a function `myHoughCircles` that detects circles using the Hough transform.
- Draw all detected circles on the image.
- Visualize the accumulator (show slices for different radii) and the accumulator slice at the radius with maximum votes.
- Comment on how the results vary with different parameters (threshold, radius range, etc.)

(8 Points)

3. Mean Shift: Implement the mean shift algorithm and use it to find peaks in the Hough accumulator.

- Implement the function `myMeanShift` and use it to find peaks in the accumulator from Task 2. You should load the image `coins.jpg` and use your implementation of `myHoughCircles` in Task 2 to get the accumulator of the image.
- Draw all detected circles on the image.
- Analyze the effect of the bandwidth parameter on peak detection.

(For more details: *D. Comaniciu and P. Meer. Mean Shift: A Robust Approach Toward Feature Space Analysis. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2002.*)

(8 Points)

Please list the names of your group members in the README. Upload all code and documentation to the corresponding Sciebo folder for the sheet (e.g. `sheet03` for this exercise).