CS 4650/7650 ECE 4655/7655 Digital Image Processing 2025 Homework 4: Detection and Segmentation of the structures of interest [100 pts + 10 bonus points]

Out: Thursday Oct 16, 2025
Due: Thursday Oct 30 (Midnight 11:59 PM)

Assignment Objectives:

Detection and segmentation of objects or structures of interest is an important and challenging task for image analysis. Characterization of object properties (i.e., count, shape, color, texture, position, spatial layout) heavily relies on detection and segmentation. The goals of this assignment are to understand

1. Spatial filtering for image pre-processing (e.g. noise removal, smoothing, etc.)
2. Scale-space blob detection using Laplacian of Gaussian filter
3. Clustering-based image segmentation using the K-means method

Beyond the correctness of the code, the quality of the results and discussion of the results will determine the score for this homework.

Part 1: Blob Detection

Potential detection errors that you should try to minimize:

- False negative: an object of interest is not detected
- False positive: detection does not correspond to a real object
- False merge (under-segmentation): single detection for multiple neighboring objects
- False split (over-segmentation): multiple detections for a single object

Task 1: Preprocessing (10 points)

Use Linear (i.e. Gaussian) or non-linear (i.e. median) filtering to smooth the input images and/or to remove noise. You can use built-in filtering functions

a. Show original and filtered images.
b. Briefly discuss why you picked a particular filter and particular filter parameters.

Task 2: Blob Detection (40 points)

1. Convert image to grayscale: either through rgb to gray conversion or by picking a specific color channel.
2. Use blob detection to detect the center of objects of interest: check scale-space Laplacian of Gaussian (LoG) filter from Lecture slides (Lec9_BlobDetection).
   Note-1: you are allowed to use built-in Python, OpenCV, SciPy functions for convolution, correlation, and filter specification.
3. Report: show
   a. original image;
   b. output of the LoG filter;
   c. binary detection mask obtained by thresholding the LoG output (Hint: LoG output takes negative values for bright blobs and positive values for dark blobs);
   d. Answer: Questions to address in the report
      i. At what scale are the nuclei (purple stained regions) detected in test image 1?
      ii. At what scale are the nuclei (purple stained regions) detected in test image 2?
   e. Brief discussion of the parameters (i.e. standard deviation of LoG filter) and outputs (i.e., are the results satisfactory why/why not; was a particular feature better than the others or not? Why?)

   Note-2: The correct filter scale should produce a high response at the center of the cells, not on boundaries.

Part 2: K-means clustering

Implement the following tasks and report the results for the given test images

Task1: Cell Segmentation (50 points)

1. Segment cells/nuclei/particles from the background: use k-means clustering method from Lecture slides (Lec10_Segmentation).
   Note-1: you are allowed to use built-in K-means functions in Python, OpenCV, scikit (i.e. cv.kmeans).
2. Use two different feature sets, show, and discuss your results.
   Note-2: As a feature vector (input to k-means), you can use intensity, color, or any other feature you think is useful.
   Note-3: Most of the built-in K-means functions partition and [n-by-p] input data into k clusters. Here is the number of data points and p is the length of the feature vector. In order to use these functions on image data you will have to reshape your

[rows-by-cols- by-channels] image matrix into [n-by-p] input format, and you will also have to reshape the [n-by-1] output into [rows-by-cols] image format.

3. Report: For each of the test images listed, show
   a. Original image,
   b. Pre-processed image,
   c. Output of K-means clustering: This will be a multi-class output
   d. Nuclei mask (binary mask corresponding to nuclei (purple stained blobs) in the test images). Generate the nuclei mask from the K-means clustering output.
   e. Brief discussion of the outputs and associated programs (i.e., are the results satisfactory why/why not; was a particular feature better than the others or not? Why?)

Task 2: Bonus [10 pts]: Detect/segment transparent red blobs in test image #2.

Reports that follow the correct structure will receive an additional 5 points.

References

1. Slides for Lecture 9: Blob detection.
2. Slides for Lecture 10: Clustering-based segmentation.
3. https://docs.opencv.org/3.4/d1/d5c/tutorial_py_kmeans_opencv.html