EE610 Image Processing

Assignment 4 – Deep Learning and Image Segmentation

Instructions:

1. Form a group of two people. Only one person needs to submit the assignment, and both with get equal points.
2. Submit your assignment, preferably a single .ipynb file, by Nov 6, 2022, 11:59pm on Moodle at: https://moodle.iitb.ac.in/mod/assign/view.php?id=97616
3. Include copious comments about your thoughts in choosing the operations for a particular problem, and your observations about what worked and what did not work, including possible reasons.
4. Comment every line of code to demonstrate understanding of how the code works. Write as much code as you can on your own, but you can use functions from the aforementioned libraries.
5. Cite sources referred at the place where they were used, and include a reference section at the end.

Problems:

1. Study the challenge statement of MoNuSeg challenge, read the data and display a few training and testing images and their masks (black background, and a random color for each cell nucleus). [1]
    a. Training data is in folders Tissue-Images and Binary-Mask at: https://drive.google.com/drive/folders/1WtpstmwwkweNwkxMfcMgIhbiSNYjclvk?usp=sharing
    b. Testing data is in folders Tissue-Images and Binary-Mask at: https://drive.google.com/drive/folders/1ucC1swyylBtaOxzqbH0z4A4Z3B6VBZYV?usp=sharing
2. Code a basic UNet for semantic segmentation in TF+Keras or Pytorch for the binary nucleus vs. non-nucleus segmentation. You may leave a small margin of pixels all around the image from the output. [4]
3. Train the UNet using Dice loss, and show Dice score on validation and test data. [2]
4. Show a few output maps. [1]
5. Modify this baseline architecture to study the impact of changing:
    a. Number of blocks in the UNet architecture [1]
    b. Number of filters per layer in the UNet architecture [1]
    c. Loss function as a combination of Dice and MS-SSIM with a trade-off parameter [2]
6. Use watershed segmentation on the probability map produced by the UNet to segment individual nuclei, and show results on test images with touching or overlapping nuclei. Experiment with various parameters of watershed algorithms to try to improve the results [3]