GNR607 ASSIGNMENT Autumn 2024

Note: Built-in functions permitted for image file reading, image file writing, image display and selected numerical functions like eigenvalue-eigenvector computation, matrix inverse,
List of Topics

1. Generate the correlation matrix of a multiband image and compute its eigenvalues and eigenvectors. (Built-in function permitted for computing eigenvectors and eigenvalues. Students must write the code to compute the covariance matrix and generation of principal components). Compute **all** the principal component images and display them along with the input multiband image.

2. Convert an RGB image into its hue, saturation and intensity components. Modify hue or saturation and redisplay the RGB image after HSI to RGB conversion.

3. Implement Hough transform on a thresholded image to detect lines using both slope/intercept and $\rho-\theta$ approaches and display the accumulator array.

4. Enhance the given low-contrast image using histogram equalization and histogram specification methods and compare the results.

5. Enhance the given low-contrast image using linear, log and exponential contrast enhancement.

6. Display a gray scale image, its pseudo-color display for a user-specified gray scale range, and its negative. Display the histogram to facilitate thresholds and gray level range. Threshold the gray level image using three user-specified thresholds and display the result.

7. Smooth an image using k-nearest neighbour algorithm with k given by the user and compare with simple averaging filter result for the same window size.

8. Smooth an image using Sigma filter and compare with simple averaging filter result for the same window size.

9. Compute intensity gradient using Sobel operator and threshold it using a user-specified threshold to produce an edge image. Display the input and edge images alongside each other.

10. Compute the Fourier transform of an image and generate a low pass filtered image using Gaussian and Butterworth filters.

11. Compute the Fourier transform of an image and compute a high boost filtered image. Built-in Fast Fourier Transform algorithm can be used for Fourier transform computation. Rest of the code should be written by the group.

12. Pan fusion: Fuse multispectral image and PAN image using RGB-HSI method

13. Implement gray scale dilation, erosion, opening, closing and compare with inbuilt functions.

14. Implement K-means algorithm on a multispectral image and compare with inbuilt function for the same number of clusters.

15. Implement binary dilation, erosion, opening, closing on a 2-level image and compare with inbuilt functions.

16. Calculate NDVI of a multispectral image, display its histogram (rescale it to 0 to 255 for display purpose) and density-slice the NDVI data into 3 classes (vegetation, water, and others). Display the classified image.

17. Implement minimum distance to mean and parallelogram classifiers and compare classification results.

18. Classify an image using Maximum Likelihood classifier and compare the result with minimum distance to mean classifier. What difference did the use of covariance matrix make?

19. Implement a moving window GLCM algorithm and generate ASM, CON, ENT, IDM and MEAN features for each window position. Cluster the texture images using K-Means algorithm. (Built-in function for K-Means permitted).

20. Compute a moving window SUM-DIFF algorithm and generate texture features equivalent to ASM, CON, ENT, IDM of GLCM method. Cluster the texture images using K-Means algorithm. (Built-in function for K-Means permitted).

21. Simulate a rotated scaled version of an image and register it with the original image. Show the input image and the registered image alongside each other.

22. Given a multiband image of N bands, compute principal components and generate the approximate version of the input image by inverse principal component transform with 2, 3, …, N-1 components.

23. Generate pan-sharpened version of the input image using the principal component transform approach.

24. Generate zero-crossing based edge detector output for user-specified value of sigma for Gaussian smoothing. Smoothing window size is to be determined based on 3-sigma rule for given value of sigma.

25. Use hit-miss transform method to detect the presence of given binary pattern in a 2-level image.

26. Compute divergence and J-M distance for a randomly selected subset of training samples (20%-50% of the total training samples) and compare these distances when computed with 100% training samples.