**GNR 602 Programming Assignment Topics (2023):**

1. Implement a support vector machine classifier with number of classes > 2, and compare the results of one versus one and one versus rest methods. (LibSVM library or any other suitable library can be used). User specified nonlinear kernel and slack parameters (as supported by the library) may be incorporated. One versus one and one versus rest strategies should be implemented and the results should be compared by the student group.
2. Implementation of clustering using Expectation Maximization (EM) algorithm and comparison with Fuzzy C-Means.
3. Implement FCM clustering algorithm and evaluate the cluster quality for different values of ‘C’. (Minimum and maximum values of C to be specified by user)
4. Implement SIFT or SURF for image matching.
5. Implement Histogram of Oriented Gradients method for extracting all instances of a particular object in the (high resolution) input image.
6. Implement a neural network ensemble using bagging
7. Implement a single layer perceptron classifier (input layer + output layer without any hidden layer) with a polynomial function for nonlinear transformation of the input. Compare this result with the result when no nonlinear transformation of the input is done.
8. Implement Fuzzy Kohenen network (use FCM model into the learning rate and updating strategies of the Kohonen network)
9. Implement Kohonen Self-Organizing Map, with user-specified grid matrix size, and a multispectral image as input. Generate a coded image using the trained SOM as a code book.
10. Implement region growing segmentation by seed selection using morphological operators (requires skills in pointers and tree structures).
11. Implement wavelet transform based image smoothing. Any Daubechies wavelet (db-4 / db-6 …) can be chosen as mother wavelet.
12. Implement wavelet transform based edge enhancement. Any Daubechies wavelet (db-4 / db-6 …) can be chosen as mother wavelet.
13. Implement dimensionality reduction by the noise-adjusted principal component transform method applied to hyperspectral images.
14. Implementation of ICA and compare with MNF (MNF implementation optional)
15. Implement LDA and compare with MNF for dimensionality reduction.
16. Spectral unmixing using K-Means and Fully Constrained Least squares (FCLS)
17. Implement genetic algorithm for band selection in hyperspectral images.
18. Implement genetic algorithm for initialization of weights of a multilayer perceptron network (no backpropagation required).
19. Implement Harris corner detector with and without preprocessing of the (high resolution) input image with Gaussian smoothing filter (sigma parameter is user specified).
20. Implement Canny edge detector with user specified sigma parameter and upper and lower thresholds for edge detection in a remote sensing image.
21. Pan-sharpening of a multispectral image using wavelet transform.
22. Pan-sharpening of a multispectral image using Bovey’s transformation and compare the results with IHS and mean pan sharpening
23. Pan-sharpening of a multispectral image using Gram Schmidt transformations and compare the results with IHS and mean pan sharpening
24. Implement PCA based pan-sharpening algorithm (assume geo-referencing of the datasets is already done).
25. Fuse multispectral image and PAN image using RGB-HSI method.
26. Fuse multispectral image and PAN image using BDSD method.
27. Implement road detection from satellite images using SNAKE algorithm.
28. Implementation of marker-based watershed segmentation.
29. Segment a remote sensed image using spectral clustering.
30. Segment a remote sensed image using hierarchical clustering.
31. Segment a remote sensed image using DBSCAN.
32. Implementation of Mean-shift segmentation of Laws texture features.
33. Implement Mean Shift Segmentation of GLCM feature based for satellite images.
34. Implement quadtree based image segmentation algorithm by region splitting strategy (Normally requires skills in implementing dynamically growing tree structures like *pointers in C language.* Pointer-less implementations are also possible).
35. Implement SIFT based feature matching for registering two images.
36. Implement random-forest based classifier to classify a hyperspectral remote sensing image
37. Implement a Naïve Bayesian classifier to classify a multispectral remote sensing image.
38. Implement the Pure Pixel Index method of finding endmembers and display the abundance map of the classes represented by the endmembers.
39. Implement Laplacian of Gaussian based 2nd derivative operator and detect edges based on zero-crossing criterion in a remote sensed image.
40. Implement salt and pepper noise removal using morphological operators.
41. Implement normalized cross-correlation based template matching method to identify all instances of an object in an image.
42. Implement one iteration of the relaxation labelling algorithm using the online probability updating rule. Initial probabilities may be generated using K-Means clustering algorithm (K is specified by user). Compatibility coefficients will be supplied by the user in a matrix along with the input multispectral image.
43. Implement the seeded region growing algorithm for segmentation of an image (requires skills in pointers and tree structures). Use Tophat transform to generate seed points.
44. Implement seed-point based region growing, assuming randomly selected seed points.
45. Classify an image using Sum-Difference histogram as texture features using SVM/RF/ANN classifier.
46. Segment an image using Kmeans clustering then select samples and extract Laws’ texture feature for classification. (Value of K to be specified by user). SVM/RF/ANN can be used as a classifier.
47. Segment an image using Kmeans clustering then select samples and extract GLCM texture feature for classification. (Value of K to be specified by user). SVM/RF/ANN can be used as a classifier
48. Implement a pyramid based multiresolution GLCM feature based classifier. Extract features like contrast, dissimilarity from windows of different sizes and use it for classification.
49. Implementing Fast GLCM for Remote sensing images. Use the matrix thus generated to extract features and classify.
50. Classify a remote sensed image for land use land cover using KNN, nearest neighbours are computed using Minimum distance classifier, Euclidian, city block, Quasi-Euclidean metrics. Compare the classifications. First use elbow method to determine the optimal number of classes and use the optimal value of k for classification.
51. Using Fisher’s Linear Discriminant classify a remote sensing image and compare the outputs of LDA and PCA based classifiers.
52. Classify a remote sensing image using a kernel PCA based classifier.
53. Use dynamic Local Binary Pattern (DLBP) for textural feature extraction for satellite image change detection.
54. Classify a remote sensed image using GLCM texture features and SVM. See if the window size has any effect on classification.
55. Classify a remote sensed image using various features from LBGLCM(where we first use LBP and then use GLCM) to extract features. See if the value of d(displacement/distance) has any effect on classification. SVM/RF can be used as a classifier
56. Classify a remote sensed image using various Laws texture energy measures using SVM/ANN. Generate an image each for each Laws’ feature and then calculate the various central moments for each image. Use these as features for classification.
57. For a multispectral remote sensed image apply PCA for dimensionality reduction then for PC1 apply GLCM to extract features by varying window size 9\*9, 11\*11, 13\*13, 15\*15, 17\*17 and 19\*19. Classify the image using the features thus collected.
58. Classify a multispectral remote sensed image using a spectral and spatial classifier. Here intensity value of spectral band is used as a spectral component and to obtain spatial component, GLCM features are extracted from first LDA components.
59. Extract features from fast GLCM where the new matrix is generated by adding 'd' new colums and subtraction of first 'd' columns from earlier window. GLCM features are extracted from the thus created window and are used to classify the image.
60. Classify a remote sensed image for land use land cover using KNN where the nearest neighbours are computed using different distance metrics like Euclidian, City block, Quasi-Euclidean. Compare the classifications
61. Classify a remote sensed image for land use land cover using Fisher's Linear Discriminant classifier. Compare the output with LDA and PCA based classifiers.
62. Classify a remote sensing image using a kernel PCA based classifier.
63. Create a model to detect changes in multi-temporal satellite images. It uses Principal Component Analysis (PCA) and K-means clustering techniques over difference image
64. Extract edges for an image, then extract texture features from edge extracted image then use it segment the image
65. Classify the hyperspectral image using spectral Information Divergence where the spectral signatures for various components are given.
66. Segment a remote sensed image using multiple thresholds. Use genetic algorithm for selection of thresholds. Use either OTSU/Kapur/Tsali’s as a fitness function. Here the threshold should be such that 0<T1< …<T5<256.
67. Classify the hyperspectral image using spectral Angle mapper where the spectral signature for various components is given.
68. Cluster a remote sensed image using ISODATA classifier.
69. Implement a minimum noise fraction for a remote sense image.