E9 246 Advanced Image Processing

Assignment 1

Due Date: 05/02/2026

General Instructions:

- You may use Google Colab and any open-source deep learning libraries (e.g., PyTorch, TensorFlow).
- Along with your code, submit a brief report with all results and key observations.
- Place all files into a single zip file and submit it. Name the zip file as `AIP1_YourName`.

---

1 Local Binary Pattern (LBP) for Texture Analysis (Marks = 6+6+8 = 20)

In this question, you will study the effectiveness of Local Binary Pattern (LBP) descriptors for texture recognition under geometric transformations, particularly image rotation. All experiments must be conducted on the subset of the *Describable Textures Dataset (DTD)*, which is available here.

Apply the standard Local Binary Pattern (LBP) operator to each training image using a 3×3 neighborhood. Extract LBP codes, compute the corresponding LBP histograms, and store the histogram feature vector for each training image.

(a) Apply the same feature extraction and histogram computation procedure to the test images. Perform texture classification by matching each test histogram to the closest training histogram using Euclidean distance (1-NN). Report the classification accuracy.

(b) Rotate the test images by multiple angles and repeat the previous experiment. Report the classification accuracy for each rotation angle and analyze the effect of rotation on standard LBP-based texture recognition.

(c) Implement a rotation-invariant LBP descriptor as described in the paper available here. Use this rotation-invariant representation to extract features from both the original and rotated images. Perform histogram-based matching (Euclidean distance, 1-NN) and report the classification accuracy. Compare and analyze the results obtained using standard LBP and rotation-invariant LBP descriptors.

Note: All LBP implementations must be written from scratch using NumPy. Do not use any built-in or library-provided LBP functions. Your report must include both quantitative results (e.g., tables/plots of accuracy) and qualitative analysis.

2 Adaptation to Sketch Domain (Marks = 4+6+6+4 = 20)

In this question, you will analyze the behavior of a deep convolutional neural network pretrained on natural images when adapted to a significantly different visual domain, namely sketches. You will use a subset of 100 classes from the ImageNet-Sketch dataset introduced by Wang *et al.*, available here.

(a) Load a ResNet18 model pretrained on ImageNet. Fine-tune the network on the provided ImageNet-Sketch subset. Report training and test accuracy.

(b) For 5–10 test images, visualize a few feature maps from early, middle, and late layers. Do this for the ImageNet-pretrained ResNet18 model (before fine-tuning) and for your fine-tuned model, and briefly describe the differences. (*Optional reference:* feature-map visualization using forward hooks is shown here.)

(c) Fine-tune ResNet18 again under a constrained setting where *all parameters are frozen except Batch Normalization layers.* Update only BatchNorm parameters. Report performance and compare it with the results from part (a).

(d) What do you observe? How does restricting learning to BatchNorm parameters affect adaptation to the sketch domain? Provide an explanation for the observed behavior.

Note: You may use pretrained models available in PyTorch or TensorFlow. Clearly specify the training protocol, including which layers are frozen or updated in each experiment. Include relevant visualizations, quantitative results, and qualitative analysis in your report.