Assignment#2: Training and Visualizing a CNN on CIFAR

Task 1: CNN Design and Training

This task focuses on designing and training a convolutional neural network (CNN) for image classification.

Model Requirements

- Custom CNN (**no** pretrained models)
- At least **3 convolutional layers**
- At least **1 pooling layer**
- Nonlinear activations (e.g., ReLU)
- Fully connected classification head

You may use PyTorch (skeleton code is provided in Canvas → Modules → JupterNotebooks → build_cnn.ipynb) or TensorFlow.

Training Setup

Dataset: CIFAR-10

You must specify and report:

- Loss function
- Optimizer
- Learning rate
- Batch size
- Number of epochs
- Any regularization or data augmentation used

Expected performance:

- CIFAR-10: ~65–75% test accuracy is sufficient. Accuracy is not graded competitively.

Deliverables for Task 1

- Final test accuracy
- Training and validation loss curves
- Brief description of model architecture and training setup

Task 2: Feature Map Visualization

This task focuses on understanding what the CNN learns internally. If you use Jupyternotebook, you can add the code for this task in the same notebook you used for Task 1. You are welcome to experiment with more visualization than the following required.

Part A: Feature Maps from the First Convolutional Layer

1. Select **3 test images from different classes**.
2. For each image:
   - Pass it through the trained CNN.
   - Extract feature maps from the **first convolutional layer**.
3. Visualize **at least 8 feature maps per image**.

Include:

- The original input image
- The corresponding feature maps
- Clear labels for layers and channels

In your report, briefly discuss:

- What patterns the filters detect (e.g., edges, colors, textures)
- How different feature maps respond to the same input

Part B: Maximally Activating Images for Selected Filters

- Choose **one convolutional layer** (early or middle layer).
- Select **3 individual filters** from that layer.
- For each filter:
  - Find the **top 5 test images** that produce the highest activation.
    - Clarification on "activation": Each convolutional filter produces a 2D feature map (after ReLU), with one activation value per spatial location. To measure how strongly a filter responds to an image, this 2D feature map must be reduced to a single scalar value. You may define the activation of a filter for an image as either:
      - the maximum value in the feature map (strongest local response),
      - or the mean value of the feature map (overall response strength).
    - You must clearly state which definition you use.
  - Visualize these images.

In your report, discuss:

- Common visual patterns among the selected images
- Whether the filter appears to respond to general features or class-specific patterns

Report Requirements

Length: 5–6 pages (including figures)

Required Sections:

1. Model Architecture and Training Setup
2. Training Results
3. Feature Map Visualization (Early Layer)
4. Maximally Activating Images
5. Brief Discussion and Reflection

All figures must be clearly labeled and referenced in the text.

Submission

- Canvas
  - PDF report through file upload
  - Link to your Github repo/project (more details below) in Comments
- Github
  - Create a Github project that includes the following:
    - Name of your repository: CSCI611_<Firstname>_<Lastname>
    - Within your repository, create project/folder: Assignment_2
    - Within your project/folder, include the following:
      - Source code (Python, or C++, or Jupyternotebook which includes your code, notes and all execution traces)
      - README with instructions to run your code
      - Screenshots of how your code is successfully executed and generating required outputs, graphs and tables etc. (if you are using Jupyternotebook, all of them should be stored in the notebook – so no need additional screenshots)
      - The PDF report (same as the one submitted to Canvas)
- Make sure you either **make your Github repo public, or invite me as a collaborator**. Here is an instruction on inviting collaborators. https://docs.github.com/en/account-and-profile/setting-up-and-managing-your-personal-account-on-github/managing-access-to-your-personal-repositories/inviting-collaborators-to-a-personal-repository
  - You can find me on Github using my email: bshen2@csuchico.edu or my Github account name: boshen-csuchico.

Rubrics:

- CNN design and training [30%]
- Feature map visualization (early layer) [30%]
- Maximally activating images [20%]
- Clarity and quality of analysis [10%]
- Source code and execution traces (within Jupyternotebook or screenshots) [10%]