Task 3 Report: Feature Extraction and Comparison from 3D Knee CT Scan

Objective

To design and implement a complete pipeline for 3D feature extraction and comparison from knee CT scans, segmented into anatomical regions: **Tibia**, **Femur**, and **Background**. The features are extracted using a **3D-inflated version of a pre-trained 2D DenseNet121**, and their similarities are evaluated using **cosine similarity**.

Pipeline Components

1. Segmentation-Based Splitting

- A **3D segmentation mask** is loaded and used to isolate three anatomical regions:
    - **Tibia (green)** – label 1
    - **Femur (red)** – label 2
    - **Background** – all other voxels
- The CT volume and the mask are aligned to extract per-region volumes for analysis.

2. 2D to 3D Model Conversion

- A pretrained **2D DenseNet121** is loaded from torchvision.models.
- All Conv2D layers are **inflated into Conv3D**:
    - The 2D weights (out_ch, in_ch, H, W) are repeated along a new depth axis → (out_ch, in_ch, D, H, W)
    - The weights are normalized by dividing by D to maintain activation scale.

3. Feature Extraction

- Each **3D region volume** (Tibia, Femur, Background) is passed through the **3D DenseNet121**.
- Feature maps are extracted from:
    - Last convolutional layer
    - Third-last convolutional layer
    - Fifth-last convolutional layer
- **Global Average Pooling (GAP)** is applied to convert each feature map to a fixed-length feature vector.

4. Feature Comparison

- Pairwise **cosine similarity** is computed between feature vectors:
    - Tibia ↔ Femur
    - Tibia ↔ Background
    - Femur ↔ Background
- Similarity is computed **separately for each of the 3 feature levels** (last, third-last, fifth-last).

5. Result Organization

- Cosine similarity scores are **saved into a CSV file**.
    - **Rows**: Each image (or region pair)
    - **Columns**: Similarity for each of the 3 layers
- Example format:

Pair	Fifth-Last	Third-Last	Last
Tibia-Femur	0.78	0.84	0.88
Tibia-Background	0.32	0.36	0.41
Femur-Background	0.28	0.34	0.39

Improvements with More Time

1. Fine-Tuning the 3D CNN on Medical Data

- **Current Limitation:** The DenseNet121 model is pre-trained on natural images (ImageNet), which are quite different from medical scans.
- **Improvement:** Fine-tune the inflated 3D DenseNet on a medical imaging dataset (e.g., **MedicalNet**, **MRNet**, or custom annotated knee CTs) to

improve feature relevance and accuracy.

2. Use of True 3D Pretrained Models

- Instead of inflating 2D models, leverage pretrained 3D models such as:
    - **MedicalNet** (pretrained on multiple medical datasets)
    - **MONAI**-based networks
    - **3D ResNet, 3D DenseNet** from torchvision or video_classification

3. Segmentation Model Integration

- Currently, masks are provided.
- Add a step to **auto-segment the CT scans** using:
    - 3D U-Net
    - nnU-Net

4. Augmentation and Preprocessing Enhancements

- Add 3D data augmentation techniques to increase generalization:
    - Random crop

    - Rotation
    - Elastic deformation
    - Intensity scaling
- Normalize intensities using medical-specific heuristics (e.g., clipping Hounsfield Units).