AI Engineer Assessment – Semantic Segmentation Pipeline

Objective

Build a full semantic segmentation pipeline that can identify and segment objects (e.g., cars, pedestrians, roads) in urban scenes using the provided dataset. This task is designed to evaluate your practical skills in building, training, and evaluating a semantic segmentation model, as well as your understanding of ML engineering principles.

General Instructions:

1. You don't need to re-invent the wheel, you can use already available models from torch or any other place/package
2. No need to improve accuracy just dummy training is sufficient

Dataset

Use the dataset provided here:

Download Link: https:/drive.google.com/file/d/1AZ9UGqRyrxzqSyfHC5mfRISgoQCvxuex/view

Folder structure:

```
/content/data/
  train/
    -image_name.jpg
    -image_name _mask.png

    ...
  valid/
    -image_name.jpg
    -image_name _mask.png
    ...
```

Your Task

Develop a semantic segmentation model using any deep learning framework. The pipeline should support:
- Loading and preprocessing the data (Think of scale of data is huge)

- Training a segmentation model
- Inference on new/unseen images

Deliverables

You are expected to submit the following:

1. Code (zipped version of the repo) with: (Think of multi-GPU availability)
   - Custom Dataset class or DataLoader
   - Training pipeline with proper augmentation (use resize, normalization, and blurr) and batching
   - Model architecture (pretrained models allowed, e.g., DeepLabV3, UNet)
   - Inference script for predicting segmentation masks

2. Notebook or Python script to demonstrate:
   - End-to-end training and evaluation
   - Inference on a sample image

3. README.md file including:
   - Setup instructions and dependencies
   - Explanation of your model choice and architecture
   - Evaluation results
   - Optional: Insights on failure cases and ideas for improvement

4. (Optional):
   - Dockerfile or environment.yml for reproducibility
   - Brief write-up on deployment: how would you expose the model (e.g., REST API, batch job, edge deployment)