PUPPYOPS - PS 1
Inference

Note: Evaluation script is supposed to be run inside the model Training folder only, its already present in the folder named as evaluate.py

Step 1: Data Preparation

- Use the original dataset that has been provided to us
- Create an empty folder named edited dataset.
- Extract trainid images from labels and put it in a new folder called gtFine.(Use preprocess.ipynb)
- Now put gtFine in the edited dataset folder.
- Copy paste images folder from original dataset to edited dataset folder and rename the image folder as leftimg8bit.

Step 2: Data Preprocessing

- Create a new folder called data
- To perform train test val split, the source path will be the edited dataset folder and destination path will be a folder called data where all the train data will be stored (Use preprocess.ipynb)

Step 3: Model Preparation

- Edit the config file (file is present inside model training folder)
- Open the config folder and go to cityscapes.yaml
- Change the path as the location of data and the cache file as location of data\data_cache\
- Make sure the batch size and epoch counts are upto the requirement (batch size as 1 and epoch as 15)

Step 4: Training and evaluation

Open the terminal at model training folder and execute the following code:-

- *For training*
  python train.py --config ./config/Cityscapes.yaml

- *For evaluating*
  python evaluate.py --config ./config/Cityscapes.yaml

- *For predicting test*
  python predict.py --config ./config/Cityscapes.yaml --split "test"

- *For predicting val*
  python predict.py --config ./config/Cityscapes.yaml --split "val"

Step 5: Testing on test data

We were given test data and then we put it in train file which is present in leftimg8bit folder present in Images folder (Images -> leftimg8bit -> train -> 3 folders of test data given to us)
We have included a folder for testing custom images named “Run Test Images”, which has the trained models saved as well as the script to run the test.

- *For testing*
  python testmax.py --config ./config/predict_test.yaml --split “train”

Step 6: Creating video results (Testing)
By using Making_Test_Video_results.ipynb we will convert the predicted images and segmented images into a concatenated result file where original video is on the left and segmented video is on the right side.
Input has two paths:-
1st path: location of city in original dataset
2nd path: location of segmented images of that city (it will be stored in \predictions\DeepLabV3plus\EfficientNetV2M-shuffle\train\rgb)

Provide destination path according to your feasibility