Computer Vision 2023
(CSE344/ CSE544/ ECE344/ ECE544)
Assignment-4

Max Marks (UG/PG): 60/ 60 **Due Date**: 25-April-2024, 11:59 PM

Instructions

- Keep collaborations at high-level discussions. Copying/plagiarism will be dealt with strictly.
- Your submission should be a single zip file **Roll_Number_HW[n].zip**. Include only the **relevant files** arranged with proper names. A single **.pdf report** explaining your codes with relevant graphs, visualization and solution to theory questions.
- Remember to **turn in** after uploading on Google Classroom. No justifications would be taken regarding this after the deadline.
- Start the assignment early. Resolve all your doubts from TAs during their office hours **two days before the deadline.**
- Kindly **document** your code. Don't forget to include all the necessary plots, and figures in your report. **This assignment will be evaluated based on your report and code**, so do not forget to include the necessary details in your report and submitted code.
- All [**PG**] questions, if any, are **optional for UG** students but are **mandatory for PG** students. UG students will get BONUS marks for solving that question.
- All [**BONUS**] questions, if any, are optional for all the students. As the name suggests, BONUS marks will be awarded to all the students who solve these questions.
- Your submission **must include a single python (*.py*) file for each question**. You can submit *.ipynb* along with the *.py* files. Failing to follow the naming convention or not submitting the python files will incur a **penalty**.

1. (30 points) **Open-Vocabulary Semantic Segmentation:** Open-vocabulary semantic segmentation aims to segment an image into semantic regions according to text descriptions, which may not have been seen during training. More details are available here.

   1. (5 points) Refer to the paper Image Segmentation Using Text and Image Prompts and its Github repository. Install dependencies as given in the README.md and download pre-trained weights.

      (a) (1 point) Initialize the segmentation model with given pre-trained weights.

      (b) (1 point) For a provided example image in the repo generate a segmentation mask.

(c) (1 point) For the same image, try out various prompts to segment all objects present in the picture.

(d) (2 points) Using each of the segmentation heatmaps provided as the output by the CLIPSeg model, visualize the segmented part of the original input image.

2. (25 points) Download the Indian Driving Dataset. Refer to the Indian Driving Dataset Home Page or Indian Driving Dataset Paper to read about the label structure.

   IDD and Cityscapes dataset masks contain multiple classes, out of which you are suggested to work for the classes "Road, Sidewalk, Person, Rider, Motorbike, Bicycle, Car, Truck, Bus, Train, Wall, Fence, Traffic Sign, Traffic Light, Pole, Building, Vegetation, Sky".

   (a) (3 point) Download the dataset and create a dataloader for the same. You may also reuse it from your own first assignment. Visualize the data distribution across the provided dataset.

   (b) (2 points) Visualize two images along with their mask for each class. Color code the classes present in the mask properly.

   (c) (5 points) Using the above 2 images for each class, try out multiple text prompts to get the segmentation masks for the actual class.

   (d) (5 points) Compare your masks with the ground truth masks and report your best text prompt. Justify.

   (e) (10 points) Use the above text prompts to segment **50** image in the dataset. Compute their mAP score and comment on model performance in comparison with *DeepLabv3Plus* model from Assignment 1.

2. (30 points) **Visual question answering:** Visual Question Answering is the task of answering open-ended questions based on an image. They output natural language responses to natural language questions.

   1. (5 points) Refer to the paper BLIP: Bootstrapping Language-Image Pre-training for Unified Vision-Language Understanding and Generation, and its GitHub repository. Follow README.md to install all the dependencies and download pre-trained weights.

      (a) (1 point) initialize model with the pre-trained weights.

      (b) (1 point) For the given sample image of human and dog, generate an answer to the question **"Where is the dog present in the image?"**.

      (c) (1 point) For the same image, generate an answer to the question **"Where is the man present in the image?"**.

      (d) (1 point) Comment on the output and accuracy of the answer for the previous two questions.

      (e) (1 point) Think of a question that generates an answer describing what man is doing in the image.

   2. (25 points) We have provided a sample dataset containing images, questions related to them and their probable answers as annotations in the folder.

(a) (10 points) Generate answers for each question in *sample_questions.json* for each image.

(b) (5 point) Calculate accuracy of the predicted answers obtained for each image with respect to the ground truth annotations.

(c) (10 points) Pick any 5 images randomly (not necessarily from sample dataset) and report a question for which given model completely fails.