Homework #2

Deep Learning for Computer Vision

NTU, Fall 2020

109/10/20

109/11/10 (Tue.) 02:00 AM due

Outline

- Task description & Implementation details
  - Problem 1: Image Classification
  - Problem 2: Semantic Segmentation
- Grading
- Homework policy

Outline

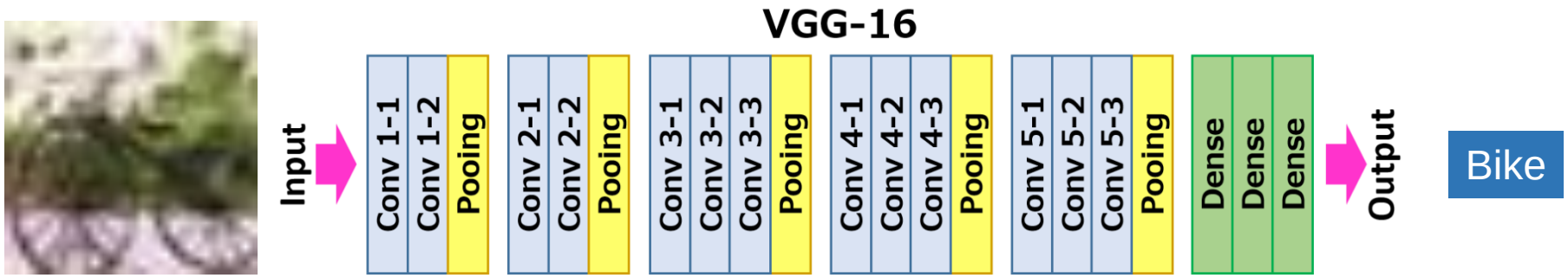
- Task description & Implementation details
  - Problem 1: Image Classification
  - Problem 2: Semantic Segmentation
- Grading
- Homework policy

Problem 1: Image classification

Task Definition

- In this assignment, you will need to perform **image classification**. which predicts labels to each image with CNN models.
  - Input : RGB image
  - Output : classification label

Baseline model



- You can use pre-trained vgg16 (or called feature extractor) as the backbone of your CNN model.

  ***Please use ImageNet pre-trained backbone.**

- You're also allowed to use other networks to fulfill the task.

Dataset

- It consists of 25000 colour images with size 32x32 pixels of 50 classes.
- We split the dataset into
  - train/
    - 22500 images
    - images are named ‘x_y.png’ → x: class label; y:number
  - Validation/
    - 2500 images
    - Naming rules are the same as train/
  - You **CANNOT** use validation data for training purposes.

Outline

- Task description & Implementation details
  - Problem 1: Image Classification
  - Problem 2: Semantic Segmentation
- Grading
- Homework policy

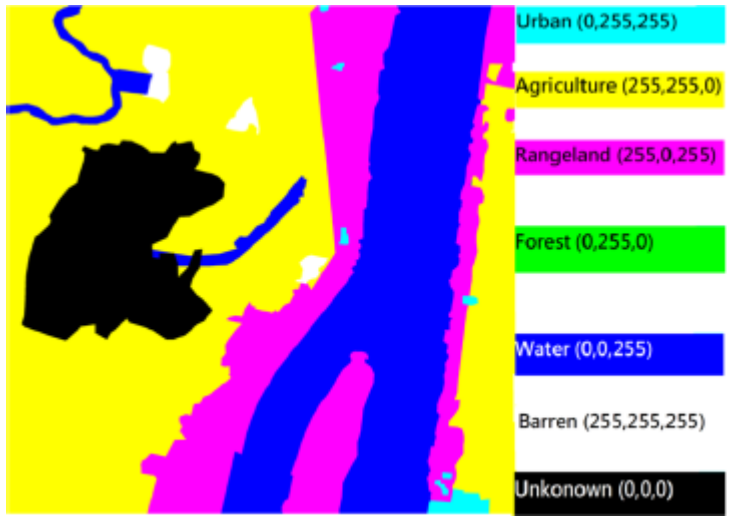
Problem 2: Semantic Segmentation

Task Definition

- In this assignment, you will need to perform **semantic segmentation**, which predicts a label to each pixel with CNN models.
  - Input : RGB image
  - Output : Semantic Segmentation/Prediction

Image

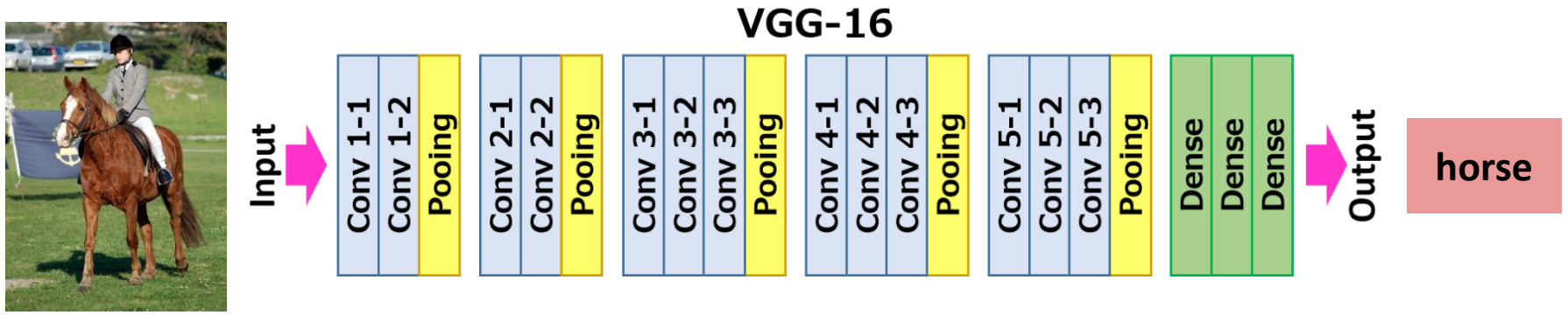
CNN



Semantic Segmentation Prediction

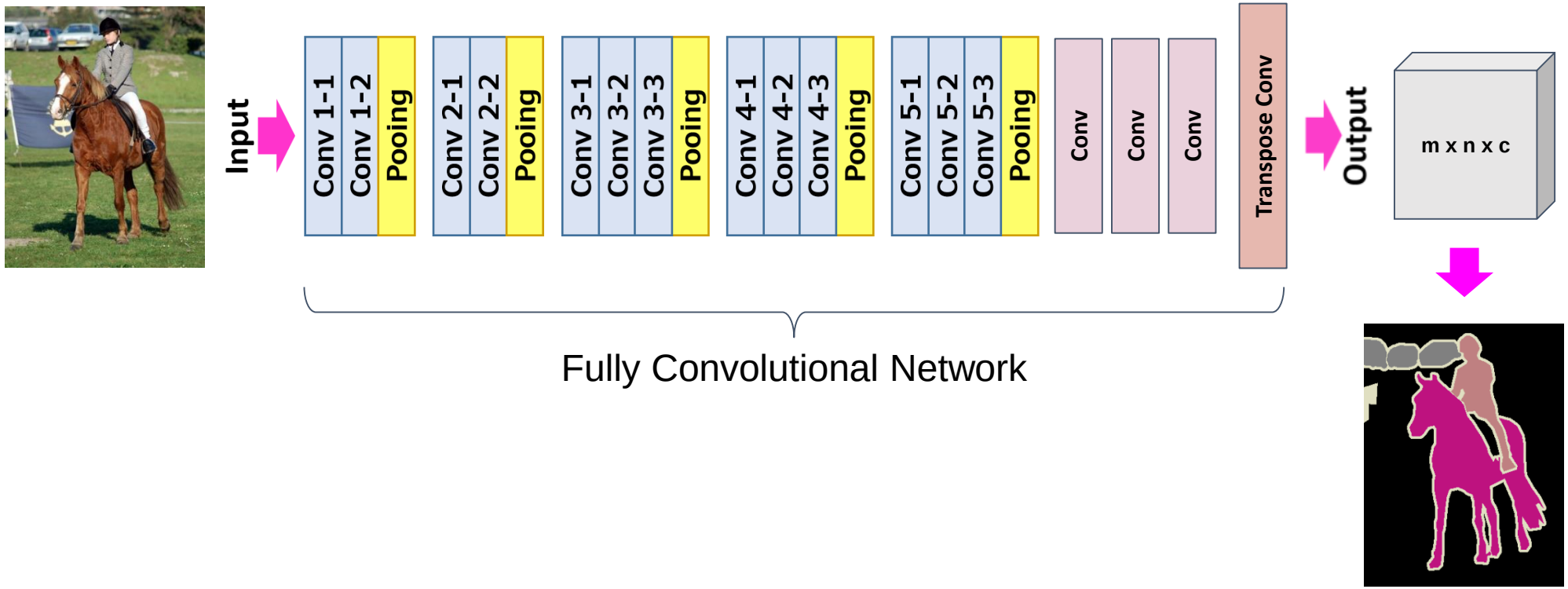
Semantic Segmentation

Convolutional Network - VGG16 for Image Classification



Reference : Long et al., “Fully Convolutional Networks for Semantic Segmentation”, CVPR 2015 (Best Paper) [link]

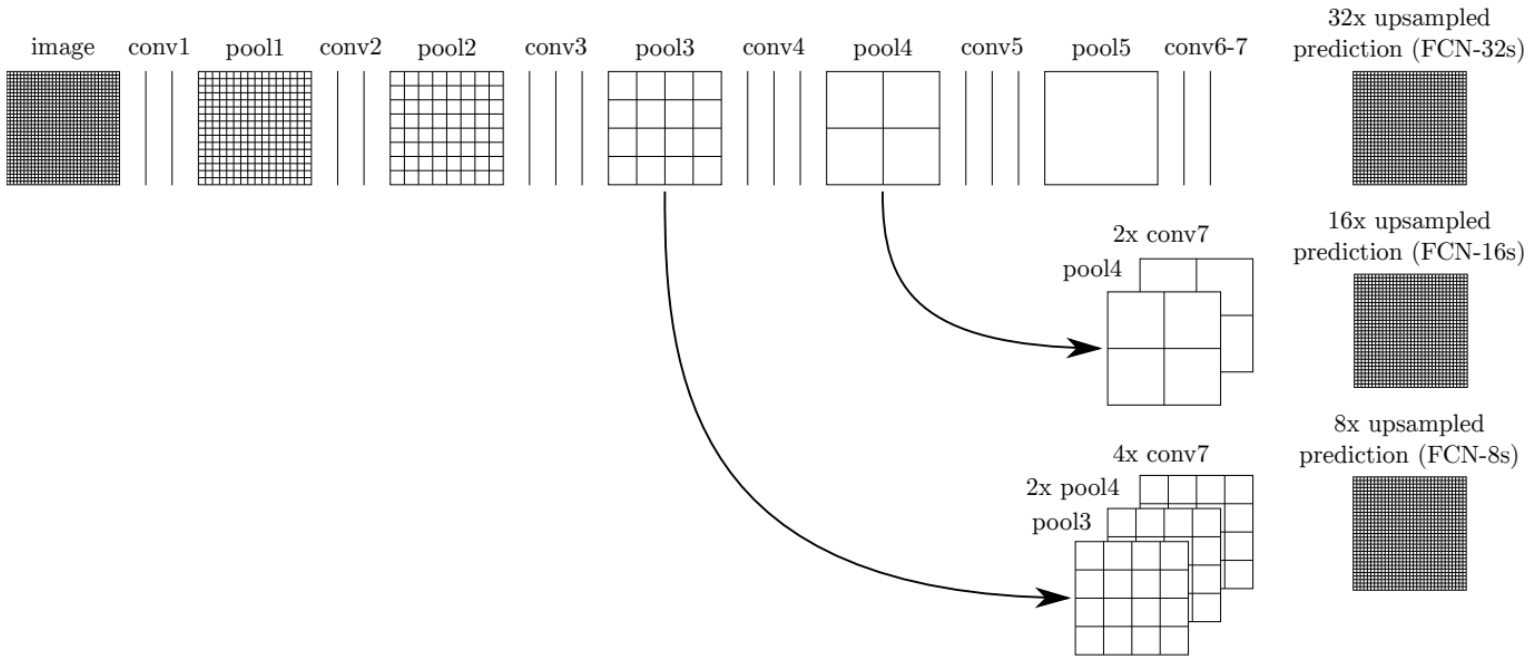
Fully Convolutional Network - FCN 32s



Reference : Long et al., “Fully Convolutional Networks for Semantic Segmentation”, CVPR 2015 (Best Paper) [link]

Semantic Segmentation

Fully Convolutional Network - FCN 32s / 16s / 8s



Reference : Long et al., “Fully Convolutional Networks for Semantic Segmentation”, CVPR 2015 (Best Paper) [link]

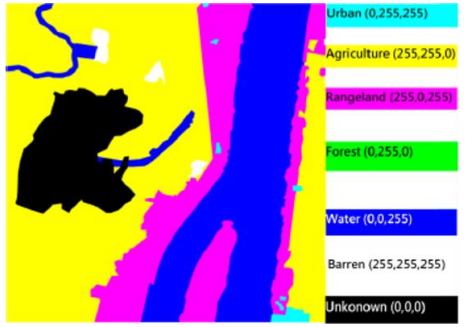
Problem 2: Semantic Segmentation

- In this assignment, you will need to implement two segmentation models and provide discussions in the report.
  1. **VGG16 + FCN32s (baseline model)**
     Implement VGG16-FCN32s model to perform segmentation.
     The results of this model should pass the baseline performance.
  2. **An improved model**
     Implement an improved model to perform segmentation. The performance of this model should be better than that of the baseline model. You may choose any model different from VGG16-FCN32s, such as FCN16s, FCN8s, U-Net, SegNet, etc.

Dataset

Image

CNN



Semantic Segmentation Prediction

- Image size: 512x512
- Mask size: 512x512
- There are **7** possible classes for each pixel.
  - Urban (0, 255 255)
  - Agriculture(255, 255, 0)
  - Rangeland (255, 0, 255)
  - Forest(0, 255, 0)
  - Water(0, 0, 255)
  - Barren(255, 255, 255)
  - Unknown(0, 0, 0)

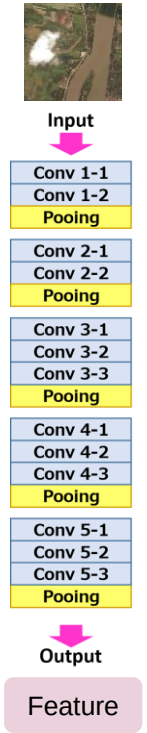
- train/
  - Contains 2000 image-mask (ground truth) pairs
  - Satellite images are named 'xxxx_sat.jpg’
  - Mask images (ground truth) are named 'xxxx_mask.png'
- validation/
  - Contains 257 image-mask pairs
  - Naming rules are the same as train/
  - You **CANNOT** use validation data for training purposes.

Implementation Details

VGG16 Architecture

***Please use ImageNet pre-trained backbone.**

- You will need to load the pretrained weights of following layers:



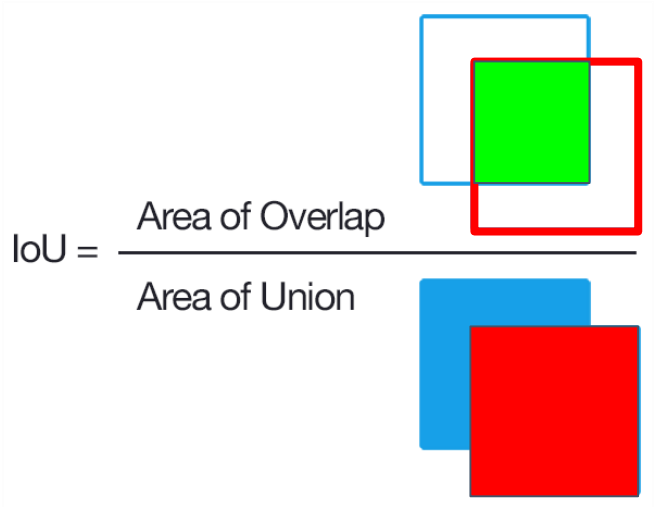
Name	Layer(Size, filter)
block1_conv1	Conv2D(64, 3x3)
block1_conv2	Conv2D(64, 3x3)
block1_pool	MaxPool2D(, 2x2)
block2_conv1	Conv2D(128, 3x3)
block2_conv2	Conv2D(128, 3x3)
block2_pool	MaxPool2D(, 2x2)
block3_conv1	Conv2D(256, 3x3)
block3_conv2	Conv2D(256, 3x3)
block3_conv3	Conv2D(256, 3x3)

Name	Layer(Size, filter)
block3_pool	MaxPool2D(, 2x2)
block4_conv1	Conv2D(512, 3x3)
block4_conv2	Conv2D(512, 3x3)
block4_conv3	Conv2D(512, 3x3)
block4_pool	MaxPool2D(, 2x2)
block5_conv1	Conv2D(512, 3x3)
block5_conv2	Conv2D(512, 3x3)
block5_conv3	Conv2D(512, 3x3)
block5_pool	MaxPool2D(, 2x2)

Model Evaluation

- Evaluation metric: **mean Intersection over Union (mIoU)**
  - For each class, IoU is defined as:

    True Positive / (True Positive + False Positive + False Negative)
  - mean IoU is calculated by averaging over all classes **except Unknown(0,0,0)**.
  - mIoU is calculated over **all test images**.



Outline

- Task description & Implementation details
  - Problem 1: Image Classification
  - Problem 2: Semantic Segmentation
- Grading
- Homework policy

Grading - Problem 1: Image classification(30%)

- Model Performance(20%)
  - Accuracy should be above the baseline score to get points
    - On the validation set(15%): 0.7
    - On the test set(5%): 0.7
- TAs will execute your code to check if you pass the baseline.
- Only TAs have the test data

Grading - Problem 1: Image classification

- Report(10%)

1. (2%)Print the network architecture of your model.
2. (2%)Report accuracy of model on the validation set.
3. (6%)Visualize the classification result on validation set by implementing t-SNE on **output features of the second last layer**. Briefly explain your result of tSNE visualization.

*You can use scikit-learn to fulfill t-SNE. [link]

e.g.

Grading - Problem 2: Semantic segmentation(70%)

- Model Performance(40%)

  mIoU score should be above the baseline score to get points

  - On the validation set:
    - Simple baseline (15%): **0.635**
    - Strong baseline (5%): **0.675**
  - On the test set:
    - Simple baseline (15%): **0.625**
    - Strong baseline (5%): **0.67**
- TAs will execute your code to check if you pass the baseline.
- Only TAs have the test data

Grading - Problem 2: Semantic segmentation

- Report(30%)

1. (5%) Print the network architecture of your VGG16-FCN32s model.
2. (5%) Show the predicted segmentation mask of “validation/0010_sat.jpg”, “validation/0097_sat.jpg”, “validation/0107_sat.jpg” during the early, middle, and the final stage during the training stage. (For example, results of 1st, 10th, 20th epoch)
3. (5%) Implement an improved model which performs better than your baseline model. Print the network architecture of this model.
4. (5%) Show the predicted segmentation mask of “validation/0010_sat.jpg”, “validation/0097_sat.jpg”, “validation/0107_sat.jpg” during the early, middle, and the final stage during the training process of this improved model.
5. (10%) Report mIoU score of both models on the validation set. Discuss the reason why the improved model performs better than the baseline one. You may conduct some experiments and show some evidences to support your reasoning.

Tools

- mIoU:
  - We provide the code to calculate mIoU score.
  - Usage:

    python3 mean_iou_evaluate.py <-g ground_truth_directory> <-p prediction_directory>
- Visualization:
  - We provide the code to draw semantic segmentation map on RGB image.
  - Usage:

    python3 viz_mask.py < --img_path path_to_the_rgb_image> < --seg_path path_to_the_segmentation>

Outline

- Task description & Implementation details
  - Problem 1: Image Classification
  - Problem 2: Semantic Segmentation
- Grading
- Homework policy

Deadline and Academic Honesty

- Report and source code deadline: **109/11/10 (Tue.) 02:00 AM (GMT+8)**
- Late policy : Up to 3 free late days in a semester. After that, late homework will be deducted 30% each day.
- Taking any unfair advantages over other class members (or letting anyone do so) is strictly prohibited. Violating university policy would result in F for this course.
- Students are encouraged to discuss the homework assignments, but you must complete the assignment by yourself. TA will compare the similarity of everyone's homework. Any form of cheating or plagiarism will not be tolerated, which will also result in F for students with such misconduct.

Collaboration Policy

- Searching for online materials or discussing with fellow classmates are highly encouraged. However, you must provide the code or solution by yourself.
- Please specify, if any, the references for any parts of your HW solution in your report (e.g., the name and student ID of your collaborators and/or the Internet URL you consult with). If you complete the assignment all by yourself, you must also specify “no collaborators”.

Submission

- Click the following link and sign in to your GitHub account to get your submission repository:

  https://classroom.github.com/a/-1WYWH3Q

- By default, we will only grade your last submission before the deadline (**NOT** your last submission). Please e-mail the TAs if you'd like to submit another version of your repository and let us know which commit to grade.

Submission

- **DLCV-Fall-2020/hw2** on your GitHub repository should include the following files:
  - hw2_<studentID>.pdf
  - hw2_1.sh (for classification model)
  - hw2_2.sh (for baseline segmentation model)
  - hw2_2_best.sh (for improved segmentation model)
  - your python files (both Training code & Testing code)
  - your model files (can be loaded by your python file)
- Don’t upload your dataset.
- If any of the file format is wrong, you will get zero point.

Trained Model

- If your model is larger than GitHub's maximum capacity (100MB), you can upload your model to another cloud service (e.g., Dropbox). However, you script file should be able to download the model **automatically**.
  - (Dropbox tutorial: https://drive.google.com/file/d/1XOz69Mgxo67IZNQWnRSjT2eZAZtpUAgZ/view)
- Do not delete your trained model before the TAs disclose your homework score and before you make sure that your score is correct.
- Use the **wget** command in your script to download you model files. Do not use the curl command.

Bash Script

- TA will run your code as shown below:
  - bash hw2_1.sh $1 $2
    - $1: testing images directory (images are named 'xxxx.png’)
    - $2: path of folder where you want to output your prediction file(csv.)
- You **should** name your output **csv file** as ‘test_pred.csv’
- The output csv file **should** have the same format as ‘val_gt.csv’
- Note that you should **NOT** hard code any path in your file or script
- Your testing code have to be finished in **10 mins.**

Bash Script (con’d)

- TA will run your code as shown below:
  - bash hw2_2.sh $1 $2
  - bash hw2_2_best.sh $1 $2
    - $1: testing images directory (images are named 'xxxx_sat.jpg')
    - $2: output images directory (You must not create this directory in your code.)
- You **should** name your output semantic segmentation map as 'xxxx_mask.png’
- You should save the predicted semantic segmentation maps to the output directory ($2).
- Note that you should **NOT** hard code any path in your file or script
- Your testing code have to be finished in **10 mins.**

Bash Script (con’d)

- You must **not** use commands such as rm, **sudo, CUDA_VISIBLE_DEVICES**, cp, mv, mkdir, cd, pip or other commands to change the Linux environment.
- In your submitted script, please use the command **python3** to execute your testing python files.
  - For example: python3 test.py < -- img_dir $1> < -- save_dir $2>
- We will execute you code on **Linux** system, so try to make sure your code can be executed on Linux system before submitting your homework.

Packages

- python:3.6
- numpy: 1.18.1
- pytorch: 1.4.0
- torchvision: 0.5.0
- cudatoolkit: 10.1
- scikit-learn: 0.21.3
- scipy: 1.5.2
- pandas: 1.1.3
- matplotlib: 3.3.2
- imageio: 2.9.0
- pillow: 8.0.0
- and other standard python packages
- **E-mail or ask TA first if you want to import other packages.**

Packages

- Do not use imshow() or show() in your code or your code will crash.
- Use os.path.join to deal with path as often as possible.

Penalty

- If we can not reproduce your accuracy or mIoU score on the validation set, you will get 0 points in model performance and you will receive a 30% penalty in your report score.
- If we can not execute your code, we will give you a chance to make minor modifications to your code. After you modify your code,
  - If we can execute your code and reproduce your results on the validation set, you will still receive a 30% penalty in your homework score.
  - If we can run your code but cannot reproduce your accuracy or mIoU score on the validation set, you will get 0 points in model performance and you will receive a 30% penalty in your report score.
  - If we still cannot execute your code, you will get 0 in this problem.

Reminder

- Please start working on this homework as early as possible.
- The training may take a few hours on a GPU or days on CPUs.
- Please read and follow the HW rules carefully.
- If not sure, please ask your TAs!

How to find help

- Google !
- Use TA hours (please check course website for time/location)
- Post your questions under hw2 FAQ section in FB group
- Contact TAs by e-mail: ntudlcv@gmail.com

DOs and DONTs for the TAs (& Instructor)

- Do NOT send private messages to TAs via Facebook.
  - TAs are happy to help, but they are not your tutors 24/7.
- TAs will NOT debug for you, including addressing coding, environmental, library dependency problems.
- TAs do NOT answer questions not related to the course.
- If you cannot make the TA hours, please email the TAs to schedule an appointment instead of stopping by the lab directly.