1. 개발 환경

: MMSegmentation을 위한 개발환경 설정 문서이며, Ubuntu OS를 기준으로 작성되었음. 또한 Nvidia 관련 프로그램들은 (CUDA, cuDNN, Nvidia Driver) Nvidia Quadro RTX 6000 기준으로 작성되었으며, GPU가 다를 경우 버전확인이 필요함

1. 사용 OS: Ubuntu 18.04
2. GPU: Nvidia Quadro RTX 6000
3. MMSegmentation Build 및 이를 위한 환경 설정
4. Build-essential 설치

$ sudo apt-get update

$ sudo apt-get install build-essential

1. CUDA 설치 (11.2.1)

$wget https://developer.download.nvidia.com/compute/cuda/11.1.1/local\_installers/cuda\_11.1.1\_455.32.00\_linux.run

$ sudo sh cuda\_11.1.1\_455.32.00\_linux.run



* Install 후에 나온대로 환경변수를 설정해준다.

다른 버전의 CUDA를 사용할 경우 Nvidia 홈페이지에서 원하는 버전 검색 후 다운로드

* + - <https://developer.nvidia.com/cuda-toolkit-archive>

1. cuDNN 설치 (8.2.1)
   * + <https://developer.nvidia.com/rdp/cudnn-download> 홈페이지 회원 가입 후 cuDNN 다운로드
     + 앞에서 설치한 CUDA 버전에 맞는 cuDNN 다운로드





* + - cuDNN 다운로드 폴더로 이동

$ tar xvfz cudnn-11.3-linux-x64-v8.2.1.32.tgz

* + - Header 및 library 이동

$ sudo cp cuda/include/cudnn\*.h /usr/local/cuda/include

$ sudo cp cuda/lib64/libcudnn\* /usr/local/cuda/lib64

$ sudo chmod a+r /usr/local/cuda/include/cudnn\*.h /usr/local/cuda/lib64/libcudnn\*

1. Anaconda 설치 (4.2)
2. 설치 파일은 <https://www.anaconda.com/products/individual#linux> 또는 wget을 이용하여 다운로드 받는다.

$ wget https://repo.anaconda.com/archive/Anaconda3-2021.11-Linux-x86\_64.sh

1. 설치

$ sudo sh Anaconda3-2021.11-Linux-x86\_64.sh

1. 환경 설정

$ conda create -n mmlab python=3.7

$ conda activate mmlab

1. PyTorch설치 (4의 콘다 환경에서 설치한다)

$ conda install pytorch torchvision torchaudio cudatoolkit=11.3 -c pytorch

1. MMCV 설치

$ pip install mmcv-full -f <https://download.openmmlab.com/mmcv/dist/cu113/torch1.10.0/index.html>

1. MMSegmentation 설치

$ git clone <https://github.com/open-mmlab/mmsegmentation.git>

$ cd mmsegmentation

$ pip install -e .

* MMSegmentation에 필요한 dependency들이 설치 됨

1. 훈련 환경 설정
   1. Configuration 설정

: segformer\_mit-b3\_512x512\_160k\_ade20k.py를 사용하거나 이를 바탕으로 적절하게 수정한다. 특히, 백그라운드를 학습하기 위해 reduce\_zero\_label=True로 설정해준다. 본 문서에서는 NIR데이터와 RGB데이터의 학습을 위한 config를 각각 segformer\_nir\_dist.py, segformer\_rgb\_dist.py 로 저장하였다.

* 1. Pre-trained 모델 다운로드 및 설정

$ wget https://download.openmmlab.com/mmsegmentation/v0.5/segformer/segformer\_mit-b3\_512x512\_160k\_ade20k/segformer\_mit-b3\_512x512\_160k\_ade20k\_20210726\_081410-962b98d2.pth

: 다운로드 후, 저장된 모델을 checkpoints 폴더로 옮긴다 (3-C, 3-D 참조)

데이터 학습

* 1. Segmentation NIR Training (gpu 2개 사용)

$ dist\_train.sh ../configs/segformer/segformer\_nir\_dist.py 2 --work-dir ../results/segformer\_nir\_dist --load-from ../checkpoints/segformer\_mit-b3\_512x512\_160k\_ade20k\_20210726\_081410-962b98d2.pth

* 학습 종료 후 --work-dir 에서 명시해준 ../results/segformer\_nir\_dist 폴더에 Weights 파일과 학습 로그가 저장됨
  1. Segmentation RGB Training (gpu 2개 사용)

$ dist\_train.sh ../configs/segformer/segformer\_rgb\_dist.py 2 --work-dir ../results/segformer\_rgb\_dist --load-from ../checkpoints/segformer\_mit-b3\_512x512\_160k\_ade20k\_20210726\_081410-962b98d2.pth

* 학습 종료 후 --work-dir 에서 명시해준 ../results/segformer\_rgb\_dist 폴더에 Weights 파일과 학습 로그가 저장됨
  1. 참고사항
* 보다 자세한 기능에 대한 내용은 MMSegmentation 참고

https://github.com/open-mmlab/mmsegmentation