

AlPhaPose

2018250033 유규빈
2018250056 함상진

Requirements

1

Python 3.7, Anaconda 3.10, Cuda 11.3

2

Cython, python-package setuptools,
Numpy, torchvision, Pytorch

Requirements

Windows

The installation process is same as above. But note that Windows users may face problem when installing cuda extension. Thus we disable the cuda extension in the setup.py by default. The affect is that models ended with "-dcn" is not supported. If you force to make cuda extension by modify [this line](#) to True, you should install Visual Studio due to the problem mentioned [here](#). We recommend Windows users to run models like FastPose, FastPose-duc, etc., as they also provide good accuracy and speed.

For Windows user, if you meet error with PyYaml, you can download and install it manually from here: <https://pyyaml.org/wiki/PyYAML>. If your OS platform is `Windows`, make sure that Windows C++ build tool like visual studio 15+ or visual c++ 2015+ is installed for training.

Code Installation

Conda 명령어를 이용한 가상환경 생성 및 활성화

```
conda create -n alphapose python=3.7 -y  
conda activate alphapose
```

파이토치 설치

```
conda install pytorch torchvision torchaudio pytorch-cuda=11.3 -c pytorch -c nvidia
```



PackagesNotFoundError: The following packages are not available from current channels

Code Installation

AlphaPose git clone

```
git clone https://github.com/MVIG-SJTU/AlphaPose.git
```

libyaml-dev -> PyYAML

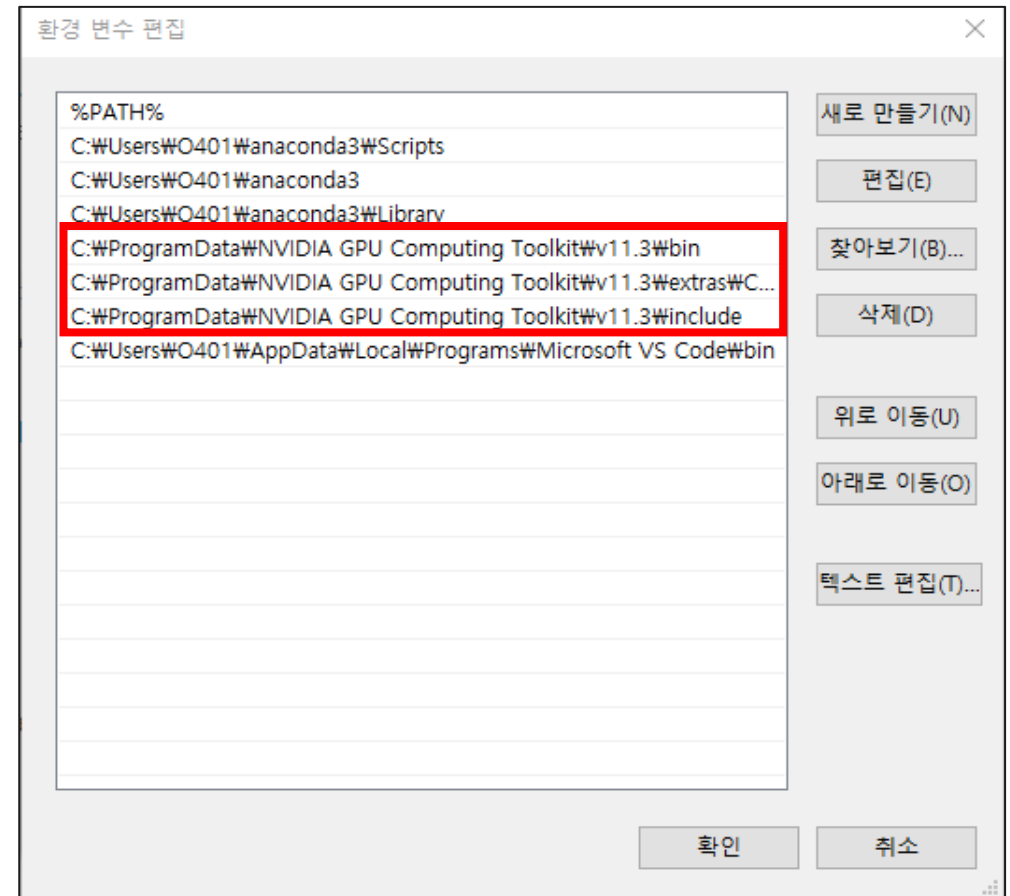
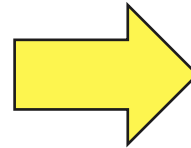
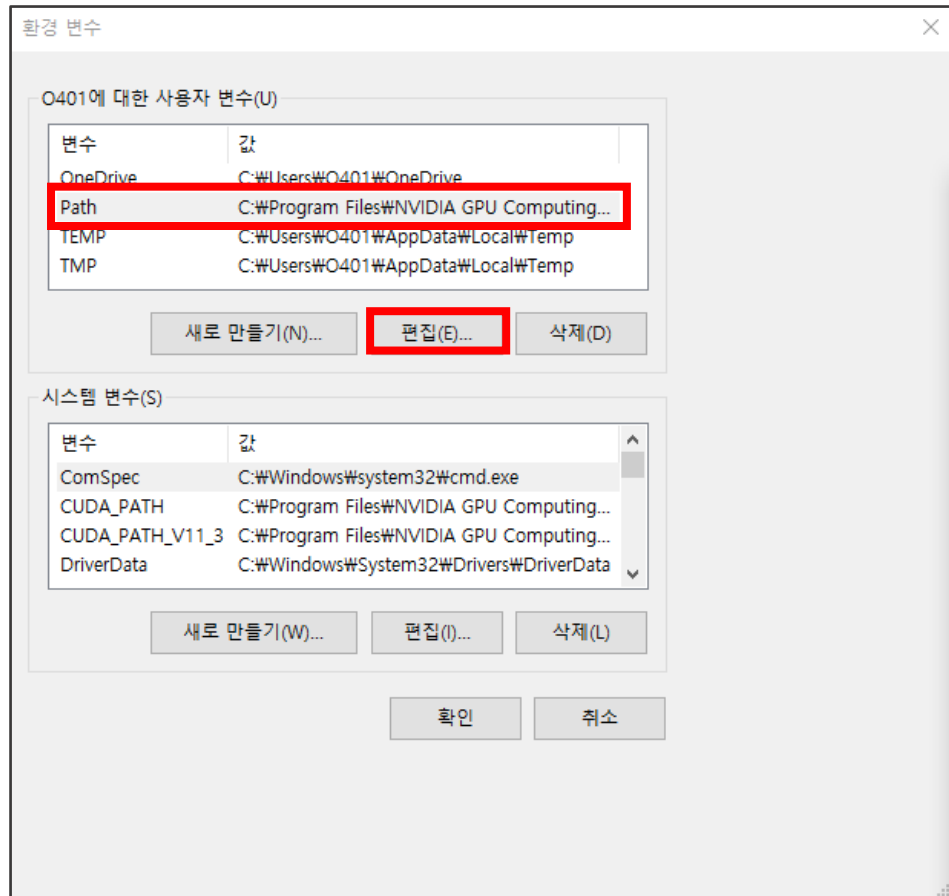
```
sudo apt-get install libyaml-dev
```



```
pip install PyYAML
```

Code Installation

CUDA 환경변수 설정



Code Installation

첫 번째 빌드 시도

```
python setup.py build develop
```



error: Setup script exited with Beginning with Matplotlib 3.6, Python 3.8 or above is required.
You are using Python 3.7.16.

Code Installation

두 번째 빌드 시도

```
python setup.py build develop
```



setup.py

```
if platform.system() != 'windows':
```



```
if platform.system() == 'windows':
```


Models

001 >> yolov3-spp.weights

object detection model을 수동으로 다운로드
detector / yolo / data 디렉터리에 저장

002 >> (Optional) YOLOX

detector로 YOLOX를 사용하고 싶다면 다운로드 후
detector / yolox / data 디렉터리에 저장

Models

003 >> pose models 다운로드

MODEL_ZOOmd을 참고하여 모델 다운로드

Strongly recommended 카테고리에 있던 Multi Domain Models중 2번째 모델을 선택
pretrained_models 디렉터리에 저장

004 >> Pose tracking

Recommended 인 human-ReID based tracking을 선택

다운로드 후 Alphapose/trackers/weights 디렉터리에 저장

사용하려면 --pose_track 이라는 추가적인 flag 사용

실행

1

GETTING_STARTED.md 문서 참고

2

적당한 config file을 찾지 못함

3

Quick Start의 colab example을 참고

실행

001 >> 실행문 작성

```
python script/demo_inference.py
    --cfg config/coco/resnet/256x192_res50_lr1e-3_1x.yaml (config 파일)
    --checkpoint pretrained_models/multit_domain_fast50_dcn_combined_256x192.pth (훈
    려된 모델)
    --indir examples/demo/ (입력 데이터)
    --save_img (출력 형식)
    --pose_track (트래커 사용)
```

002 >> 문제 발생

size mismatch for duc2.conv.weight: copying a param with shape torch.Size([1024, 256, 3, 3])
from checkpoint, the shape in current model is torch.Size([512, 256, 3, 3]). 라는 오류 메시지
모델크기와 체크포인트에서 복사하는 매개변수의 크기가 달라서 생기는 오류

003

>>

실행문 재작성

```
python script/demo_inference.py
    --cfg config/coco/resnet/256x192_res50_lr1e-3_1x.yaml (config 파일)
    --checkpoint pretrained_models/fast_res50_256x192.pth (코랩 코드 참고)
    --indir examples/demo/ (입력 데이터)
    --save_img (출력 형식)
    --pose_track (트래커 사용)
```

config 파일과 checkpoint 파일의 호환성 문제로 예상
코랩 예시코드에서 사용한 checkpoint를 찾아 다운로드 후 실행

실행



실행



실행



결론

001 >> Install PyTorch3D (Optional, only for visualization)

Code installation의 마지막 단계
현재 Optional이고 conda로 설치가 되지 않아 넘어간 부분이지만
앞으로 우리가 작업할 때 도움이 될 수 있는 부분이므로 확인 필요

002 >> Models

코랩의 예시코드로 실행 가능하게 만들기 위한 모델(체크포인트)만을 사용했음
앞서 발생한 size mismatch 문제를 해결하고
MODEL ZOO.md에서 더 다양하고 성능이 좋은 모델을 사용

003 >> 웹캠 사용

아직 웹캠을 구입하지 못해 실시간 성능을 확인하지 못함