

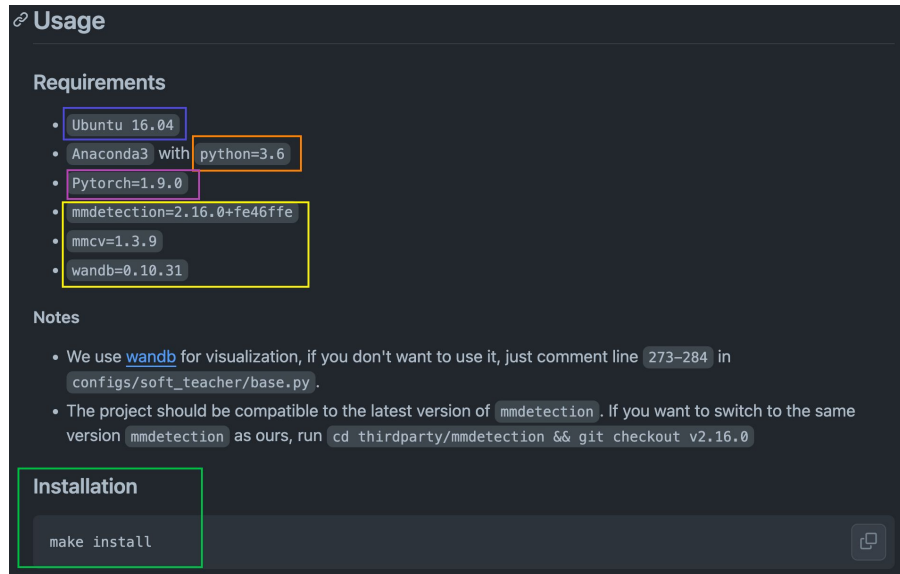
Virtual Environment

Challenges beyond pip and conda

1. Python version
2. CUDA version
3. Linux kernel version
4. Incompatible package version
5. ...

This happens a lot when you reproduce the codebase from prior work.

Exhausted when solving all these? Why not make the whole challenge disappear!



The image shows a dark-themed screenshot of a README file. At the top, there is a section titled "Usage" with a link icon. Below it is a "Requirements" section with a list of dependencies: Ubuntu 16.04, Anaconda3 with python=3.6, Pytorch=1.9.0, mmdetection=2.16.0+fe46ffe, mmcv=1.3.9, and wandb=0.10.31. Each item is enclosed in a colored box (blue, orange, purple, yellow, grey, yellow). Below the requirements is a "Notes" section with two bullet points. The first bullet point mentions using wandb for visualization and refers to lines 273-284 in a file named configs/soft_teacher/base.py. The second bullet point discusses compatibility with the latest version of mmdetection and provides a command to checkout version v2.16.0. At the bottom, there is an "Installation" section with a code block containing the command "make install" and a copy icon.

```
Usage
```

Requirements

- Ubuntu 16.04
- Anaconda3 with python=3.6
- Pytorch=1.9.0
- mmdetection=2.16.0+fe46ffe
- mmcv=1.3.9
- wandb=0.10.31

Notes

- We use wandb for visualization, if you don't want to use it, just comment line 273-284 in configs/soft_teacher/base.py .
- The project should be compatible to the latest version of mmdetection . If you want to switch to the same version mmdetection as ours, run cd thirdparty/mmdetection && git checkout v2.16.0

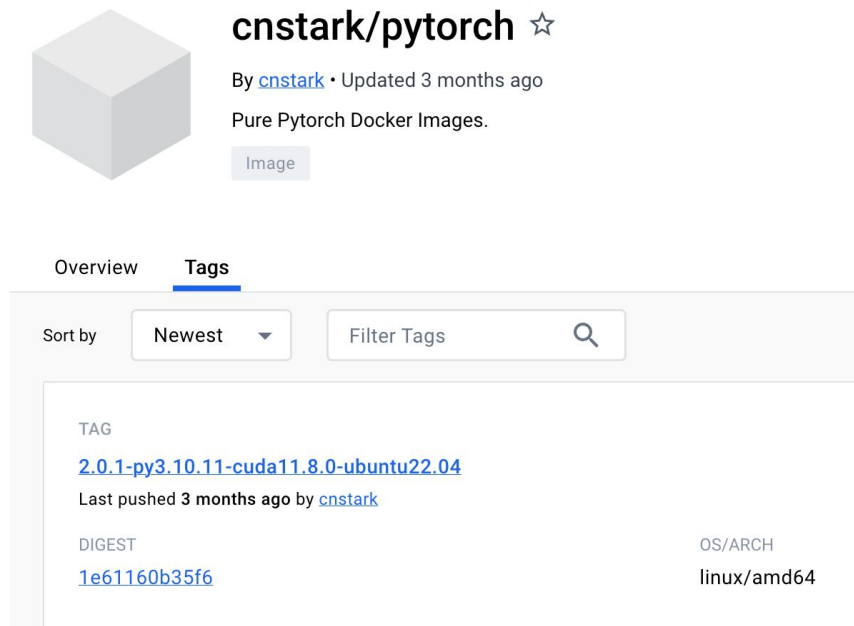
Installation

```
make install
```

Docker (My Workflow)

Docker is an open-source platform that automates the deployment of applications inside software containers

1. Select an docker image with suitable
 - a. Linux Version
 - b. CUDA version
 - c. System-level packages
 - d. Python packages (torch+cuda)



The screenshot shows the Docker Hub interface for the repository `cnstark/pytorch`. It includes a 3D cube icon, the repository name, the author `cnstark`, the update time (3 months ago), and a description: "Pure Pytorch Docker Images." There is an "Image" button. Below this, there are tabs for "Overview" and "Tags", with "Tags" being the active tab. The "Tags" section includes a "Sort by" dropdown set to "Newest" and a "Filter Tags" search box. The main content area displays a table of tags with columns for TAG, DIGEST, and OS/ARCH.

| TAG | DIGEST | OS/ARCH |
|--|------------------------------|-------------|
| 2.0.1-py3.10.11-cuda11.8.0-ubuntu22.04 | 1e61160b35f6 | linux/amd64 |

Docker (My Workflow)

Docker is an open-source platform that automates the deployment of applications inside software containers

2. Pull the image to the local machine.
3. Create the container and link local folders to folders within the container.

Now only the change on the selected folders will be projected back to outside.

```
augustovq at August0vQ-UDnew in ~/Documents/Helper  
λ docker pull cnstark/pytorch:2.0.1-py3.10.11-cuda11.8.0-ubuntu22.04  
2.0.1-py3.10.11-cuda11.8.0-ubuntu22.04: Pulling from cnstark/pytorch  
445a6a12be2b: Extracting [=====] 27.85MB/29.54MB  
f5e061a59d50: Download complete  
1fbc487da933: Download complete  
927548b89dc5: Downloading [>] 31.63MB/2.431GB  
105bd9c63fd5: Download complete
```

```
docker2.bash × [▶] [□] [⋮]  
Helper > docker2.bash  
1 #!/usr/bin/bash  
2  
3 # 1 PORTAL_1: lab portal 8888:8888  
4 # 2 PORTAL_2: vscode portal 8022:22  
5 # 3 NAME: container name "test"  
6 # 4 FOLDER: folder name /home/augustovq/Documents/Project/Test  
7 # 5 IMAGE: image name augustovq/ufoym:latest  
8 # 6 COMMAND: command to run  
9  
10 docker run --gpus all -i -t -u 0 -p ${1:-8888:8888} -p ${2:-8022:22} --name=${3:-"test"} -v ${4:-/home/augustovq/Documents/Project/Test}:/project -v /home/augustovq/Documents/Helper:/helper -v /home/augustovq/Documents/DATA:/data --ipc=host ${5:-augustovq/nvidia-cuda:latest} ${6:-bash}  
11 # jupyter lab --no-browser --ip=0.0.0.0 --allow-root --LabApp.allow_origin='*' --LabApp.root_dir='/project' --NotebookApp.token='' --NotebookApp.password=''
```

Docker (My Workflow)

4. (advanced) understand docker image.

Docker images are stacked, that means for example:

Layer 1-6 are for ubuntu 22.04;

Layer 7-11 are for cuda;

Layer 12-14 are for python packages;

Later installations are based on top of the previous ones.

| IMAGE LAYERS ? | | |
|-----------------------------|--|-----------|
| 1 | ARG RELEASE | 0 B |
| 2 | ARG LAUNCHPAD_BUILD_ARCH | 0 B |
| 3 | LABEL org.opencontainers.image.ref.name=ubuntu | 0 B |
| 4 | LABEL org.opencontainers.image.version=22.04 | 0 B |
| 5 | ADD file ... in / | 28.17 MB |
| 6 | CMD ["/bin/bash"] | 0 B |
| 7 | ENV NVIDIA_DRIVER_CAPABILITIES=compute,utility | 0 B |
| 8 | RUN /bin/sh -c apt update | 105.11 MB |
| 9 | ARG PYTHON_VERSION | 0 B |
| 10 | RUN [1 PYTHON_VERSION=3.10.11 /bin/sh -c | 65.19 MB |
| 11 | ARG PYTORCH_VERSION | 0 B |
| 12 | ARG PYTORCH_VERSION_SUFFIX | 0 B |
| 13 | ARG TORCHVISION_VERSION | 0 B |
| 14 | ARG TORCHVISION_VERSION_SUFFIX | 0 B |

Docker (My Workflow)

5. (advanced) customize docker image.

Since you cannot always find a nice image which meets all of the requirements, I usually customize my own image on top of a simple base image:

nvidia/cuda, of which you can select from all combinations of CUDA version X linux kernel version.

But there is nothing else in the image.



nvidia/cuda ☆

By [nvidia](#) • Updated 8 days ago

CUDA and cuDNN images from gitlab.com/nvidia/cuda

Image

| | | |
|---|--|---|
| <p>TAG 12.3.1-devel-ubi8 Last pushed 8 days ago by svccomputepackagin363</p> <p>DIGEST fb0ac5fcdfdb 17c78dede901 000c9b25e97b</p> | <p>OS/ARCH linux/amd64 linux/arm64 linux/ppc64le</p> | <p>TAG 12.1.0-cudnn8-devel-ubuntu20.04 Last pushed 22 days ago by svccomputepackagin363</p> <p>DIGEST a29da661f2d8 da5f69611ae7</p> |
| <p>TAG 12.3.1-devel-rockylinux8 Last pushed 8 days ago by svccomputepackagin363</p> <p>DIGEST 15af66ec04f1 e4d8d533befe</p> | <p>OS/ARCH linux/amd64 linux/arm64</p> | <p>TAG 11.3.1-cudnn8-runtime-centos7 Last pushed 22 days ago by svccomputepackagin363</p> <p>DIGEST 2ffa68597097</p> |
| <p>TAG 12.3.1-devel-rockylinux9 Last pushed 8 days ago by svccomputepackagin363</p> <p>DIGEST 122cb04f5e43 6ce44ba6b44a</p> | <p>OS/ARCH linux/amd64 linux/arm64</p> | <p>TAG 12.1.0-devel-ubuntu22.04 Last pushed 22 days ago by svccomputepackagin363</p> <p>DIGEST 3c6687b5b582 998c1e2e6c0c</p> |

Docker (My Workflow)

6. (advanced) build docker image.

- a. Choose a base image
- b. Update ubuntu and install packages

```
1 # Use only pytorch based deepo image
2 FROM nvidia/cuda:12.1.1-devel-ubuntu22.04
3
4 ARG DEBIAN_FRONTEND=noninteractive
5
6 # Update ubuntu, install python, pip, and other dependencies
7 RUN apt update && apt upgrade -y && \
8     # Install basic dependencies
9     apt install -y \
10    # Basic
11    git ffmpeg libsm6 libxext6 ntp \
12    # Terminal
13    fish zsh tmux screen && \
14    # Change default shell to fish
15    chsh -s $(which fish) && \
16    # System cleanup
17    apt-get update && apt-get clean -y && \
18    # Install dependencies
19    apt-get install -y \
20    # Basic
21    build-essential curl wget python3 python3-pip nodejs vim nano exa bat systemd cargo && \
22    # Package 1: microsoft-prod
23    # wget https://packages.microsoft.com/config/ubuntu/20.04/packages-microsoft-prod.deb -O
24    # dpkg -i packages-microsoft-prod.deb && \
25    # rm packages-microsoft-prod.deb && \
26    # rm -rf /var/lib/apt/lists/* && \
27    # Package 2: lsd
28    # cargo install lsd && \
29    # Make downloads directory
30    mkdir downloads && \
31    # Cleanup
32    # apt update && apt install -y --allow-change-held-packages cuda-libraries-12-2 cuda-libr
33    libcublas-dev-12-2 && \
34    apt autoremove -y && apt autoclean -y
```

Docker (My Workflow)

6. (advanced) build docker image.

- Choose a base image
- Update ubuntu and install packages (git, ffmpeg, zsh, fish, etc.)
- Install Anaconda (or mamba)
- Install python packages

Everytime I start a new subproject, I just run my customized image and do everything inside container.

```
35 # Install anaconda latest
36 ENV CONDA_DIR /opt/conda
37 RUN wget --quiet https://repo.anaconda.com/archive/Anaconda3-2023.09-0-Linux-x86_64.sh -O ~/anaconda.sh && \
38     /bin/bash ~/anaconda.sh -b -p /opt/conda
39
40 # Put conda in path so we can use conda activate
41 ENV PATH=$CONDA_DIR/bin:$PATH
42
43 RUN rm /root/anaconda.sh && \
44     # Update conda
45     conda init fish && \
46     conda install conda-build -y && \
47     conda update conda -y && \
48     conda update conda-build -y && \
49     conda update --all && \
50     # Update pip
51     pip install --upgrade pip && \
52     # Install pip packages
53     pip install gdown \
54     jupyter ipykernel jupyter \
55     # torch torchvision torchaudio torchmetrics pytorch-lightning \
56     black isort autopep8 && \
57     gdown 1TNyg6Qri0szwErbPMYF2s42SXrRaawgs && \
58     # lsd configurations
59     mkdir ~/.config/lsd && \
60     mv config.yaml ~/.config/lsd/ && \
61     cd downloads && \
62     curl https://raw.githubusercontent.com/oh-my-fish/oh-my-fish/master/bin/install > install && \
63     fish install --noninteractive --yes && \
64     fish -c "omf install lambda" && \
65     fish -c "omf update" && \
66     # Cleanup
67     apt update && apt upgrade -y && \
68     apt autoremove && apt autoclean -y
```


Docker (My Workflow)

The major disadvantage of docker is the requirement of SUDO permission, which means basically it cannot be deployed to cluster.

But by using the same ENV on the cluster in a docker image we can avoid some debugging due to transformation, and export the package list then reinstall them on cluster.

```
augustovq at August0vq-UDnew in ~/Documents/Helper
└─┬ sudo docker images
  [sudo] password for augustovq:
  REPOSITORY          TAG          IMAGE ID      CREATED      SIZE
  augustovq/vidia-cuda latest       f4c9d8732633 7 weeks ago 17.1GB
  nvidia/cuda         12.2.2-devel-ubuntu22.04 f2c3ae91f29c 7 weeks ago 7.1GB
  cnstark/pytorch    2.0.1-py3.10.11-cuda11.8.0-ubuntu22.04 0c016883bc4b 3 months ago 5.18GB
  nvidia/cuda         12.1.1-devel-ubuntu22.04 9df437e8fe97 5 months ago 7.03GB
```

```
Welcome to fish, the friendly interactive shell
Type help for instructions on how to use fish
● └─ root at ccdd328d1524 in /project (main ✓)
  └─ # whoami
    root
  └─ #
○ └─ root at ccdd328d1524 in /project (main ✓)
  └─ #
```

Within container

Virtual Environment (Summary)

1. I use Docker, but other softwares are also good
2. Pull the docker image from online or customize your own docker image that completely meets the requirement
3. Run the image to create the container, and do everything in the container, keep your local environment clean and tidy.
4. Wrap-up the whole project simply, then either transfer to cluster or archive it.

Docker is way more than that (might took several weeks to learn), but for our research purpose the above steps are already enough.

This will only take you 20 minutes to copy-and-do, and another 2 to 3 hours to understand.

And after that, experiments becomes 5x easier and well-organized.

Random opinions

Things that I'm passionate about

- **tmux** rather than screen for multiplexing and preventing any issues coming from connection dropping
 - (screen also work but not as pretty imho)
- the **fish** shell has nice autocompletions built-in
 - (it's non POSIX compatible not a perfect drop-in replacement for bash/zsh but it's not too bad)

My very own Roman empire

- **mamba** is *really* nice and fast at solving environments and works just like conda
 - (if you don't wanna make the switch you can just use [mamba's solver in conda](#))

Let's try to create a large environment by saving the environment definition file from above to `env.yml` and installing directly from there.

```
time conda env update --file env.yml

real 10m51.233s
user 10m4.853s
sys 0m12.286s
```

```
time mamba env update --file env.yml

real 1m0.634s
user 0m45.550s
sys 0m4.051s
```

From <https://towardsdatascience.com/conda-too-slow-try-mamba-c29faf1e64cc>

10x faster!

