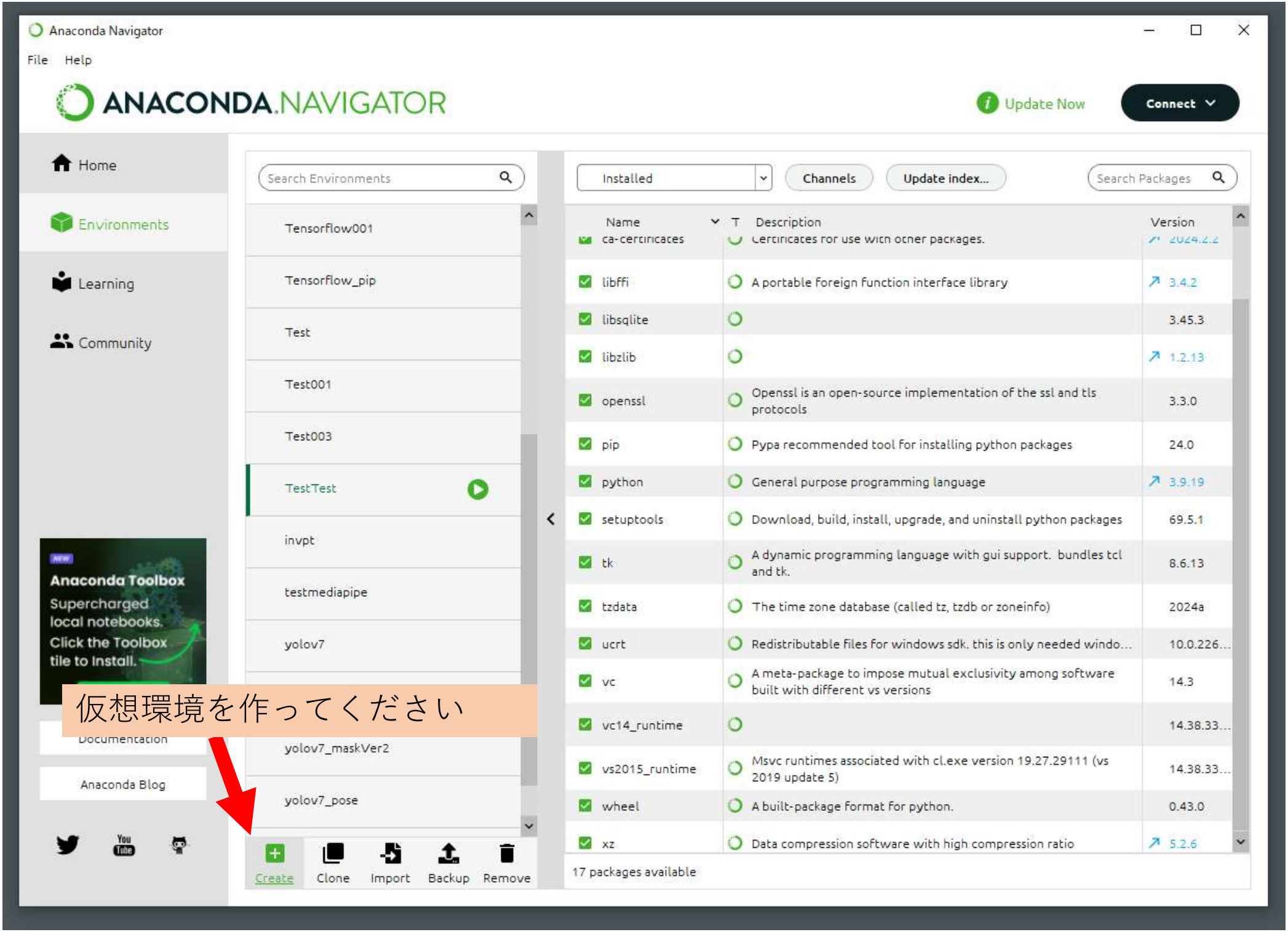
Yolov7-maskのインストール
(仮想環境の構築)

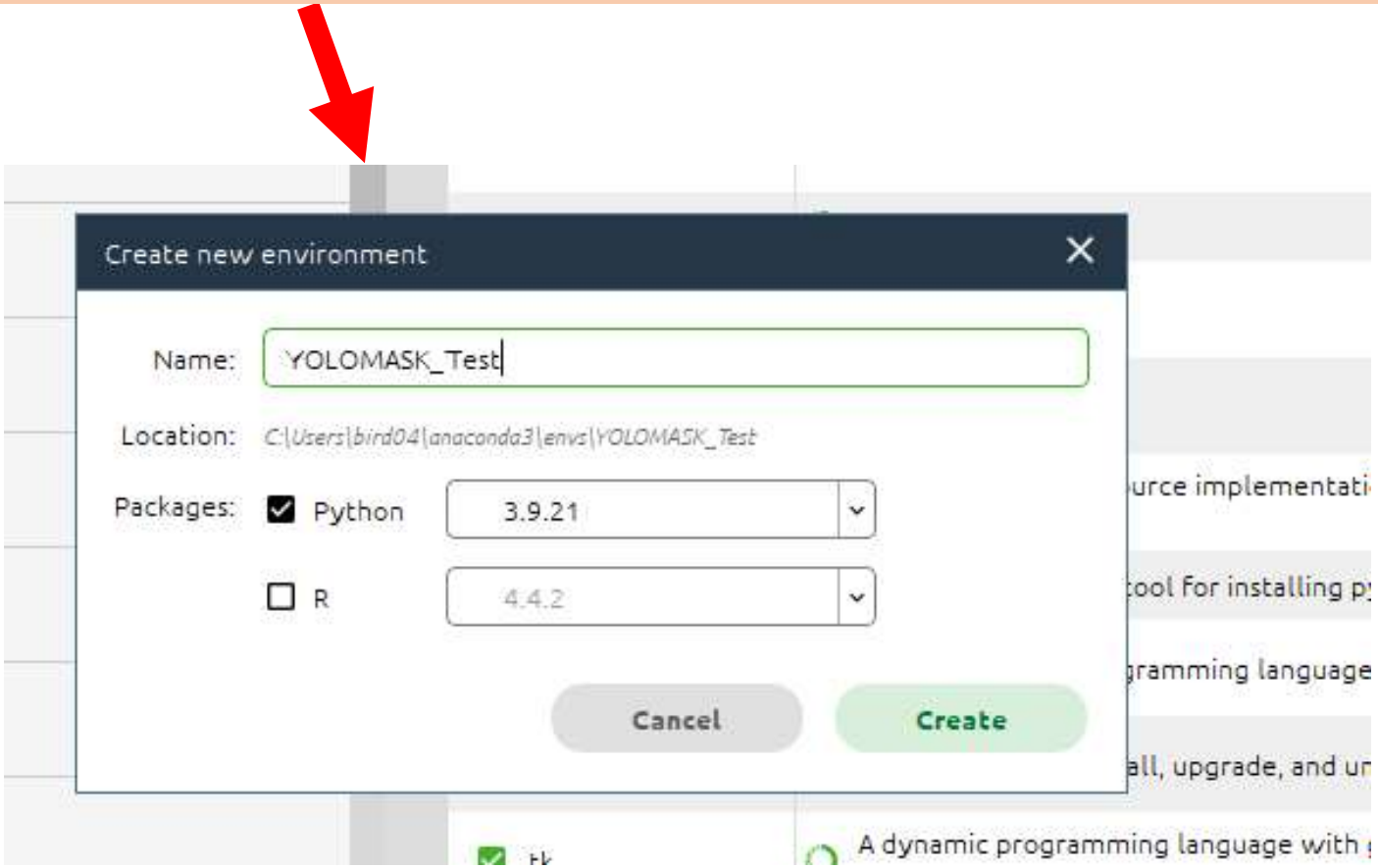
「NvidiaのGPU」，「anaconda」がインストールされて
いることが前提で，話を進めます．

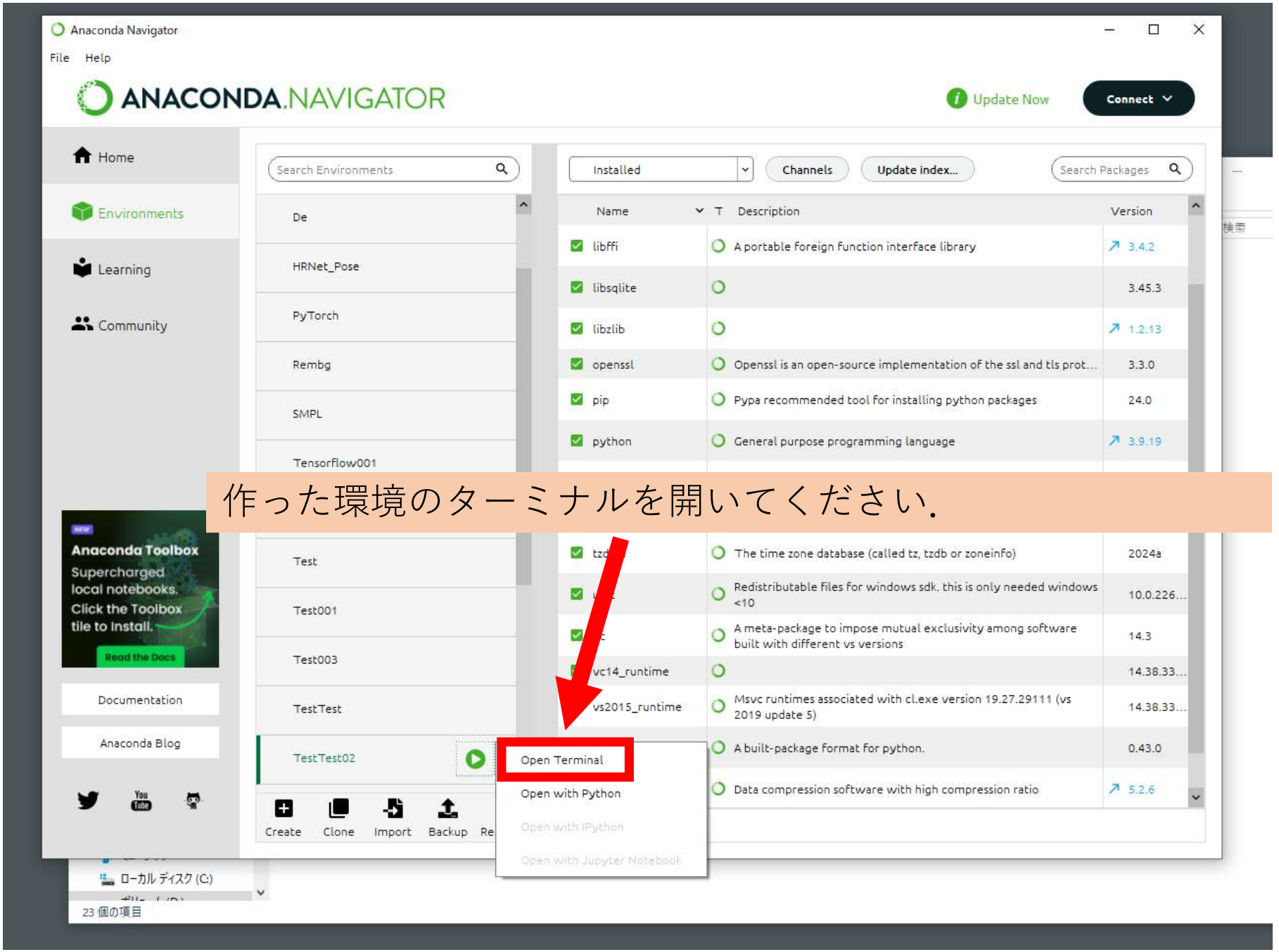
https://www.youtube.com/watch?v=tq0GI4FahWU&t=1709s

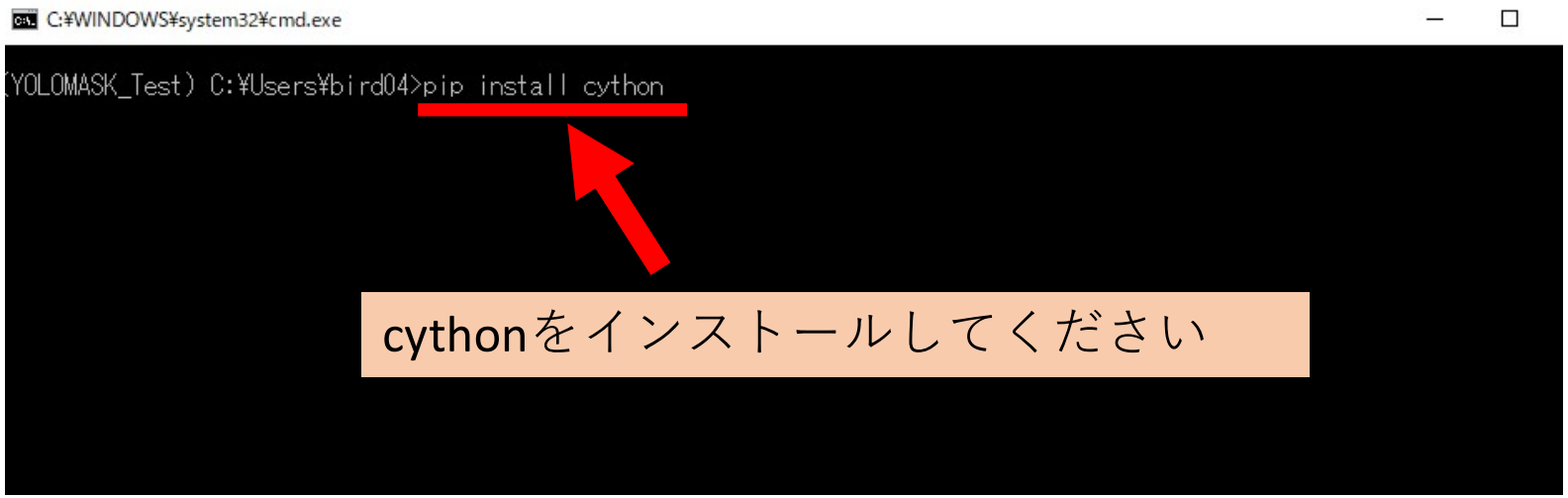
↑のチュートリアル動画を参考に説明していきますが，
動画の内容から一部修正しないとうまくいかない部分があります


Anaconda Navigator
File Help
ANACONDA.NAVIGATOR
Update Now
Connect
Home
Environments
Learning
Community
Search Environments
Tensorflow001
Tensorflow_pip
Test
Test001
Test003
TestTest
invpt
testmediapipe
yolov7
yolov7_maskVer2
yolov7_pose
Create
Clone
Import
Backup
Remove
Installed
Channels
Update index...
Search Packages
Name
T
Description
Version
ca-certificates
Certificates for use with other packages.
libffi
A portable foreign function interface library
3.4.2
libsqlite
3.45.3
libzlib
1.2.13
openssl
Openssl is an open-source implementation of the ssl and tls protocols
3.3.0
pip
Pypa recommended tool for installing python packages
24.0
python
General purpose programming language
3.9.19
setuptools
Download, build, install, upgrade, and uninstall python packages
69.5.1
tk
A dynamic programming language with gui support. bundles tcl and tk.
8.6.13
tzdata
The time zone database (called tz, tzdb or zoneinfo)
2024a
ucrt
Redistributable files for windows sdk. this is only needed windo...
10.0.226...
vc
A meta-package to impose mutual exclusivity among software built with different vs versions
14.3
vc14_runtime
14.38.33...
vs2015_runtime
Msvc runtimes associated with cl.exe version 19.27.29111 (vs 2019 update 5)
14.38.33...
wheel
A built-package format for python.
0.43.0
xz
Data compression software with high compression ratio
5.2.6
17 packages available
Anaconda Toolbox
Supercharged local notebooks.
Click the Toolbox tile to install.
Documentation
Anaconda Blog
仮想環境を作ってください

名前は何でもよいです（日本語でない方が良いと思います）.
Python3.9.21で問題なく動作しました




Anaconda Navigator
File Help
ANACONDA.NAVIGATOR
Update Now
Connect
Home
Environments
Learning
Community
Search Environments
Installed
Channels
Update index...
Search Packages
Name
T
Description
Version
libffi
A portable foreign function interface library
3.4.2
libsqlite
3.45.3
libzlib
1.2.13
openssl
Openssl is an open-source implementation of the ssl and tls prot...
3.3.0
pip
Pypa recommended tool for installing python packages
24.0
python
General purpose programming language
3.9.19
De
HRNet_Pose
PyTorch
Rembg
SMPL
Tensorflow001
Test
Test001
Test003
TestTest
TestTest02
作った環境のターミナルを開いてください.
Open Terminal
Open with Python
Open with IPython
Open with Jupyter Notebook
Create
Clone
Import
Backup
Anaconda Toolbox
Supercharged local notebooks. Click the Toolbox tile to install.
Read the Docs
Documentation
Anaconda Blog
The time zone database (called tz, tzdb or zoneinfo)
2024a
Redistributable files for windows sdk. this is only needed windows <10
10.0.226...
A meta-package to impose mutual exclusivity among software built with different vs versions
14.3
vc14_runtime
14.38.33...
vs2015_runtime
Msvc runtimes associated with cl.exe version 19.27.29111 (vs 2019 update 5)
14.38.33...
A built-package format for python.
0.43.0
Data compression software with high compression ratio
5.2.6
ローカル ディスク (C:)
23 個の項目


C:¥WINDOWS¥system32¥cmd.exe
(YOLOMASK_Test) C:¥Users¥bird04>pip install cython
cythonをインストールしてください

```
pip install torch==1.11.0 torchvision==0.12.0 torchaudio==0.11.0
```

Linux and Windows

```
# ROCM 4.5.2 (Linux only)
pip install torch==1.11.0+rocm4.5.2 torchvision==0.12.0+rocm4.5.2 torchaudio==0.11.0 --extra-index-url  htt

# CUDA 11.3
pip install torch==1.11.0+cu113 torchvision==0.12.0+cu113 torchaudio==0.11.0 --extra-index-url https://down

# CUDA 10.2
pip install torch==1.11.0+cu102 torchvision==0.12.0+cu102 torchaudio==0.11.0 --extra-inde  https://down

# CPU only
pip install torch==1.11.0+cpu torchvision==0.12.0+cpu torchaudio==0.11.0 --extra-index-url  ttps://download
```

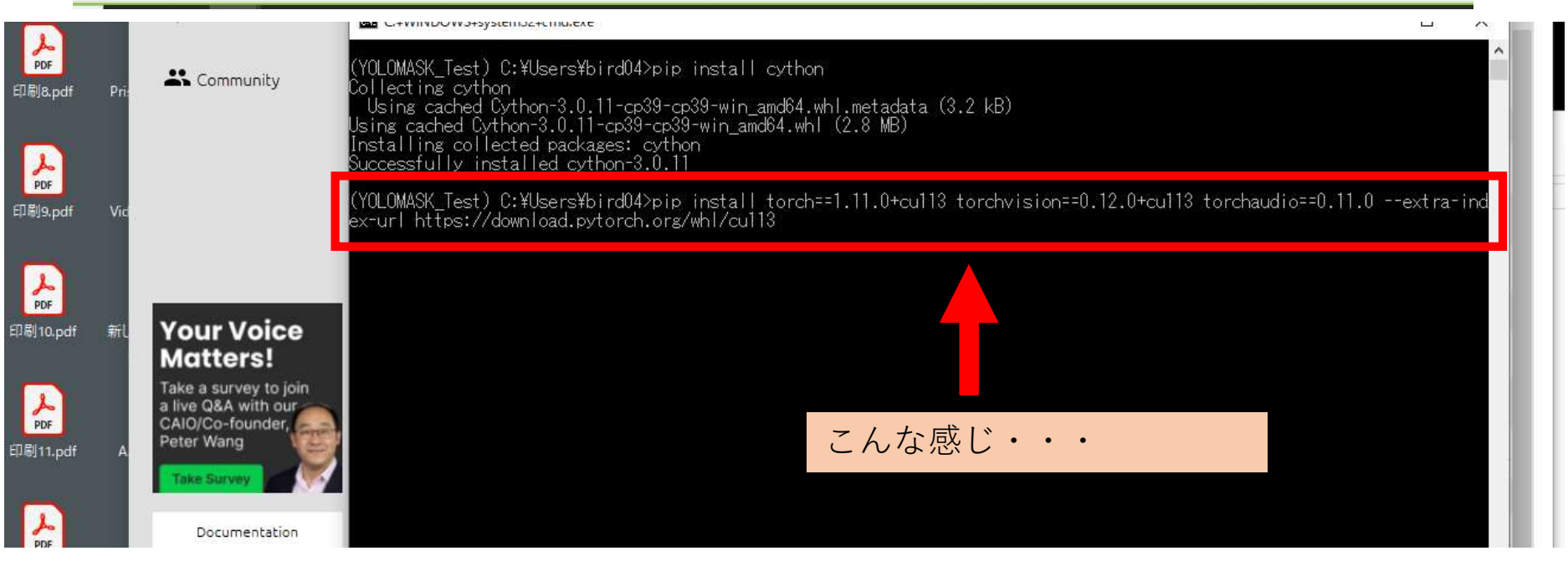
v1.10.1

Conda

OSX

```
# conda
conda install pytorch==1.10.1 torchvision==0.11.2 torchaudio==0.10.1 -c pytorch
```

Pytorchのサイトに行ってこれをインストールしてください.
このテキストをコピーして，ターミナルに張り付けてエンターキーを押すとインストールできます.



C:¥Users¥bird04¥anaconda3¥envs¥YOLOMASK_Test¥Lib¥site-packages¥torch¥nn¥modules

PCのユーザー名

Anacondaで作った
仮想環境の名前

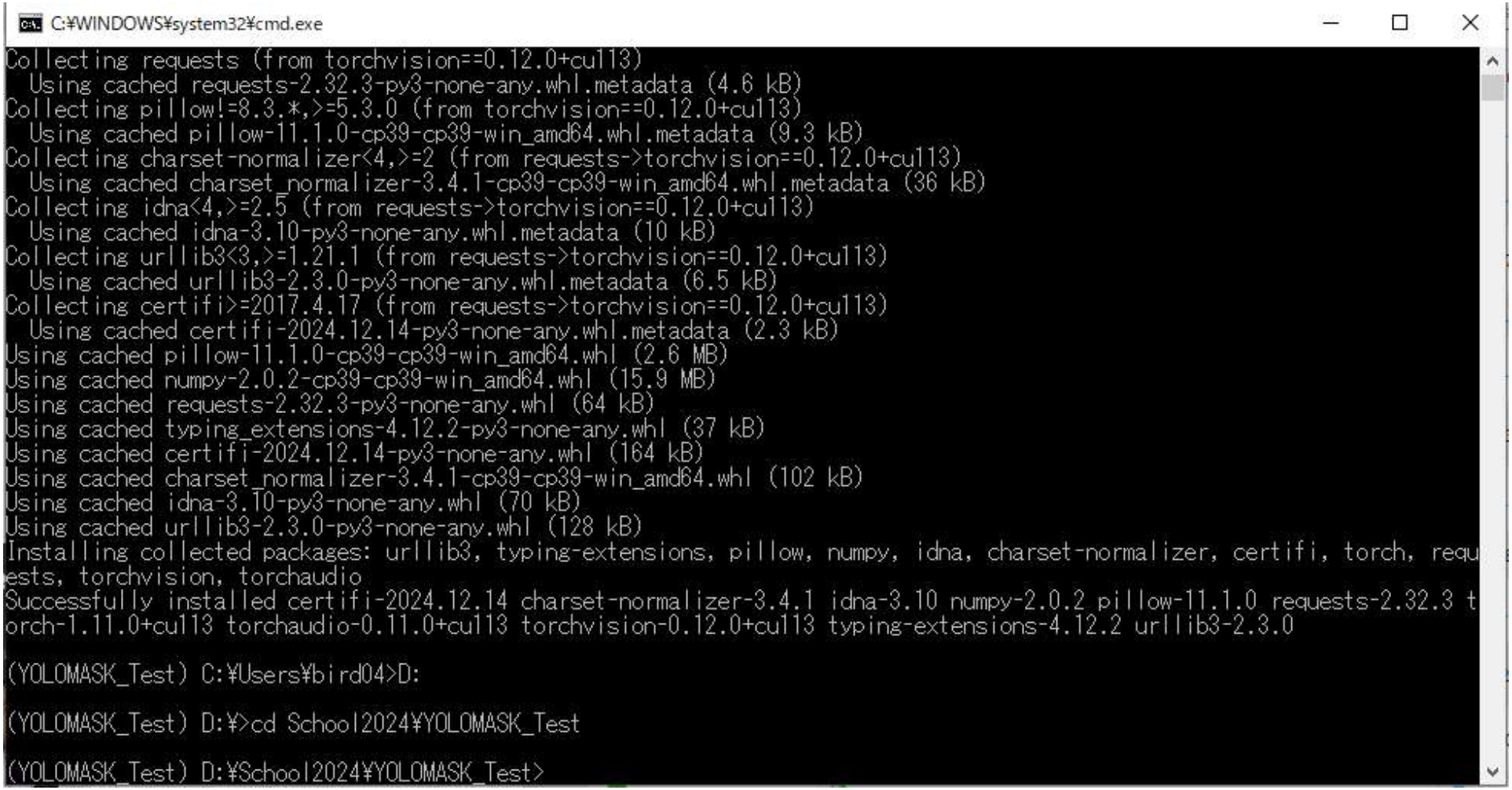
このフォルダ内になるupsampling.pyを開いてください．①から②に変更してください．

①

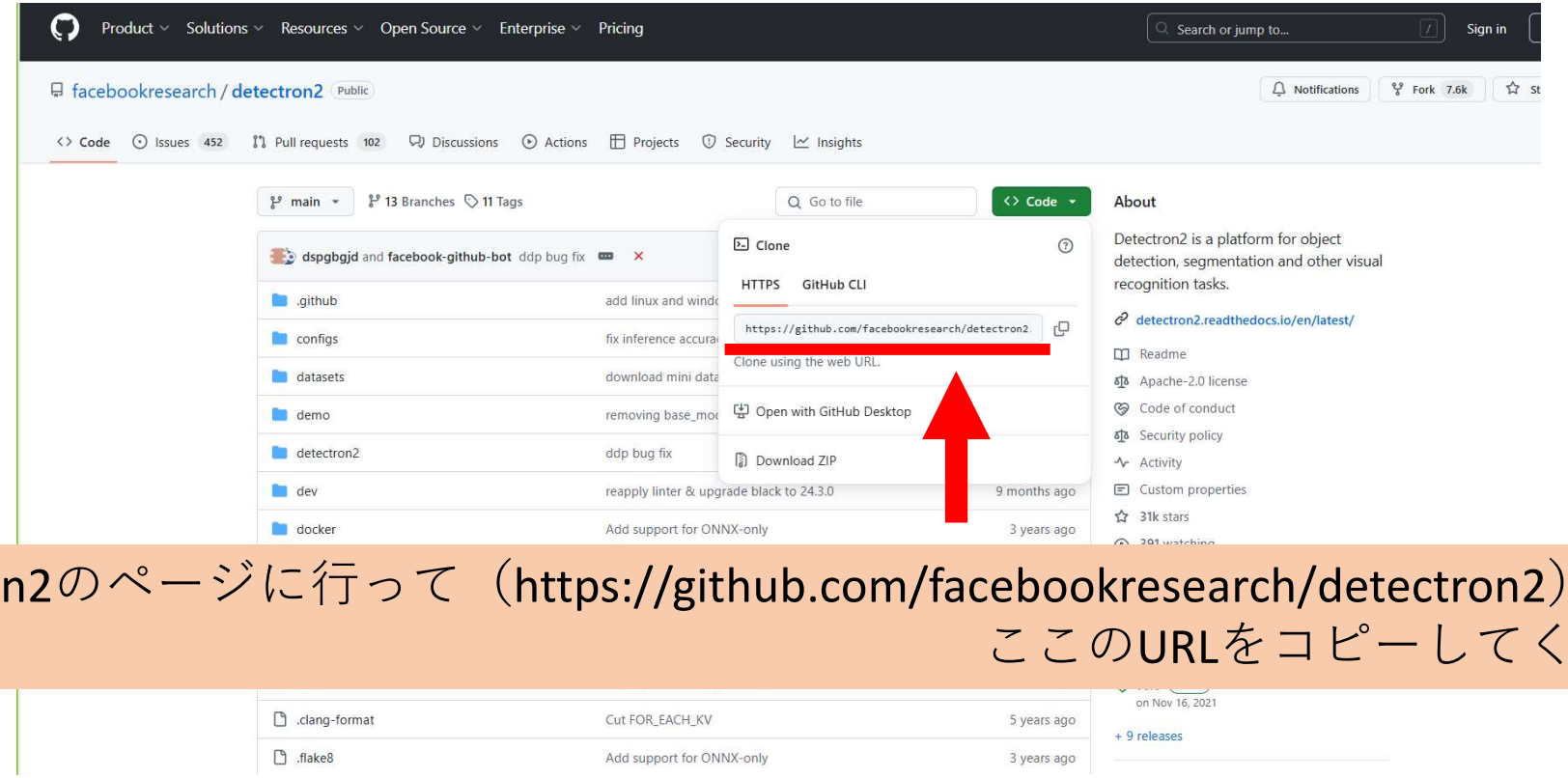
```
149         self.align_corners = align_corners
150         self.recompute_scale_factor = recompute_scale_factor
151
152     def forward(self, input: Tensor) -> Tensor:
153         return F.interpolate(input, self.size, self.scale_factor, self.mode, self.align_corners,
154                              recompute_scale_factor=self.recompute_scale_factor)
155
```

②

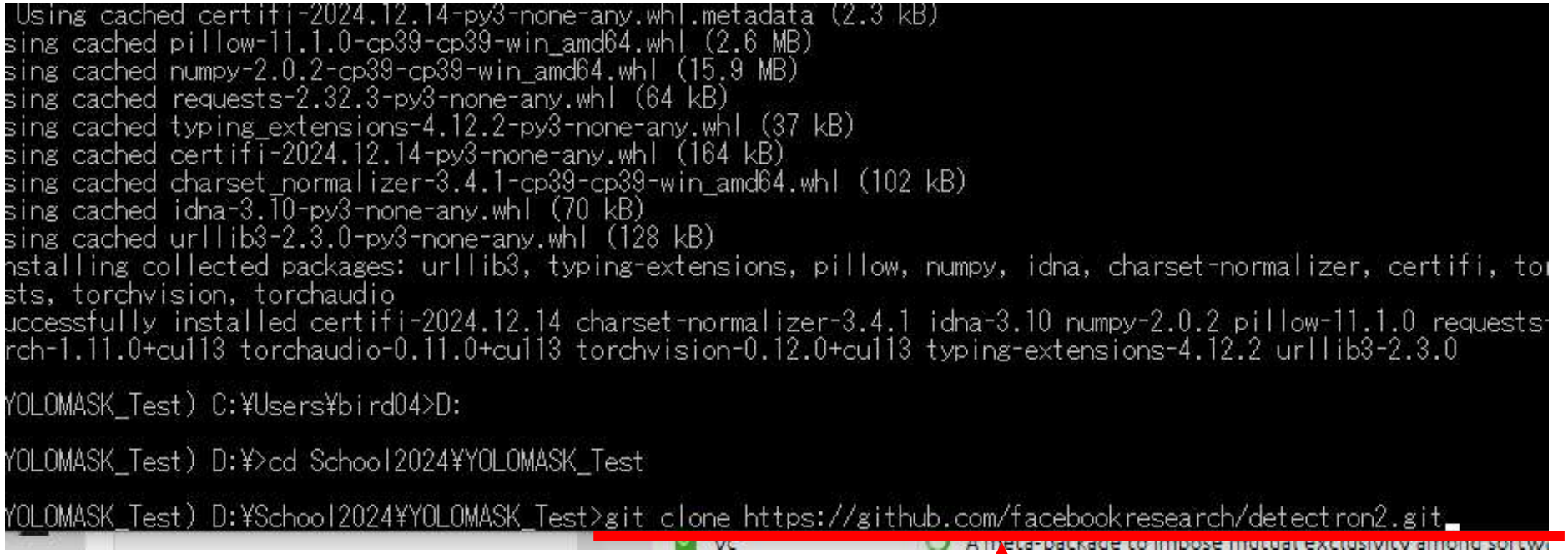
```
149         self.align_corners = align_corners
150         self.recompute_scale_factor = recompute_scale_factor
151
152     def forward(self, input: Tensor) -> Tensor:
153         return F.interpolate(input, self.size, self.scale_factor, self.mode, self.align_corners)
154
155     def extra_repr(self) -> str:
156         if self.scale_factor is not None:
```



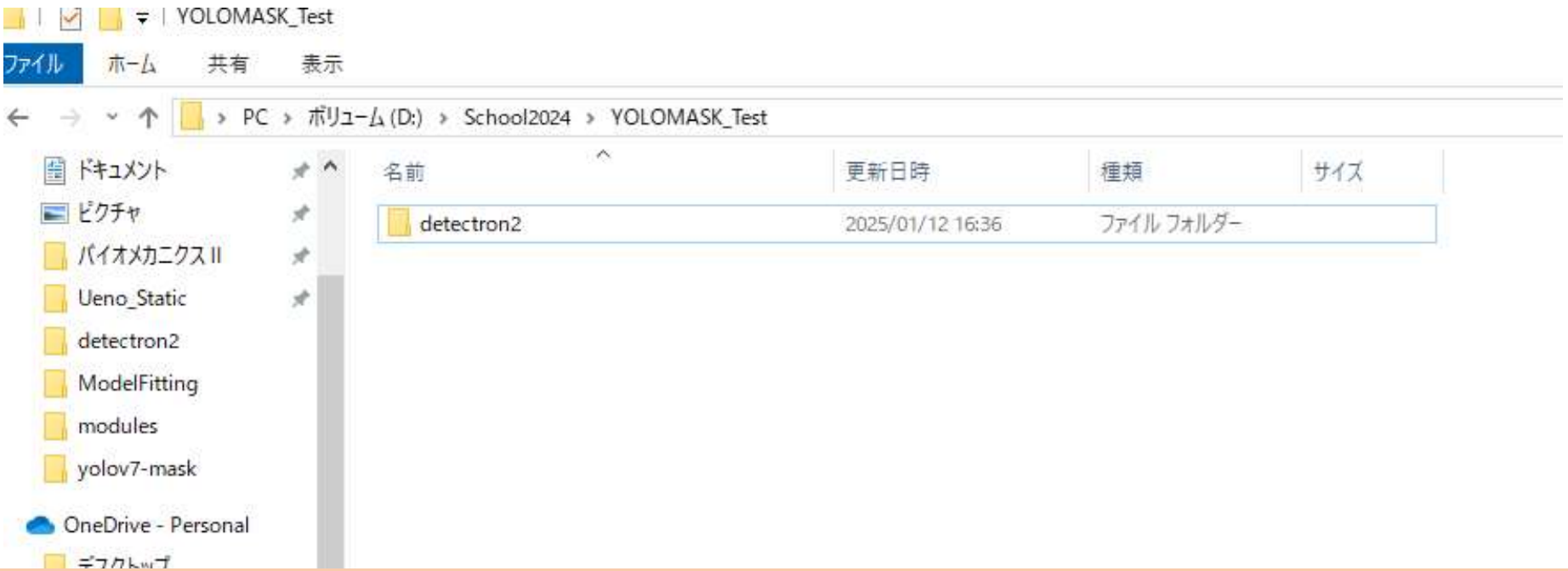
Yolo-Maskに関するプログラムを保存するフォルダを,
カレントディレクトリにしてください



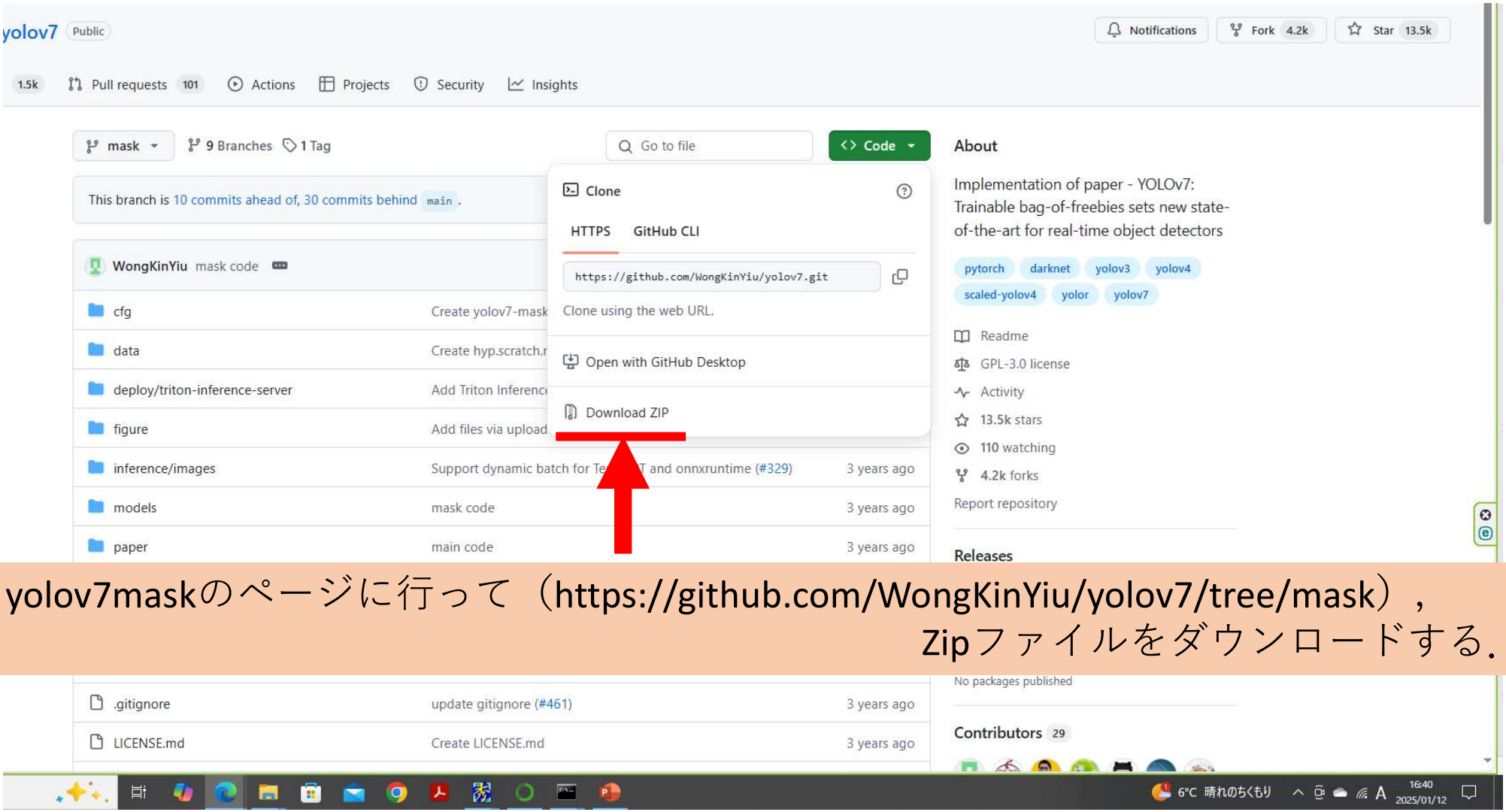
detectron2のページに行って（https://github.com/facebookresearch/detectron2），
ここのURLをコピーしてください．



git clone と打ち込んでから，コピーしたURLを張り付けて実行してください．

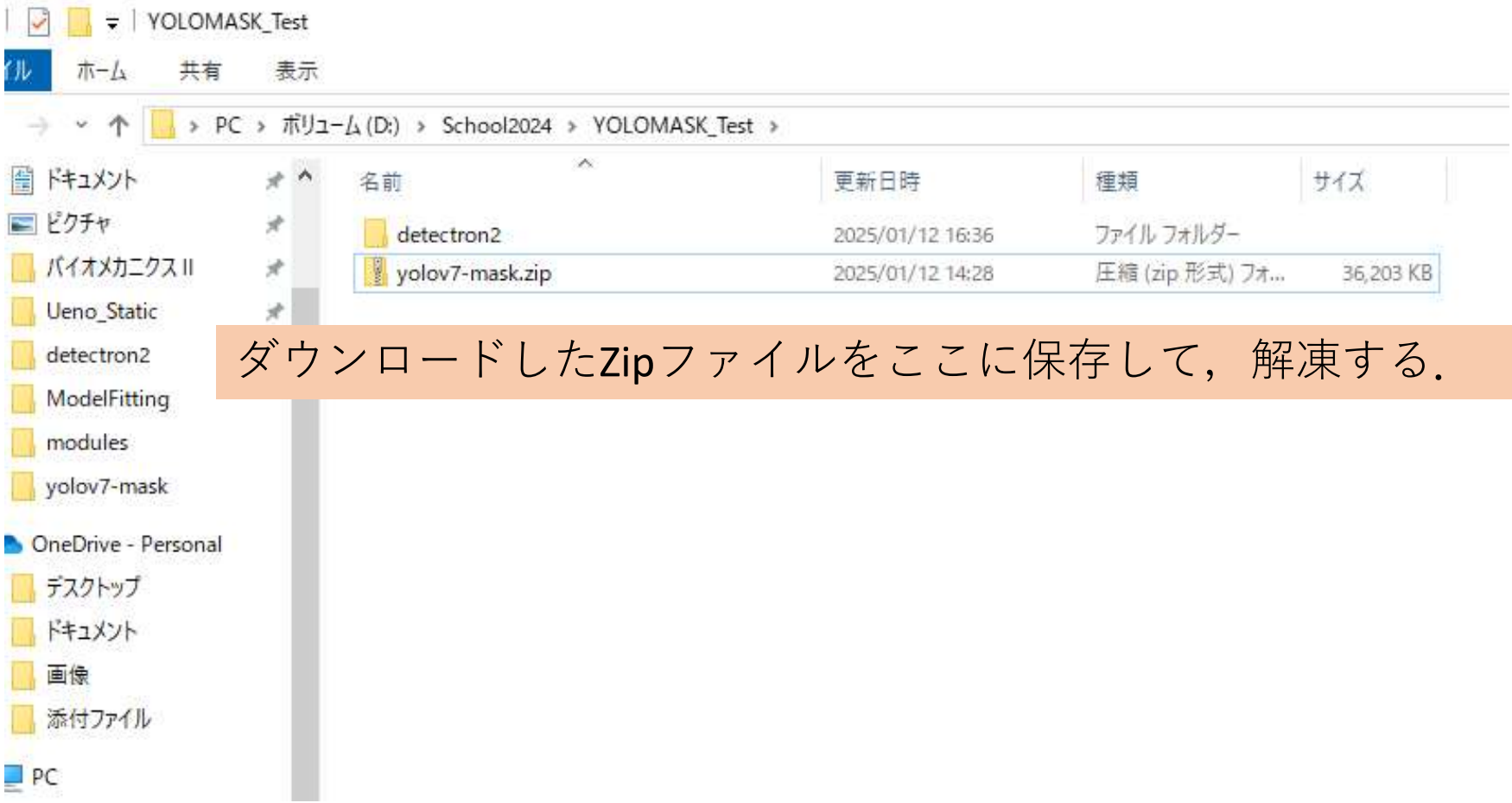


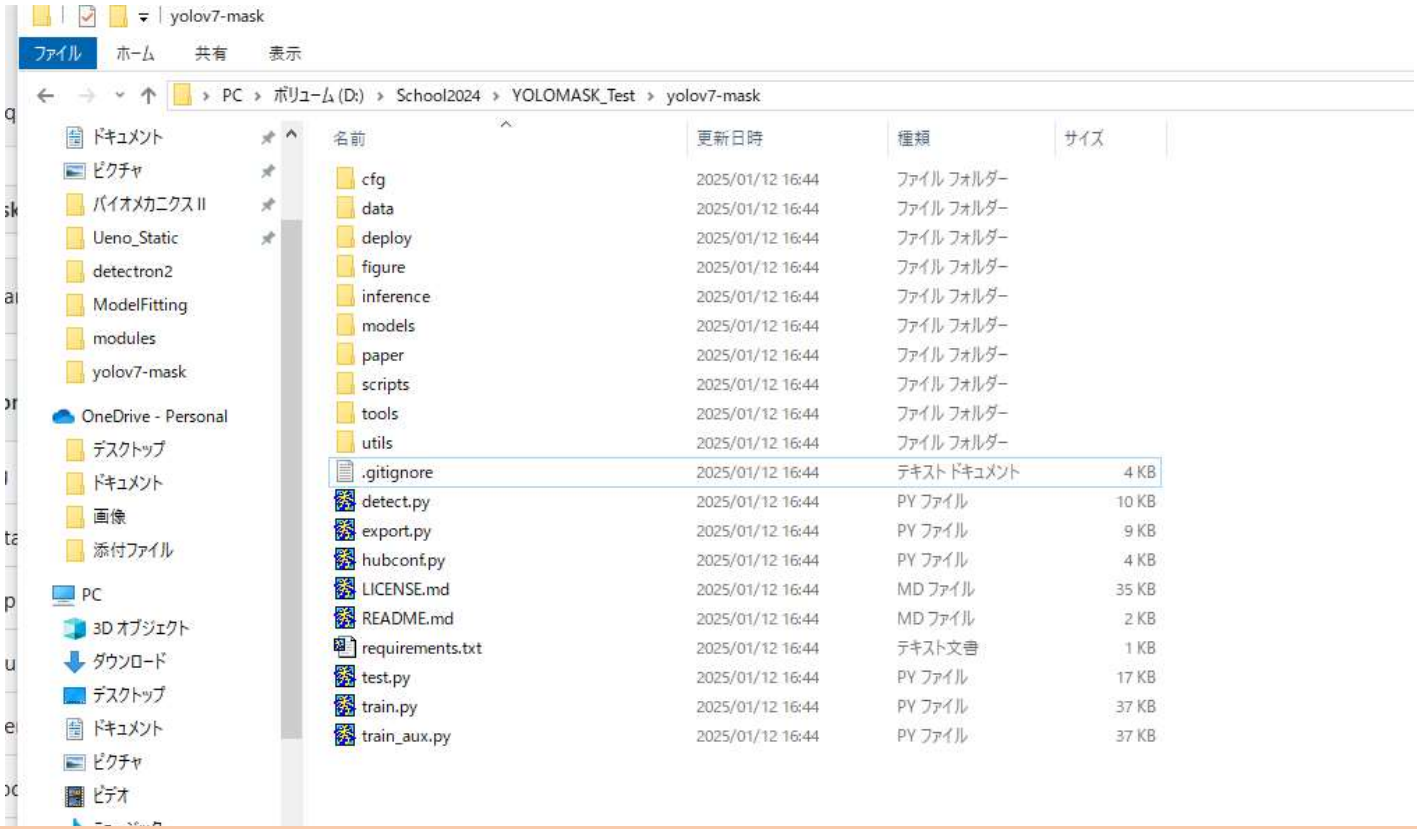
カレントディレクトリに「detectron2」ができている.



yolov7maskのページに行って（https://github.com/WongKinYiu/yolov7/tree/mask），
Zipファイルをダウンロードする.

このページの下の方にyolov7-mask.ptがあるので,
クリックしてモデルファイルをダウンロードしてください.


YOLOMASK_Test
ホーム
共有
表示
PC › ボリューム (D:) › School2024 › YOLOMASK_Test ›
ドキュメント
ピクチャ
バイオメカニクス II
Ueno_Static
detectron2
ModelFitting
modules
yolov7-mask
OneDrive - Personal
デスクトップ
ドキュメント
画像
添付ファイル
PC
名前
更新日時
種類
サイズ
detectron2
2025/01/12 16:36
ファイル フォルダー
yolov7-mask.zip
2025/01/12 14:28
圧縮 (zip 形式) フォ...
36,203 KB
ダウンロードしたZipファイルをここに保存して，解凍する．



解凍したら，yolov7-maskのプログラムの階層を1段上げてください.

cd/yolov7-mask/yolov7-mask/
から
cd/yolov7-mask/
にしてくださいということです.

また，yolov7-maskの中に，yolov7-mask.ptを入れてください.

cd/detectron2/setup.pyを開いてください.

①から②にコードを書き加えてください.

注：これをしないとエラーが出ます．とても重要です.

①

```
    if is_rocm_pytorch:
        assert torch_ver >= [1, 8], "ROCM support requires PyTorch >= 1

    # common code between cuda and rocm platforms, for hipify version [
    source_cuda = glob.glob(path.join(extensions_dir, "**", "*.cu")) +
        path.join(extensions_dir, "*.cu")
    )
    sources = [main_source] + sources

    extension = CppExtension

    extra_compile_args = {"cxx": []}
    define_macros = []

    if (torch.cuda.is_available() and ((CUDA_HOME is not None) or is_ro
        "FORCE_CUDA", "0"
    ) == "1":
        extension = CUDAExtension
        sources += source_cuda

        if not is_rocm_pytorch:
            define_macros += [("WITH_CUDA", None)]
            extra_compile_args["nvcc"] = [
                "-O3",
                "-DCUDA_HAS_FP16=1",
                "-D__CUDA_NO_HALF_OPERATORS__",
                "-D__CUDA_NO_HALF_CONVERSIONS__",
                "-D__CUDA_NO_HALF2_OPERATORS__",
            ]
        else:
            define_macros += [("WITH_HIP", None)]
            extra_compile_args["nvcc"] = []

        nvcc_flags_env = os.getenv("NVCC_FLAGS", "")
        if nvcc_flags_env != "":
            extra_compile_args["nvcc"].extend(nvcc_flags_env.split(" "))
```

②

```
    if is_rocm_pytorch:
        assert torch_ver >= [1, 8], "ROCM support requires PyTorch >= 1

    # common code between cuda and rocm platforms, for hipify version [
    source_cuda = glob.glob(path.join(extensions_dir, "**", "*.cu")) +
        path.join(extensions_dir, "*.cu")
    )
    sources = [main_source] + sources

    extension = CppExtension

    extra_compile_args = {"cxx": []}
    define_macros = []

    if (torch.cuda.is_available() and ((CUDA_HOME is not None) or is_ro
        "FORCE_CUDA", "0"
    ) == "1":
        extension = CUDAExtension
        sources += source_cuda

        if not is_rocm_pytorch:
            define_macros += [("WITH_CUDA", None)]
            extra_compile_args["nvcc"] = [
                "-O3",
                "-DCUDA_HAS_FP16=1",
                "-D__CUDA_NO_HALF_OPERATORS__",
                "-D__CUDA_NO_HALF_CONVERSIONS__",
                "-D__CUDA_NO_HALF2_OPERATORS__",
                "-DWITH_CUDA",
            ]
        else:
            define_macros += [("WITH_HIP", None)]
            extra_compile_args["nvcc"] = []

        nvcc_flags_env = os.getenv("NVCC_FLAGS", "")
        if nvcc_flags_env != "":
            extra_compile_args["nvcc"].extend(nvcc_flags_env.split(" "))
```

cd/detectron2/setup.pyのコードを変更したら,
カレントディレクトリをdetectron2にしてください.

```
Using cached numpy-2.0.2-cp39-cp39-win_amd64.whl (15.9 MB)
Using cached requests-2.32.3-py3-none-any.whl (64 kB)
Using cached typing_extensions-4.12.2-py3-none-any.whl (37 kB)
Using cached certifi-2024.12.14-py3-none-any.whl (164 kB)
Using cached charset_normalizer-3.4.1-cp39-cp39-win_amd64.whl (102 kB)
Using cached idna-3.10-py3-none-any.whl (70 kB)
Using cached urllib3-2.3.0-py3-none-any.whl (128 kB)
Installing collected packages: urllib3, typing-extensions, pillow, numpy, idna, charset-normalizer, certifi, to
ests, torchvision, torchaudio
Successfully installed certifi-2024.12.14 charset-normalizer-3.4.1 idna-3.10 numpy-2.0.2 pillow-11.1.0 requests
orch-1.11.0+cu113 torchaudio-0.11.0+cu113 torchvision-0.12.0+cu113 typing-extensions-4.12.2 urllib3-2.3.0

(YOLOMASK_Test) C:¥Users¥bird04>D:

(YOLOMASK_Test) D:¥>cd School2024¥YOLOMASK_Test

(YOLOMASK_Test) D:¥School2024¥YOLOMASK_Test>git clone https://github.com/facebookresearch/detectron2.git
Cloning into 'detectron2'...
remote: Enumerating objects: 15832, done.
remote: Counting objects: 100% (62/62), done.
remote: Compressing objects: 100% (50/50), done.
remote: Total 15832 (delta 27), reused 12 (delta 12), pack-reused 15770 (from 2)
Receiving objects: 100% (15832/15832), 6.40 MiB | 3.76 MiB/s, done.
Resolving deltas: 100% (11529/11529), done.

(YOLOMASK_Test) D:¥School2024¥YOLOMASK_Test>cd detectron2

(YOLOMASK_Test) D:¥School2024¥YOLOMASK_Test¥detectron2>
(YOLOMASK_Test) D:¥School2024¥YOLOMASK_Test¥detectron2>
(YOLOMASK_Test) D:¥School2024¥YOLOMASK_Test¥detectron2>
```

pip install –e .
と打ち込んで実行してください.

```
Using cached charset_normalizer-3.4.1-cp39-cp39-win_amd64.whl (102 kB)
Using cached idna-3.10-py3-none-any.whl (70 kB)
Using cached urllib3-2.3.0-py3-none-any.whl (128 kB)
Installing collected packages: urllib3, typing-extensions, pillow, numpy, idna, charset-normalizer, certifi, t
ests, torchvision, torchaudio
Successfully installed certifi-2024.12.14 charset-normalizer-3.4.1 idna-3.10 numpy-2.0.2 pillow-11.1.0 request
orch-1.11.0+cu113 torchaudio-0.11.0+cu113 torchvision-0.12.0+cu113 typing-extensions-4.12.2 urllib3-2.3.0

(YOLOMASK_Test) C:¥Users¥bird04>D:

(YOLOMASK_Test) D:¥>cd School2024¥YOLOMASK_Test

(YOLOMASK_Test) D:¥School2024¥YOLOMASK_Test>git clone https://github.com/facebookresearch/detectron2.git
Cloning into 'detectron2'...
remote: Enumerating objects: 15832, done.
remote: Counting objects: 100% (62/62), done.
remote: Compressing objects: 100% (50/50), done.
remote: Total 15832 (delta 27), reused 12 (delta 12), pack-reused 15770 (from 2)
Receiving objects: 100% (15832/15832), 6.40 MiB | 3.76 MiB/s, done.
Resolving deltas: 100% (11529/11529), done.

(YOLOMASK_Test) D:¥School2024¥YOLOMASK_Test>cd detectron2

(YOLOMASK_Test) D:¥School2024¥YOLOMASK_Test¥detectron2>
(YOLOMASK_Test) D:¥School2024¥YOLOMASK_Test¥detectron2>
(YOLOMASK_Test) D:¥School2024¥YOLOMASK_Test¥detectron2>pip install -e .
```

数分かかりますが，下のようになったら成功です．

カレントディレクトリをyolov7-maskにしてください.

cd/yolov7-mask/requirements.txtを開いてください．①から②へコードを変更してください．

①

```
# Usage: pip install -r requirements.txt

# Base ----------
matplotlib>=3.2.2
numpy>=1.18.5
opencv-python>=4.1.1
Pillow>=7.1.2
PyYAML>=5.3.1
requests>=2.23.0
scipy>=1.4.1
torch>=1.7.0,!=1.12.0
torchvision>=0.8.1,!=0.13.0
tqdm>=4.41.0
protobuf<4.21.3

# Logging ----------
tensorboard>=2.4.1
# wandb

# Plotting ----------
pandas>=1.1.4
seaborn>=0.11.0

# Export ----------
# coremltools>=4.1  # CoreML export
# onnx>=1.9.0  # ONNX export
# onnx-simplifier>=0.3.6  # ONNX simplifier
# scikit-learn==0.19.2  # CoreML quantization
# tensorflow>=2.4.1  # TFLite export
# tensorflowjs>=3.9.0  # TF.js export
# openvino-dev  # OpenVINO export
```

これを変更

これを消す

②

```
# Usage: pip install -r requirements.txt

# Base ----------
matplotlib>=3.2.2
numpy<2
opencv-python>=4.1.1
Pillow>=7.1.2
PyYAML>=5.3.1
requests>=2.23.0
scipy>=1.4.1
tqdm>=4.41.0
protobuf<4.21.3

# Logging ----------
tensorboard>=2.4.1
# wandb

# Plotting ----------
pandas>=1.1.4
seaborn>=0.11.0

# Export ----------
# coremltools>=4.1  # CoreML export
# onnx>=1.9.0  # ONNX export
# onnx-simplifier>=0.3.6  # ONNX simplifier
# scikit-learn==0.19.2  # CoreML quantization
# tensorflow>=2.4.1  # TFLite export
# tensorflowjs>=3.9.0  # TF.js export
# openvino-dev  # OpenVINO export

# Extras ----------
ipython  # interactive notebook
```

変更したら，pip install –r requirements.txt
をして，必要なものをインストールする．

```
Using cached portalocker-3.1.1-py3-none-any.whl (19 kB)
Using cached importlib_metadata-8.5.0-py3-none-any.whl (26 kB)
Using cached MarkupSafe-3.0.2-cp39-cp39-win_amd64.whl (15 kB)
Using cached pywin32-308-cp39-cp39-win_amd64.whl (6.6 MB)
Using cached zipp-3.21.0-py3-none-any.whl (9.6 kB)
Installing collected packages: pywin32, antlr4-python3-runtime, zipp, tomli, termcolor, tensorboard-data-server, tabulat
e, six, pyyaml, pyparsing, protobuf, portalocker, platformdirs, pathspec, packaging, mypy-extensions, MarkupSafe, kiwiso
lver, grpcio, fonttools, cycler, contourpy, colorama, cloudpickle, absl-py, yacs, werkzeug, tqdm, python-dateutil, omega
conf, importlib-resources, importlib-metadata, click, matplotlib, markdown, iopath, hydra-core, black, tensorboard, pyco
cotools, fvcore, detectron2
  DEPRECATION: Legacy editable install of detectron2==0.6 from file:///D:/School2024/YOLOMASK_Test/detectron2 (setup.py
develop) is deprecated. pip 25.0 will enforce this behaviour change. A possible replacement is to add a pyproject.toml o
r enable --use-pep517, and use setuptools >= 64. If the resulting installation is not behaving as expected, try using --
config-settings editable_mode=compat. Please consult the setuptools documentation for more information. Discussion can b
e found at https://github.com/pypa/pip/issues/11457
  Running setup.py develop for detectron2
Successfully installed MarkupSafe-3.0.2 absl-py-2.1.0 antlr4-python3-runtime-4.9.3 black-24.10.0 click-8.1.8 cloudpickle
-3.1.0 colorama-0.4.6 contourpy-1.3.0 cycler-0.12.1 detectron2 fonttools-4.55.3 fvcore-0.1.5.post20221221 grpcio-1.69.0
hydra-core-1.3.2 importlib-metadata-8.5.0 importlib-resources-6.5.2 iopath-0.1.9 kiwisolver-1.4.7 markdown-3.7 matplotli
b-3.9.4 mypy-extensions-1.0.0 omegaconf-2.3.0 packaging-24.2 pathspec-0.12.1 platformdirs-4.3.6 portalocker-3.1.1 protob
uf-5.29.3 pycocotools-2.0.8 pyparsing-3.2.1 python-dateutil-2.9.0.post0 pywin32-308 pyyaml-6.0.2 six-1.17.0 tabulate-0.9
.0 tensorboard-2.18.0 tensorboard-data-server-0.7.2 termcolor-2.5.0 tomli-2.2.1 tqdm-4.67.1 werkzeug-3.1.3 yacs-0.1.8 zi
pp-3.21.0

(YOLOMASK_Test) D:¥School2024¥YOLOMASK_Test¥detectron2>cd..

(YOLOMASK_Test) D:¥School2024¥YOLOMASK_Test>cd yolov7-mask

(YOLOMASK_Test) D:¥School2024¥YOLOMASK_Test¥yolov7-mask>pip install -r requirements_
```

.txt
をつけてください．

yolov7-maskのフォルダの中に「1.jpg」，「segment.py」
を入れてください.

「python segment.py --source 1.jpg」と打ち込んで，実行してください.

```
>>> import detectron2
>>> import torch
>>> torch.cuda.is_available()
True
>>> exit()

(YOLOMASK_Test) D:¥School2024¥YOLOMASK_Test¥yolov7-mask>python segment.py --source 1.jpg
Namespace(weights='yolov7-mask.pt', source='1.jpg', img_size=640, conf_thres=0.25, iou_thres=0.45,
e=False, project='runs/detect', name='exp', hyp='data/hyp.scratch.mask.yaml', seed=1, thickness=1,
=False, showfps=False)
C:¥Users¥bird04¥anaconda3¥envs¥YOLOMASK_Test¥lib¥site-packages¥torch¥functional.py:568: UserWarning
an upcoming release, it will be required to pass the indexing argument. (Triggered internally at
k¥pytorch¥pytorch¥builder¥windows¥pytorch¥aten¥src¥ATen¥native¥TensorShape.cpp:2228.)
  return _VF.meshgrid(tensors, **kwargs)  # type: ignore[attr-defined]
Output saved:  runs¥detect¥exp¥1.jpg

(YOLOMASK_Test) D:¥School2024¥YOLOMASK_Test¥yolov7-mask>python segment.py --source 1.jpg
```

実行後，「runs」フォルダの奥に，画像ができていれば成功です．

こんな画像．．．

ここまで，できたならば，yolo-maskの仮想環境が完成しました.

yolov7-maskのフォルダの中に「Segment-human.py」を保存してください.