ZeroRotate

erenzhou

Nov 16, 2021

GET STARTED

ZeroRotate is an open-source PyTorch benchmark for performing scalable rotation detection on various datasets, which is maintained by Yue Zhou and Xue Yang with Shanghai Jiao Tong University supervised by Prof. Jiang Xue.

CHAPTER

ONE

PREREQUISITES

- Linux or macOS (Windows is in experimental support)
- Python 3.6+
- PyTorch 1.3+
- CUDA 9.2+ (If you build PyTorch from source, CUDA 9.0 is also compatible)
- GCC 5+
- opencv-python 4.5.1+
- MMCV 1.3.14+
- MMDetection 2.18+

CHAPTER

TWO

INSTALLATION

2.1 Prepare environment

1. Create a conda virtual environment and activate it.

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

2. Install PyTorch and torchvision following the official instructions, e.g.,

```
conda install pytorch torchvision -c pytorch
```

Note: Make sure that your compilation CUDA version and runtime CUDA version match. You can check the supported CUDA version for precompiled packages on the PyTorch website.

`E.g.1` If you have CUDA 10.1 installed under `/usr/local/cuda` and would like to install PyTorch 1.5, you need to install the prebuilt PyTorch with CUDA 10.1.

```
conda install pytorch cudatoolkit=10.1 torchvision -c pytorch
```

`E.g. 2` If you have CUDA 9.2 installed under `/usr/local/cuda` and would like to install PyTorch 1.3.1., you need to install the prebuilt PyTorch with CUDA 9.2.

```
conda install pytorch=1.3.1 cudatoolkit=9.2 torchvision=0.4.2 -c pytorch
```

If you build PyTorch from source instead of installing the prebuilt package, you can use more CUDA versions such as 9.0.

2.2 Install MMDetection

It is recommended to install MMDetection with MIM, which automatically handle the dependencies of OpenMMLab projects, including mmcv and other python packages.

```
pip install openmim
mim install mmdet
```

Or you can still install MMDetection manually:

1. Install mmcv-full.

```
pip install mmcv-full -f https://download.openmmlab.com/mmcv/dist/{cu_version}/
↪{torch_version}/index.html
```

(continues on next page)

(continued from previous page)

Please replace `{cu_version}` and `{torch_version}` in the url to your desired one. For example, to install the latest `mmcv-full` with `CUDA 11.0` and `PyTorch 1.7.0`, use the following command:

```
pip install mmcv-full -f https://download.openmmlab.com/mmcv/dist/cu110/torch1.7.0/
↪index.html
```

See here for different versions of MMCV compatible to different PyTorch and CUDA versions.

Optionally you can compile mmcv from source if you need to develop both mmcv and mmdet. Refer to the guide for details.

2. Install MMDetection.

   You can simply install mmdetection with the following command:

```
pip install mmdet
```

   or clone the repository and then install it:

```
git clone https://github.com/open-mmlab/mmdetection.git
cd mmdetection
pip install -r requirements/build.txt
pip install -v -e .  # or "python setup.py develop"
```

Note:

a. When specifying `-e` or `develop`, MMDetection is installed on dev mode , any local modifications made to the code will take effect without reinstallation.

b. If you would like to use `opencv-python-headless` instead of `opencv-python`, you can install it before installing MMCV.

c. It is best to use `opencv-python` greater than `4.5.1` because its angle representation has been changed in `4.5.1`. The following experiments are all run with `4.5.3`.

2.3 Install ZeroRotate

You can simply install ZeroRotate with the following command:

```
pip install rmmdet
```

or clone the repository and then install it:

```
git clone https://github.com/zytx121/mmrotate.git
cd mmrotate
pip install -r requirements.txt
pip install -v -e .
```

2.4 A from-scratch setup script

Assuming that you already have CUDA 10.1 installed, here is a full script for setting up MMDetection with conda.

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

conda install pytorch==1.6.0 torchvision==0.7.0 cudatoolkit=10.1 -c pytorch -y

# install the latest mmcv
pip install mmcv-full -f https://download.openmmlab.com/mmcv/dist/cu101/torch1.6.0/index.
↪html

# install the latest mmdetection
pip install mmdet==2.18.0

# install ZeroRotate
git clone https://github.com/zytx121/mmrotate.git
cd mmrotate
pip install -r requirements.txt
pip install -v -e .
```

CHAPTER

THREE

BENCHMARK AND MODEL ZOO

3.1 Baselines

3.1.1 RetinaNet OBB/HBB

Please refer to RetinaNet-OBB/HBB for details.

3.1.2 Faster R-CNN OBB

Please refer to Faster R-CNN OBB for details.

3.1.3 RoI Transformer

Please refer to RoI Transformer for details.

3.1.4 R3Det

Please refer to R3Det for details.

3.1.5 Gliding Vertex

Please refer to Gliding Vertex for details.

3.1.6 S2A-Net

Please refer to S2A-Net for details.

3.1.7 ReDet

Please refer to ReDet for details.

3.1.8 Oriented R-CNN

Please refer to Oriented R-CNN for details.

3.1.9 GWD

Please refer to GWD for details.

3.1.10 KFIoU

Please refer to KFIoU for details.

3.1.11 Deformable DETR OBB

3.1.12 KLD

CHAPTER

FOUR

TRAIN

```
# train
CUDA_VISIBLE_DEVICES=0 PORT=29500 \
./tools/dist_train.sh configs/rretinanet/rretinanet_obb_r50_fpn_1x_dota_v3.py 1
```

CHAPTER

FIVE

TEST

```
CUDA_VISIBLE_DEVICES=0 PORT=29500 \
./tools/dist_test.sh configs/rretinanet/rretinanet_obb_r50_fpn_1x_dota_v3.py \
        work_dirs/rretinanet_obb_r50_fpn_1x_dota_v3/epoch_12.pth 1 --mAP
```

CHAPTER

SIX

INFERENCE & SUBMIT

```
CUDA_VISIBLE_DEVICES=0 PORT=29500 \
./tools/dist_test.sh configs/rretinanet/rretinanet_obb_r50_fpn_1x_dota_v3.py \
        work_dirs/rretinanet_obb_r50_fpn_1x_dota_v3/epoch_12.pth 1 --format-only\
        --eval-options submission_dir=work_dirs/rretinanet_obb_r50_fpn_1x_dota_v3/Task1_
↪results
```

CHAPTER

SEVEN

CROP IMAGES

For DOTA dataset, please crop the original images into 1024×1024 patches with an overlap of 200 by run

```
python tools/split/img_split.py --base_json \
       tools/split/split_configs/split_configs/dota1_0/ss_trainval.json

python tools/split/img_split.py --base_json \
       tools/split/split_configs/dota1_0/ss_test.json
```

Please change path in `ss_trainval.json`, `ss_test.json` to your path. (Forked from BboxToolkit, which is faster then DOTA_Devkit.)

CHAPTER

EIGHT

INDICES AND TABLES

- genindex
- search