幻灯片 1

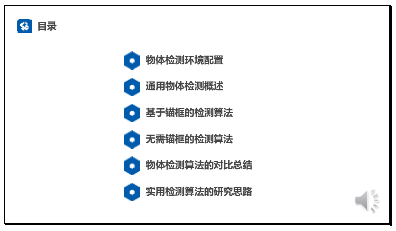
深蓝学院

# 通用物体检测

主讲人 张士峰

中国科学院自动化研究所
模式识别国家重点实验室博士

幻灯片 2



目录

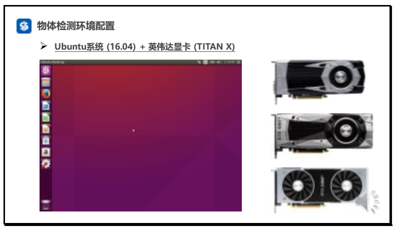
- 物体检测环境配置
- 通用物体检测概述
- 基于锚框的检测算法
- 无需锚框的检测算法
- 物体检测算法的对比总结
- 实用检测算法的研究思路

幻灯片 3

目录

- 物体检测环境配置
- 通用物体检测概述
- 基于锚框的检测算法
- 无需锚框的检测算法
- 物体检测算法的对比总结
- 实用检测算法的研究思路

幻灯片 4



## 物体检测环境配置

➢ Ubuntu系统 (16.04) + 英伟达显卡 (TITAN X)

幻灯片 5

## 物体检测环境配置

环境管理软件Anaconda

物体检测平台Detectron2

代码调试软件PyCharm

幻灯片 6

## 物体检测环境配置

环境管理软件Anaconda

物体检测平台Detectron2

代码调试软件PyCharm

幻灯片 7

## 物体检测环境配置

环境管理软件Anaconda

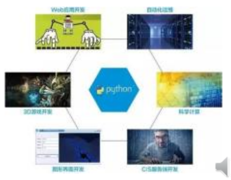
物体检测平台Detectron2

代码调试软件PyCharm

幻灯片 8

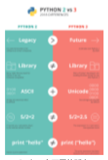
## 编程语言Python简介

- Python编程语言
- 深度学习最常用语言
- 各行各业都在用Python
- 各式各样的包
- PyTorch
- TensorFlow
- MXNet



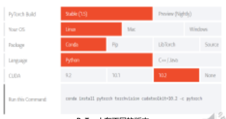
幻灯片 9

## 环境管理软件Anaconda



Python有不同的版本

**如何让一个系统有不同的Python环境?**



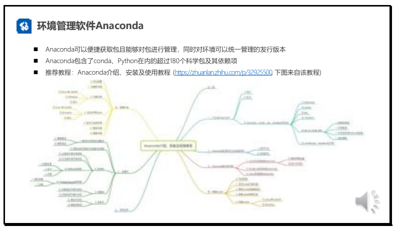
| | | | | |
|---|---|---|---|---|
| PyTorch Build | Stable (1.5) | | Preview (Nightly) | |
| Your OS | Linux | Mac | Windows | |
| Package | Conda | Pip | LibTorch | Source |
| Language | Python | | C++ / Java | |
| CUDA | 9.2 | 10.1 | 10.2 | None |

Run this Command:

```
conda install pytorch torchvision cudatoolkit=10.2 -c pytorch
```

PyTorch有不同的版本

幻灯片 10

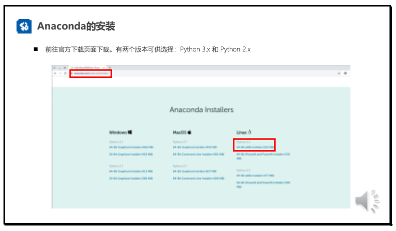


## 环境管理软件Anaconda

- Anaconda可以便捷获取包且能够对包进行管理，同时对环境可以统一管理的发行版本
- Anaconda包含了conda、Python在内的超过180个科学包及其依赖项
- 推荐教程：Anaconda介绍、安装及使用教程（https://zhuanlan.zhihu.com/p/32925500 下图来自该教程）

https://zhuanlan.zhihu.com/p/32925500

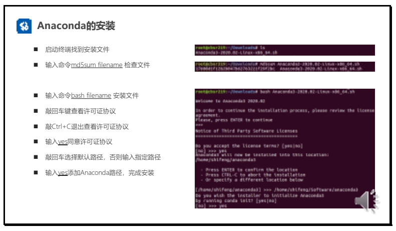
幻灯片 11



## Anaconda的安装

- 前往官方下载页面下载，有两个版本可供选择：Python 3.x 和 Python 2.x

幻灯片 12



## Anaconda的安装

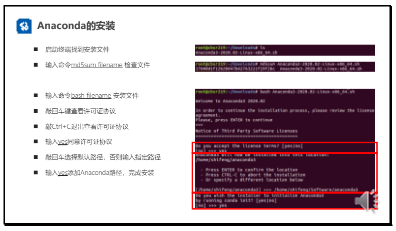
- 启动终端找到安装文件
- 输入命令md5sum filename 检查文件
- 输入命令bash filename 安装文件
- 敲回车键查看许可证协议
- 敲Ctrl+C退出查看许可证协议
- 输入yes同意许可证协议
- 敲回车选择默认路径，否则输入指定路径
- 输入yes添加Anaconda路径，完成安装

```
root@cbsr219:~/Downloads# ls
Anaconda3-2020.02-Linux-x86_64.sh
```

```
root@cbsr219:~/Downloads# md5sum Anaconda3-2020.02-Linux-x86_64.sh
17600d1f12b2b047b6276321f29f2bc  Anaconda3-2020.02-Linux-x86_64.sh
```

```
root@cbsr219:~/Downloads# bash Anaconda3-2020.02-Linux-x86_64.sh

Welcome to Anaconda3 2020.02

In order to continue the installation process, please review the license
agreement.
Please, press ENTER to continue
>>>
Notice of Third Party Software Licenses
========================================

Do you accept the license terms? [yes|no]
[no] >>> yes
Anaconda3 will now be installed into this location:
/home/shifeng/anaconda3

  - Press ENTER to confirm the location
  - Press CTRL-C to abort the installation
  - Or specify a different location below

[/home/shifeng/anaconda3] >>> /home/shifeng/Software/anaconda3
Do you wish the installer to initialize Anaconda3
by running conda init? [yes|no]
[no] >>> yes
```

幻灯片 13



## Anaconda的安装

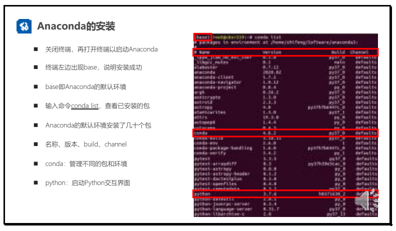
- 启动终端找到安装文件
- 输入命令md5sum filename 检查文件
- 输入命令bash filename 安装文件
- 敲回车键查看许可证协议
- 敲Ctrl+C退出查看许可证协议
- 输入yes同意许可证协议
- 敲回车选择默认路径，否则输入指定路径
- 输入yes添加Anaconda路径，完成安装

```
root@cbsr219:~/Downloads# ls
Anaconda3-2020.02-Linux-x86_64.sh
```

```
root@cbsr219:~/Downloads# md5sum Anaconda3-2020.02-Linux-x86_64.sh
17600d1f12b2b047b62763221f29f2bc  Anaconda3-2020.02-Linux-x86_64.sh
```

```
root@cbsr219:~/Downloads# bash Anaconda3-2020.02-Linux-x86_64.sh

Welcome to Anaconda3 2020.02

In order to continue the installation process, please review the license
agreement.
Please, press ENTER to continue
>>>
Notice of Third Party Software Licenses
=======================================
Do you accept the license terms? [yes|no]
[no] >>> yes
Anaconda3 will now be installed into this location:
/home/shifeng/anaconda3

  - Press ENTER to confirm the location
  - Press CTRL-C to abort the installation
  - Or specify a different location below

[/home/shifeng/anaconda3] >>> /home/shifeng/Software/anaconda3
Do you wish the installer to initialize Anaconda3
by running conda init? [yes|no]
[no] >>> yes
```

幻灯片 14

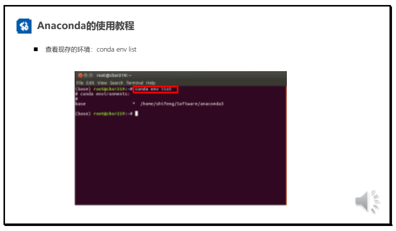


## Anaconda的安装

- 关闭终端，再打开终端以启动Anaconda
- 终端左边出现base，说明安装成功
- base即Anaconda的默认环境
- 输入命令conda list，查看已安装的包
- Anaconda的默认环境安装了几十个包
- 名称、版本、build、channel
- conda：管理不同的包和环境
- python：启动Python交互界面

```
(base) root@cbsr219:~# conda list
# packages in environment at /home/shifeng/Software/anaconda3:
#
# Name                    Version                   Build  Channel
_ipyw_jlab_nb_ext_conf    0.1.0                    py37_0    defaults
_libgcc_mutex             0.1                        main    defaults
alabaster                 0.7.12                   py37_0    defaults
anaconda                  2020.02                  py37_0    defaults
anaconda-client           1.7.2                    py37_0    defaults
anaconda-navigator        1.9.12                   py37_0    defaults
anaconda-project          0.8.4                      py_0    defaults
argh                      0.26.2                   py37_0    defaults
asn1crypto                1.3.0                    py37_0    defaults
astroid                   2.3.3                    py37_0    defaults
astropy                   4.0              py37h7b6447c_0    defaults
atomicwrites              1.3.0                    py37_1    defaults
attrs                     19.3.0                     py_0    defaults
autopep8                  1.4.4                      py_0    defaults
colorama                  0.4.3                      py_0    defaults
conda                     4.8.2                    py37_0    defaults
conda-build               3.18.11                  py37_0    defaults
conda-env                 2.6.0                         1    defaults
conda-package-handling    1.6.0            py37h7b6447c_0    defaults
conda-verify              3.4.2                      py_1    defaults
pytest                    5.3.5                    py37_0    defaults
pytest-arraydiff          0.3              py37h39e3cac_0    defaults
pytest-astropy            0.8.0                      py_0    defaults
pytest-astropy-header     0.1.2                      py_0    defaults
pytest-doctestplus        0.5.0                      py_0    defaults
pytest-openfiles          0.4.0                      py_0    defaults
pytest-remotedata         0.3.2                    py37_0    defaults
python                    3.7.6                h0371630_2    defaults
python-dateutil           2.8.1                      py_0    defaults
python-jsonrpc-server     0.3.4                      py_0    defaults
python-language-server    0.31.7                   py37_0    defaults
python-libarchive-c       2.8                     py37_13    defaults
```

幻灯片 15

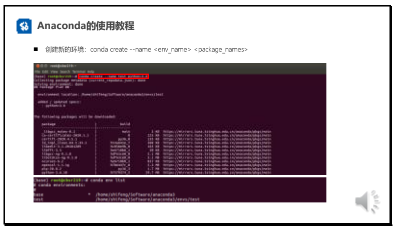


## Anaconda的使用教程

- 查看现存的环境：conda env list

```
root@cbsr219: ~
File  Edit  View  Search  Terminal  Help
(base) root@cbsr219:~# conda env list
# conda environments:
#
base                  *  /home/shifeng/Software/anaconda3

(base) root@cbsr219:~# 
```

幻灯片 16

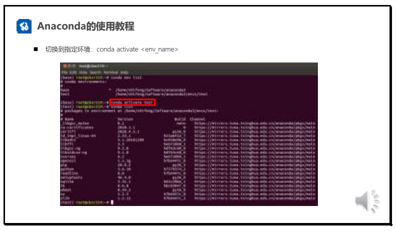


## Anaconda的使用教程

- 创建新的环境：conda create --name <env_name> <package_names>

```
(base) root@cbsr219:~# conda create --name test python=3.6
[illegible]
(base) root@cbsr219:~# conda env list
# conda environments:
#
base                  *  /home/shifeng/Software/anaconda3
test                     /home/shifeng/Software/anaconda3/envs/test
```

幻灯片 17

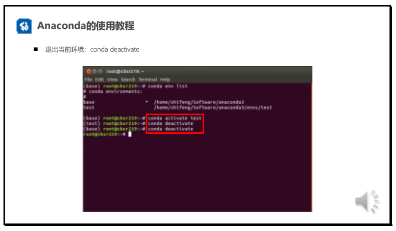


## Anaconda的使用教程

- 切换到指定环境：conda activate <env_name>

```
(base) root@cbsr219:~# conda env list
# conda environments:
#
base                  *  /home/shifeng/Software/anaconda3
test                     /home/shifeng/Software/anaconda3/envs/test

(base) root@cbsr219:~# conda activate test
[illegible]
```

幻灯片 18

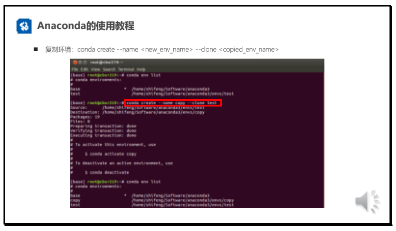


## Anaconda的使用教程

- 退出当前环境：conda deactivate

```
(base) root@cbsr219:~# conda env list
# conda environments:
#
base                  *  /home/shifeng/Software/anaconda3
test                     /home/shifeng/Software/anaconda3/envs/test

(base) root@cbsr219:~# conda activate test
(test) root@cbsr219:~# conda deactivate
(base) root@cbsr219:~# conda deactivate
root@cbsr219:~# 
```

幻灯片 19

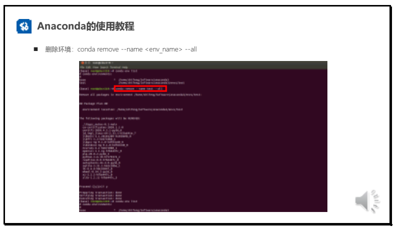


## Anaconda的使用教程

- 复制环境：conda create --name <new_env_name> --clone <copied_env_name>

```
root@cbsr219: ~
File Edit View Search Terminal Help
(base) root@cbsr219:~# conda env list
# conda environments:
#
base                  *  /home/shifeng/Software/anaconda3
test                     /home/shifeng/Software/anaconda3/envs/test

(base) root@cbsr219:~# conda create --name copy --clone test
Source:      /home/shifeng/Software/anaconda3/envs/test
Destination: /home/shifeng/Software/anaconda3/envs/copy
Packages: 19
Files: 0
Preparing transaction: done
Verifying transaction: done
Executing transaction: done
#
# To activate this environment, use
#
#     $ conda activate copy
#
# To deactivate an active environment, use
#
#     $ conda deactivate

(base) root@cbsr219:~# conda env list
# conda environments:
#
base                  *  /home/shifeng/Software/anaconda3
copy                     /home/shifeng/Software/anaconda3/envs/copy
test                     /home/shifeng/Software/anaconda3/envs/test
```

幻灯片 20

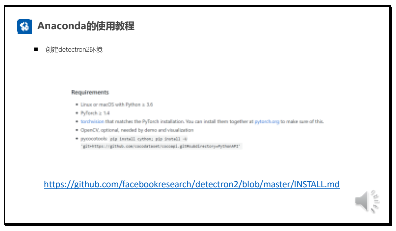


## Anaconda的使用教程

- 删除环境：conda remove --name <env_name> --all

```
(base) root@cbsr219:~# conda env list
# conda environments:
#
base                  *  /home/shifeng/Software/anaconda3
test                     /home/shifeng/Software/anaconda3/envs/test

(base) root@cbsr219:~# conda remove --name test --all

Remove all packages in environment /home/shifeng/Software/anaconda3/envs/test:

## Package Plan ##

  environment location: /home/shifeng/Software/anaconda3/envs/test

The following packages will be REMOVED:

  [illegible]

Proceed ([y]/n)? y

Preparing transaction: done
Verifying transaction: done
Executing transaction: done
(base) root@cbsr219:~# conda env list
# conda environments:
#
base                  *  /home/shifeng/Software/anaconda3
```

幻灯片 21



## Anaconda的使用教程

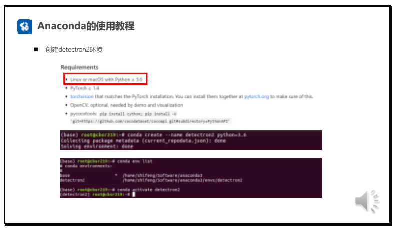
- 创建detectron2环境

**Requirements**

- Linux or macOS with Python ≥ 3.6
- PyTorch ≥ 1.4
- torchvision that matches the PyTorch installation. You can install them together at pytorch.org to make sure of this.
- OpenCV, optional, needed by demo and visualization
- pycocotools: `pip install cython; pip install -U 'git+https://github.com/cocodataset/cocoapi.git#subdirectory=PythonAPI'`

https://github.com/facebookresearch/detectron2/blob/master/INSTALL.md

幻灯片 22



## Anaconda的使用教程

- 创建detectron2环境

**Requirements**

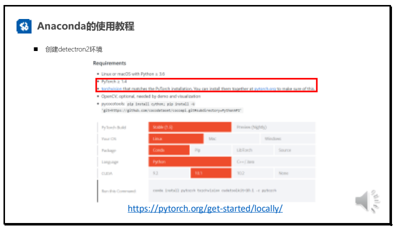
- Linux or macOS with Python ≥ 3.6
- PyTorch ≥ 1.4
- torchvision that matches the PyTorch installation. You can install them together at pytorch.org to make sure of this.
- OpenCV, optional, needed by demo and visualization
- pycocotools: `pip install cython; pip install -U 'git+https://github.com/cocodataset/cocoapi.git#subdirectory=PythonAPI'`

```
(base) root@cbsr219:~# conda create --name detectron2 python=3.6
Collecting package metadata (current_repodata.json): done
Solving environment: done
```

```
(base) root@cbsr219:~# conda env list
# conda environments:
#
base                  *  /home/shifeng/Software/anaconda3
detectron2               /home/shifeng/Software/anaconda3/envs/detectron2

(base) root@cbsr219:~# conda activate detectron2
(detectron2) root@cbsr219:~# 
```

- conda create --name detectron2 python=3.6
- conda activate detectron2

幻灯片 23



## Anaconda的使用教程

- 创建detectron2环境

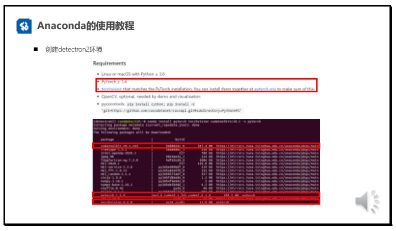
**Requirements**

- Linux or macOS with Python ≥ 3.6
- PyTorch ≥ 1.4
- torchvision that matches the PyTorch installation. You can install them together at pytorch.org to make sure of this.
- OpenCV, optional, needed by demo and visualization
- pycocotools: `pip install cython; pip install -U 'git+https://github.com/cocodataset/cocoapi.git#subdirectory=PythonAPI'`

| | | | | |
|---|---|---|---|---|
| PyTorch Build | [✓ Stable (1.5)] | | Preview (Nightly) | |
| Your OS | [✓ Linux] | Mac | | Windows |
| Package | [✓ Conda] | Pip | LibTorch | Source |
| Language | [✓ Python] | | C++ / Java | |
| CUDA | 9.2 | [✓ 10.1] | 10.2 | None |
| Run this Command: | conda install pytorch torchvision cudatoolkit=10.1 -c pytorch | | | |

https://pytorch.org/get-started/locally/

幻灯片 24



## Anaconda的使用教程

- 创建detectron2环境

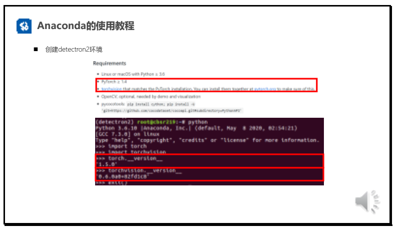
**Requirements**

- Linux or macOS with Python ≥ 3.6
- PyTorch ≥ 1.4
- torchvision that matches the PyTorch installation. You can install them together at pytorch.org to make sure of this.
- OpenCV, optional, needed by demo and visualization
- pycocotools: `pip install cython; pip install -U 'git+https://github.com/cocodataset/cocoapi.git#subdirectory=PythonAPI'`

```
(detectron2) root@cbsr219:~# conda install pytorch torchvision cudatoolkit=10.1 -c pytorch
Collecting package metadata (current_repodata.json): done
Solving environment: done
The following packages will be downloaded:

    package                    |            build
[illegible]
```

- conda install pytorch torchvision cudatoolkit=10.1 -c pytorch

幻灯片 25



## Anaconda的使用教程

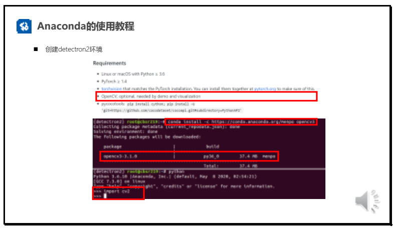
- 创建detectron2环境

**Requirements**

- Linux or macOS with Python ≥ 3.6
- PyTorch ≥ 1.4
- torchvision that matches the PyTorch installation. You can install them together at pytorch.org to make sure of this.
- OpenCV, optional, needed by demo and visualization
- pycocotools: `pip install cython; pip install -U 'git+https://github.com/cocodataset/cocoapi.git#subdirectory=PythonAPI'`

```
(detectron2) root@cbsr219:~# python
Python 3.6.10 |Anaconda, Inc.| (default, May  8 2020, 02:54:21)
[GCC 7.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import torch
>>> import torchvision
>>> torch.__version__
'1.5.0'
>>> torchvision.__version__
'0.6.0a0+82fd1c8'
>>> exit()
```

幻灯片 26



## Anaconda的使用教程

- 创建detectron2环境

**Requirements**

- Linux or macOS with Python ≥ 3.6
- PyTorch ≥ 1.4
- torchvision that matches the PyTorch installation. You can install them together at pytorch.org to make sure of this.
- OpenCV, optional, needed by demo and visualization
- pycocotools: `pip install cython; pip install -U 'git+https://github.com/cocodataset/cocoapi.git#subdirectory=PythonAPI'`

```
(detectron2) root@cbsr219:~# conda install -c https://conda.anaconda.org/menpo opencv3
Collecting package metadata (current_repodata.json): done
Solving environment: done
The following packages will be downloaded:

    package                    |            build
    opencv3-3.1.0              |           py36_0        37.4 MB  menpo
                                           Total:        37.4 MB
(detectron2) root@cbsr219:~# python
Python 3.6.10 |Anaconda, Inc.| (default, May  8 2020, 02:54:21)
[GCC 7.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import cv2
>>> 
```

- conda install --channel https://conda.anaconda.org/menpo opencv3

幻灯片 27

## Anaconda的使用教程

- 创建detectron2环境

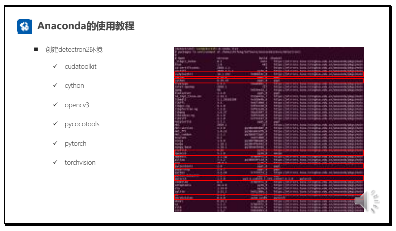
**Requirements**

- Linux or macOS with Python ≥ 3.6
- PyTorch ≥ 1.4
- torchvision that matches the PyTorch installation. You can install them together at pytorch.org to make sure of this.
- OpenCV, optional, needed by demo and visualization
- pycocotools: `pip install cython; pip install -U 'git+https://github.com/cocodataset/cocoapi.git#subdirectory=PythonAPI'`

```
[illegible]
```

- pip install cython; pip install -U 'git+https://github.com/cocodataset/cocoapi.git#subdirectory=PythonAPI'

幻灯片 28



## Anaconda的使用教程

- 创建detectron2环境
  - ✓ cudatoolkit
  - ✓ cython
  - ✓ opencv3
  - ✓ pycocotools
  - ✓ pytorch
  - ✓ torchvision

```
[illegible]
```

幻灯片 29

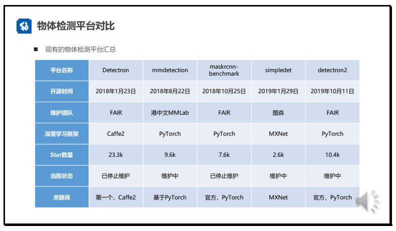
## 物体检测环境配置

环境管理软件Anaconda

物体检测平台Detectron2

代码调试软件PyCharm

幻灯片 30

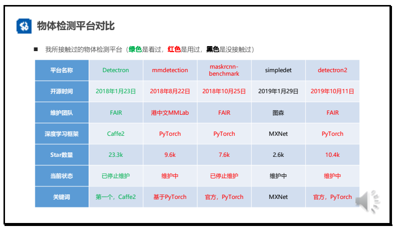


## 物体检测平台对比

- 现有的物体检测平台汇总

| 平台名称 | Detectron | mmdetection | maskrcnn-benchmark | simpledet | detectron2 |
|---|---|---|---|---|---|
| 开源时间 | 2018年1月23日 | 2018年8月22日 | 2018年10月25日 | 2019年1月29日 | 2019年10月11日 |
| 维护团队 | FAIR | 港中文MMLab | FAIR | 图森 | FAIR |
| 深度学习框架 | Caffe2 | PyTorch | PyTorch | MXNet | PyTorch |
| Star数量 | 23.3k | 9.6k | 7.6k | 2.6k | 10.4k |
| 当前状态 | 已停止维护 | 维护中 | 已停止维护 | 维护中 | 维护中 |
| 关键词 | 第一个，Caffe2 | 基于PyTorch | 官方，PyTorch | MXNet | 官方，PyTorch |

幻灯片 31

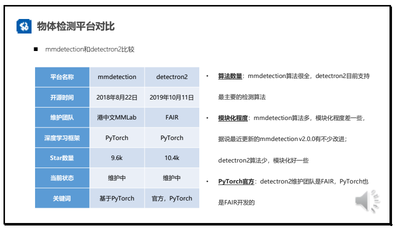


## 物体检测平台对比

- 我所接触过的物体检测平台（**绿色**是看过，**红色**是用过，**黑色**是没接触过）

| 平台名称 | Detectron | mmdetection | maskrcnn-benchmark | simpledet | detectron2 |
|---|---|---|---|---|---|
| 开源时间 | 2018年1月23日 | 2018年8月22日 | 2018年10月25日 | 2019年1月29日 | 2019年10月11日 |
| 维护团队 | FAIR | 港中文MMLab | FAIR | 图森 | FAIR |
| 深度学习框架 | Caffe2 | PyTorch | PyTorch | MXNet | PyTorch |
| Star数量 | 23.3k | 9.6k | 7.6k | 2.6k | 10.4k |
| 当前状态 | 已停止维护 | 维护中 | 已停止维护 | 维护中 | 维护中 |
| 关键词 | 第一个，Caffe2 | 基于PyTorch | 官方，PyTorch | MXNet | 官方，PyTorch |

幻灯片 32



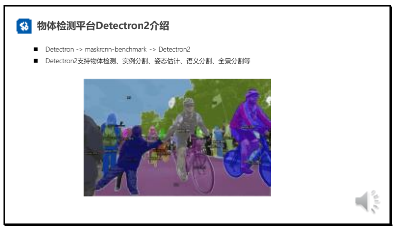
## 物体检测平台对比

- mmdetection和detectron2比较

| 平台名称 | mmdetection | detectron2 |
|---|---|---|
| 开源时间 | 2018年8月22日 | 2019年10月11日 |
| 维护团队 | 港中文MMLab | FAIR |
| 深度学习框架 | PyTorch | PyTorch |
| Star数量 | 9.6k | 10.4k |
| 当前状态 | 维护中 | 维护中 |
| 关键词 | 基于PyTorch | 官方，PyTorch |

- **算法数量**：mmdetection算法很全，detectron2目前支持最主要的检测算法
- **模块化程度**：mmdetection算法多，模块化程度差一些，据说最近更新的mmdetection v2.0.0有不少改进；detectron2算法少，模块化好一些
- **PyTorch官方**：detectron2维护团队是FAIR，PyTorch也是FAIR开发的

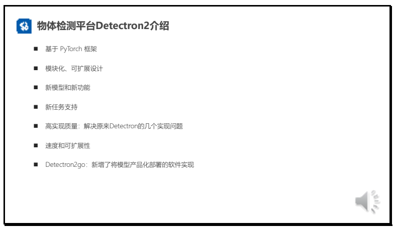
幻灯片 33



## 物体检测平台Detectron2介绍

- Detectron -> maskrcnn-benchmark -> Detectron2
- Detectron2支持物体检测、实例分割、姿态估计、语义分割、全景分割等

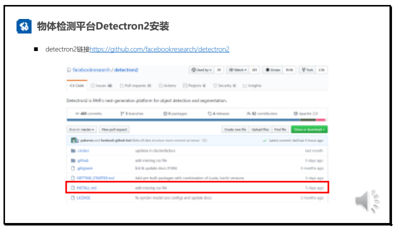
幻灯片 34



## 物体检测平台Detectron2介绍

- 基于 PyTorch 框架
- 模块化、可扩展设计
- 新模型和新功能
- 新任务支持
- 高实现质量：解决原来Detectron的几个实现问题
- 速度和可扩展性
- Detectron2go：新增了将模型产品化部署的软件实现

幻灯片 35



## 物体检测平台Detectron2安装

- detectron2链接https://github.com/facebookresearch/detectron2

facebookresearch / detectron2

Used by 19 | Watch 281 | Unstar 10.4k | Fork 2.2k

Code | Issues 44 | Pull requests 11 | Actions | Projects 0 | Security 0 | Insights

Detectron2 is FAIR's next-generation platform for object detection and segmentation.

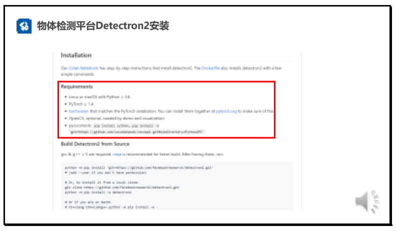
488 commits | 5 branches | 0 packages | 4 releases | 82 contributors | Apache-2.0

Branch: master | New pull request | Create new file | Upload files | Find file | Clone or download

| | | |
|---|---|---|
| .circleci | updates in docker/docs | last month |
| .github | add missing css file | 5 days ago |
| .gitignore | lint & update docs (PSRN) | 3 months ago |
| GETTING_STARTED.md | Add pre-built packages with combination of {cuda, torch} versions | 4 days ago |
| INSTALL.md | add missing css file | 5 days ago |
| LICENSE | fix syncbn model zoo configs and update docs | 3 months ago |

https://github.com/facebookresearch/detectron2

幻灯片 36



## 物体检测平台Detectron2安装

### Installation

Our Colab Notebook has step-by-step instructions that install detectron2. The Dockerfile also installs detectron2 with a few simple commands.

#### Requirements

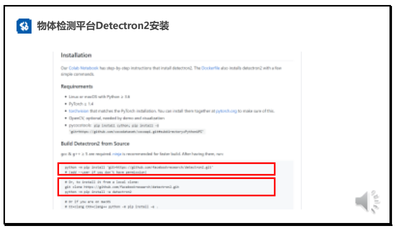
- Linux or macOS with Python ≥ 3.6
- PyTorch ≥ 1.4
- torchvision that matches the PyTorch installation. You can install them together at pytorch.org to make sure of this.
- OpenCV, optional, needed by demo and visualization
- pycocotools: `pip install cython; pip install -U 'git+https://github.com/cocodataset/cocoapi.git#subdirectory=PythonAPI'`

#### Build Detectron2 from Source

gcc & g++ ≥ 5 are required. ninja is recommended for faster build. After having them, run:

```
python -m pip install 'git+https://github.com/facebookresearch/detectron2.git'
# (add --user if you don't have permission)

# Or, to install it from a local clone:
git clone https://github.com/facebookresearch/detectron2.git
python -m pip install -e detectron2

# Or if you are on macOS
# CC=clang CXX=clang++ python -m pip install -e .
```

幻灯片 37



## 物体检测平台Detectron2安装

Installation

Our Colab Notebook has step-by-step instructions that install detectron2. The Dockerfile also installs detectron2 with a few simple commands.

Requirements

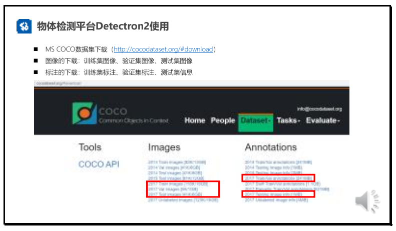
- Linux or macOS with Python ≥ 3.6
- PyTorch ≥ 1.4
- torchvision that matches the PyTorch installation. You can install them together at pytorch.org to make sure of this.
- OpenCV, optional, needed by demo and visualization
- pycocotools: `pip install cython; pip install -U 'git+https://github.com/cocodataset/cocoapi.git#subdirectory=PythonAPI'`

Build Detectron2 from Source

gcc & g++ ≥ 5 are required. ninja is recommended for faster build. After having them, run:

```
python -m pip install 'git+https://github.com/facebookresearch/detectron2.git'
# (add --user if you don't have permission)

# Or, to install it from a local clone:
git clone https://github.com/facebookresearch/detectron2.git
python -m pip install -e detectron2

# Or if you are on macOS
# CC=clang CXX=clang++ python -m pip install -e .
```

幻灯片 38



## 物体检测平台Detectron2使用

- MS COCO数据集下载（http://cocodataset.org/#download）
- 图像的下载：训练集图像、验证集图像、测试集图像
- 标注的下载：训练集标注、验证集标注、测试集信息

Tools

COCO API

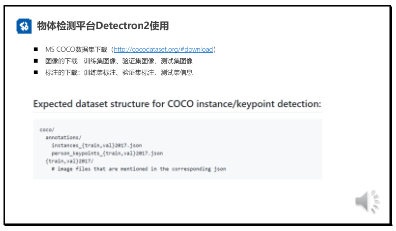
Images

2014 Train images [83K/13GB]
2014 Val images [41K/6GB]
2014 Test images [41K/6GB]
2015 Test images [81K/12GB]
2017 Train images [118K/18GB]
2017 Val images [5K/1GB]
2017 Test images [41K/6GB]
2017 Unlabeled images [123K/19GB]

Annotations

2014 Train/Val annotations [241MB]
2014 Testing Image info [1MB]
2015 Testing Image info [2MB]
2017 Train/Val annotations [241MB]
2017 Stuff Train/Val annotations [1.1GB]
2017 Panoptic Train/Val annotations [821MB]
2017 Testing Image info [1MB]
2017 Unlabeled Image info [4MB]

幻灯片 39



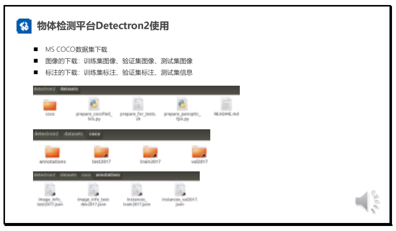
## 物体检测平台Detectron2使用

- MS COCO数据集下载（http://cocodataset.org/#download）
- 图像的下载：训练集图像、验证集图像、测试集图像
- 标注的下载：训练集标注、验证集标注、测试集信息

Expected dataset structure for COCO instance/keypoint detection:

```
coco/
  annotations/
    instances_{train,val}2017.json
    person_keypoints_{train,val}2017.json
  {train,val}2017/
    # image files that are mentioned in the corresponding json
```

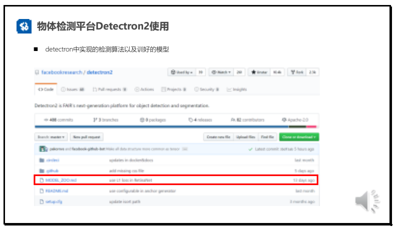
幻灯片 40



## 物体检测平台Detectron2使用

- MS COCO数据集下载
- 图像的下载：训练集图像、验证集图像、测试集图像
- 标注的下载：训练集标注、验证集标注、测试集信息

幻灯片 41



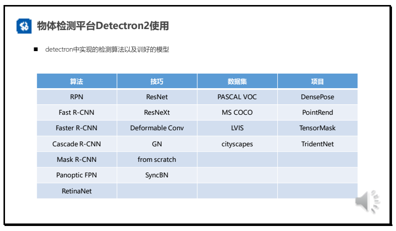
## 物体检测平台Detectron2使用

- detectron中实现的检测算法以及训好的模型

Detectron2 is FAIR's next-generation platform for object detection and segmentation.

```
488 commits    3 branches    0 packages    4 releases    82 contributors    Apache-2.0
Branch: master   New pull request        Create new file   Upload files   Find file   Clone or download
ppwwyyxx and facebook-github-bot Make all data structure more common as tensor     Latest commit 5 hours ago
.circleci        updates in docker&docs            last month
.github          add missing css file              5 days ago
MODEL_ZOO.md     use L1 loss in RetinaNet          13 days ago
README.md        use configurable in anchor generator    last month
setup.cfg        update isort path                 3 months ago
```

幻灯片 42

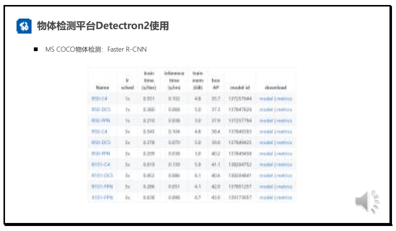


## 物体检测平台Detectron2使用

- detectron中实现的检测算法以及训好的模型

| 算法 | 技巧 | 数据集 | 项目 |
|---|---|---|---|
| RPN | ResNet | PASCAL VOC | DensePose |
| Fast R-CNN | ResNeXt | MS COCO | PointRend |
| Faster R-CNN | Deformable Conv | LVIS | TensorMask |
| Cascade R-CNN | GN | cityscapes | TridentNet |
| Mask R-CNN | from scratch | | |
| Panoptic FPN | SyncBN | | |
| RetinaNet | | | |

幻灯片 43

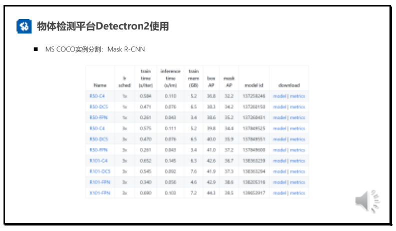


## 物体检测平台Detectron2使用

- MS COCO物体检测：Faster R-CNN

| Name | lr sched | train time (s/iter) | inference time (s/im) | train mem (GB) | box AP | model id | download |
|---|---|---|---|---|---|---|---|
| R50-C4 | 1x | 0.551 | 0.102 | 4.8 | 35.7 | 137257644 | model \| metrics |
| R50-DC5 | 1x | 0.380 | 0.068 | 5.0 | 37.3 | 137847829 | model \| metrics |
| R50-FPN | 1x | 0.210 | 0.038 | 3.0 | 37.9 | 137257794 | model \| metrics |
| R50-C4 | 3x | 0.543 | 0.104 | 4.8 | 38.4 | 137849393 | model \| metrics |
| R50-DC5 | 3x | 0.378 | 0.070 | 5.0 | 39.0 | 137849425 | model \| metrics |
| R50-FPN | 3x | 0.209 | 0.038 | 3.0 | 40.2 | 137849458 | model \| metrics |
| R101-C4 | 3x | 0.619 | 0.139 | 5.9 | 41.1 | 138204752 | model \| metrics |
| R101-DC5 | 3x | 0.452 | 0.086 | 6.1 | 40.6 | 138204841 | model \| metrics |
| R101-FPN | 3x | 0.286 | 0.051 | 4.1 | 42.0 | 137851257 | model \| metrics |
| X101-FPN | 3x | 0.638 | 0.098 | 6.7 | 43.0 | 139173657 | model \| metrics |

幻灯片 44

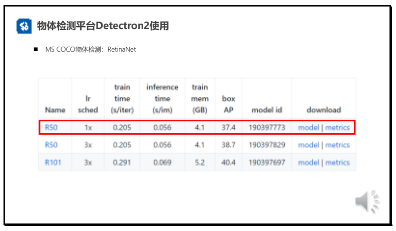


## 物体检测平台Detectron2使用

- MS COCO实例分割：Mask R-CNN

| Name | lr sched | train time (s/iter) | inference time (s/im) | train mem (GB) | box AP | mask AP | model id | download |
|---|---|---|---|---|---|---|---|---|
| R50-C4 | 1x | 0.584 | 0.110 | 5.2 | 36.8 | 32.2 | 137259246 | model \| metrics |
| R50-DC5 | 1x | 0.471 | 0.076 | 6.5 | 38.3 | 34.2 | 137260150 | model \| metrics |
| R50-FPN | 1x | 0.261 | 0.043 | 3.4 | 38.6 | 35.2 | 137260431 | model \| metrics |
| R50-C4 | 3x | 0.575 | 0.111 | 5.2 | 39.8 | 34.4 | 137849525 | model \| metrics |
| R50-DC5 | 3x | 0.470 | 0.076 | 6.5 | 40.0 | 35.9 | 137849551 | model \| metrics |
| R50-FPN | 3x | 0.261 | 0.043 | 3.4 | 41.0 | 37.2 | 137849600 | model \| metrics |
| R101-C4 | 3x | 0.652 | 0.145 | 6.3 | 42.6 | 36.7 | 138363239 | model \| metrics |
| R101-DC5 | 3x | 0.545 | 0.092 | 7.6 | 41.9 | 37.3 | 138363294 | model \| metrics |
| R101-FPN | 3x | 0.340 | 0.056 | 4.6 | 42.9 | 38.6 | 138205316 | model \| metrics |
| X101-FPN | 3x | 0.690 | 0.103 | 7.2 | 44.3 | 39.5 | 139653917 | model \| metrics |

幻灯片 45

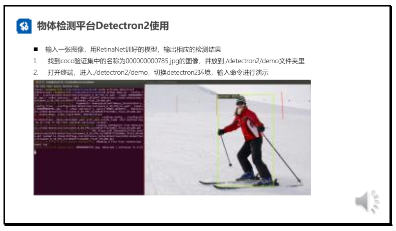


## 物体检测平台Detectron2使用

- MS COCO物体检测：RetinaNet

| Name | lr sched | train time (s/iter) | inference time (s/im) | train mem (GB) | box AP | model id | download |
|---|---|---|---|---|---|---|---|
| R50 | 1x | 0.205 | 0.056 | 4.1 | 37.4 | 190397773 | model \| metrics |
| R50 | 3x | 0.205 | 0.056 | 4.1 | 38.7 | 190397829 | model \| metrics |
| R101 | 3x | 0.291 | 0.069 | 5.2 | 40.4 | 190397697 | model \| metrics |

幻灯片 46



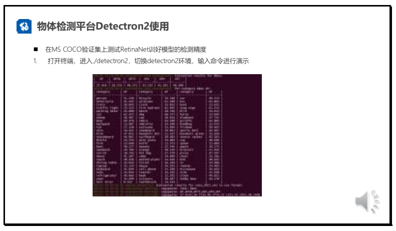
## 物体检测平台Detectron2使用

- 输入一张图像，用RetinaNet训好的模型，输出相应的检测结果

1. 找到coco验证集中的名称为000000000785.jpg的图像，并放到./detectron2/demo文件夹里
2. 打开终端，进入./detectron2/demo，切换detectron2环境，输入命令进行演示

- conda activate detectron2
- cd ./detectron2/demo
- python demo.py --config-file ../configs/COCO-Detection/retinanet_R_50_FPN_1x.yaml --input 000000000785.jpg [--other-options] --opts MODEL.WEIGHTS detectron2://COCO-Detection/retinanet_R_50_FPN_1x/190397773/model_final_bfca0b.pkl

幻灯片 47



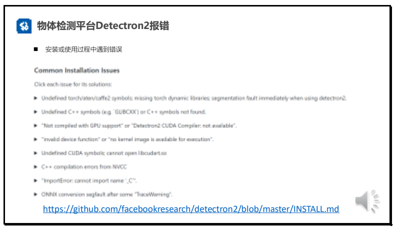
## 物体检测平台Detectron2使用

- 在MS COCO验证集上测试RetinaNet训好模型的检测精度

1. 打开终端，进入./detectron2，切换detectron2环境，输入命令进行演示

- conda activate detectron2
- cd ./detectron2
- python ./tools/train_net.py --config-file ./configs/COCO-Detection/retinanet_R_50_FPN_1x.yaml --eval-only MODEL.WEIGHTS detectron2://COCO-Detection/retinanet_R_50_FPN_1x/190397773/model_final_bfca0b.pkl

幻灯片 48



## 物体检测平台Detectron2报错

- 安装或使用过程中遇到错误

**Common Installation Issues**

Click each issue for its solutions:

- Undefined torch/aten/caffe2 symbols; missing torch dynamic libraries; segmentation fault immediately when using detectron2.
- Undefined C++ symbols (e.g. `GLIBCXX`) or C++ symbols not found.
- "Not compiled with GPU support" or "Detectron2 CUDA Compiler: not available".
- "invalid device function" or "no kernel image is available for execution".
- Undefined CUDA symbols; cannot open libcudart.so
- C++ compilation errors from NVCC
- "ImportError: cannot import name '_C'".
- ONNX conversion segfault after some "TraceWarning".

https://github.com/facebookresearch/detectron2/blob/master/INSTALL.md

幻灯片 49

## 物体检测环境配置

环境管理软件Anaconda
物体检测平台Detectron2
代码调试软件PyCharm

幻灯片 50

## 代码调试软件PyCharm介绍

- PyCharm是一种Python IDE，其带有一整套可以帮助用户在使用Python语言开发时提高其效率的工具，比如调试、语法高亮、Project管理、代码跳转、智能提示、自动完成、单元测试、版本控制等。

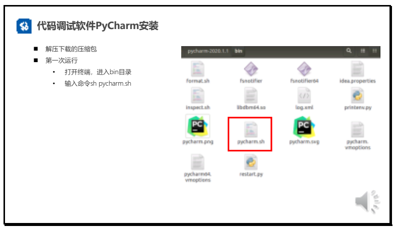
PC
PyCharm
COMMUNITY 2016.2

幻灯片 51

## 代码调试软件PyCharm安装

- Pycharm提供免费的社区版与付费的专业版，专业版额外增加了一些功能，个人学习使用社区版足够
- 如果是学生或老师，有学校的邮箱，可以免费使用功能更多的专业版
- 下载链接https://www.jetbrains.com/pycharm/download/#section=linux

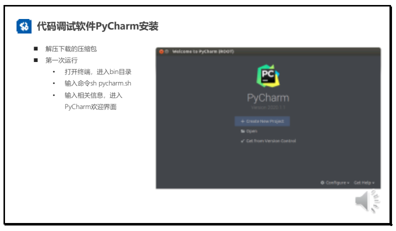
幻灯片 52



## 代码调试软件PyCharm安装

- 解压下载的压缩包
- 第一次运行
  - 打开终端，进入bin目录
  - 输入命令sh pycharm.sh

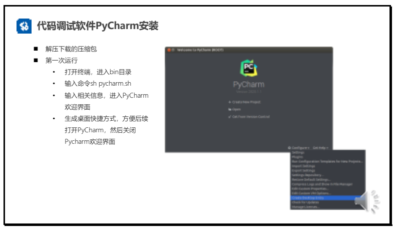
幻灯片 53



## 代码调试软件PyCharm安装

- 解压下载的压缩包
- 第一次运行
  - 打开终端，进入bin目录
  - 输入命令sh pycharm.sh
  - 输入相关信息，进入PyCharm欢迎界面

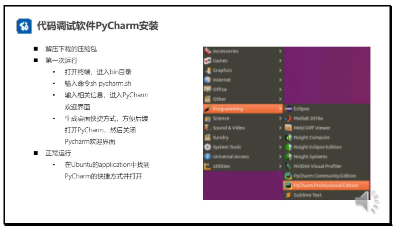
幻灯片 54



## 代码调试软件PyCharm安装

- 解压下载的压缩包
- 第一次运行
  - 打开终端，进入bin目录
  - 输入命令sh pycharm.sh
  - 输入相关信息，进入PyCharm欢迎界面
  - 生成桌面快捷方式，方便后续打开PyCharm，然后关闭Pycharm欢迎界面

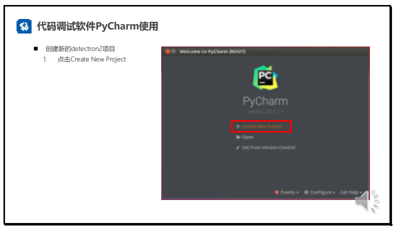
幻灯片 55



## 代码调试软件PyCharm安装

- 解压下载的压缩包
- 第一次运行
  - 打开终端，进入bin目录
  - 输入命令sh pycharm.sh
  - 输入相关信息，进入PyCharm欢迎界面
  - 生成桌面快捷方式，方便后续打开PyCharm，然后关闭Pycharm欢迎界面
- 正常运行
  - 在Ubuntu的application中找到PyCharm的快捷方式并打开

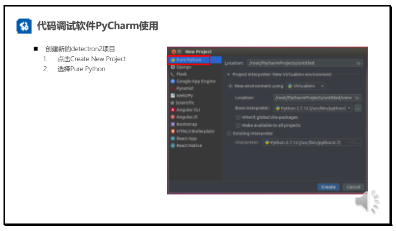
幻灯片 56



## 代码调试软件PyCharm使用

- 创建新的detectron2项目
  1. 点击Create New Project

幻灯片 57

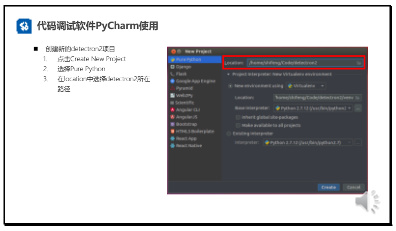


## 代码调试软件PyCharm使用

- 创建新的detectron2项目
  1. 点击Create New Project
  2. 选择Pure Python

New Project
Pure Python
Django
Flask
Google App Engine
Pyramid
Web2Py
Scientific
Angular CLI
AngularJS
Bootstrap
HTML5 Boilerplate
React App
React Native
Location: /root/PycharmProjects/untitled
Project Interpreter: New Virtualenv environment
New environment using Virtualenv
Location: /root/PycharmProjects/untitled/venv
Base interpreter: Python 2.7.12 (/usr/bin/python2
Inherit global site-packages
Make available to all projects
Existing interpreter
Interpreter: Python 2.7.12 (/usr/bin/python2.7)
Create
Cancel

幻灯片 58

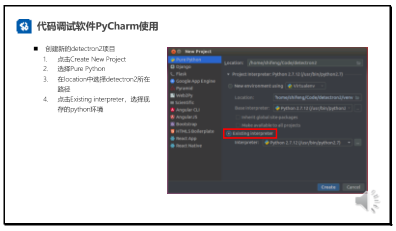


## 代码调试软件PyCharm使用

- 创建新的detectron2项目
  1. 点击Create New Project
  2. 选择Pure Python
  3. 在location中选择detectron2所在路径

Location: /home/shifeng/code/detectron2

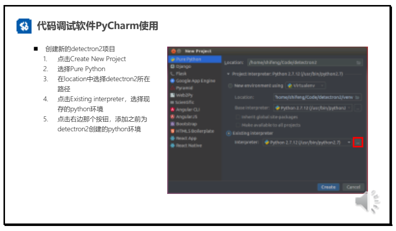
幻灯片 59



## 代码调试软件PyCharm使用

- 创建新的detectron2项目
  1. 点击Create New Project
  2. 选择Pure Python
  3. 在location中选择detectron2所在路径
  4. 点击Existing interpreter，选择现存的python环境

幻灯片 60



## 代码调试软件PyCharm使用

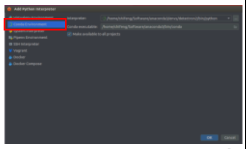
- 创建新的detectron2项目
  1. 点击Create New Project
  2. 选择Pure Python
  3. 在location中选择detectron2所在路径
  4. 点击Existing interpreter，选择现存的python环境
  5. 点击右边那个按钮，添加之前为detectron2创建的python环境

Location: /home/shifeng/code/detectron2
Project Interpreter: Python 2.7.12 (/usr/bin/python2.7)
Existing interpreter
Interpreter: Python 2.7.12 (/usr/bin/python2.7)

幻灯片 61

## 代码调试软件PyCharm使用

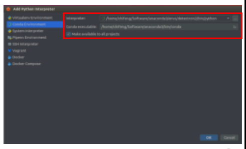
- 创建新的detectron2项目
  1. 点击Create New Project
  2. 选择Pure Python
  3. 在location中选择detectron2所在路径
  4. 点击Existing interpreter，选择现存的python环境
  5. 点击右边那个按钮，添加之前为detectron2创建的python环境
  6. 选择Conda Environment



幻灯片 62

## 代码调试软件PyCharm使用

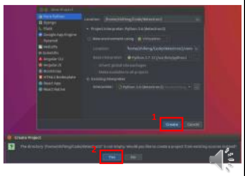
- 创建新的detectron2项目
  1. 点击Create New Project
  2. 选择Pure Python
  3. 在location中选择detectron2所在路径
  4. 点击Existing interpreter，选择现存的python环境
  5. 点击右边那个按钮，添加之前为detectron2创建的python环境
  6. 选择Conda Environment
  7. 找到detectron2环境下的python路径，选择make available to all pro，点OK返回



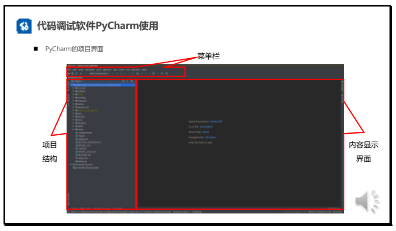
幻灯片 63

## 代码调试软件PyCharm使用

- 创建新的detectron2项目
  1. 点击Create New Project
  2. 选择Pure Python
  3. 在location中选择detectron2所在路径
  4. 点击Existing interpreter，选择现存的python环境
  5. 点击右边那个按钮，添加之前为detectron2创建的python环境
  6. 选择Conda Environment
  7. 找到detectron2环境下的python路径，选择make available to all pro，点OK返回
  8. 点Create，最后点Yes，成功创建项目



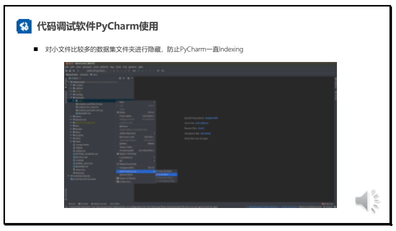
幻灯片 64



## 代码调试软件PyCharm使用

- PyCharm的项目界面

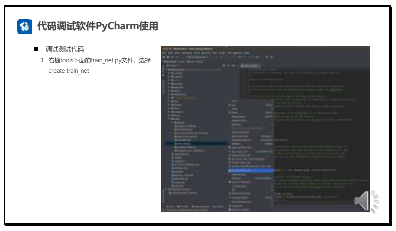
幻灯片 65



## 代码调试软件PyCharm使用

- 对小文件比较多的数据集文件夹进行隐藏，防止PyCharm一直Indexing

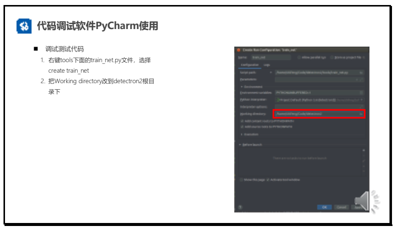
幻灯片 66



## 代码调试软件PyCharm使用

- 调试测试代码
  1. 右键tools下面的train_net.py文件，选择create train_net

幻灯片 67



## 代码调试软件PyCharm使用

- 调试测试代码
  1. 右键tools下面的train_net.py文件，选择create train_net
  2. 把Working directory改到detectron2根目录下

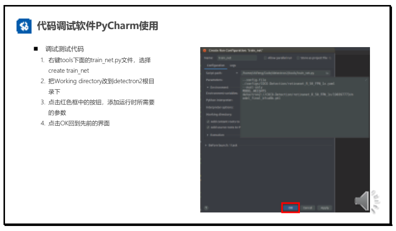
幻灯片 68

## 代码调试软件PyCharm使用

- 调试测试代码
  1. 右键tools下面的train_net.py文件，选择create train_net
  2. 把Working directory改到detectron2根目录下
  3. 点击红色框中的按钮，添加运行时所需要的参数

--config-file ./configs/COCO-Detection/retinanet_R_50_FPN_1x.yaml --eval-only MODEL.WEIGHTS detectron2://COCO-Detection/retinanet_R_50_FPN_1x/190397773/model_final_bfca0b.pkl

幻灯片 69



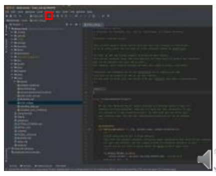
## 代码调试软件PyCharm使用

- 调试测试代码
  1. 右键tools下面的train_net.py文件，选择create train_net
  2. 把Working directory改到detectron2根目录下
  3. 点击红色框中的按钮，添加运行时所需要的参数
  4. 点击OK回到先前的界面

幻灯片 70

## 代码调试软件PyCharm使用

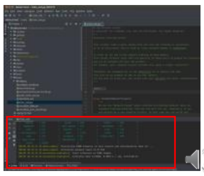
- 调试测试代码
  1. 右键tools下面的train_net.py文件，选择create train_net
  2. 把Working directory改到detectron2根目录下
  3. 点击红色框中的按钮，添加运行时所需要的参数
  4. 点击OK回到先前的界面
  5. 点击三角形按钮运行测试代码



幻灯片 71

## 代码调试软件PyCharm使用

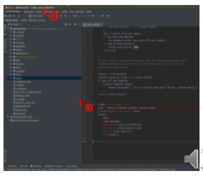
- 调试测试代码
  1. 右键tools下面的train_net.py文件，选择create train_net
  2. 把Working directory改到detectron2根目录下
  3. 点击红色框中的按钮，添加运行时所需要的参数
  4. 点击OK回到先前的界面
  5. 点击三角形按钮运行测试代码
  6. 测试代码开始运行，直至结束



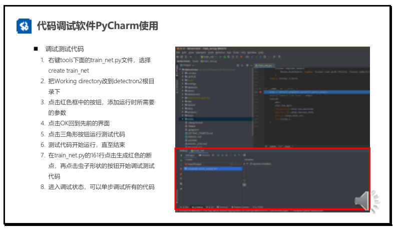
幻灯片 72

## 代码调试软件PyCharm使用

- 调试测试代码
  1. 右键tools下面的train_net.py文件，选择create train_net
  2. 把Working directory改到detectron2根目录下
  3. 点击红色框中的按钮，添加运行时所需要的参数
  4. 点击OK回到先前的界面
  5. 点击三角形按钮运行测试代码
  6. 测试代码开始运行，直至结束
  7. 在train_net.py的161行点击生成红色的断点，再点击虫子形状的按钮开始调试测试代码



幻灯片 73



## 代码调试软件PyCharm使用

- 调试测试代码
  1. 右键tools下面的train_net.py文件，选择create train_net
  2. 把Working directory改到detectron2根目录下
  3. 点击红色框中的按钮，添加运行时所需要的参数
  4. 点击OK回到先前的界面
  5. 点击三角形按钮运行测试代码
  6. 测试代码开始运行，直至结束
  7. 在train_net.py的161行点击生成红色的断点，再点击虫子形状的按钮开始调试测试代码
  8. 进入调试状态，可以单步调试所有的代码