


YOLOv5项目实战-TensorRT加速

课程演示环境: Windows10; cuda 10.2; cudnn7.6.5;
VisualStudio2019; Opencv3.4.0

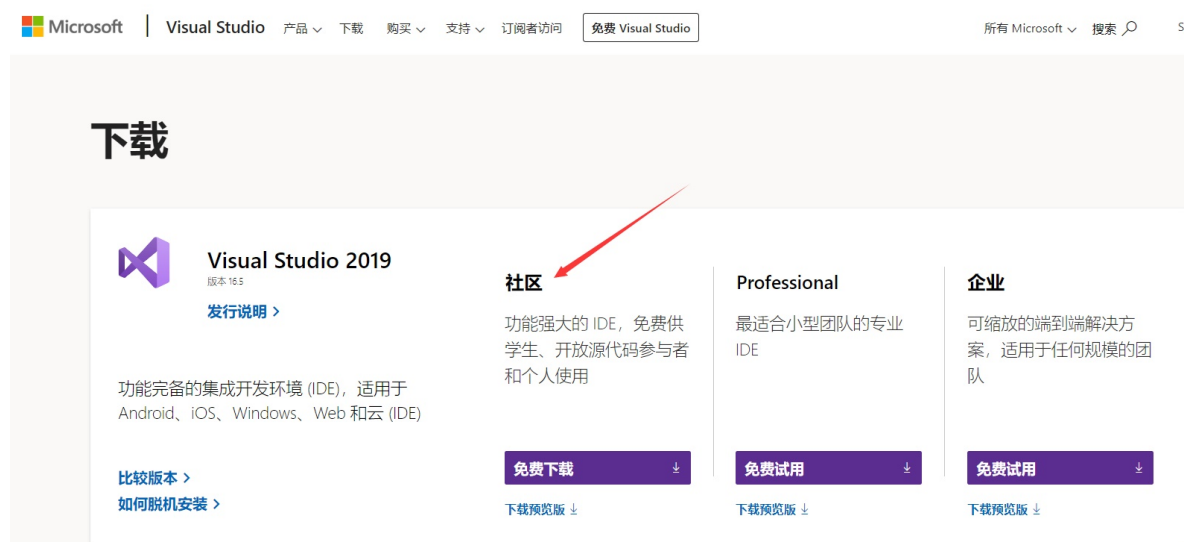
1 软件安装

名称	修改日期	类型	大小
 cuda_10.2.89_441.22_win10.exe	2020/4/28 11:29	应用程序	2,556,900...
 cudnn-10.2-windows10-x64-v7.6.5.32.zip	2020/4/28 12:06	360压缩 ZIP 文件	284,026 KB
 opencv-3.4.0-vc14_vc15.exe	2020/4/28 13:19	应用程序	174,194 KB
 vs_community__1888582249.1552317504.exe	2020/4/28 12:16	应用程序	1,362 KB

1) 首先安装Visual Studio 2019

下载Visual Studio 社区版


下载链接: <https://visualstudio.microsoft.com/zh-hans/downloads/>



Microsoft | Visual Studio 产品 下载 购买 支持 订阅者访问 免费 Visual Studio

所有 Microsoft 搜索

下载



Visual Studio 2019
版本 16.5
[发行说明 >](#)

功能完备的集成开发环境 (IDE)，适用于 Android、iOS、Windows、Web 和云 (IDE)

[比较版本 >](#)
[如何脱机安装 >](#)

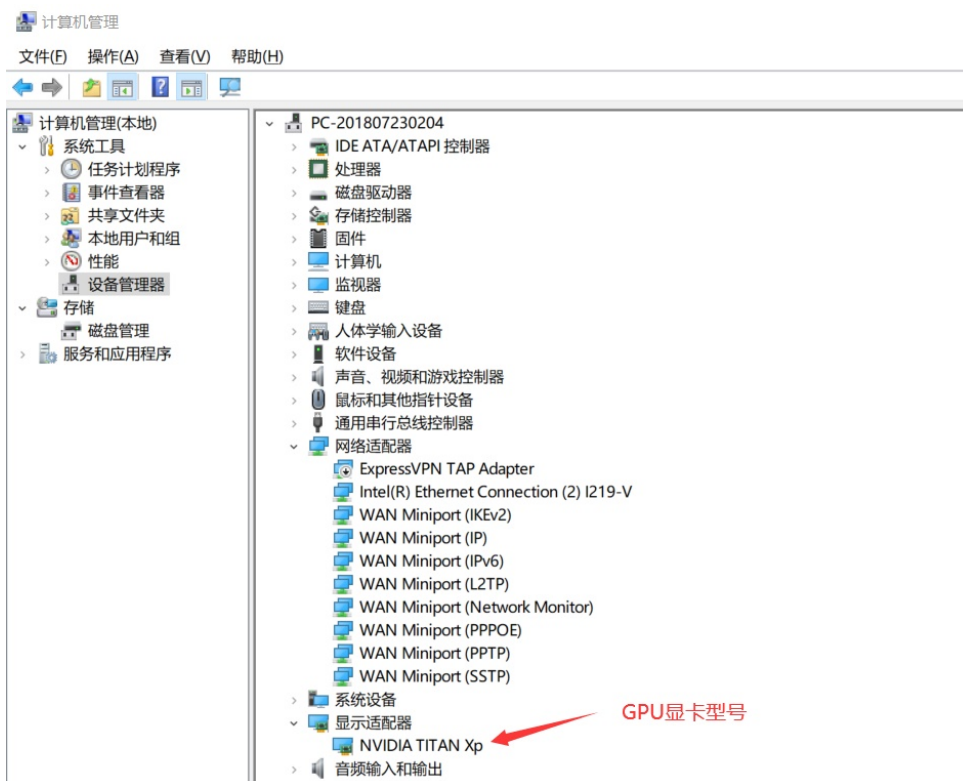
社区
功能强大的 IDE，免费供学生、开放源代码参与者和个人使用
[免费下载](#)
[下载预览版](#)

Professional
最适合小型团队的专业 IDE
[免费试用](#)
[下载预览版](#)

企业
可缩放的端到端解决方案，适用于任何规模的团队
[免费试用](#)
[下载预览版](#)

2) 下载和安装nvidia显卡驱动

首先要在设备管理器中查看你的显卡型号，比如在这里可以看到我的显卡型号为Titan XP。



NVIDIA 驱动下载: <https://www.nvidia.cn/Download/index.aspx?lang=cn>

下载对应你的英伟达显卡驱动。

驱动程序下载

NVIDIA > 驱动程序下载

NVIDIA 助力远程办公

随时随地工作 尽享自由生活

了解详情 >

NVIDIA 驱动程序下载

选项 1: 手动查找适用于我的 NVIDIA 产品的驱动程序。

产品类型: TITAN

产品系列: NVIDIA TITAN Series

产品家族: NVIDIA TITAN Xp

操作系统: Windows 10 64-bit

下载类型: Studio 驱动程序 (SD) ?

语言: Chinese (Simplified)

下载之后就是简单的下一步直到完成。

完成之后, 在cmd中输入执行:

```
nvidia-smi
```

如果有错误:

'nvidia-smi' 不是内部或外部命令, 也不是可运行的程序 或批处理文件。

把C:\Program Files\NVIDIA Corporation\NVSMI添加到环境变量的path中。再重新打开cmd窗口。

如果输出下图所示的显卡信息, 说明你的驱动安装成功。

```

C:\Windows\System32>nvidia-smi
Thu Apr 30 17:22:50 2020

+-----+
| NVIDIA-SMI 441.22                Driver Version: 441.22          CUDA Version: 10.2     |
+-----+-----+
| GPU Name                   TCC/WDDM    Bus-Id        Disp.A   Volatile Uncorr. ECC  |
| Fan  Temp  Perf  Pwr:Usage/Cap     Memory-Usage  GPU-Util  Compute M. |
|====+=====+====+=====+=====+=====+=====+=====+=====+
|   0  TITAN Xp              WDDM       00000000:65:00:0  On      N/A              |
| 23%   26C    P8         11W / 250W      727MiB / 1228MiB      1%          Default  |
+-----+-----+

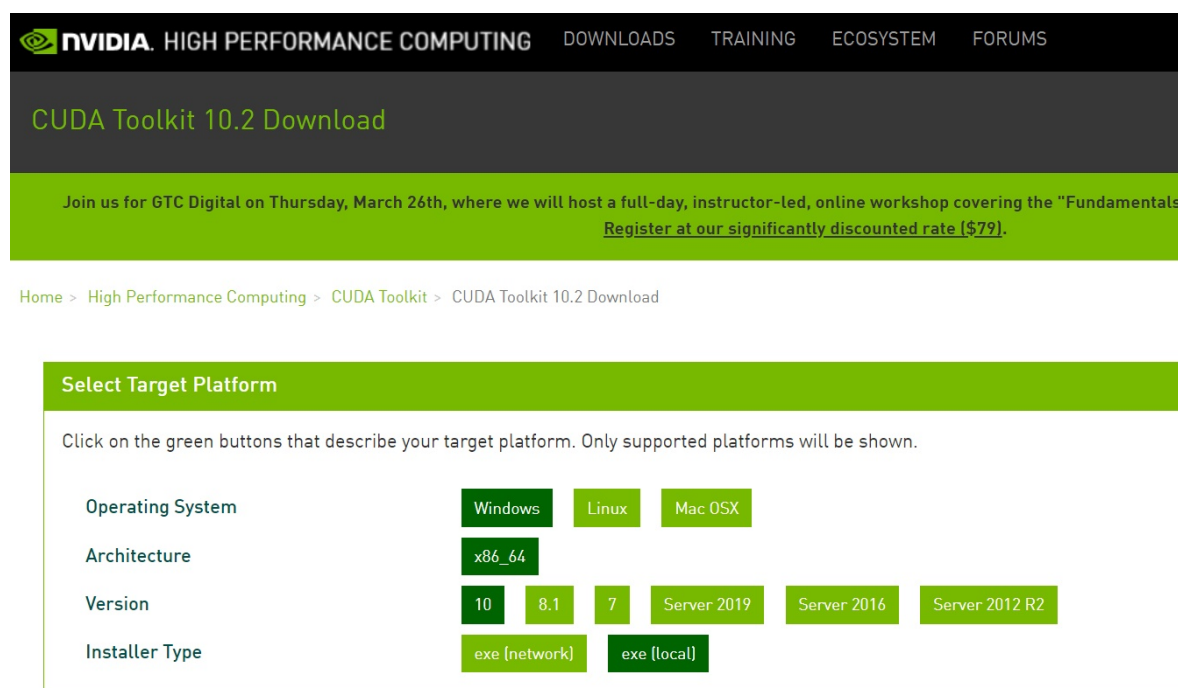
Processes:
+-----+
| GPU  PID  Type  Process name                      GPU Memory |
|=====+=====+=====+=====+=====+=====+=====+=====+=====+
|   0   1160  C+G   Insufficient Permissions           N/A         |
|   0   7316  C+G   C:\Windows\explorer.exe            N/A         |
|   0   7412  C+G   ...t_cw5nlh2txyewy\ShellExperienceHost.exe N/A         |
|   0   7624  C+G   C:\Program Files\Typora\Typora.exe  N/A         |
|   0   8220  C+G   ...dows.Cortana_cw5nlh2txyewy\SearchUI.exe N/A         |
|   0   9064  C+G   ...hell.Experiences.TextInput.InputApp.exe N/A         |
|   0  11856  C+G   ...ta\Roaming\360se6\Application\360se.exe N/A         |
+-----+

```

3) 下载CUDA

CUDA用的是10.2版本

cuda下载链接: https://developer.nvidia.com/cuda-downloads?target_os=Windows&target_arch=x86_64&target_version=10&target_type=exelocal



The image shows the NVIDIA CUDA Toolkit 10.2 Download page. It features a green header with the NVIDIA logo and navigation links. Below the header, there's a section for the CUDA Toolkit 10.2 Download, followed by a promotional banner for GTC Digital. The main content area is titled "Select Target Platform" and includes instructions to click on green buttons describing the target platform. The buttons are organized by Operating System (Windows, Linux, Mac OSX), Architecture (x86_64), Version (10, 8.1, 7, Server 2019, Server 2016, Server 2012 R2), and Installer Type (exe (network), exe (local)).

下载后得到文件: cuda_10.2.89_441.22_win10.exe

4) 下载cuDNN

cuda地址: <https://developer.nvidia.com/cudnn>

需要有账号

cuDNN Download

NVIDIA cuDNN is a GPU-accelerated library of primitives for deep neural networks.

☒ I Agree To the Terms of the [cuDNN Software License Agreement](#)

Note: Please refer to the [Installation Guide](#) for release prerequisites, including supported GPU architectures and compute capabilities, before downloading.

For more information, refer to the cuDNN Developer Guide, Installation Guide and Release Notes on the [Deep Learning SDK Documentation](#) web page.

Download cuDNN v7.6.5 [November 18th, 2019], for CUDA 10.2

Library for Windows, Mac, Linux, Ubuntu and RedHat/Centos(x86_64architecture)

[cuDNN Library for Windows 7](#)

[cuDNN Library for Windows 10](#)

[cuDNN Library for Linux](#)

[cuDNN Runtime Library for Ubuntu18.04 \(Deb\)](#)

[cuDNN Developer Library for Ubuntu18.04 \(Deb\)](#)

[cuDNN Code Samples and User Guide for Ubuntu18.04 \(Deb\)](#)

下载后得到文件: cudnn-10.2-windows10-x64-v7.6.5.32.zip

5) 安装cuda

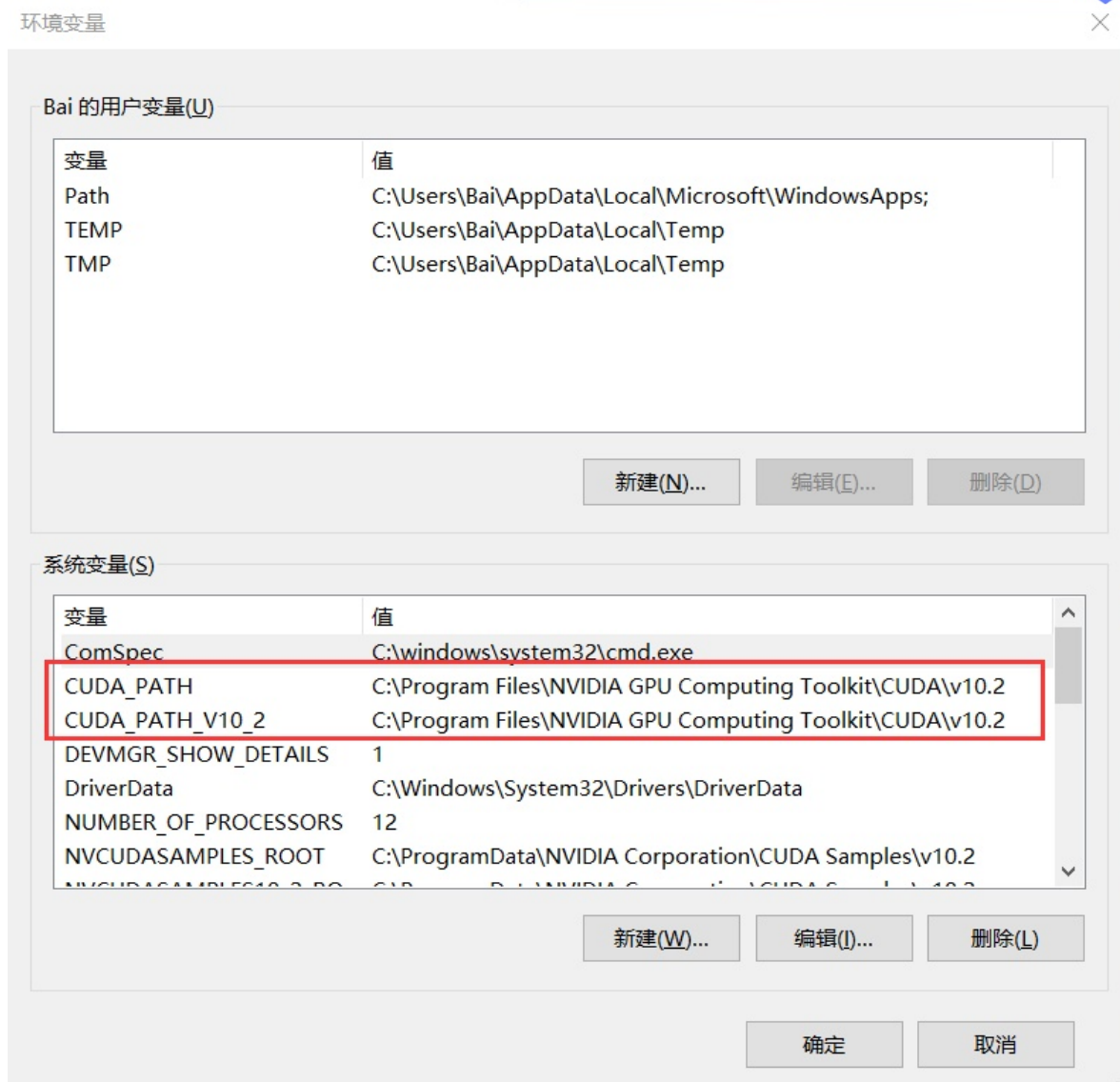
(1) 将cuda运行安装，建议默认路径





安装时可以勾选Visual Studio Integration

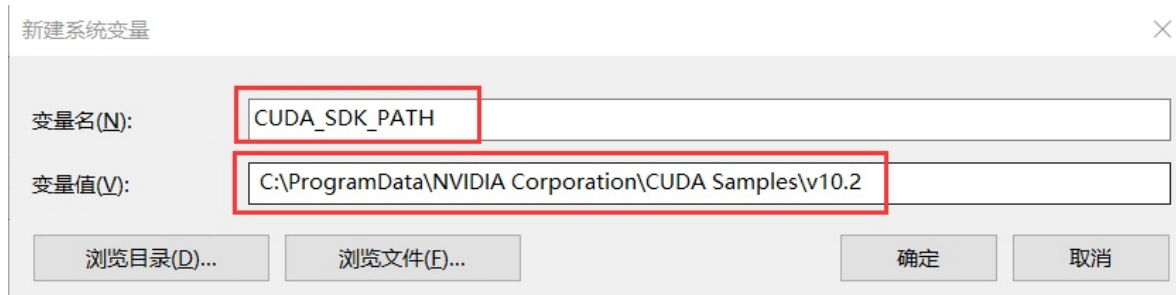
(2) 安装完成后设置环境变量



计算机上点右键，打开属性->高级系统设置->环境变量，可以看到系统中多了CUDA_PATH和CUDA_PATH_V10_2两个环境变量。

接下来，还要在系统中添加以下几个环境变量：这是默认安装位置的路径：C:\ProgramData\NVIDIA Corporation\CUDA Samples\v10.2

CUDA_SDK_PATH = C:\ProgramData\NVIDIA Corporation\CUDA Samples\v10.2
 CUDA_LIB_PATH = %CUDA_PATH%\lib\x64
 CUDA_BIN_PATH = %CUDA_PATH%\bin
 CUDA_SDK_BIN_PATH = %CUDA_SDK_PATH%\bin\win64
 CUDA_SDK_LIB_PATH = %CUDA_SDK_PATH%\common\lib\x64



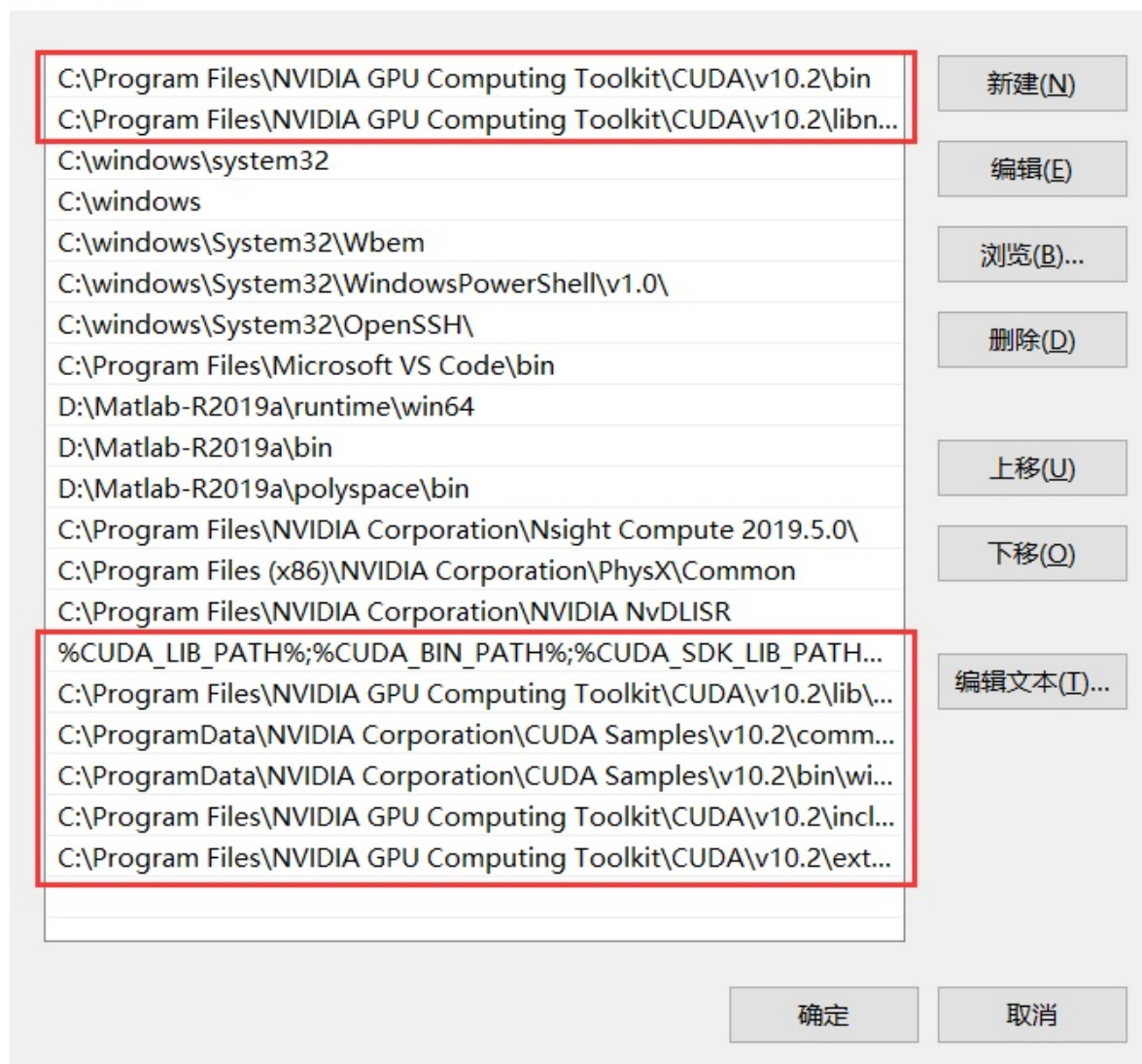
在系统变量 Path 的末尾添加：

%CUDA_LIB_PATH%;%CUDA_BIN_PATH%;%CUDA_SDK_LIB_PATH%;%CUDA_SDK_BIN_PATH%;

再添加如下5条（默认安装路径）：

C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v10.2\lib\x64 C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v10.2\include C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v10.2\extras\CUPTI\lib64 C:\ProgramData\NVIDIA Corporation\CUDA Samples\v10.2\bin\win64 C:\ProgramData\NVIDIA Corporation\CUDA Samples\v10.2\common\lib\x64

编辑环境变量



6) 安装cuDNN

复制cudnn文件

对于cudnn直接将其解开压缩包，然后需要将bin,include,lib中的文件复制粘贴到cuda的文件夹下

C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v10.2

7) CUDA安装测试

最后测试cuda是否配置成功：

打开CMD执行：

```
nvcc -V
```

即可看到cuda的信息

```
C:\Users\Bai>nvcc -V
nvcc: NVIDIA (R) Cuda compiler driver
Copyright (c) 2005-2019 NVIDIA Corporation
Built on Wed_Oct_23_19:32:27_Pacific_Daylight_Time_2019
Cuda compilation tools, release 10.2, V10.2.89
```

8) 安装OpenCV

下载opencv3.4: <https://opencv.org/opencv-3-4.html>

注意: 不要下载最新版本 (不要高于4.0版本) !

接着只需要将其解压缩, 然后配置环境变量就行了。

Download

Documentation

Sources

Win pack

iOS pack

Android pack

运行exe (其实是解压), 将压缩包解压到相应目录, 如: C:\Program Files (x86)\opencv

在系统变量 Path 的末尾添加: C:\Program Files (x86)\opencv\build\x64\vc15\bin

9) 安装Anaconda

Anaconda 是一个用于科学计算的 Python 发行版, 支持 Linux, Mac, Windows, 包含了众多流行的科学计算、数据分析的 Python 包。

1) 下载安装包

Anaconda下载Windows版: <https://www.anaconda.com/products/individual>

2) 然后安装anaconda

3) 添加Anaconda国内镜像配置

清华TUNA提供了 Anaconda 仓库的镜像, 运行以下命令:


```
conda config --add channels  
https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/free/
```

```
conda config --add channels  
https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main/
```

```
conda config --set show_channel_urls yes
```

10) 安装pytorch

注意：安装pytorch 1.7以上的版本 创建虚拟环境，环境名字可自己确定，这里本人使用pytorch1.7作为环境名：

```
conda create -n pytorch1.7 python=3.8
```

安装成功后激活pytorch1.7环境：

```
conda activate pytorch1.7
```

在所创建的pytorch环境下安装pytorch的1.7版本, 执行命令：

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

注意：10.2处应为自己电脑上的cuda版本号

离线安装：

下载网址: <https://mirrors.tuna.tsinghua.edu.cn/anaconda/cloud/pytorch/win-64/>

版本: pytorch-1.7.1-py3.8_cuda102_cudnn7_0.tar.bz2

```
conda install --offline pytorch-1.7.1-py3.8_cuda102_cudnn7_0.tar.bz2
```

2 yolov5项目克隆和安装

1) 克隆yolov5项目

安装Git软件 (<https://git-scm.com/downloads>) , 克隆项目到本地 (如d:)

```
git clone https://github.com/ultralytics/yolov5.git
```

或直接下载V4.0版本的源码

2) 安装所需库

使用清华镜像源：

在yolov5路径下执行：

```
pip install -i https://pypi.tuna.tsinghua.edu.cn/simple -r requirements.txt
```

注意: simple 不能少, 是 https 而不是 http

3) 下载预训练权重文件

下载yolov5s.pt, yolov5m.pt, yolov5l.pt, yolov5x.pt权重文件, 并放置在weights文件夹下

百度网盘下载链接:

链接: <https://pan.baidu.com/s/14O704m9olHx8KK38Pf3RuQ> 提取码: y47n

4) 安装测试

测试图片:

在yolov5路径下执行

```
python detect.py --source ./data/images/ --weights weights/yolov5s.pt --conf 0.4
```

3 TensorRT安装

参考[官网安装教程](#)

<https://docs.nvidia.com/deeplearning/tensorrt/install-guide/index.html>

1) 下载安装包:

1. Go to: <https://developer.nvidia.com/tensorrt>.
2. 点击 **立即下载(Download Now)**
3. 选择合适的TensorRT版本
4. Select the check-box to agree to the license terms.
5. Click the package you want to install. Your download begins.

本人使用的版本: TensorRT-7.0.0.11.Windows10.x86_64.cuda-10.2.cudnn7.6.zip

2) 配置环境变量

1. 新建文件夹, 命名为tensorrt_tar, 然后将下载的压缩文件拷贝进来解压
2. 解压得到TensorRT-7.0.0.11的文件夹, 将里边的lib绝对路径添加到环境变量中, 即

E:\tensorrt_tar\TensorRT-7.0.0.11\lib

3. 将TensorRT解压位置\lib下的dll文件复制到C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v10.2\bin目录下

3) 安装pycuda

如果要使用python接口的tensorrt, 则需要安装pycuda

```
pip install pycuda
```

4) 测试TensorRT示例代码

1. 配置VS2019

用VS2019打开sampleMNIST示例 (E:\tensorrt_tar\TensorRT-7.0.0.11\samples\sampleMNIST) a. 将E:\tensorrt_tar\TensorRT-7.0.0.11\lib加入 项目->属性->VC++目录->可执行文件目录

b.将E:\tensorrt_tar\TensorRT-7.0.0.11\lib加入 VC++目录->库目录

c. 将E:\tensorrt_tar\TensorRT-7.0.0.11\include加入C/C++ --> 常规 --> 附加包含目录

d.将nvinfer.lib、nvinfer_plugin.lib、nvonnxparser.lib和nvparsers.lib加入链接器->输入->附加依赖项

E:\tensorrt_tar\TensorRT-7.0.0.11\lib*.lib

2. 下载pgm文件

到tensorrt目录下的data文件夹找到对应数据集的download_pgms.py，然后运行。运行的时候没输出，等一会看到文件夹下有了x.pgm文件就说明下载好了。即执行：

```
python E:\tensorrt_tar\TensorRT-7.0.0.11\data\mnist\download_pgms.py
```

将下载的x.pgm文件放置到E:\tensorrt_tar\TensorRT-7.0.0.11\data\mnist

备注：PGM 是便携式灰度图像格式(portable graymap file format)，在黑白超声图像系统中经常使用PGM格式的图像。

3. 编译后可执行得到测试结果

4 YOLOv5的TensorRT加速

1) 克隆tensorrtx

```
git clone https://github.com/wang-xinyu/tensorrtx.git
```

2) 下载文件dirent.h

下载文件dirent.h, 下载地址 <https://github.com/tronkko/dirent>

放置到 tensorrtx/include文件夹下，文件夹需新建

3) 生成yolov5s.wts文件

// 下载权重文件yolov5s.pt // 将文件tensorrtx/yolov5/gen_wts.py 复制到ultralytics/yolov5 // ensure the file name is yolov5s.pt and yolov5s.wts in gen_wts.py // go to ultralytics/yolov5 执行

```
python gen_wts.py
```

// a file 'yolov5s.wts' will be generated.

// copy文件'yolov5s.wts' 文件到tensorrtx/yolov5/build目录下

4) 修改CMakeList.txt

修改D:\tensorrtx\yolov5下的CMakeList.txt文件，修改后的CMakeList.txt见网盘。

注意：使用时需要根据自己电脑上的软件位置做相应的修改。

5) 编译tensorrtx/yolov5

(1) 安装cmake

下载地址<https://cmake.org/>

(2) 执行cmake-gui来配置project

(3) 点击 Configure并设置环境

(4) 点击Finish,等待Configure done

(5) 点击Generate并等待Generate done

(6) 点击Open Project

注意：使用Release模式

(7) 生成解决方案

6) 执行TensorRT加速后的yolov5命令

D:\tensorrtx\yolov5\build\Release目录下，执行

```
yolov5.exe -s
```

// serialize model to plan file i.e. yolov5s. engine

```
yolov5.exe -d ../samples
```

// deserialize plan file and run inference, the images in samples will be processed.

7) INT8量化加速

1. 准备校准图片（calibration images），可以从你的训练集随机选择 1000张图片。对于coco, 可以从百度网盘下载校准图片集 [coco_calib.zip](#)

2. unzip it in tensorrtx\yolov5\build

3. set the macro `USE_INT8` in yolov5.cpp

然后，使用camke-gui重新编译

File->Delete Cache

4. serialize the model and test

D:\tensorrtx\yolov5\build\Release目录下，执行

```
yolov5.exe -s
```

```
yolov5.exe -d ../samples
```