

第0步:

先安装anaconda（我是之前安装过anaconda3）

这里是安装GPU类型，不是CPU

第1步:

确定电脑的GPU型号，查看是否支持CUDA

第2步:

查看tensorflow安装版本所要求的CUDA版本和Cmake

我的:

tensorflow1.2

cuda: 8.1.44: https://developer.download.nvidia.cn/compute/cuda/8.0/secure/prod/local_installers/cuda_8.0.44_win10.exe?JF84ksLF8CnxDDCQa-w4HPaWNmDNYPaaNxHmUII9IsDQb-ay5UnrloNCIjmwnSBmm11n9bO6YQhOOINgSbXQyMoq_gEJUKDFPHSVYmMmk2uWQzIfenP1bDwNBv6YM8jtzTbnVf8NLXu3YZA96I-VNUAEEs5_kr1sBvSRYW8ugW6hCQ

Windows下cuda注意选择自定义安装--》只安装cuda，因为典型安装会安装其他组件，造成黑屏。

安装好之后，输入:

```
nvcc -V
```

如果显示如下，则安装成功

```
(tensorflow) C:\Users\Lenovo>nvcc -V
nvcc: NVIDIA (R) Cuda compiler driver
Copyright (c) 2005-2017 NVIDIA Corporation
Built on Fri_Sep__1_21:08:32_Central_Daylight_Time_2017
Cuda compilation tools, release 9.0, V9.0.176
```

cuda: [cudnn-8.0-windows10-x64-v5.1](#)

1. 下载CUDNN后,复制粘贴cuDNN里面的下面三个文件到CUDA的相应同名文件（我使用的）

- cudnn\cuda\bin => CUDA\v10.0\bin
- cudnn\cuda\include => CUDA\v10.0\include
- cudnn\lib\x64 => CUDA\v10.0\lib\x64

cuDNN解压文件		刚刚CUDA安装的路径
D:\cudnn\cudnn-10.0-windows10-x64-v7.3.1.20\cuda\bin	复制cuDNN路径中的文件到CUDA路径中	C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v10.0\bin
D:\cudnn\cudnn-10.0-windows10-x64-v7.3.1.20\cuda\include	复制cuDNN路径中的文件到CUDA路径中	C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v10.0\include
D:\cudnn\cudnn-10.0-windows10-x64-v7.3.1.20\cuda\lib\x64	复制cuDNN路径中的文件到CUDA路径中	C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v10.0\lib\x64

把cuda/bin路径加入系统环境变量中

2. 或者进入该软件包，使用bash

```
cat include/cudnn.h | grep CUDNN_MAJOR -A 2
```

如果显示如下图就说明安装成功

```
liaowen1992@DESKTOP-LFSUJDQ:~/mnt/d/Software/CUDNN 7.0.5/cuda$ cat include/cudnn.h | grep CUDNN_MAJOR -A 2
#define CUDNN_MAJOR 7
#define CUDNN_MINOR 0
#define CUDNN_PATCHLEVEL 5
...
#define CUDNN_VERSION (CUDNN_MAJOR * 1000 + CUDNN_MINOR * 100 + CUDNN_PATCHLEVEL)
#include "driver_types.h"
liaowen1992@DESKTOP-LFSUJDQ:~/mnt/d/Software/CUDNN 7.0.5/cuda$
```

奇怪的是我没有用bazel编译

第3步

anaconda安装tensorflow

在ANACONDA里创建名为TENSORFLOW的环境（你可以叫他任何名字，这里我叫这个环境为TENSORFLOW。（可能不了解python版本，可以或者直接先创建环境，再用conda安装）

```
conda create -n tensorflow pip python=3.6
```

这里pip python=3.6的意思是在名为tensorflow的环境里搭建版本是3.6的python。

2) 发出以下命令以激活 conda 环境:

activate tensorflow

如果想要检测**tensorflow**的确用**gpu**来做运算了, 请用以下脚本测试

```
import tensorflow as tf
# Creates a graph.
a = tf.constant([1.0, 2.0, 3.0, 4.0, 5.0, 6.0], shape=[2, 3], name='a')
b = tf.constant([1.0, 2.0, 3.0, 4.0, 5.0, 6.0], shape=[3, 2], name='b')
c = tf.matmul(a, b)
# Creates a session with log_device_placement set to True.
sess = tf.Session(config=tf.ConfigProto(log_device_placement=True))
# Runs the op.
print(sess.run(c))
```

测试结果如下图



The screenshot shows a terminal window with TensorFlow logs. The logs indicate that the MatMul operation is being executed on the GPU. The output of the operation is a 2x2 matrix: [[22. 28.] [49. 64.]].

图中

device:GPU:0

的意思就是说该运算用到了GPU。

如果想手动声明让电脑用CPU运算, 请参考官方文档

主要参考: <https://zhuanlan.zhihu.com/p/37086409>

