Library installation for PSPnet and Mark-rcnn

Good Luck :)

---

1. CPU version

PLEASE NOTE, THE FOLLOWING IS THE INSTALLATION INSTRUCTION FOR THE CPU VERSION OF MXNET. FOR STUDENTS WITH NVIDIA GPU (GRAPHICS CARD) AND USE WINDOWS, YOU CAN HAVE THE OPTION OF INSTALLING A GPU VERSION OF MXNET (SEE SECTION 2). GPU VERSION IS COMPLETELY OPTIONAL, BUT WILL MAKE THINGS RUN A LOT FASTER.

1.1 Create a new environment

Better to create a new environment for the following installation process

```
conda create -n py36_mxnet_cpu python=3.6
```

1.2 install within the current environment, by typing the following, hitting enter after, and typing y when asked if you wish to proceed the code below:
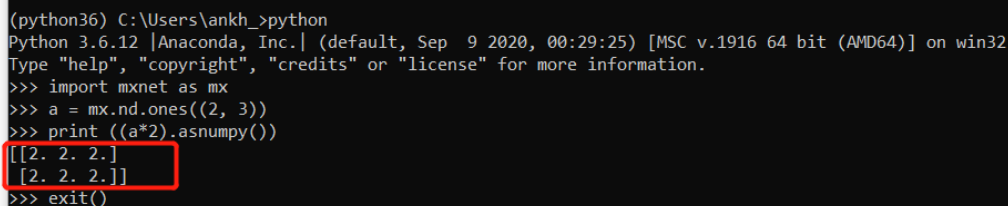
```
pip install mxnet
```

1.3 Next validate your MXNET installation with python. Open the Python terminal by typing and pressing enter on:

```
Python
```

```
import mxnet as mx
```

```
a = mx.nd.ones((2, 3))
```

```
print ((a*2).asnumpy()
```

```
(python36) C:\Users\ankh_>python
Python 3.6.12 |Anaconda, Inc.| (default, Sep  9 2020, 00:29:25) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> import mxnet as mx
>>> a = mx.nd.ones((2, 3))
>>> print ((a*2).asnumpy())
[[2. 2. 2.]
 [2. 2. 2.]]
>>> exit()
```

It should look as above. If you can print successfully, you have successfully installed mxnet. You can then close python with the command below

```
exit()
```

1.4 If you have experienced any red errors while running this code or do in future steps, please download the latest version of Microsoft Visual Studio C++ from the following page and install it,

before re-running the code above from step 3 onwards: https://visualstudio.microsoft.com/visual-cpp-build-tools/.

1.5 Next install gluoncv with this command and hitting enter:

```
pip install gluoncv
```

1.6 You can use "conda list" to see all the library you have installed successfully then then use python to validate again:

```
python
```

```
import gluoncv
```

```
import mxnet as mx
```

```
a = mx.nd.ones((2, 3))
```

```
b = a*2 +1
```

```
b.asnumpy()
```

then to close:

```
exit()
```

2. GPU version

If you have a NVIDIA GPU (Graphics Card) and use Windows, you can install a GPU version. **This entire step is completely optional** but will make things run faster (about 20 times faster).

Note: Mxnet GPU version is only available for Windows users with NVIDIA GPU card, not available for Mac users or Windows users with AMD GPU card. If you don't have a NVIDIA GPU card this guide will not work for you. Please use the standard installation instructions. (you can find further installation instruction here: https://mxnet.apache.org/versions/1.4.1/install/windows_setup.html)
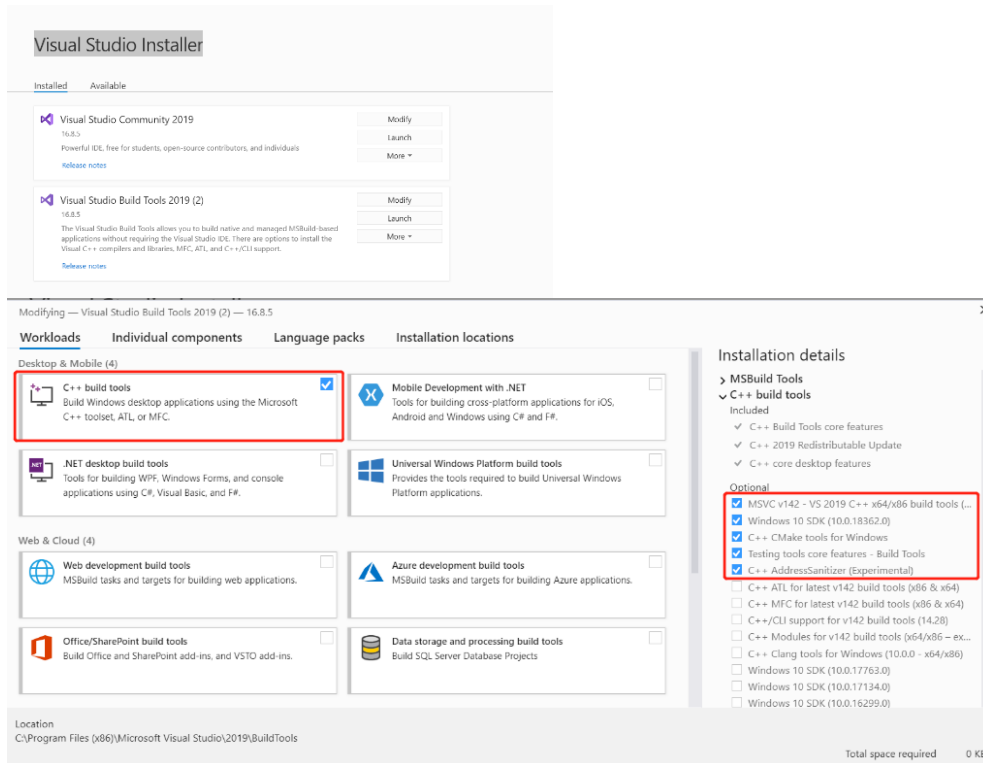
2.1 check your GPU:

Computer desktop—— right click or:
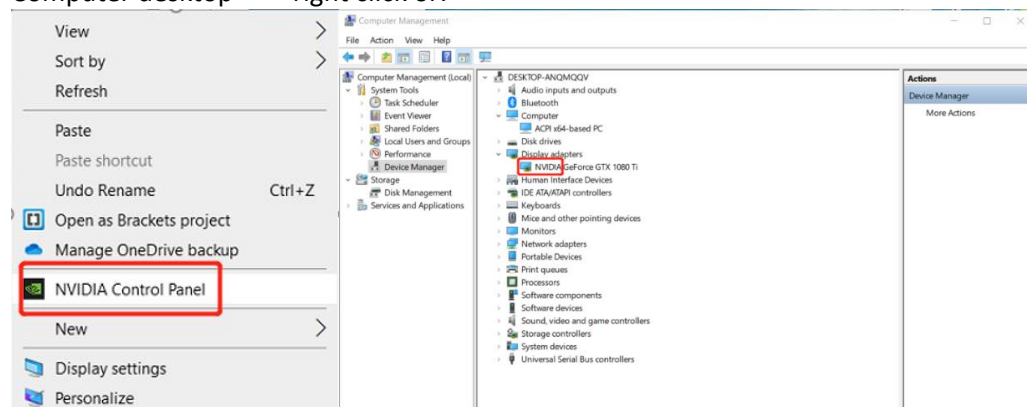
2.2 Install Microsoft Visual Studio 2019, 2017 or 2015. community version is fine.
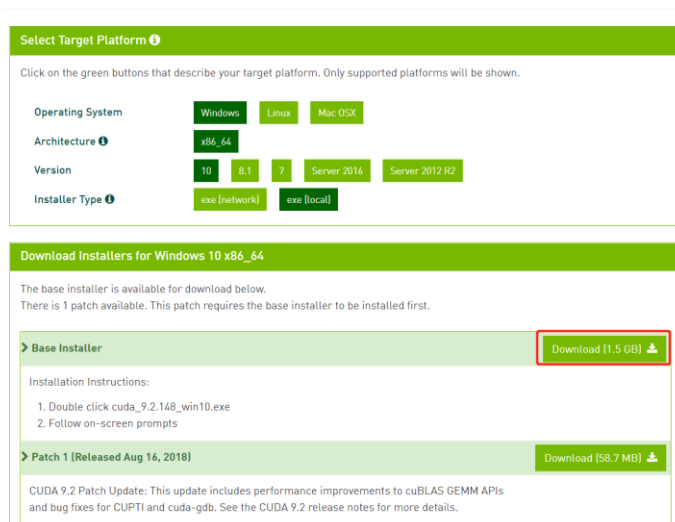https://visualstudio.microsoft.com/zh-hans/downloads/

2.3 Install visual Studio Build Tools

2.4 install CUDA:

 https://developer.nvidia.com/cuda-toolkit-archive

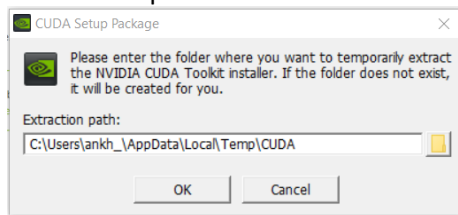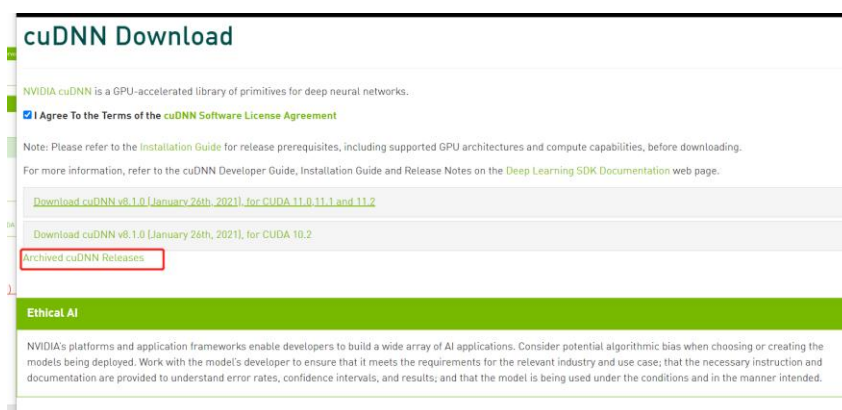here I choose 9.2 version (10 or 10.1 is also fine)

The default path is fine:

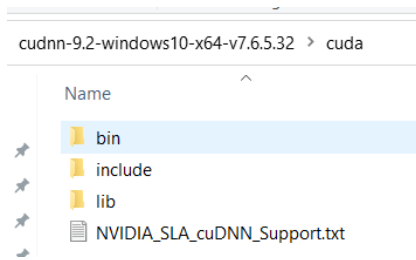

2.5 install CUDNN:

Need to register on the website at the first time

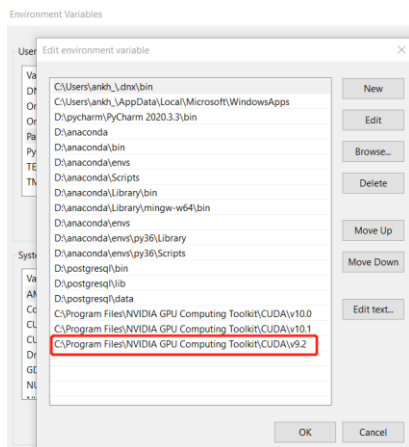Must choose a corresponding version for CUDA

Here I chose CUDNN for CUDA 9.2

Copy these three documents into the installation path of CUDA (usually C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\"VERSION NUMBER")

2.6 add "" into Environment Variables

2.7 Create a new environment

Better create a new environment for the following installation process

```
conda create -n py36_mxnet_gpu python=3.6
```

2.8 install mxnet in anaconda prompt:

```
pip install mxnet-cu92
```

note: -cu92 is the version corresponding to your CUDA version.

If you installed CUDA10.0, it should be pip install mxnet-cu100

If you installed CUDA10.1, it should be pip install mxnet-cu101

```
(python36) C:\Users\ankh_>pip install mxnet-cu92
Collecting mxnet-cu92
  Using cached mxnet_cu92-1.5.0-py2.py3-none-win_amd64.whl (340.5 MB)
Collecting graphviz<0.9.0,>=0.8.1
  Using cached graphviz-0.8.4-py2.py3-none-any.whl (16 kB)
Collecting numpy<1.17.0,>=1.8.2
  Using cached numpy-1.16.6-cp36-cp36m-win_amd64.whl (11.9 MB)
Collecting requests<2.19.0,>=2.18.4
  Using cached requests-2.18.4-py2.py3-none-any.whl (88 kB)
Requirement already satisfied: certifi>=2017.4.17 in d:\anaconda\envs\python36\lib\site-packages (from requests<2.19.0
=2.18.4->mxnet-cu92) (2020.12.5)
Collecting chardet<3.1.0,>=3.0.2
  Using cached chardet-3.0.4-py2.py3-none-any.whl (133 kB)
Collecting idna<2.7,>=2.5
  Using cached idna-2.6-py2.py3-none-any.whl (56 kB)
Collecting urllib3<1.23,>=1.21.1
  Using cached urllib3-1.22-py2.py3-none-any.whl (132 kB)
Installing collected packages: urllib3, idna, chardet, requests, numpy, graphviz, mxnet-cu92
Successfully installed chardet-3.0.4 graphviz-0.8.4 idna-2.6 mxnet-cu92-1.5.0 numpy-1.16.6 requests-2.18.4 urllib3-1.2

(python36) C:\Users\ankh_>
```

2.9 validate your MXNET installation with python:

```
(python36) C:\Users\ankh_>python
Python 3.6.12 |Anaconda, Inc.| (default, Sep  9 2020, 00:29:25) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> import mxnet as mx
>>> a = mx.nd.ones((2,3),mx.gpu())
>>> b = a*2 +1
>>> b.asnumpy()
array([[3., 3., 3.],
       [3., 3., 3.]], dtype=float32)
>>> exit()

(python36) C:\Users\ankh_>
```

import mxnet as mx
a = mx.nd.ones((2,3),mx.gpu())
b = a*2 +1
b.asnumpy()

then type and enter the following if it was a success to close python:

exit()

2.10 install gluoncv

pip install gluoncv

You can use "conda list" to see all the library you have installed successfully then then use python to validate again:

python

import gluoncv

import mxnet as mx

a = mx.nd.ones((2,3),mx.gpu())

b = a*2 +1

b.asnumpy()

exit()

Here are some of the combinations which can install mxnet GPU version successfully according to my experiment:

(1) window10, python3.6, GPU: NVIDIA GeForce RTX3080,   CUDA: 10.1;   cuDNN v8.0.5 (November 9th, 2020), for CUDA 10.1； pip install mxnet-cu101

(1) Windows 10 , python3.6 ;   GPU: NVIDIA GeForce RTX2070;   CUDA: 10.0;   cuDNN v7.6.5 (November 5th, 2019), for CUDA 10.0; pip install mxnet-cu100

(2) Window10; python 3.6 ;   GPU: GeForce GTX 1080 TI ;   CUDA: 10.1;   cuDNN v8.0.5 (November 9th, 2020), for CUDA 10.1;   pip install mxnet-cu101

(3) Window10; python 3.6 ;   GPU: GeForce GTX 1070 ;   CUDA: 9.2;   cuDNN v7.6.5 (November 5th, 2019), for CUDA 9.2;   pip install mxnet-cu92