

For those wishing to use TensorFlow and Keras on their own GPU (provided it is an NVIDIA GPU), at their own risk:

- After installing anaconda with python 3.x (<https://www.anaconda.com/distribution/>);
- Since the last version of Anaconda supports python 3.7 it is necessary to downgrade to python 3.6 (I'm not sure if TensorFlow works with 3.7 version, free to test it):
 - open anaconda shell (as admin if you work with windows) and run the following command to install Python 3.6 in the root environment (<https://docs.anaconda.com/anaconda/user-guide/faq/#anaconda-faq-35>)
conda install python = 3.6
or
sudo conda install python = 3.6
- Download and install the CUDA nvidia toolkit, I strongly suggest version 9.0 since I'm sure it works (<https://developer.nvidia.com/cuda-90-download-archive>);
- download and install the CUDNN (<https://developer.nvidia.com/rdp/cudnn-download>), you must login, also with a google account;
- Add path environment variables:
...\\NVIDIA GPU Computing Toolkit\\CUDA\\v9.0
...\\NVIDIA GPU Computing Toolkit\\CUDA\\v9.0\\extras\\CUPTI\\libx64
...\\NVIDIA GPU Computing Toolkit\\CUDA\\v9.0\\bin
...\\NVIDIA GPU Computing Toolkit\\CUDA\\v9.0\\libnvvp
- At this point you need to install TensorFlow and Keras, simply run these commands in the anaconda shell (as admin if you work with windows):
conda install -c anaconda tensorflow-gpu
conda install -c conda-forge keras
if you use linux or mac don't forget to add sudo before the commands:
sudo conda install -c anaconda tensorflow-gpu
sudo conda install -c conda-forge keras
- try it:
 - open the anaconda shell;
 - run python (you should see the version);
 - type:
import keras
if you see "Using TensorFlow backend" is a good sign;
 - type:
from tensorflow.python.client import device_lib
print(device_lib.list_local_devices())
if you read your GPU, you have correctly installed tensorflow and keras