

summary

简单介绍一下

论文简单过了一遍，主要的时间都花在配置环境上了，过程中踩的坑以及解决方案总结如下：

GitHub上的代码作为算法的演示以 `ipynb` 文件的形式给出，跑起来还是相对容易的，并且在repo的 `README.md` 文件中作者很贴心地给出了三种运行方式：

- anaconda (conda)

```
Here is the list of libraries you need to install to execute the code:
```

- python = 3.6
- `pytorch` = 0.4
- numpy
- scipy
- matplotlib
- scikit-image
- jupyter

```
All of them can be installed via conda (anaconda), e.g.
```

```
conda install jupyter
```

```
or create an conda env with all dependencies via environment file
```

```
conda env create -f environment.yml
```

- docker

```
Docker image
```

Alternatively, you can use a Docker image that exposes a Jupyter Notebook with all required dependencies. To build this image ensure you have both `docker` and `nvidia-docker` installed, then run

```
nvidia-docker build -t deep-image-prior .
```

After the build you can start the container as

```
nvidia-docker run --rm -it --ipc=host -p 8888:8888 deep-image-prior
```

you will be provided an URL through which you can connect to the Jupyter notebook.

- google colab

```
Google Colab
```

To run it using Google Colab, click [here](#) and select the notebook to run. Remember to uncomment the first cell to clone the repository into colab's environment.

conda和colab的方式我都试了一下，具体的一个一个说。

anaconda

一些anaconda的介绍和使用也可以提两句，最重要的操作莫过于换源（我一般用[tuna镜像](#)）和环境管理，查看文档[点这里](#)，也可以[点这里](#)看别人的总结

实际上用的是conda搭建环境，当然用pip也可以，区别在于conda会自动解决依赖，pip要手动解决（我记得pip能解决一些）

说回正题，Python（解释器）、numpy、jupyter之流安装起来都非常容易，搭建环境的核心在于安装pytorch，官网上给了详细的[文档](#)，当然为了下载速度还需要[国内镜像](#)

第一个问题

第一个问题是尽管pytorch有 `cpu-only` 和 `gpu` 的版本，但是要跑起来这里的代码只能用 `gpu` 版本，作者使用的环境是 `pytorch0.4 + cuda9.0`，但是cuda9.0和我的显卡不兼容，所以最后我选了最新的 `pytorch1.7(stable)+cuda11.0+cuDNN11.2`（Python版本还是3.6），幸运的是代码也能运行

在安装cuda的时候一定要注意版本，首先要满足显卡兼容性（按照经验是不能用早于显卡出厂时间的版本），然后要满足框架（比如pytorch最高支持cuda11.0，而cuda的最新版本号是11.2），这种对应关系在pytorch等的官网上可以查到，当然还可以用百度搜索，当然还可以看镜像站上具体包的包名

用 `nvidia-smi` 检查显卡驱动（这里显示的cuda版本是最新版，不用管它）：

```
Windows PowerShell
PS C:\Users\linxinhui> nvidia-smi
Mon Feb  8 08:56:08 2021
```

NVIDIA-SMI 461.40		Driver Version: 461.40		CUDA Version: 11.2	
GPU	Name	TCC/WDDM	Bus-Id	Disp.A	Volatile Uncorr. ECC
Fan	Temp	Perf	Pwr:Usage/Cap	Memory-Usage	GPU-Util Compute M. MIG M.
0	GeForce RTX 2060	WDDM	00000000:01:00:0	Off	N/A
N/A	39C	P8	10W / N/A	164MiB / 6144MiB	0% Default N/A

```
Processes:
GPU  GI  CI  PID  Type  Process name  GPU Memory Usage
   ID  ID
No running processes found
PS C:\Users\linxinhui>
```

用 `nvcc -V` 检查cuda版本：

```
Windows PowerShell
PS C:\Users\linxinhui> nvcc -V
nvcc: NVIDIA (R) Cuda compiler driver
Copyright (c) 2005-2020 NVIDIA Corporation
Built on Thu_Jun_11_22:26:48_Pacific_Daylight_Time_2020
Cuda compilation tools, release 11.0, V11.0.194
Build cuda_11.0_bu.relgpu_drvr445TC445_37.28540450_0
PS C:\Users\linxinhui>
```

检查cuda能否与显卡建立连接的最严谨的方法是运行 `devicequery.exe` 和 ``，这也是nvidia官网guide中的方法

还有通过pytorch检查的，见下。

第二个问题

解决cuda和cuDNN的问题之后就可以用conda配置环境了，conda有导出环境的功能，但实质上是一个“配置单”，该下载的包还是要重新下载，而且由于上述问题，pytorch和cuda不会匹配，所以无视作者给的 `environment.yml`，手动配置conda，首先创建环境：

```
C:\Windows\system32\cmd.exe - conda create --name deep-image-prior python=3.6
C:\Users\linxinhui>conda create --name deep-image-prior python=3.6
Solving environment: done

==> WARNING: A newer version of conda exists. <=
  current version: 4.5.11
  latest version: 4.5.2
Please update conda by running

  $ conda update --base --c defaults conda

# Package Plan #
  environment location: C:\Users\linxinhui\Anaconda3\envs\deep-image-prior
  added / updated specs:
    - python=3.6

The following packages will be downloaded:

  package                                     build                                1.3 MB defaults
  sqlite=3.33.0                               h2a8688b_0                          1.3 MB defaults
  certifi=2020.12.5                           py36ha9532_0                        144 KB defaults
  pip=20.3.3                                  py36ha9532_0                         2.1 MB defaults
  wc2015_runtime=14.27.29016                  h5a58377_2                          2.2 MB defaults
  wheel=0.36.2                                 py36ha9532_0                         3.1 MB defaults
  python=3.6.12                                h350082f_2                          17.8 MB defaults
  setuptools=52.0.0                           py36ha9532_0                         813 KB defaults
  wincertstore=0.2                             py36h7e430ca_0                      13 KB defaults
  libffi=3.11                                 h2828b97_4                          152 KB defaults
  vc=14.2                                      h2ff491_1                             8 KB defaults

Total: 24.5 MB

The following NEW packages will be INSTALLED:

  certifi: 2020.12.5-py36ha9532_0 defaults
  pip: 20.3.3-py36ha9532_0 defaults
  python: 3.6.12-h350082f_2 defaults
  setuptools: 52.0.0-py36ha9532_0 defaults
  sqlite: 3.33.0-h2a8688b_0 defaults
  wc: 14.2-h2ff491_1 defaults
  wc2015_runtime: 14.27.29016-h5a58377_2 defaults
  wheel: 0.36.2-py36ha9532_0 defaults
  wincertstore: 0.2-py36h7e430ca_0 defaults
  libffi: 3.11-h2828b97_4 defaults

Proceed ([y]/n)?
```

conda会自动解决包依赖的问题，打印所有需要下载的包。

第二个问题就是conda下载速度慢的问题，当下载量很大时，很有可能出现无法下载的情况：

```
C:\Windows\system32\cmd.exe
C:\Users\linzhihui>conda env list

# conda environments:
#
base                    * C:\Users\linzhihui\Anaconda3

C:\Users\linzhihui>conda create --name deep-image-prior python=3.6
Collecting package metadata (repodata.json): done
Solving environment: done

## Package Plan ##

environment location: C:\Users\linzhihui\Anaconda3\envs\deep-image-prior
added / updated specs:
- python=3.6

The following packages will be downloaded:



| package           | build          | size    | channel  |
|-------------------|----------------|---------|----------|
| certifi-2020.12.5 | pp36ha96532_0  | 144 KB  | defaults |
| pip-20.3.0        | pp36ha96532_0  | 2.1 MB  | defaults |
| python-3.6.12     | h55008f2_2     | 17.8 MB | defaults |
| setuptools-52.0.0 | pp36ha96532_0  | 875 KB  | defaults |
| vc-14.2           | h21ff451_1     | 3 KB    | defaults |
| wheel-0.36.2      | py36h1ba10_0   | 31 KB   | defaults |
| wincertstore-0.2  | py36h7fa50ca_0 | 12 KB   | defaults |
| zlib-1.2.11       | h036d9f7_4     | 152 KB  | defaults |
| Total:            |                | 21.0 MB |          |



The following NEW packages will be INSTALLED:

certifi      pkgs/main/win-64::certifi-2020.12.5-py36ha96532_0
pip          pkgs/main/win-64::pip-20.3.0-py36ha96532_0
python       pkgs/main/win-64::python-3.6.12-h55008f2_2
setuptools   pkgs/main/win-64::setuptools-52.0.0-py36ha96532_0
vc           pkgs/main/win-64::vc14-h21ff451_1
vcruntime    pkgs/main/win-64::vcruntime-14.27.29010-h558377_2
wheel        pkgs/main/conda-forge::wheel-0.36.2-py36h1ba10_0
wincertstore pkgs/main/win-64::wincertstore-0.2-py36h7fa50ca_0
zlib         pkgs/main/win-64::zlib-1.2.11-h036d9f7_4

Proceed ([y]/n)? y

Downloading and Extracting Packages
zlib-1.2.11                               152 KB |#####| 100%
wheel-0.36.2                             31 KB |#####| 100%
wincertstore-0.2                         12 KB |#####| 100%
setuptools-52.0.0                       875 KB |#####| 100%
vc-14.2                                 3 KB |#####| 100%
python-3.6.12                           17.8 MB |#####| 88%
pip-20.3.0                              2.1 MB |#####| 82%
certifi-2020.12.5                       144 KB |#####| 100%

CondaError: Downloaded bytes did not match Content-Length
url: http://mirrors.tuna.tsinghua.edu.cn/anaconda/pkg/win-64/python-3.6.12-h55008f2_2.tar.bz2
target_path: C:\Users\linzhihui\Anaconda3\pkgs\python-3.6.12-h55008f2_2.tar.bz2
Content-Length: 18647292
downloaded bytes: 10363922

C:\Users\linzhihui>
```

conda没有断点续传功能，这时候需要手动删除已下载的不完整的包（必要时重启），然后重试下载：

```
C:\Windows\system32\cmd.exe
C:\Users\linzhihui>conda create --name deep-image-prior python=3.6

setuptools pkgs/main/win-64::setuptools-52.0.0-py36ha96532_0
zlib        pkgs/main/win-64::zlib-1.2.11-h036d9f7_4
vc          pkgs/main/win-64::vc14-h21ff451_1
vcruntime   pkgs/main/win-64::vcruntime-14.27.29010-h558377_2
wheel       pkgs/main/conda-forge::wheel-0.36.2-py36h1ba10_0
wincertstore pkgs/main/win-64::wincertstore-0.2-py36h7fa50ca_0
zlib        pkgs/main/win-64::zlib-1.2.11-h036d9f7_4

Proceed ([y]/n)? y

Downloading and Extracting Packages
zlib-1.2.11                               152 KB |#####| 100%
wheel-0.36.2                             31 KB |#####| 100%
wincertstore-0.2                         12 KB |#####| 100%
setuptools-52.0.0                       875 KB |#####| 100%
vc-14.2                                 3 KB |#####| 100%
python-3.6.12                           17.8 MB |#####| 82%
pip-20.3.0                              2.1 MB |#####| 82%
certifi-2020.12.5                       144 KB |#####| 100%

CondaError: Downloaded bytes did not match Content-Length
url: https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkg/win-64/python-3.6.12-h55008f2_2.tar.bz2
target_path: C:\Users\linzhihui\Anaconda3\pkgs\python-3.6.12-h55008f2_2.tar.bz2
Content-Length: 18647292
downloaded bytes: 10363922

C:\Users\linzhihui>conda create --name deep-image-prior python=3.6
Collecting package metadata (repodata.json): done
Solving environment: done
WARNING: conda.gateways.disk.delete.unlink or rename_to trash(140): Could not remove or rename C:\Users\linzhihui\Anaconda3\pkgs\python-3.6.12-h55008f2_2\tar.bz2. Please remove this file manually (you may need to reboot to free file handles)
WARNING: conda.gateways.disk.delete.unlink or rename_to trash(140): Could not remove or rename C:\Users\linzhihui\Anaconda3\pkgs\python-3.6.12-h55008f2_2\tar.bz2. Please remove this file manually (you may need to reboot to free file handles)

## Package Plan ##

environment location: C:\Users\linzhihui\Anaconda3\envs\deep-image-prior
added / updated specs:
- python=3.6

The following packages will be downloaded:



| package       | build      | size    | channel  |
|---------------|------------|---------|----------|
| python-3.6.12 | h55008f2_2 | 17.8 MB | defaults |
| Total:        |            | 17.8 MB |          |



The following NEW packages will be INSTALLED:

certifi      pkgs/main/win-64::certifi-2020.12.5-py36ha96532_0
pip          pkgs/main/win-64::pip-20.3.0-py36ha96532_0
python       pkgs/main/win-64::python-3.6.12-h55008f2_2
setuptools   pkgs/main/win-64::setuptools-52.0.0-py36ha96532_0
vc           pkgs/main/win-64::vc14-h21ff451_1
vcruntime    pkgs/main/win-64::vcruntime-14.27.29010-h558377_2
wheel        pkgs/main/conda-forge::wheel-0.36.2-py36h1ba10_0
wincertstore pkgs/main/win-64::wincertstore-0.2-py36h7fa50ca_0
zlib         pkgs/main/win-64::zlib-1.2.11-h036d9f7_4

Proceed ([y]/n)?
```

```
C:\Windows\system32\cmd.exe - conda create --name deep-image-prior python=3.6
added / updated specs:
- python=3.6

The following packages will be downloaded:

```

package	build	size	defaults
python-3.6.12	h550b2f2_2	17.8 MB	
Total:		17.8 MB	

```

The following NEW packages will be INSTALLED:

```

package	build	size	defaults
certifi	pkgs/main/win-64::certifi-2020.12.5-py36ha05532_0		
pip	pkgs/main/win-64::pip-20.3.2-py36ha05532_0		
python	pkgs/main/win-64::python-3.6.12-h550b2f2_2		
setuptools	pkgs/main/win-64::setuptools-42.0.0-py36ha05532_0		
sqlite	pkgs/main/win-64::sqlite-3.33.0-ha28b8b_0		
vc	pkgs/main/win-64::vc-14.2-h21ff481_1		
vs2015_runtime	pkgs/main/win-64::vs2015_runtime-14.27.26010-h5e8377_2		
wheel	pkgs/main/win-64::wheel-0.36.2-pyhd3eb1b0_0		
wincertstore	pkgs/main/win-64::wincertstore-0.2-py36h7e50ca_0		
zlib	pkgs/main/win-64::zlib-1.2.11-h63d267_4		

```

Proceed ([y]/n) ?
CondaSystemExit: Exiting.

C:\Users\linxinhu\conda create --name deep-image-prior python=3.6
Collecting package metadata (repodata.json): done
Solving environment: done

## Package Plan ##

```

package	build	size	defaults
python-3.6.12	h550b2f2_2	17.8 MB	
Total:		17.8 MB	

```

The following NEW packages will be INSTALLED:

```

package	build	size	defaults
certifi	pkgs/main/win-64::certifi-2020.12.5-py36ha05532_0		
pip	pkgs/main/win-64::pip-20.3.2-py36ha05532_0		
python	pkgs/main/win-64::python-3.6.12-h550b2f2_2		
setuptools	pkgs/main/win-64::setuptools-42.0.0-py36ha05532_0		
sqlite	pkgs/main/win-64::sqlite-3.33.0-ha28b8b_0		
vc	pkgs/main/win-64::vc-14.2-h21ff481_1		
vs2015_runtime	pkgs/main/win-64::vs2015_runtime-14.27.26010-h5e8377_2		
wheel	pkgs/main/win-64::wheel-0.36.2-pyhd3eb1b0_0		
wincertstore	pkgs/main/win-64::wincertstore-0.2-py36h7e50ca_0		
zlib	pkgs/main/win-64::zlib-1.2.11-h63d267_4		

```

Proceed ([y]/n) ?
```

下载速度慢毫无疑问是网络的问题，这时候当然首先考虑换源，然而操作的过程中发现即使换了国内镜像，下载速度还是很慢，大文件几乎成功不了，换源之后再挂上VPN效果不错。

当然这并不能完全解决问题并不能解决问题，要完全解决这个问题，可以采取如下方案：

- 修改conda用于下载文件的函数的所在文件
 - 用 `python -c "from conda.gateways.connection import download; print(download.__file__)"` 找到目标文件
 - 找到 `download()` 函数，该函数的开始处加上如下代码：

```
from ...base.constants import CONDA_TEMP_EXTENSION

tmp_file_path = target_full_path + CONDA_TEMP_EXTENSION
if exists(tmp_file_path):
    print("\n[Download patch] file exists: %s", tmp_file_path)

    checksum_ok = True

    if sha256 or md5:
        builder = hashlib.new("sha256" if sha256 else "md5")
        checksum = sha256 if sha256 else md5

        with open(tmp_file_path, 'rb') as f:
            for chunk in iter(lambda: f.read(4096), b''):
                builder.update(chunk)

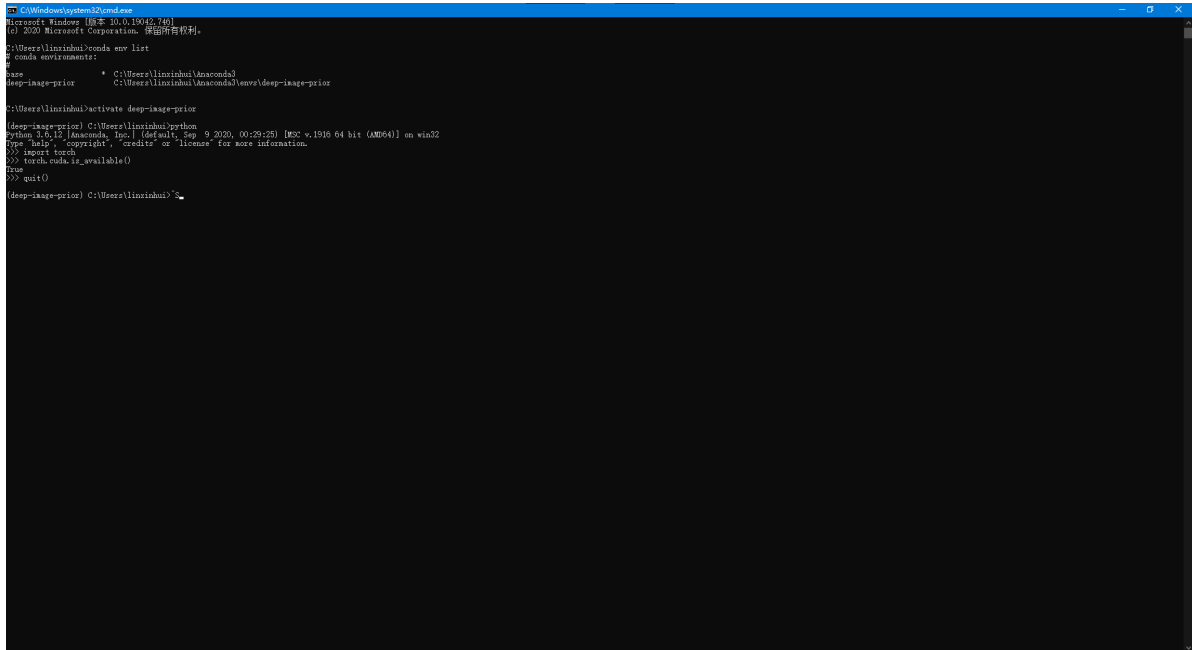
        actual_checksum = builder.hexdigest()
        if actual_checksum != checksum:
            print("\n[Download patch] cached file checksum mismatch: %s (%s
            != %s)",
                checksum_type, actual_checksum, checksum)
            checksum_ok = actual_checksum == checksum

    if checksum_ok:
        from ..disk.update import backoff_rename
        backoff_rename(tmp_file_path, target_full_path, True)
        if progress_update_callback:
            progress_update_callback(1.0)
```

```
print("\n[Download patch] using cached file instead of download",
target_full_path)
return
```

- 从镜像站手动下载需要的包，放到conda的pkgs目录下
- （重启电脑）
- 重复下载过程，应该出现 [Download patch] using cached file instead of download 等字样，下载进度条直接跑到底，稍等片刻即可

最后检查安装情况：



```
C:\Windows\system32\cmd.exe
Microsoft Windows [版本 10.0.19042.740]
(c) 2020 Microsoft Corporation。保留所有权利。

C:\Users\linzhihui>conda env list

# conda environments:
#
base                    * C:\Users\linzhihui\Anaconda3
deep-image-prior        C:\Users\linzhihui\Anaconda3\envs\deep-image-prior

C:\Users\linzhihui>activate deep-image-prior

(deep-image-prior) C:\Users\linzhihui>python
Python 3.6.12 [Anaconda, Inc.] (default, Sep 9 2020, 00:29:25) [MSC v.1910 64 bit (AMD64)] on win32
Type help() for copyright, credit or 'license()' for more information.
>>> import torch
>>> torch.cuda.is_available()
True
>>> quit()

(deep-image-prior) C:\Users\linzhihui>
```

这种方法本质上是离线安装，只适用于下载的包很少的情况（一般来说可以在线下载体积较小的包，像pytorch这样1GB多的包应该是少数情况）。

docker

暂时没用过，据说可以满足多下切换和直接调GPU资源的需求。

Google Colab

要搭梯子。

这大概是跑这篇论文的代码的最简单的方式，作者已经把环境都配置好了（查了一下git log，后期不少提交都是关于colab的，可能作者也觉得环境不好整[doge]），注意看注释（要先把相关的仓库clone到colab环境中）。

据说colab的IO瓶颈是个大问题，当然网速也是，我想如果后期需要类似的平台训练可以用百度paddle之类国内平台。

还有一些问题

- 论文不能完全看懂
这是知识储备的问题，数字图像处理、机器学习、深度学习、线性代数概率论等基础该学还得学
- 科学上网的问题
科学上网快成刚需了，ExpressVPN，NordVPN这些服务商提供的VPN服务支持多人共用一个账号，我们三个人分摊一下差不多每个月每人二十到三十

我好奇这笔支出能报销么[doge]

- 环境切换和算力的问题

需要在不同的环境（操作系统）下切换，包括Windows, Ubuntu16, Ubuntu18等，虚拟机和WSL（WSL1和WSL2中的一个甚至用不了GPU）的方案根据测试GPU性能的折损挺厉害的，双系统一次只能跑一个，也不是很好的方案，docker据说可以直接调用GPU，这个有待研究，比较靠谱的方法可能是用colab这样的平台，或者租服务器自己搭平台（费用挺高，并且我们没有相关的经验）