https://neurodata.io/help/visualization/

需要端口转发

https://blog.csdn.net/sdnuwjw/article/details/103483974

配web server环境很麻烦，npm i配不好，换了好几个镜像都不行。

问了一下周洋同学部署过这个了，只需要看他写的文档。

但是也还是要用npm安装http-server，可能是防火墙导致npm install不了

使用了代理

https://stackoverflow.com/questions/28056051/running-npm-behind-a-corporate-firewall-what-do-i-need-to-tell-the-security-tea

```
npm config set proxy http://内网里我的电脑的ip:http代理端口
```

代理端口在v2ray里面

之后不用镜像也可以安装了

启动http server cors解决跨域问题；要用外网开放端口

```
http-server --cors 127.0.0.1 -p 2333
```

之后打开 http://202.38.95.227:10026/

```
precomputed://http://192.168.2.232:2333
```

segmentation layer

upload image and segmentation using upload_segmentation.py 修改输入路径

```
python upload_SEGMENTATION.py precomputed://file:///data12T/janechen/httpServer/se
```

creat mesh:

https://github.com/google/neuroglancer/blob/master/src/neuroglancer/datasource/precomputed/meshes.md#legacy-single-resolution-mesh-format

https://github.com/google/neuroglancer/issues/293

In addition to the multi-resolution mesh format, an older single-resolution mesh format is also supported. This format is specified by either the absence of an `info` file in the mesh subdirectory or an `info` file containing a JSON object with the following members:

- `"@type"`: Must be `"neuroglancer_legacy_mesh"`.

To specify a legacy single-resolution mesh dataset that lacks an `info` file as a Neuroglancer data source, use the data source URL syntax `precomputed://FILE_URL#type=mesh`, where `FILE_URL` is the URL to the directory containing the mesh data using any supported file protocol.

The surface mesh representation for a given segmented object may be split into one or more separate fragments (e.g. corresponding to subvolumes).

Within the mesh subdirectory, for each segmented object for which a surface representation is available, there is a JSON-format metadata file named `<segment-id>:0`, where `<segment-id>` is substituted with the base-10 string representation of the segment label value. This metadata file must contain an object with a `"fragments"` property specifying the filenames (relative to the mesh subdirectory) containing the mesh data for each fragment.

meshing:

https://github.com/seung-lab/igneous

Igneous is a TaskQueue and CloudVolume based pipeline for producing and managing visualizable Neuroglancer Precomputed volumes.

Igneous is useful for downsampling, transferring, deleting, meshing, and skeletonizing large images. There are a few more esoteric functions too.

需要用的是generate meshes/skeletons for an already-existing Precomputed segmentation volume：

mip缩放：1.如果不缩放meshing的过程会很慢。2.neuroglancer上不放大的时候会读更大的mip，更快。

1. downsample the segmentation

```
igneous downsample --mip 0 --num-mips 4 --queue my_queue precomputed://file://seg
igneous execute my_queue/
```

但是问题是生成的格式是gzip压缩过的，还要解压一下。option输入不了None，可能是个bug

```
gzip -d *
```

2. create mesh

```
igneous mesh forge --dir mesh --mip 2 --queue my_queue2 precomputed://file://seg
igneous execute my_queue2
igneous mesh merge --dir mesh --magnitude 2 --queue my_queue3 precomputed://file:/
igneous execute my_queue3
```

3. create skelenton

```
% 保持和evaluation的skeletonization参数一致
igneous skeleton forge --mip 2 --scale 4 --const 500 --dust-threshold 100 --queue
igneous execute my_queue2
igneous skeleton merge --queue my_queue2 precomputed://file://seg
igneous execute my_queue2
```

解压+在skeletons的info中制订segment info的位置，add `"segment_properties": '..'`

暂时没用上的：

这个是有mesh了之后做multi resolution的，现在不需要

https://github.com/janelia-cosem/multiresolution-mesh-creator

swc support

https://github.com/google/neuroglancer/pull/212

大组里的数据：

precomputed://http://202.38.95.227:10027/LiuXinzhao0.8Jpeg2