**Remote Computing:**

**Great Lakes**

UM Slurm high performance cluster (HPC)

https://arc.umich.edu/greatlakes/

Host Name: greatlakes.arc-ts.umich.edu

Password: Level 1

**Galban Tier2**

Host Name: galban-ap-ps1a

Password: Level 2

Access either through PuTTY or Remote Desktop. We’ve had some issues with RDP stability, so it is recommended to use PuTTY. For enabling of graphical interfaces, enable X11 forwarding (Connection -> SSH -> X11 -> Enable X11 forwarding)

**MySQL**

https://documentation.its.umich.edu/?q=node/380

Host Name: login.itd.umich.edu

Password: Level 1

**Third-Party Software:**

**PuTTY**

SSH client for accessing Great Lakes, the tier2, and MySQL

Download: https://www.putty.org/

**Xming**

X11 forwarding program for use with PuTTY in displaying windows through SSH

Download: https://sourceforge.net/projects/xming/

**Elastix**

Software for image coregistration

Download: https://elastix.lumc.nl/

**TotalSegmentator**

GitHub: <https://github.com/wasserth/TotalSegmentator>

Install Python >= 3.7

Install Pytorch >= 1.12.1

Install TotalSegmentator:

pip install TotalSegmentator

Usage:

python TotalSegmentator -i ct.nii.gz -o segmentations

Options:

--ml : Saves segmentations in single file instead of individual binary files

From matlab:

system(‘cd /D <Save Path> & python <TotalSegmentator> -i <Input Nifti> -o <Save Path>’)

* <Save Path> = full path to where you want the results saved
* <TotalSegmentator> = full path to the python installation of TotalSegmentator
* <Input Nifti> = full path to the Nifti image to be processed

For GPU enabling:

Anaconda

**CUDA**

Software for GPU control

PyTorch needs CUDA v11.7 or v11.8

https://docs.nvidia.com/cuda/cuda-installation-guide-microsoft-windows/index.html

* Requires an installation of MS Visual Studio
  + <https://visualstudio.microsoft.com/free-developer-offers/>
  + Visual Studio Community

**Pytorch**

pip3 install torch torchvision torchaudio --index-url <https://download.pytorch.org/whl/cu118>

**Windows Subsystem for Linux (WSL)**

This is a virtual Linux workspace.

In cmd.exe as administrator: wsl –install

**GitHub**

**Not Working**

**Docker Desktop**

Provides a GUI for managing containers, using WSL to isolate environment versions for certain applications.

**NVIDIA PyTorch Release Containers**

<https://docs.nvidia.com/deeplearning/frameworks/pytorch-release-notes/rel-23-03.html#rel-23-03>