Documentation

**Data Preprocessing**

1. /gpfs/data/denizlab/Users/bz1030/src/preprocessing.py
   1. Read DICOM file and convert it into HDF5
   2. Knee detection from both OULU method and our own method
2. /gpfs/data/denizlab/Users/bz1030/src/utils.py
   1. Global Contrast Normalization
   2. Inversion if the image is MONOCHROME1
   3. 5th 95th histogram truncation
   4. 0.2 mm fixed resolution conversion
3. /gpfs/data/denizlab/Users/bz1030/src/detector.py
   1. Code from OULU [KneeLocalizer](https://github.com/MIPT-Oulu/KneeLocalizer) (SVM Model), Follow the repo to install
4. /gpfs/data/denizlab/Users/bz1030/src/main.py
   1. Run this file to load data from OAI\_Original to OAI\_Processed
   2. ‘python main.py 00m’, ‘00m’ stands for the month of patient data folder

Procedure:

* Run the preprocessing by “python main.py month\_file\_name” e.g. “python main.py 00m”
* The code will first read DICOM as a numpy array then perform the following preprocessing

1. Convert image to 0.2mm fixed resolution
2. Check DICOM header, if it is MONOCHROME1, then we invert the image
3. Global Contrast Normaliztion
4. 5th 95th histogram truncation
5. Use OULU’s SVM to extract left knee and right knee, use our method to extract left knee and right knee
6. If SVM return -1 in coordinates of knee, then we use our method.
7. The raw image before and after preprocessing, the left right knee image from both methods will be saved into a png figure for displaying the effects.
8. Save each knee image to HDF5, and make a record in a summary file.

**Model**

* **KLModel**
* **Model\_torch**
* **Train\_test\_split.py**

**Data Statistics**

* Total 44865 XRay image
* Total 4490 Patients (Removing those who do not have KL grade in file)
* KL Grade distribution
  + 0:0.39
  + 1:0.27
  + 2:0.18
  + 3:0.14
  + 4:0.03
* Train, validation, test is split as 7:2:1. Split are based on subjects level. For each set, only same group of subjects.