

의료 이미지 전처리

2021.05.10 ~ 2021.05.14

백지윤

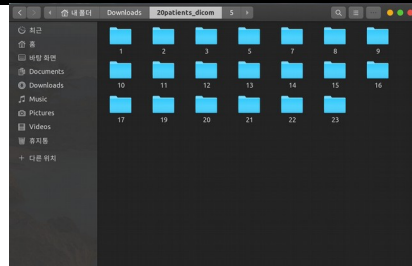
# Index

- **CT Dicom image preprocessing**
  - **images stacking** ☒
  - **resampling** ☒
  - **normalization** ☒
  - **resizing** ☒
  - **.tif format save** ☒

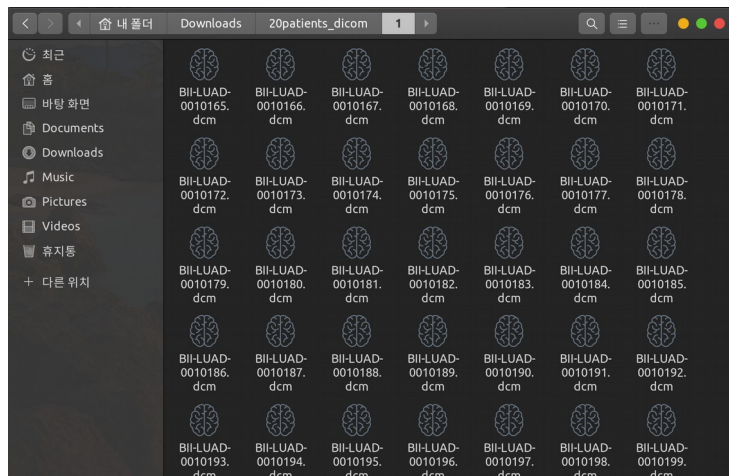
# Index

- **segmentation txt** 파일로 **mask** 파일 만들기
  -

# 20patients\_dicom 데이터 구성

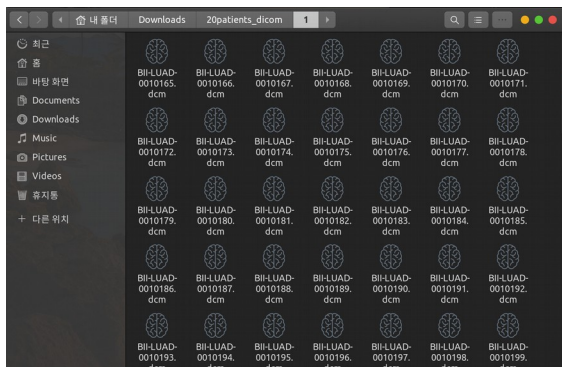


- 총 20 개의 폴더 (환자 #1 ~ 환자 #20)
- 각 폴더 내부 폐 위치 별로 연속해서 촬영한 dcm 데이터로 구성



# 20patients\_dicom 데이터 구성

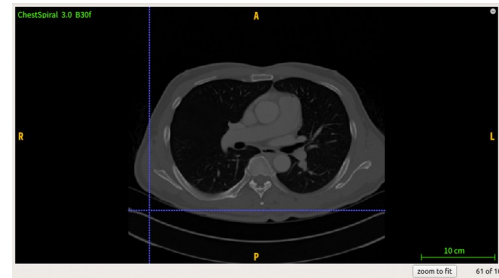
- 각 폴더 내부 폐 위치 별로 연속해서 촬영한 dcm 데이터로 구성
- dcm 파일 총 108개 → 즉 한 명의 환자 당 연속적으로 폐 위치를 바꾸어가며 108번 촬영



ex1. 52번째 순서



ex2. 61번째 순서



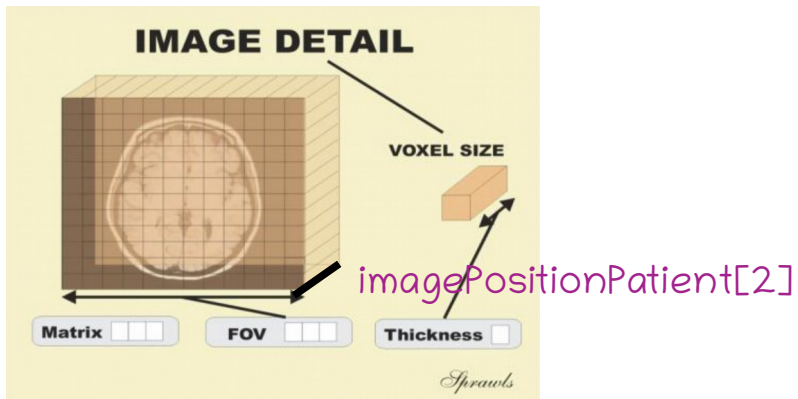
## CT Dicom image preprocessing 1) Image stacking

- Image J 같은 소프트웨어에 환자군 폴더를 그대로 업로드시키면 자동으로 연속적인 108개의 이미지를 stacking 해서 볼 수 있음
- 이 과정을 코드로 직접 작성해볼 것 !

# CT Dicom image preprocessing 1) Image stacking

- 이 과정을 코드로 직접 작성해볼 것 !
- 1) load\_scan 메소드

```
slice_thickness = np.abs(slices[0].ImagePositionPatient[2] - slices[1].ImagePositionPatient[2])
```



# CT Dicom image preprocessing 1) Image stacking

- 2) get\_pixels\_hu 메소드

HU ; Hounsfield 단위(HU)는 의료 CT 이미지의 그레이 스케일을 구성합니다. 4096개 값(12비트)의 / 검은색에서 흰색에 이르는 스케일로 그 범위는 -1024HU ~ 3071HU(0 또한 값에 포함됨)입니다. 이는 다음과 같이 정의됩니다.

-1024HU는 검은색이며 공기(폐 내부)를 나타냅니다. 0HU는 물(인체는 주로 물로 구성되어 있으므로 여기에서 피크가 큼)을 나타냅니다. 3071HU는 흰색이며 인체에서 가장 밀도가 높은 조직인 치아 에나멜을 나타냅니다. 다른 모든 조직은 이 스케일 내에 있습니다. 지방은 약 -100HU, 근육은 약 100HU이며 뼈 쪽은 200HU(소주골/하악골)에서 약 2000HU(피질골)입니다.

- Some scanners have cylindrical scanning bounds, but the output image is square. The pixels that fall outside of these bounds get the fixed value -2000. The first step is setting these values to 0, which currently corresponds to air.

```
image[image == -2000] = 0
```

```
image += np.int16(intercept) 따라서 0 → 1024 !
```

```
[[[ 0  5 18 ... 24 23 23]
 [11 20 28 ... 30 22 13]
 [23 32 36 ... 20 20 20]
 ...
 [15 22 26 ... 22 24 26]
 [ 6 17 27 ... 21 26 25]
 [ 0  7 21 ... 16 27 27]]
```

(108,512,512)



```
[[[-1024 -1019 -1006 ... -1000 -1001 -1001]
 [-1013 -1004 -996 ... -994 -1002 -1011]
 [-1001 -992 -988 ... -1004 -1004 -1004]
 ...
 [-1009 -1002 -998 ... -1002 -1000 -998]
 [-1018 -1007 -997 ... -1003 -998 -999]
 [-1024 -1017 -1003 ... -1008 -997 -997]]
```

(108,512,512)



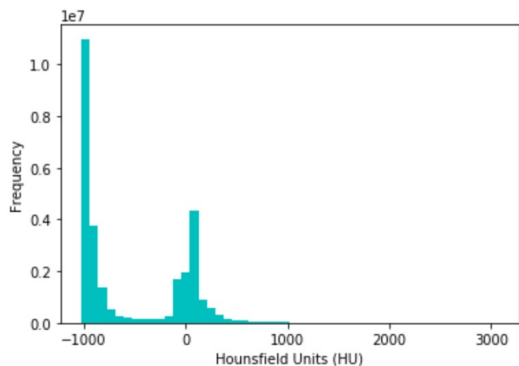
# CT Dicom image preprocessing 1) Image stacking

- HU 변환 결과

## Displaying Images

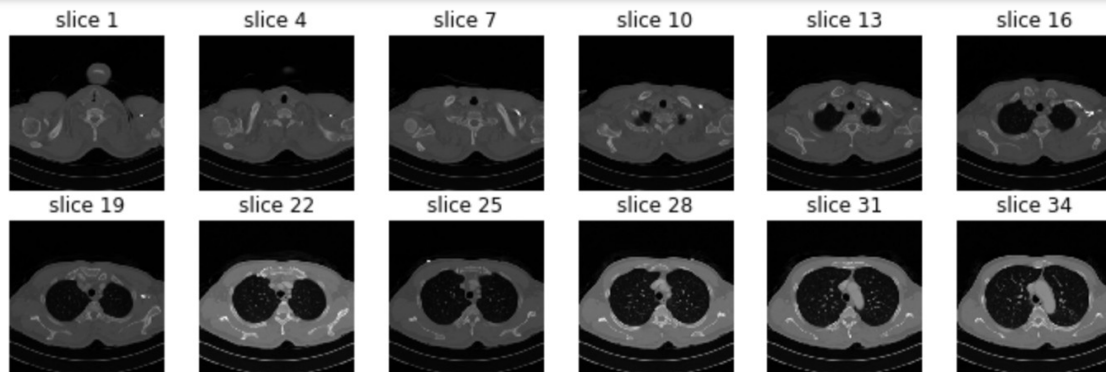
```
In [43]: file_used=output_path+"fullimages_%d.npy" % id
         imgs_to_process = np.load(file_used).astype(np.float64)

         plt.hist(imgs_to_process.flatten(), bins=50, color='c')
         plt.xlabel("Hounsfield Units (HU)")
         plt.ylabel("Frequency")
         plt.show()
```

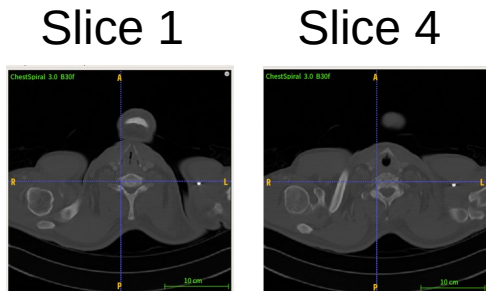


Substance	HU
Air	-1000
Lung	-500
Fat	-100 to -50
Water	0
Blood	+30 to +70
Muscle	+10 to +40
Liver	+40 to +60
Bone	+700 (cancellous bone) to +3000 (cortical bone)

# CT Dicom image preprocessing 1) Image stacking



코드로 구현한 것



소프트웨어로 확인 결과 일치 !

## CT Dicom image preprocessing 2) resampling

- 각 108개의 사진마다 slice thickness, pixel spacing 이 제각각일 수 있음 →

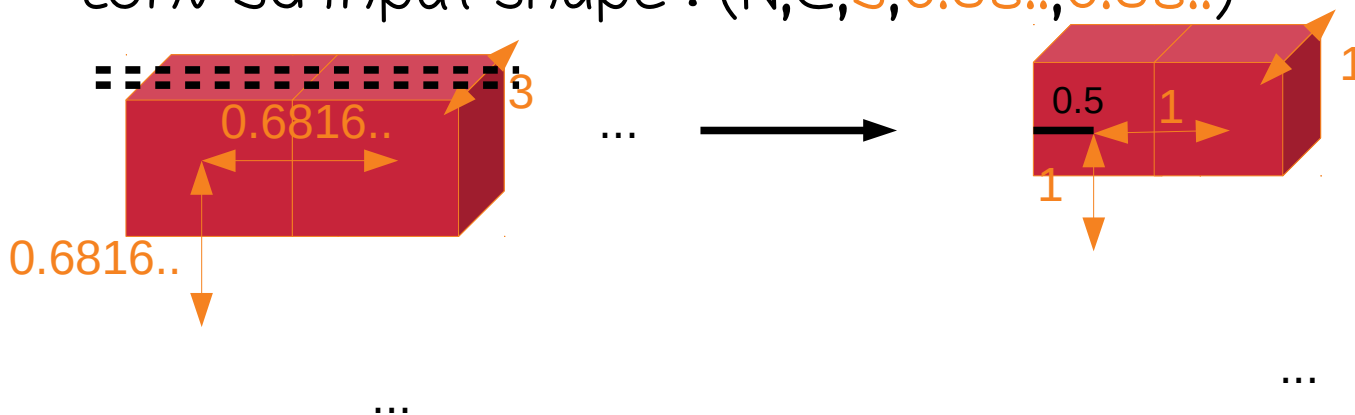
conv 3d 에 넣어주기 위해서  $[1, 1, 1]$  로 통일할 것 !

conv 3d input shape :  $(N, C, D, H, W)$

## CT Dicom image preprocessing 2) resampling

- Pixel spacing : 출력 결과 [0.6816... , 0.6816...]

conv 3d input shape : (N,C,3,0.68..,0.68..)



→ (N,c,1,1,1) 로 reshape 해주기 ! 이 과정을 코드로 직접 작성해볼 것 !

## CT Dicom image preprocessing 2) resampling

- `image = scipy.ndimage.interpolation.zoom(image, real_resize_factor)` `#real_resize_factor=[3,0.68...,0.68...]`
- `[108, 512, 512] → [324,349,349]`; 108개의 연속된 이미지인데 각각 sliding thickness 가 3인 것을 1,1,1로 분리해주는 것이므로 108\*3, 나머지 차원들도 마찬가지로

## CT Dicom image preprocessing 3) normalization

- Normalization 수행.
- 현재 HU 값 범위는 -1000 ~ 3000, ~ -1000 은 공기를 나타내고, 400~ 은 뼈를 나타내므로 관심 영역이 아님 !  
(MIN\_BOUND, MAX\_BOUND)

## CT Dicom image preprocessing 4) resizing

- original shape : (108,512,512) 108개 이미지, 512\*512 pixel
- shape after resampling : (324,349,349) 324개 이미지,  
349\*349 pixel
- -> 목표 : (324,512,512) 324개 이미지, 512\*512 pixel 로 다시 바꾸기
- `skimage.transform` 사용 !  
`resized=resize(img,(IMG_PX_SIZE,IMG_PX_SIZE))`

## CT Dicom image preprocessing 5) .tif 포맷으로 저장

- PIL 의 Image 모듈 사용

```
im.save('/home/miruware/Desktop/Dicom_prac(2021.05.14)/  
temp/patient_1/{}.tif'.format(i))
```

- 324 장 각각의 Image 를 tif 포맷으로 patient\_1 폴더에 저장하였음

