HeLP Challenge
- Cardiac Segmentation

engineering advices

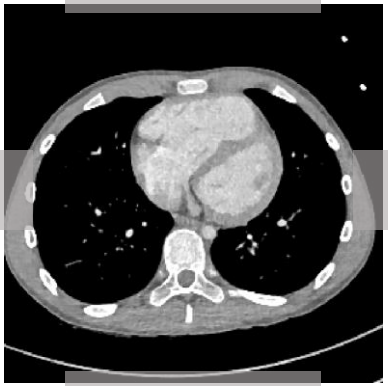
Nov. 28. 2018

Jieun Kim

Radiographic presentations

Anterior

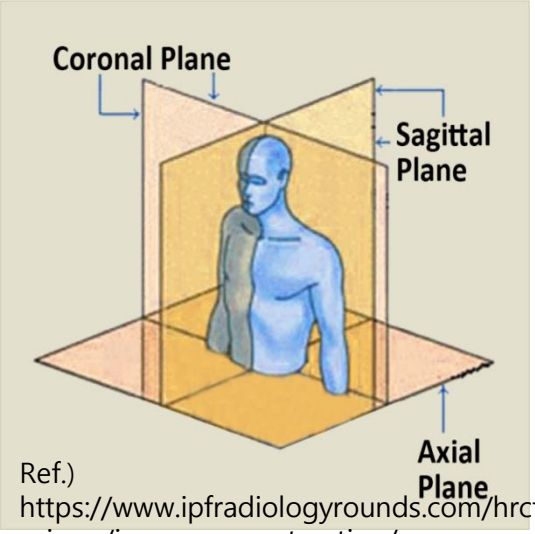
Axial plane



Right

Left

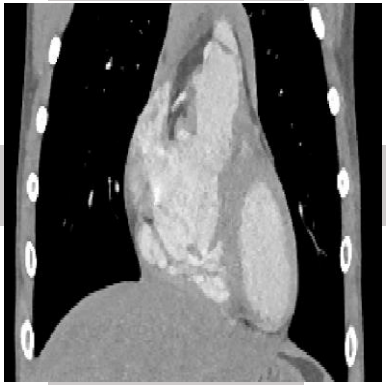
Posterior



Ref.) https://www.ipfradiologyrounds.com/hrct-primer/image-reconstruction/

Head

Coronal plane



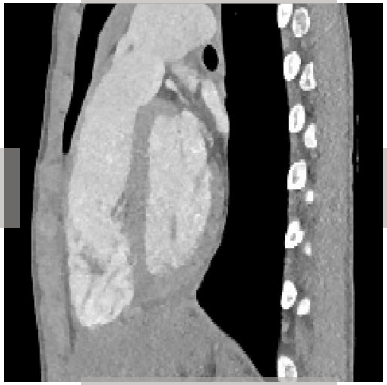
Right

Left

Foot

Head

Sagittal plane



Anterior

Posterior

Foot

MHA analyzed file format 1/3

- MHA file
  - An ITK(Insight segmentation and Registration Toolkit) MetaImage
  - Soft application of viewer : ITK SNAP, MITK, ImageJ etc.
- Load MHA file in python
  - Using simpleITK

```
import SimpleITK as sitk
```

```
imageInput = sitk.ReadImage(file_name_imageT1[n])
```

  - Get size, origin, spacing, direction

```
imageInput =sitk.ReadImage(file_name_imageT1[0])

origin3d = imageInput.GetOrigin()
spacing3d =imageInput.GetSpacing()
size3d =imageInput.GetSize()
direction3d =imageInput.GetDirection()
```

MHA analyzed file format 2/3

- Convert itk image to array

```
imageT = sitk.GetArrayFromImage(image) # covert itk image to Array
```

```
model.train_on_batch(imageT,LabelT1)
```

- Write image

```
model1.load_weights(path_model1)
result11 = model1.predict(imageT)
result1 = result11.reshape((img_rows,img_cols,img_dep))
ResultImg11 = sitk.GetImageFromArray(result1)
```

```
final_Img1.SetSpacing(originalImg.GetSpacing())
final_Img1.SetOrigin(originalImg.GetOrigin())
```

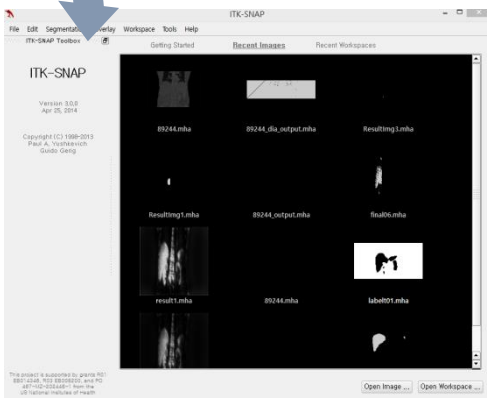
```
sitk.WriteImage(final_Img1,output_filename)
```

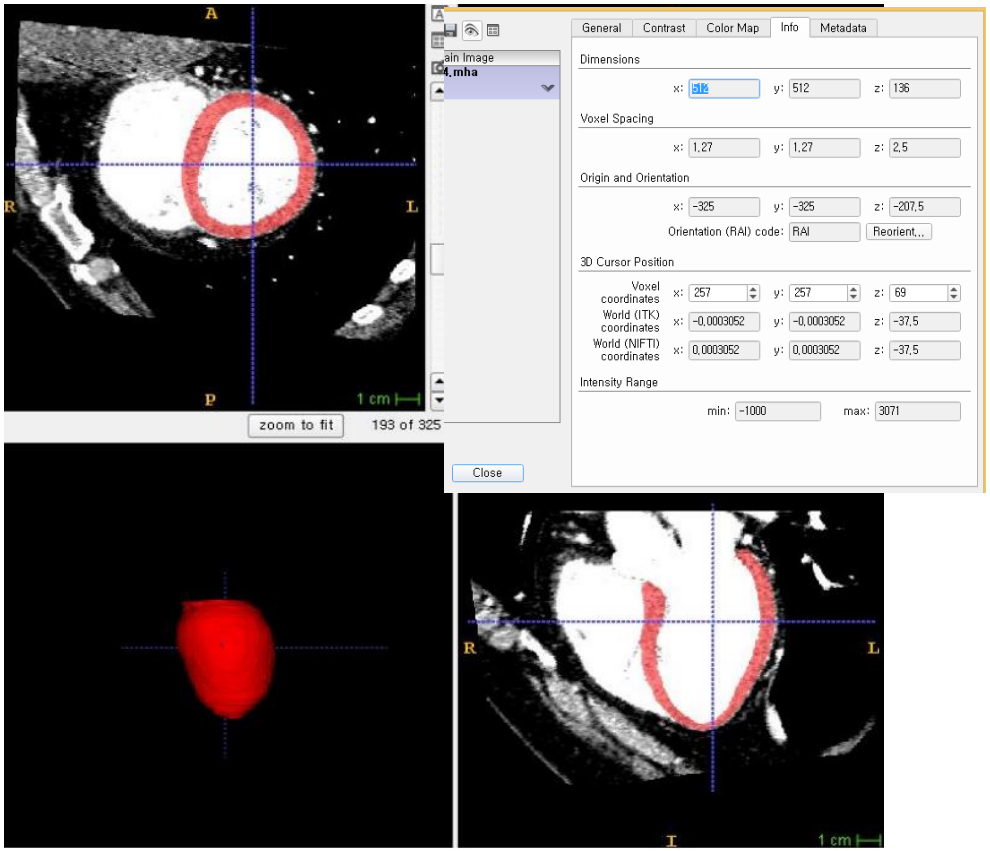
MHA analyzed file format 3/3

- Soft application of visualization

dragging

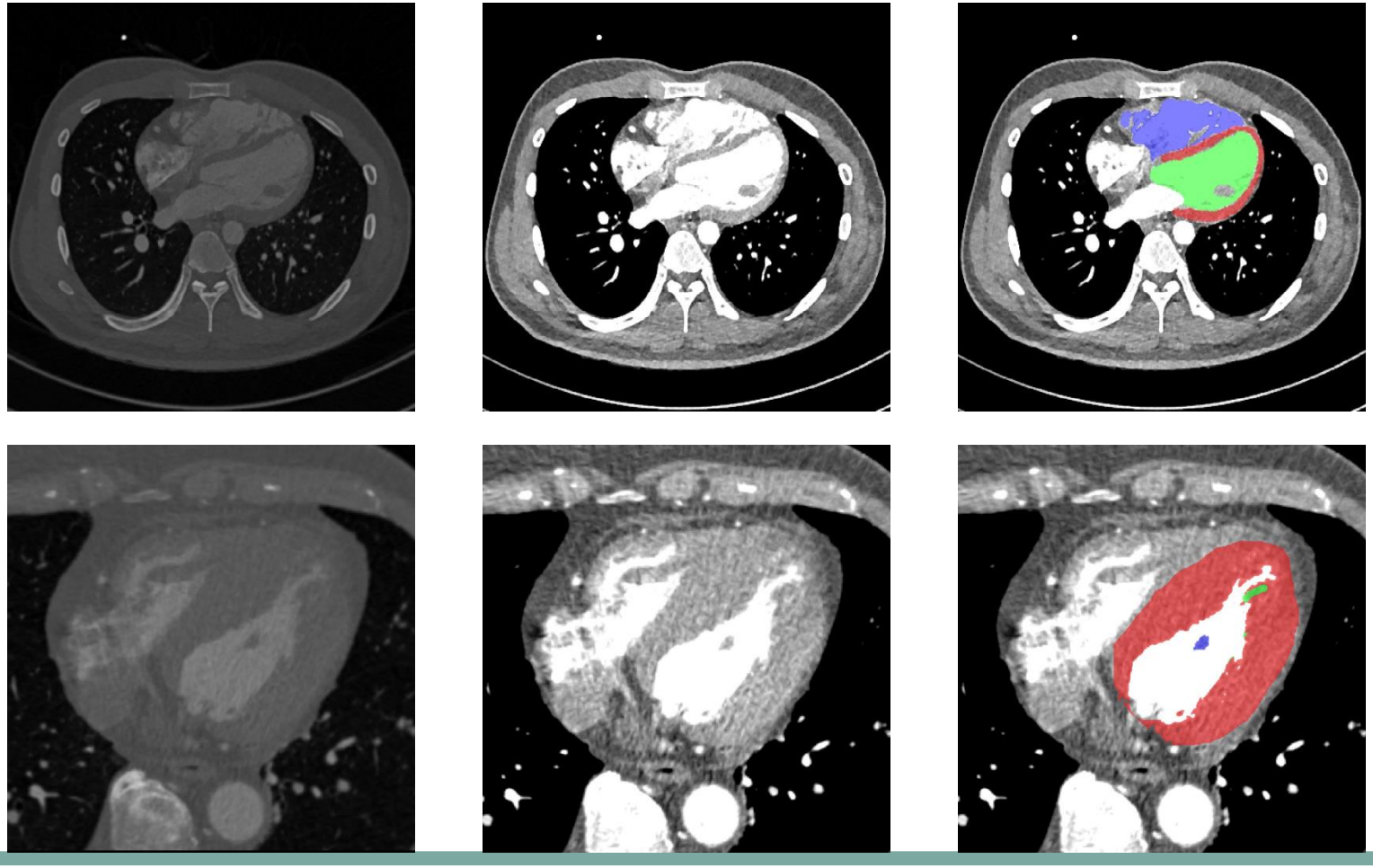
89244.mha



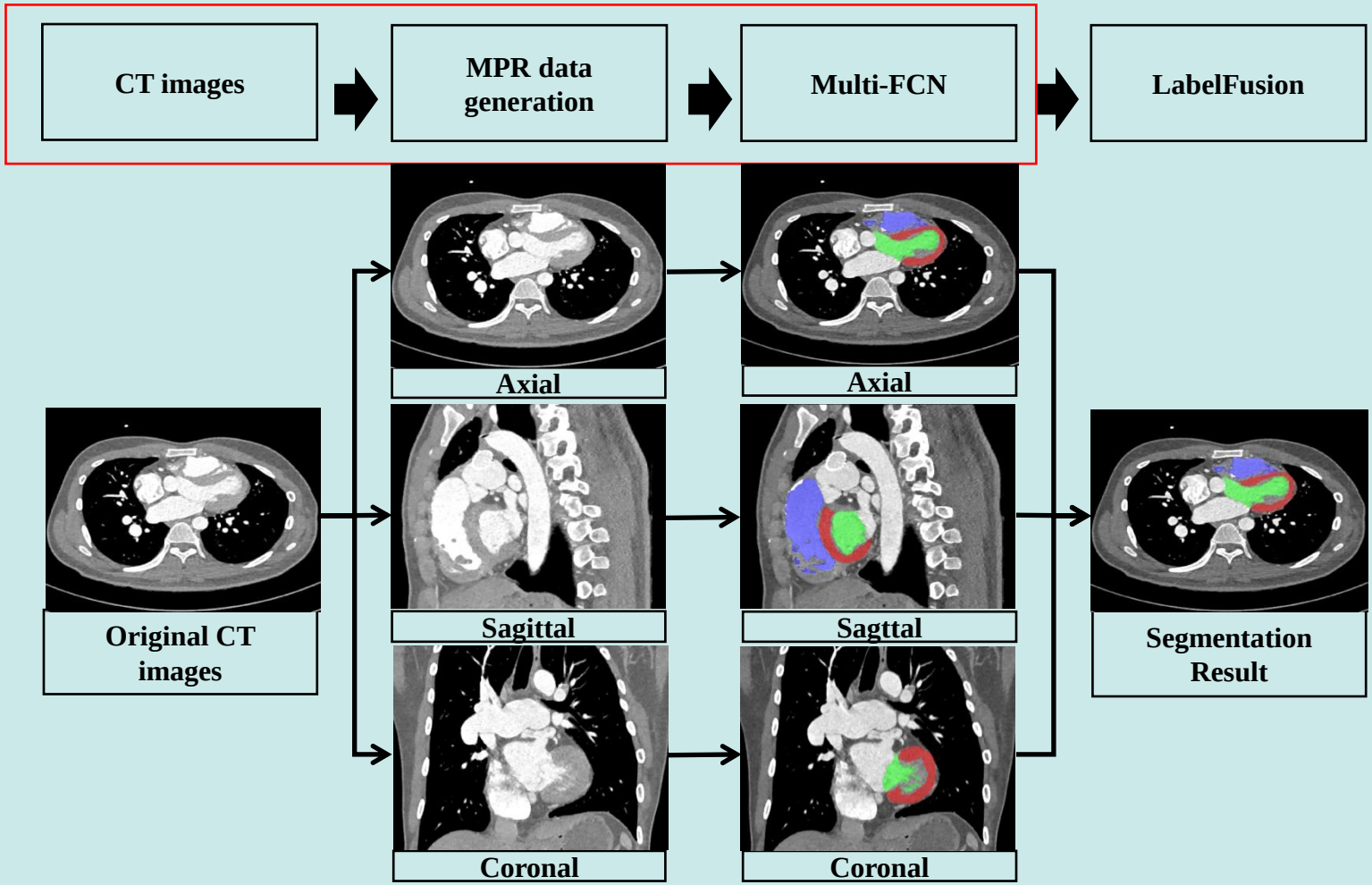


Pre-processing
-intensity windowing

- (WindowLevel, WindowWidth) = (0, 600) ~ (0, 1000)



Reference method flow chart



Reference Method 1/2

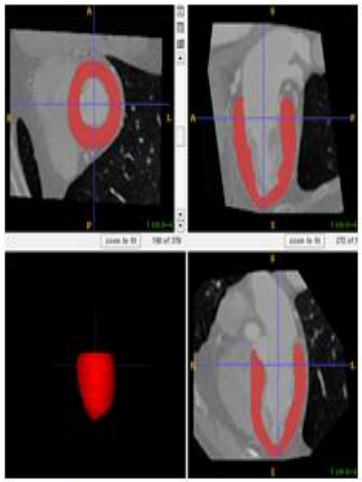
```
for epoch in range(20) :
    random.shuffle(indices_t)
    random.shuffle(indices_v)
    for n in indices_t:

        input_img = sitk.ReadImage(file_name_imageT1[n])# read image
        input_lbl = sitk.ReadImage(file_name_imageT2[n])# read label

        isocubic_img = resample_ISOCUBIC(input_img,(0.5,0.5,0.5))
        isocubim_lable=resampleMask_ISOCUBIC(input_lbl,(0.5,0.5,0.5))  →  sitk.Resample(


        isocubic_img_ww = sitk.Cast(sitk.IntensityWindowing(isocubic_img, LowT,HighT,0,255), sitk.sitkUInt8)

        size3d = isocubic_img_ww.GetSize()
        size2d = (size3d[0],size3d[1],0) #axial slices,

        outimg = sitk.Image(isocubic_img.GetSize(),sitk.sitkUInt8)
        outimg.SetDirection(isocubic_img.GetDirection())
        outimg.SetSpacing(isocubic_img.GetSpacing())
        outimg.SetOrigin(isocubic_img.GetOrigin())

        outimage_array = sitk.GetArrayFromImage(outimg)

        num_of_slices = isocubic_img_ww.GetDepth()
```

Reference Method 2/2

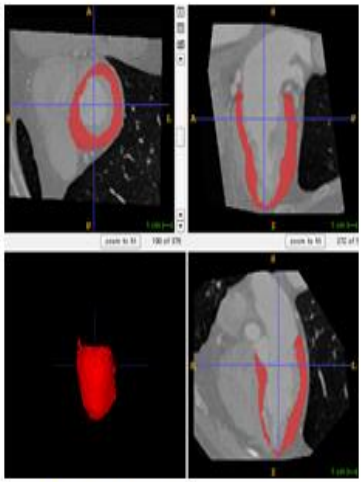
```
for i in range(num_of_slices):
    prev_idx = (0,0,i-displacement)
    cur_idx  = (0,0,i)
    next_idx = (0,0,i+displacement)

    if( prev_idx[2] <0) or (next_idx[2] >=num_of_slices):
        continue

    pre_image = sitk.Extract(isocubic_img_ww,size2d,prev_idx)
    cur_image  = sitk.Extract(isocubic_img_ww,size2d,cur_idx)
    next_image = sitk.Extract(isocubic_img_ww,size2d,next_idx)

    mask2d = sitk.Extract(isocubim_lable,size2d, cur_idx)

    #512*512로 추출한 영상들 resize하기.
    prev_image_resample = resample_resize2D(prev_image,(img_rows,img_cols))
    cur_image_resample  = resample_resize2D(cur_image,(img_rows,img_cols))
    next_image_resample = resample_resize2D(next_image,(img_rows,img_cols))


    mask2d_resample = resample_resizeMask2D(mask2d,(img_rows,img_cols)) # training할 최종 label
    mask_final = label_Mask_3ch(mask2d_resample)# 3label을 각각 label별로 채널에 나누어서 저장.
    rgbImage = sitk.Compose(prev_image_resample, cur_image_resample,next_image_resample)

    image_train_arr = sitk.GetArrayFromImage(rgbImage)
    label_train_arr = sitk.GetArrayFromImage(mask_final)

    image_training = image_train_arr.reshape((1,img_rows,img_cols,3) )
    label_training = label_train_arr.reshape((1,img_rows,img_cols,n_class))

    model.train_on_batch( image_training, label_training)
```

Result of ref. method

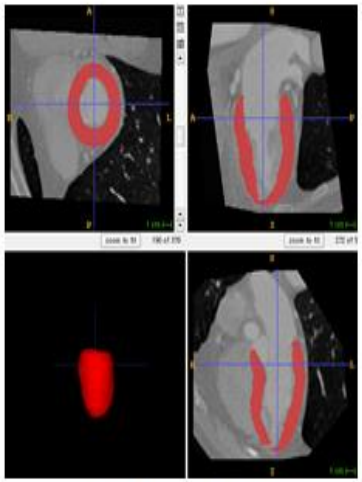
- Myocardium segmentation in cardiac CT images



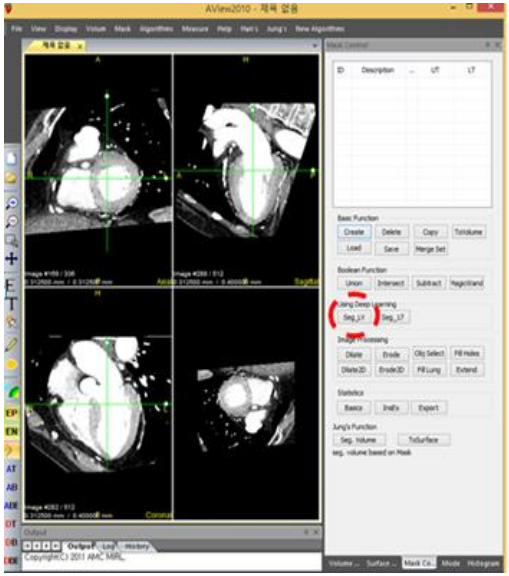
Manual Segmentation



Multi-atlas Segmentation



FCN Segmentation



$$DSC(A,B) = \frac{2 * |A \cap B|}{|A| + |B|}$$

Dice Coefficient result	Min	Max	Mean	Std
Multi-atlas	74.36674	92.10817	87.73908	3.058546
FCN	84.80109	95.0653	90.84953	2.264284

Running Time

- ▶ Multi-atlas : 40min/1case
- ▶ FCN : 2~3min/1case

TIPS : Docker image build (nvidia-docker)

- If your cuda is less than 9.0 version, do not use

```
FROM tensorflow/tensorflow:latest-gpu
```

  - It requires **CUDA 9.0 or more. (Nov.27. 2018)**
  - 1.2.0-gpu-py3
- Make good use of official docker-hub
  - https://www.tensorflow.org/install/docker
  - https://hub.docker.com/r/nvidia/cuda/
- Always commit docker-image when you modify your container.

Command : Docker image build (nvidia-docker)

- nvidia-docker exec -it [container-id] bash
- docker kill [container id] # remove container
- docker kill $(docker ps -q) #remove all container
- docker rmi –f [image id] # remove an image
- docker system prune –a # remove all image
- docker cp [host file or folder name] [container id]:[container file or folder name]

copy files

- docker commit [container name] [img name] # commit!
- docker save [img name]:[img tag] | qzip > [filename] # docker save image (.tar.gz)
- qunzip –c [filename] | docker load # docker qzip file load

Dataset structure

- CHD (congenital heart disease)
  - image
    - Diastolic :%3d_dia.mha
    - Systolic : %3d_sys.mha
  - label
    - Diastolic :%3d_dia_M.mha
    - Systolic : %3d_sys_M.mha
- HCMP (hypertrophic cardiomyopathy)
  - image
    - Abnormal: A_%3d.mha
    - Normal : N_%3d.mha
  - label
    - Abnormal: A_%3d_M.mha
    - Normal : N_%3d_M.mha

- [train]
  - /data/train/CHD/image
  - /data/train/CHD/label
  - /data/train/HCMP/image
  - /data/train/HCMP/label
- [test]
  - - /data/test/CHD
  - - /data/test/HCMP

- [output]
  - /data/output/CHD
    - '%03d_dia_output.mha'
    - '%03d_sys_output.mha'
  - /data/output/HCMP
    - 'A_%03d_output.mha'
    - 'N_%03d_output.mha

Outline
-cardiac segmentation

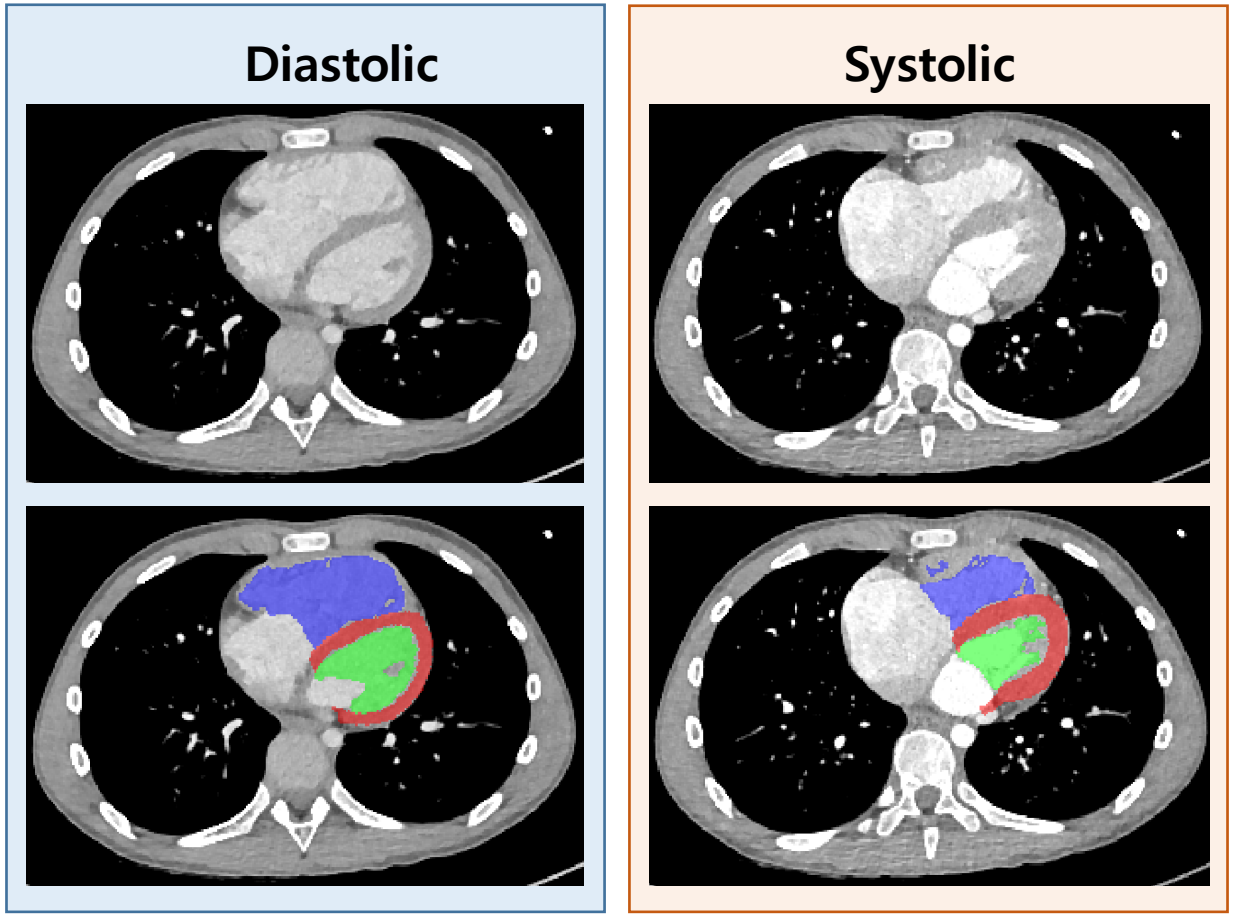
- [output]
  - 3-dimensional-1channel MHA file, (x,y,z,1)
  - Having the same structure as the test image
    - Size, origin, spacing, direction
  - Each output consisting of 3 labels
- To evaluate the performance, Dice similarity coefficient (DSC) of the cardiac segmentation will be computed.
- $DSC(R, G) = 2\,|R \cap G| \,/\, (|R|+|G|)$

Cardiac dataset (phase1)

- Three-dimensional CT scans for systolic, diastolic, abnormal and normal, respectively.
  - Train : 70 cases, test : 10cases (diastolic and systolic, respectively) → CHD
  - Train : 50 cases, test : 7 cases (abnormal)
  - Train : 12 cases, test : 3 cases (normal)

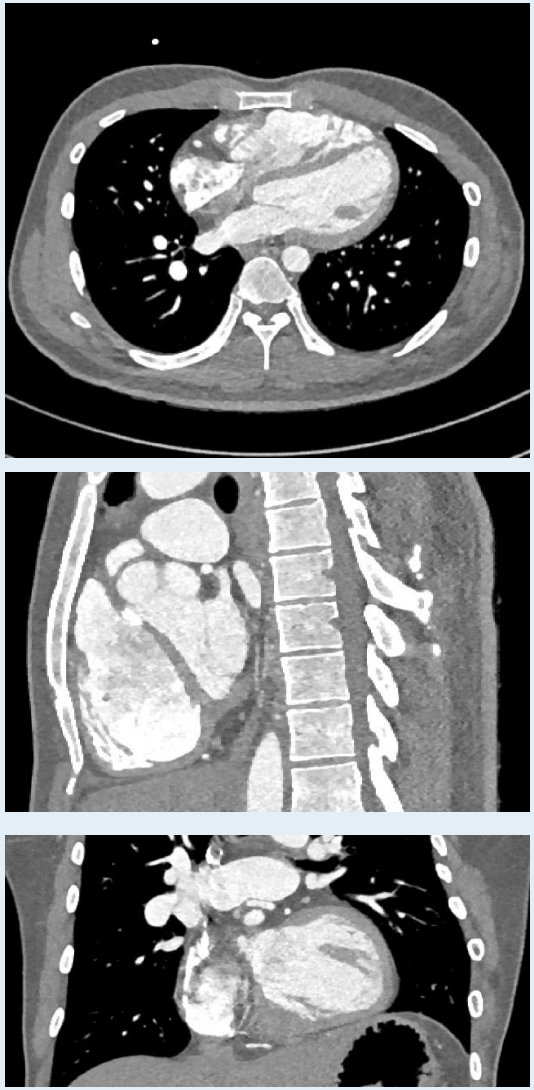
  → HCMP
- The image resolution is 512×512 voxels in-plane and the range of all slice number is diverse.
- Various range of voxel spacing
- The train dataset includes the segmentation result by experts to label and delineate

Cardiac CT data 1/2 -CHD

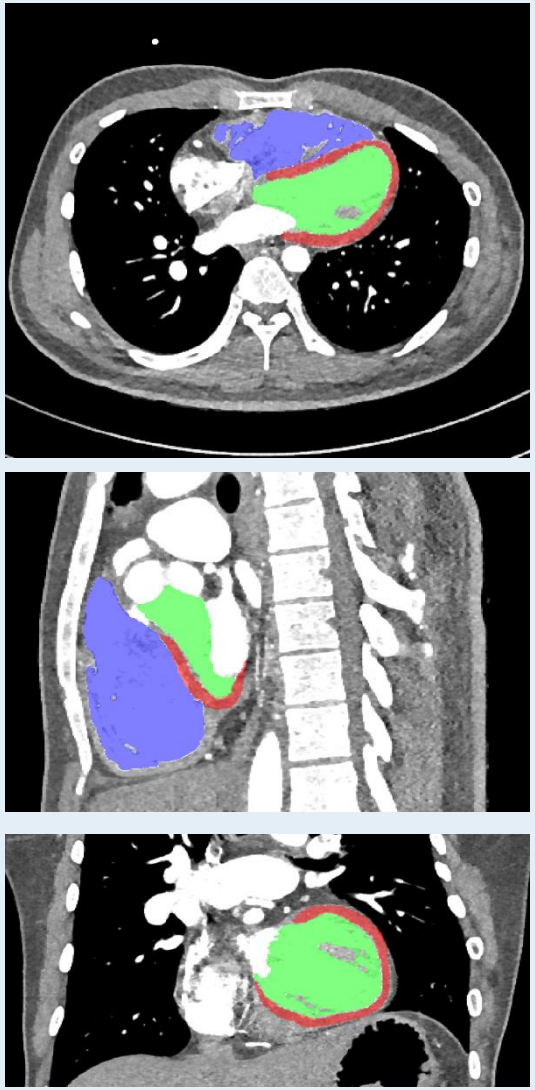


- LVM = left ventricle myocardium(1, read)
- LV = left ventricle chamber (2, green)
- RV = right ventricle chamber (3, blue)

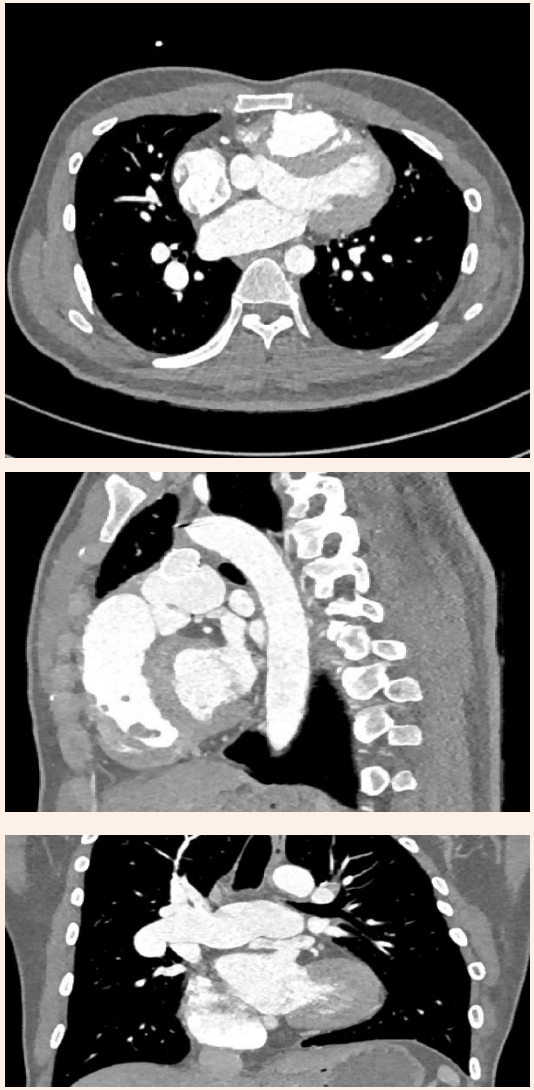
Diastolic



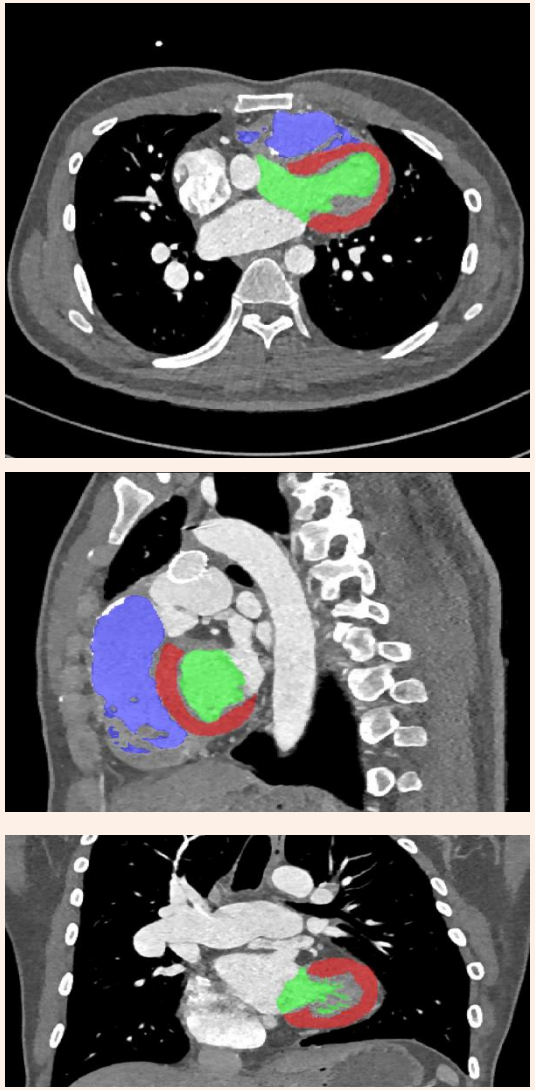
Diastolic



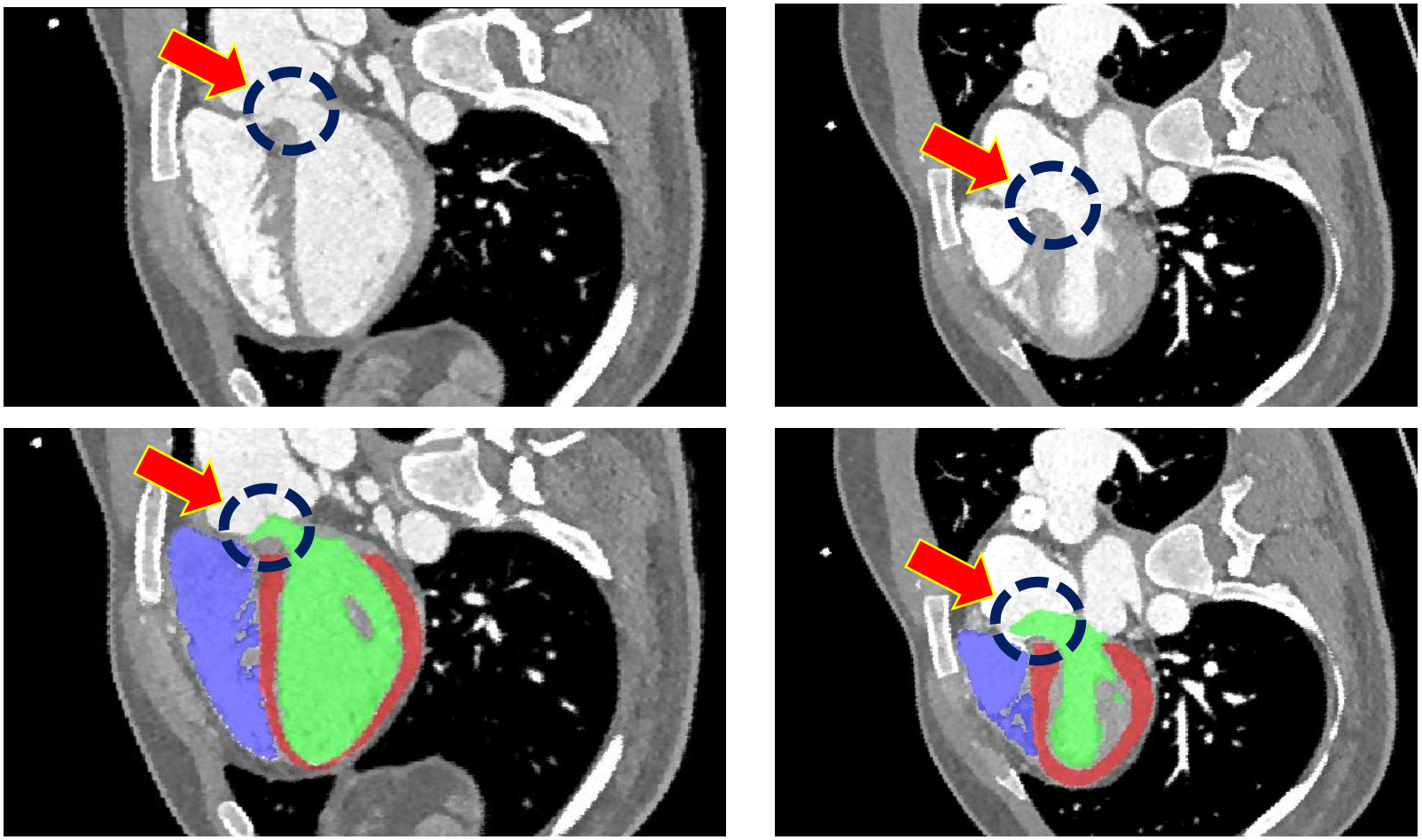
Systolic



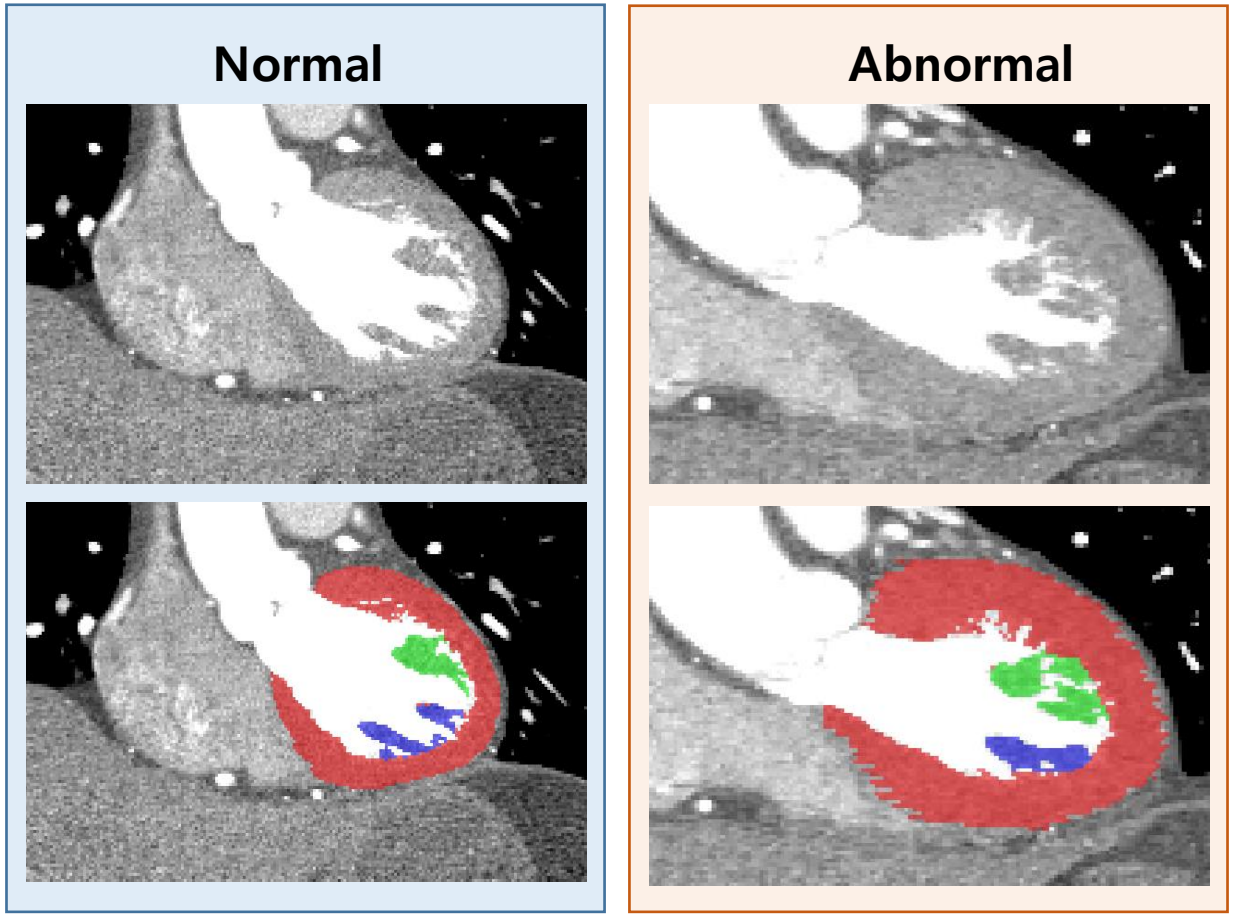
Systolic



Short axis view
- diastolic

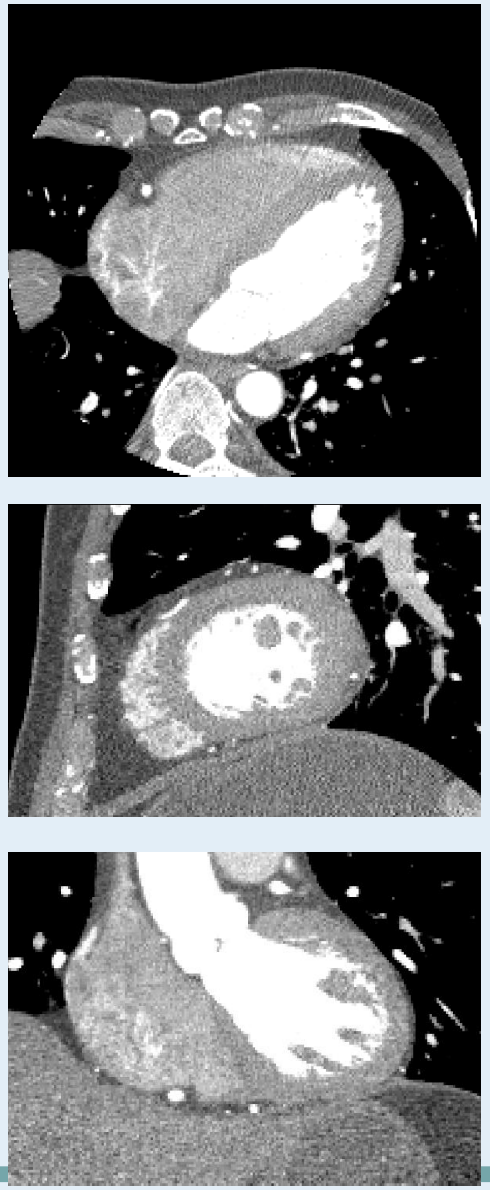


Cardiac CT data 2/2 -HCMP

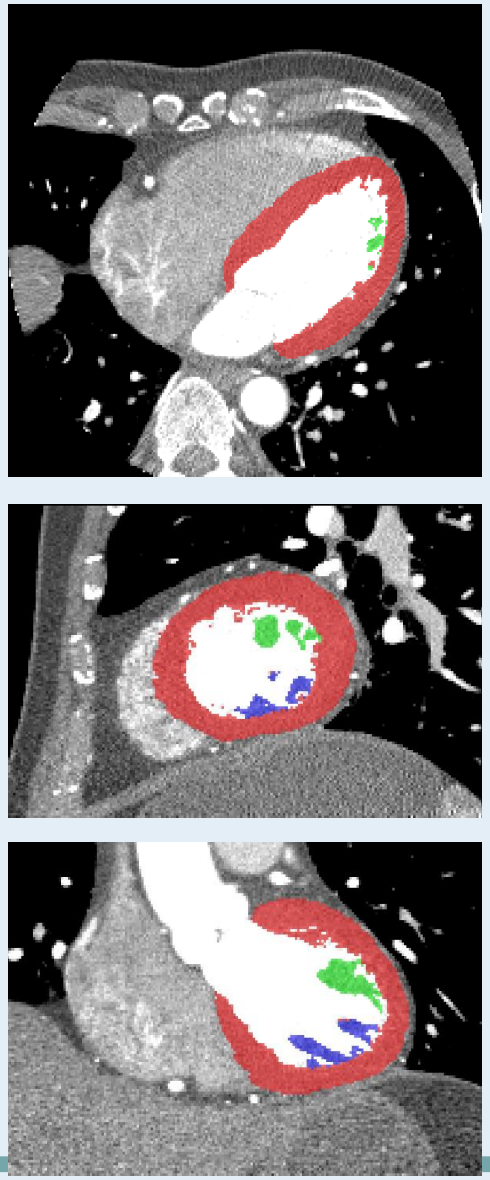


- LVM = left ventricle myocardium (1, red)
- APM = anterior papillary muscle (2, green)
- PPM = posteromedial papillary muscle (3, blue)

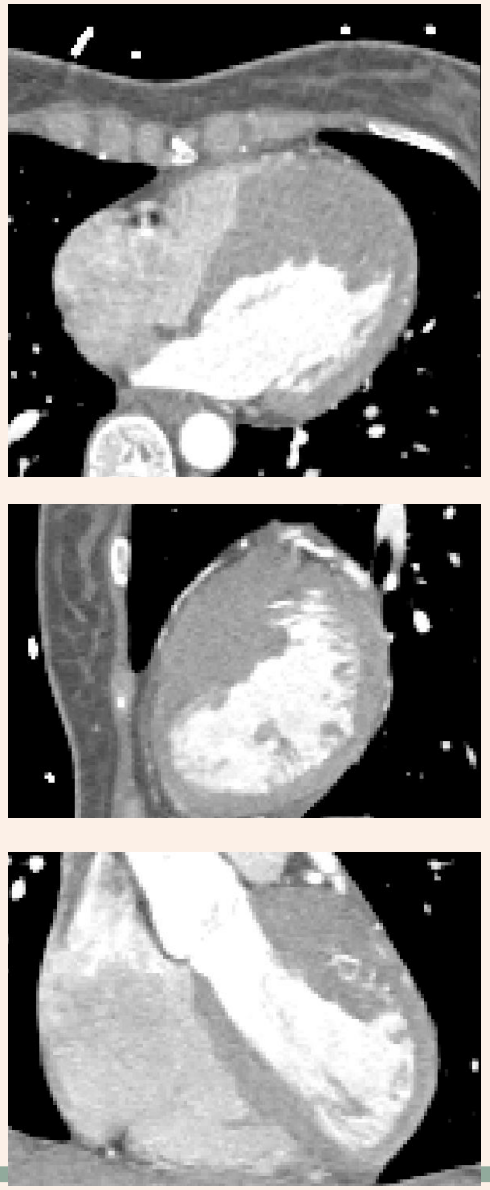
Normal



Normal



Abnormal



Abnormal