

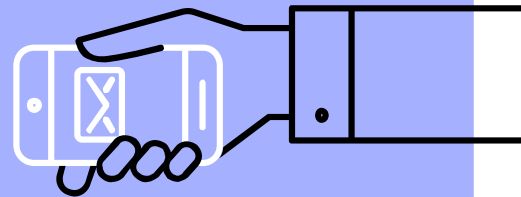
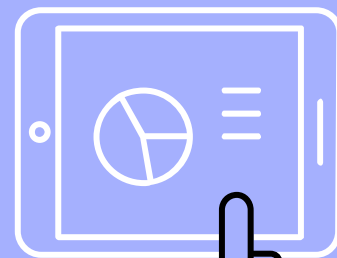


# MATLAB Flow Control I

Import and process of 3D images



盧家鋒 Chia-Feng Lu, Ph.D.  
Department of Biomedical Imaging  
and Radiological Sciences, NYCU  
[alvin4016@nycu.edu.tw](mailto:alvin4016@nycu.edu.tw)



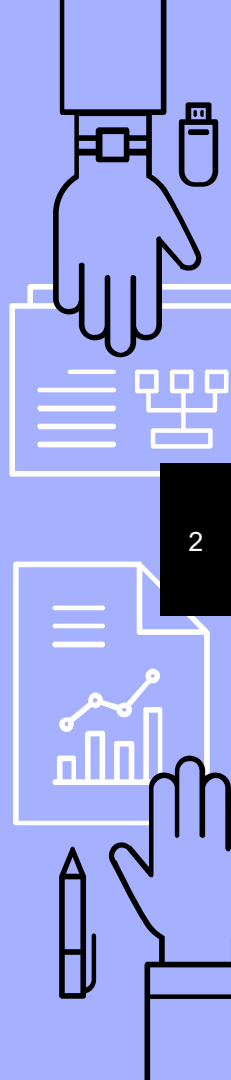


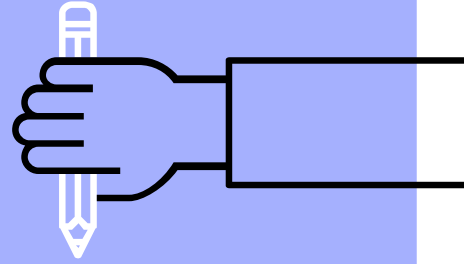
# Contents

- ▶ Flow Control: for-loop and while-loop
- ▶ Import and process of 3D images

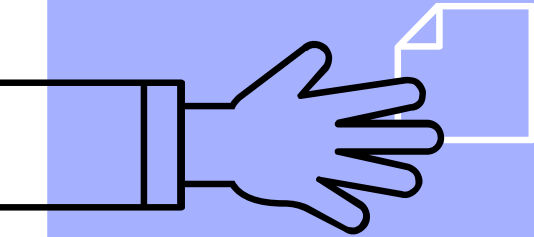
Please download the handout and materials from

[http://cflu.lab.nycu.edu.tw/CFLu\\_course\\_matlabimage.html](http://cflu.lab.nycu.edu.tw/CFLu_course_matlabimage.html)



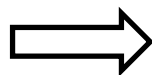
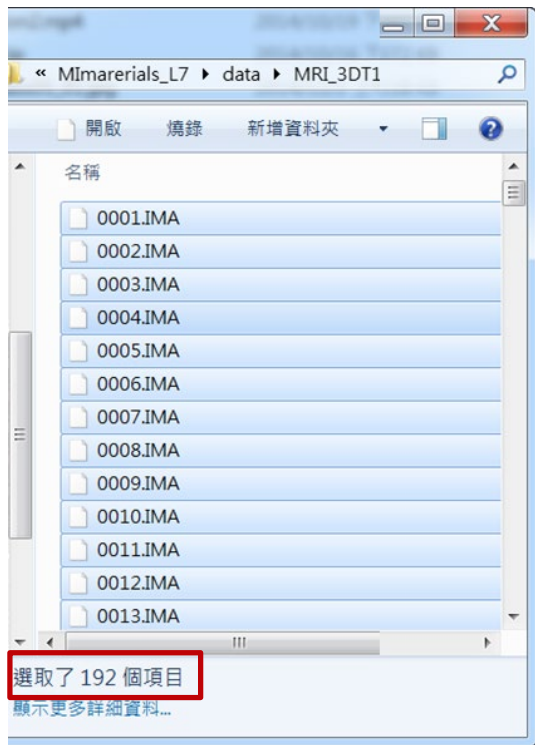


# For-loop and While-loop

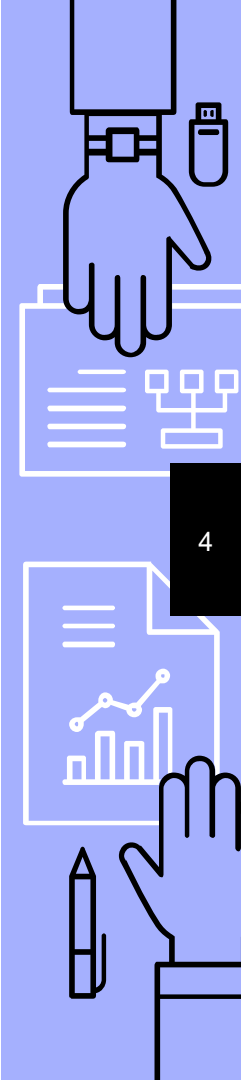


# Execution of Iterative Processes

- ▶ Read DICOM sequential images from a directory.

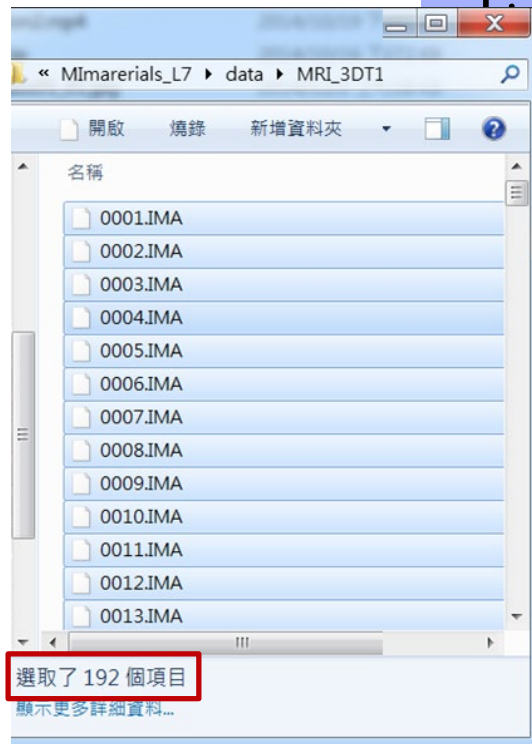


MRI  
3D T1W



# Required Commands

- ▶ Access directory information (path, content)
  - `dir`
- ▶ Read DICOM images from a directory
  - `for-loop`
  - `dicomread`
- ▶ Display images slice-by-slice
  - `for-loop`
  - `imshow`
  - `pause`



# for-loop Structure

```
for variable=expression  
    Statement  
end
```

```
for i=1:4  
    i  
end
```

```
for i=4:1  
    i  
end
```

```
for i=4:-1:1  
    i  
end
```

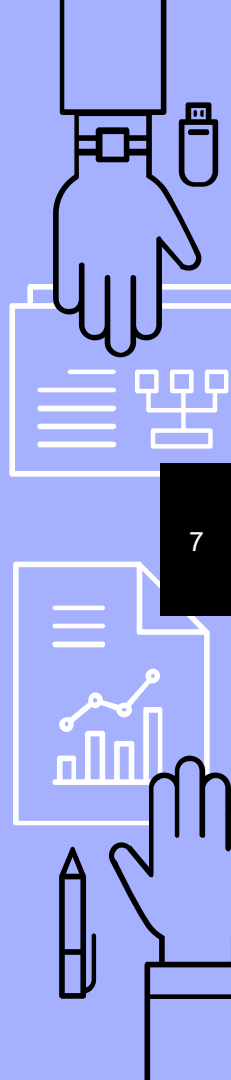
```
subjects=[1 3 4];  
for i=subjects  
    i  
end
```



# for-loop: Construct an array

- ▶ Create a 1 x 30 structure array "NC"
  - "age" field: a random number(0~50)
  - "predata" field: a 8 x 450 random array

```
for i=1:30
    NC(i).age=round(rand(1)*50);
    NC(i).predata=rand(8,450);
end
```



# Nested Structure

```
for variable1=expression1
    Statement1
    for variable2=expression2
        Statement2
    end
end
end
```

```
for i=1:2
    for j=1:3
        ind=[i,j]
    end
end
```

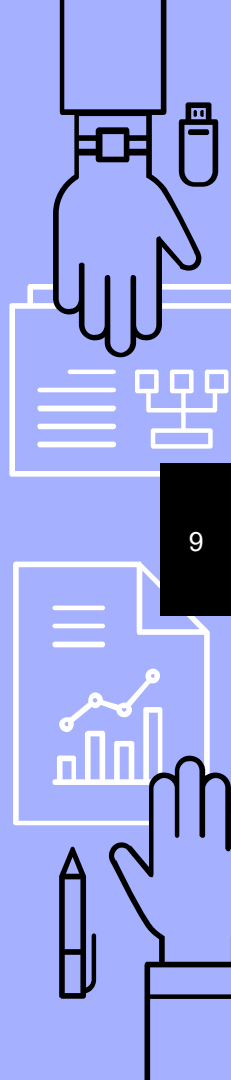
**HINT:**  
Smart indent: "Ctrl+I"



# Nested for-loop

- ▶ Execute element-by-element matrix operations

```
A=rand(5,6);  
B=rand(5,6);  
for i=1:size(A,1)  
    for j=1:size(A,2)  
        C(i,j)=A(i,j)*B(i,j);  
    end  
end
```





# Recursive Data Import

- ▶ **dir** List directory

**D=dir(FOLDERNAME)**

returns the results in an M-by-1 structure with the fields:

name -- Filename

date -- Modification date

bytes -- Number of bytes allocated to the file

isdir -- 1 if name is a directory and 0 if not

datenum -- Modification date as a MATLAB serial date number.

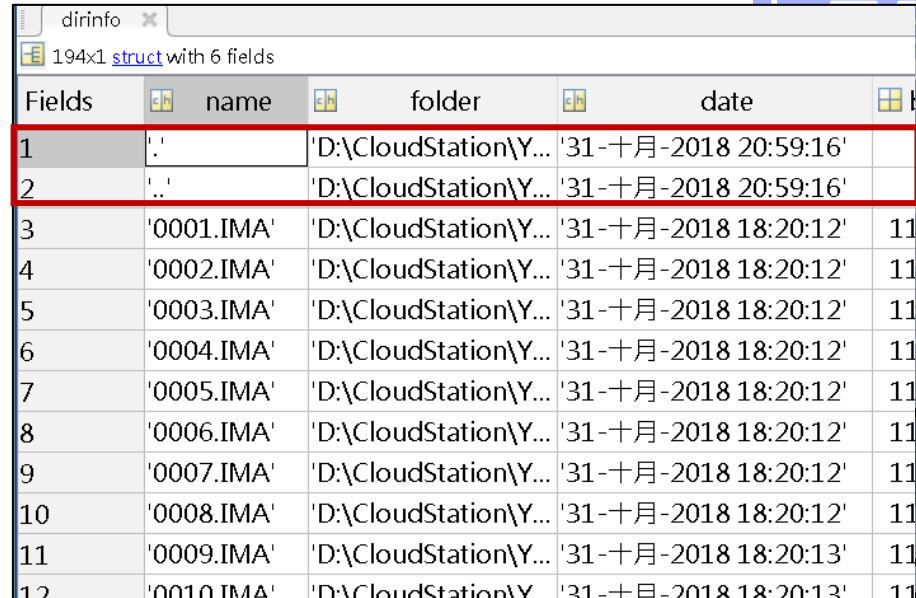
# Directory list

- ▶ `dirinfo=dir('./data\MRI_3DT1');`
- ▶ Watch out the first 2 elements!!
- ▶ `.` is the current folder.
- ▶ `..` is the upper folder. (上一層)

Note:

```
dirpath='./data/MRI_3DT1';  
%for mac OS, replace \ by /
```

194x1 [struct](#) with 6 fields



dirinfo x

194x1 struct with 6 fields

Fields	name	folder	date	
1	'.'	'D:\CloudStation\Y...	'31-十月-2018 20:59:16'	
2	'..'	'D:\CloudStation\Y...	'31-十月-2018 20:59:16'	
3	'0001.IMA'	'D:\CloudStation\Y...	'31-十月-2018 18:20:12'	11
4	'0002.IMA'	'D:\CloudStation\Y...	'31-十月-2018 18:20:12'	11
5	'0003.IMA'	'D:\CloudStation\Y...	'31-十月-2018 18:20:12'	11
6	'0004.IMA'	'D:\CloudStation\Y...	'31-十月-2018 18:20:12'	11
7	'0005.IMA'	'D:\CloudStation\Y...	'31-十月-2018 18:20:12'	11
8	'0006.IMA'	'D:\CloudStation\Y...	'31-十月-2018 18:20:12'	11
9	'0007.IMA'	'D:\CloudStation\Y...	'31-十月-2018 18:20:12'	11
10	'0008.IMA'	'D:\CloudStation\Y...	'31-十月-2018 18:20:12'	11
11	'0009.IMA'	'D:\CloudStation\Y...	'31-十月-2018 18:20:13'	11
12	'0010.IMA'	'D:\CloudStation\Y...	'31-十月-2018 18:20:13'	11



# Recursive dicomread

```
dirpath='.\data\MRI_3DT1';  
dirinfo=dir(dirpath);  
dirinfo(1:2)=[]; % remove . and ..  
img=[];  
for i=1:length(dirinfo)  
    img(:,:,i)=dicomread([dirpath filesep dirinfo(i).name]);  
end
```

Note:

```
dirpath='./data/MRI_3DT1';  
%for mac OS, replace \ by /
```

Note:

```
dirinfo(1:3)=[];  
%for mac OS, remove ., .., and .DS_Store
```



# Recursive imshow

```
figure,  
for i=1:size(img,3)  
    imshow(img(:,:,i),[],'border','tight','InitialMagnification',300)  
    pause(0.01)  
end
```

MImaterials\_L7\ImageEx07\_1.m

# While-loop Structure

**while** condition

Statement

**end**

**HINT:**

Use “Ctrl+c” to interrupt the current work in MATLAB.

```
total=0;
for i=1:100
    total=total+i;
end
```

```
total=0;
i=1;
while i<=100
    total=total+i;
    i=i+1;
end
```



# Recursive dicomread

```
dirpath='.\data\MRI_3DT1';  
dirinfo=dir(dirpath);  
dirinfo(1:2)=[]; % remove . and ..  
img=[];  
i=1;  
while i<=length(dirinfo)  
    img(:,:,i)=dicomread([dirpath filesep dirinfo(i).name]);  
    i=i+1;  
end
```

Note:

```
dirpath='./data/MRI_3DT1';  
%for mac OS, replace \ by /
```



# Recursive imshow

figure,

**i=1;**

**while** i <= size(img,3)

    imshow(img(:,:,i),[],**'border','tight','InitialMagnification',300**)

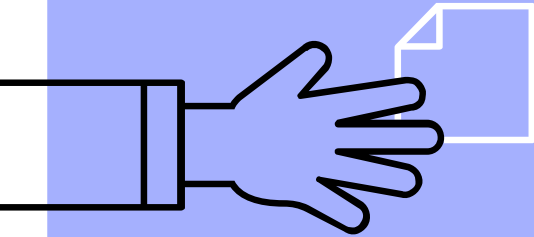
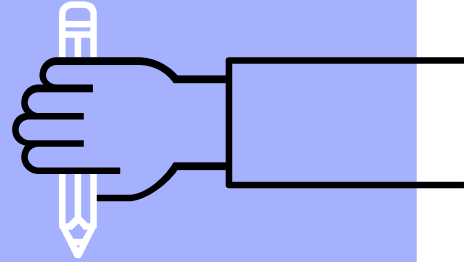
**pause(0.01)**

**i=i+1;**

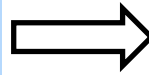
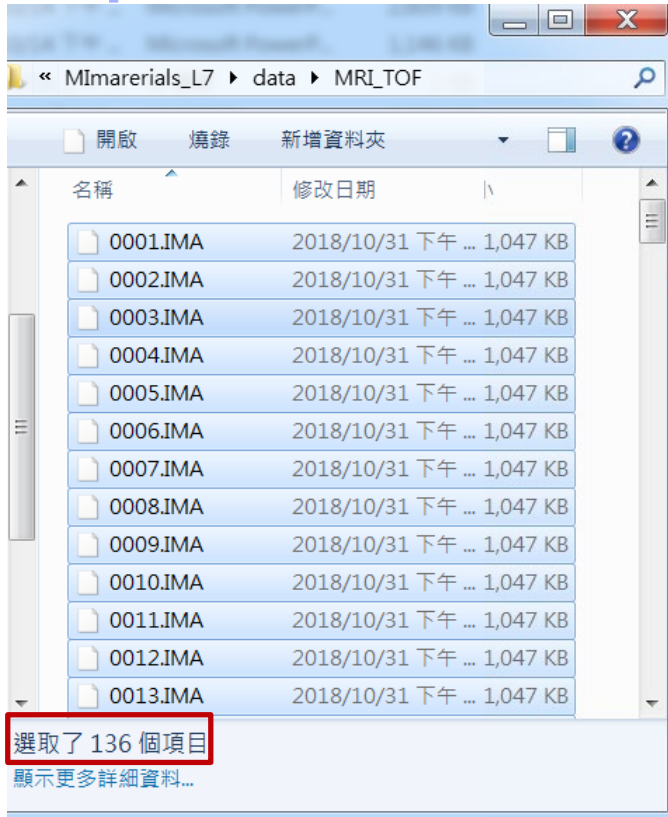
**end**



# Import and process of 3D images



# Step 1 – Read MRI TOF images



MRI  
TOF

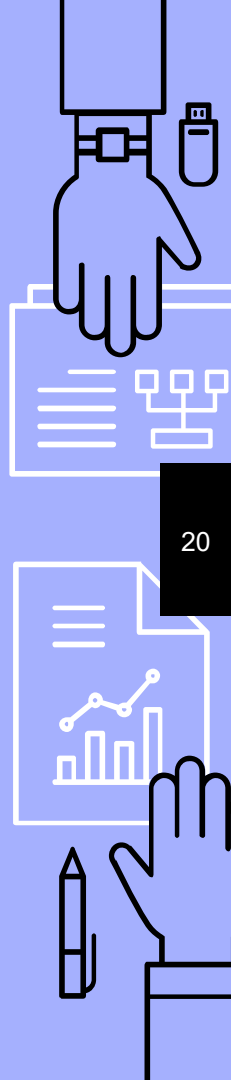
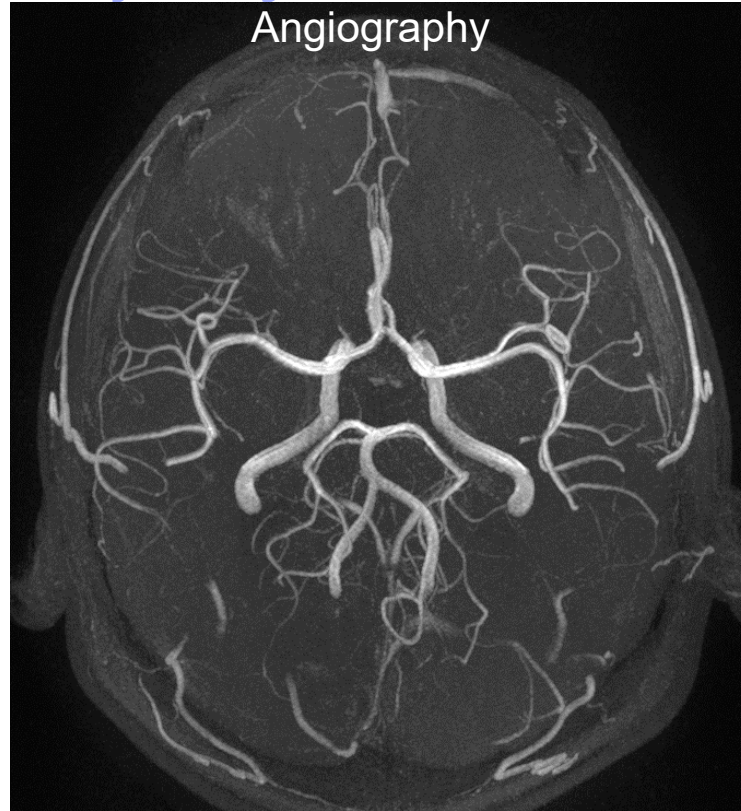
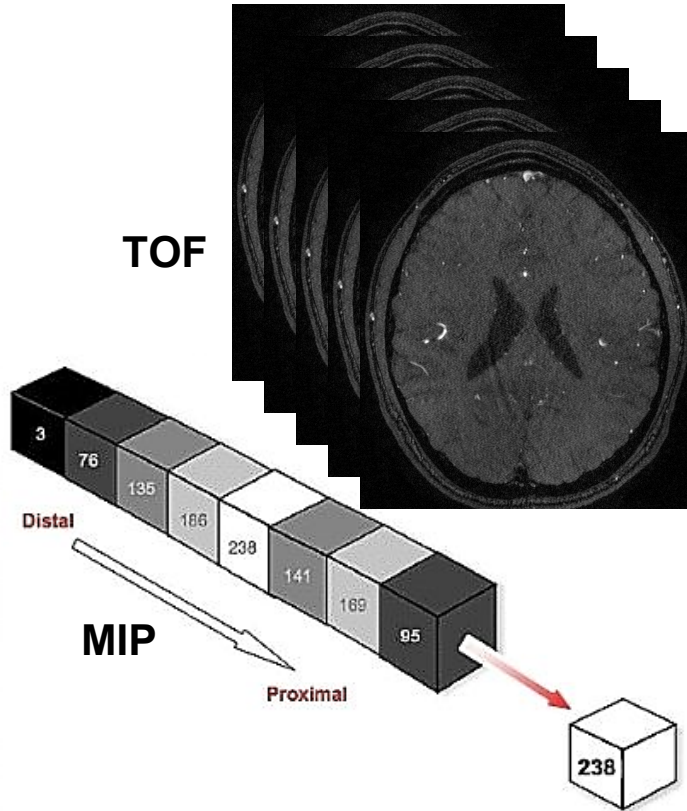
# Step 1 – Read MRI TOF images

```
4 %% Import sequential images into a variable
5 dirpath='./data\MRI_TOF';
6 dirinfo=dir(dirpath);
7 dirinfo(1:2)=[]; % remove . and ..
8 img=[];
9 for i=1:length(dirinfo)
10     img(:,:,i)=dicomread([dirpath filesep dirinfo(i).name]);
11 end
12
13 %% Display images slice by slice using for-loop
14 figure,
15 for i=1:size(img,3)
16     imshow(img(:,:,i),[],'border','tight')
17     pause(0.01)
18 end
```

Note:

```
dirpath='./data/MRI_TOF';
%for mac OS, replace \ by /
```

## Step 2 – Maximum Intensity Projection (MIP)

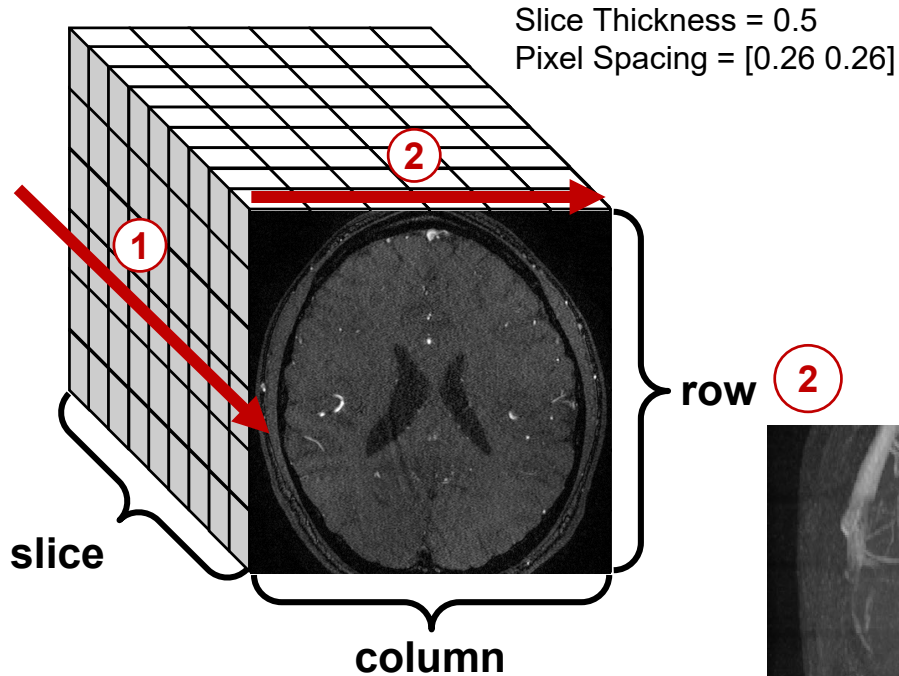


## Step 2 – Maximum Intensity Projection (MIP)

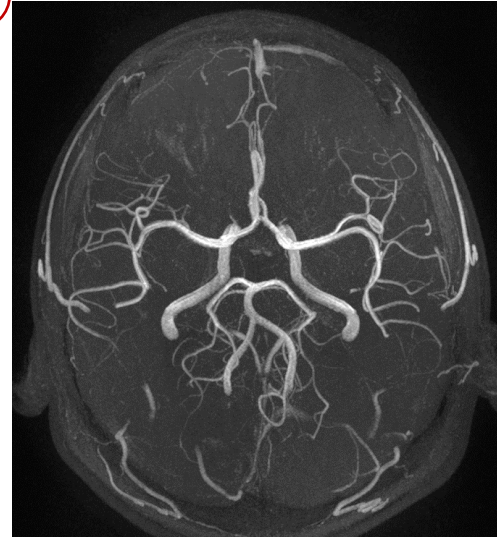
```
13 %% Display images slice by slice using for-loop
14 figure,
15 for i=1:size(img,3)
16     imshow(img(:,:,i),[],'border','tight')
17     pause(0.01)
18 end
19
20 %% Perform maximum intensity projection
21 MIPimg=max(img,[],3);
22 figure,imshow(MIPimg,[],'border','tight')
```

**help max to confirm its usage!**

# Step 3 – MIP along different directions



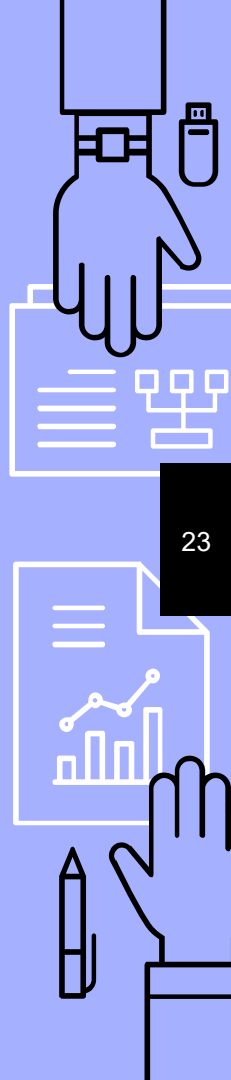
1





## Step 3 – MIP along different directions

- ▶ Useful functions
  - **squeeze** Remove singleton dimensions.
  - **flipud** Flip array in up/down direction.
  - **fliplr** Flip array in left/right direction.
  - **daspect** Data aspect ratio.



## Step 3 – MIP along different directions

```
24 %% Perform maximum intensity projection
25 - info=dicominfo([dirpath filesep dirinfo(1).name]);
26
27 - MIPimg=flipud(squeeze(max(img,[],2))); % projection along column direction
28 - figure,imshow(MIPimg,[],'border','tight')
29 - daspect([1/info.PixelSpacing(2) 1/info.SliceThickness 1])
```



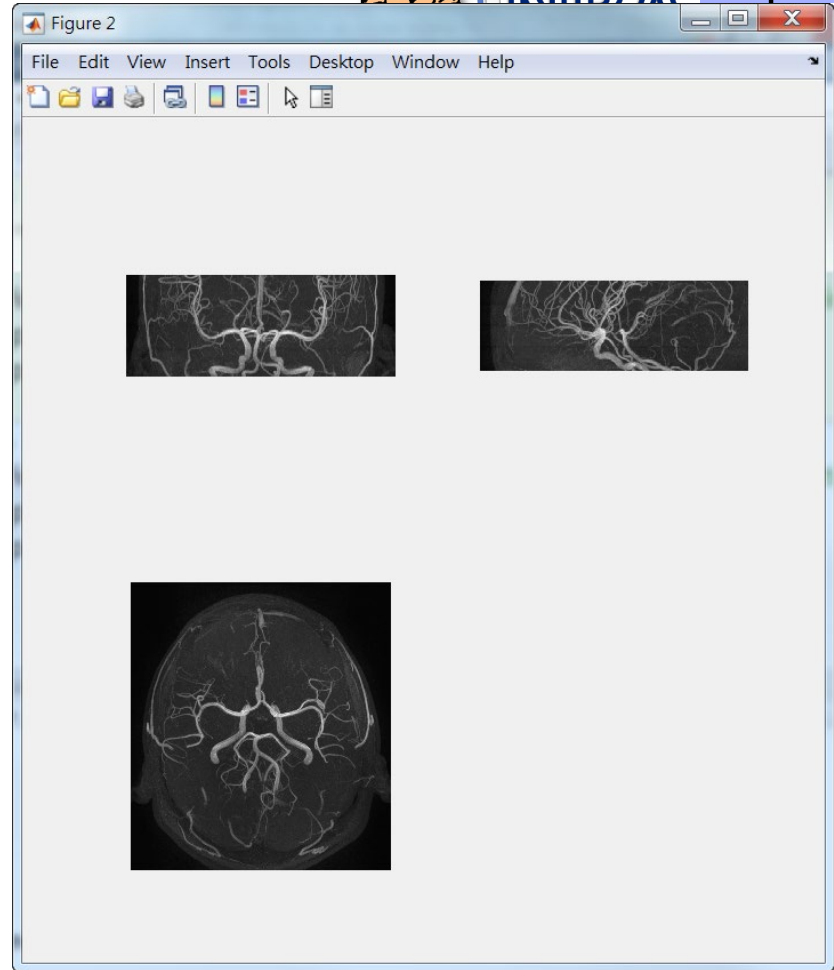


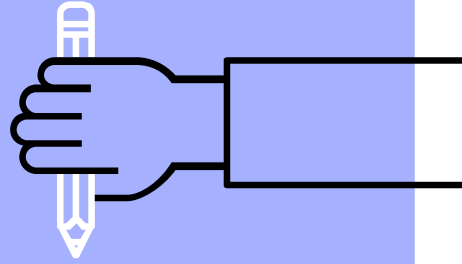


# Homework

- ▶ Perform MIP process along three directions, respectively.
- ▶ Display three MIP images using subplot as shown in the figure.

- **Modify from ImageEx07\_2.m**





THE END

[alvin4016@nycu.edu.tw](mailto:alvin4016@nycu.edu.tw)

