CS516 Medical Imaging

Final Project Part 1

Luyun Nie 002268087

Junjia Lin 002268506

In [1]:
```python
import os
from nipype.interfaces import fsl
import nibabel as nib
from skimage.filters import threshold_otsu
from matplotlib import pyplot as plt
import numpy as np
```

In [4]:
```python
# define a function transform the t1 to t2
def t1tot2(path):
    # transform the t1 to t2
    flt12 = fsl.FLIRT(out_matrix_file = str(path)+'/t1_to_t2.mat',out_file = str(path)+'/t1_to_t2.nii.gz')
    flt12.inputs.in_file = str(path)+'/t1.nii.gz'
    flt12.inputs.reference = str(path)+'/t2.nii.gz'
    flt12.inputs.output_type = "NIFTI_GZ"
    flt12.run()
```

In [3]:
```python
# scan the root file
go = os.walk(r"/Volumes/myDisk/final")
```

In [4]:
```python
# exceed the transformation in each sub file
for path,dir_list,file_list in go:
    for dir_name in dir_list:
        t1tot2(os.path.join(path,dir_name))
```

In [5]:
```python
# transform the t1 what has transformed to t2 to tof
def t1tot2totof(path):
    # transform the t1_to_t2 to tof
    flt12 = fsl.FLIRT(out_matrix_file = str(path)+'/t1_to_t2_to_tof.mat',out_file = str(path)+'/t1_to_t2_to_tof
    flt12.inputs.in_file = str(path)+'/t1_to_t2.nii.gz'
    flt12.inputs.reference = str(path)+'/tof.nii.gz'
    flt12.inputs.output_type = "NIFTI_GZ"
    flt12.run()
```

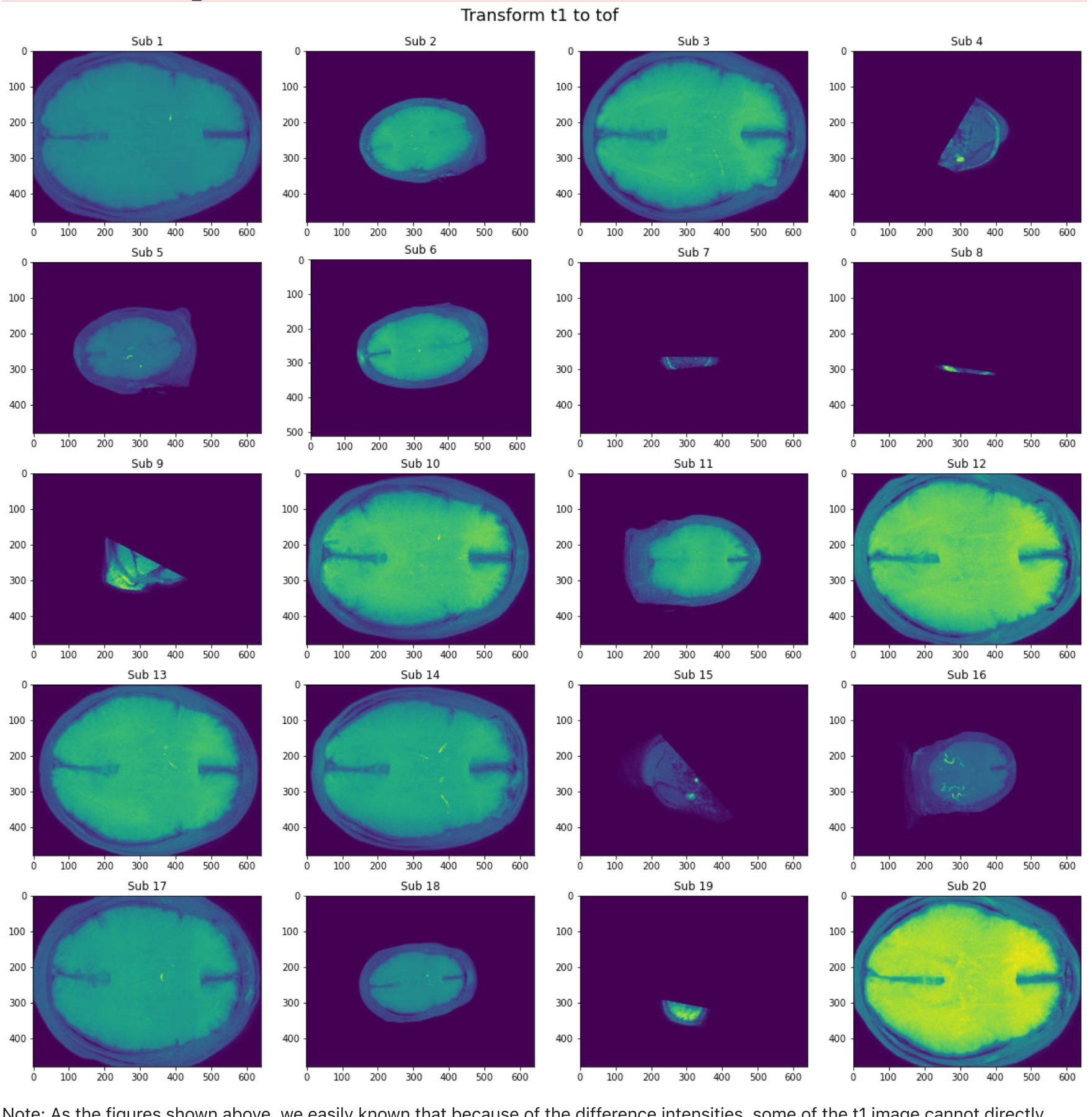
In [6]:
```python
# exceed the tranformation in each sub file
go = os.walk(r"/Volumes/myDisk/final")
for path,dir_list,file_list in go:
    for dir_name in dir_list:
        t1tot2totof(os.path.join(path,dir_name))
```

In [7]:
```python
# define a function plotting the results
def plott1tot2tot1f(path):
    img = nib.load(str(path)+'/t1_to_t2_to_tof.nii.gz')
    imgdata = img.get_data()
    i,j,k = imgdata.shape
    plt.imshow(np.max(imgdata[:,:,:],axis = 2))
```

In [8]:
```python
# plot the transformations
import os
go = os.walk(r"/Volumes/myDisk/final")
plt.figure(figsize = (20,20))
plt.suptitle('Transform t1 to tof',fontsize = 18, y=0.91)
plt.tight_layout()
m = 1
for path,dir_list,file_list in go:
    for dir_name in dir_list:
        plt.subplot(5,4,m)
        plott1tot2tot1f(os.path.join(path,dir_name))
        plt.title("Sub {}".format(m))
        m = m+1
```

```
<ipython-input-7-e3c56a3d81e5>:3: DeprecationWarning: get_data() is deprecated in favor of get_fdata(), which h
as a more predictable return type. To obtain get_data() behavior going forward, use numpy.asanyarray(img.dataob
j).

* deprecated from version: 3.0
* Will raise <class 'nibabel.deprecator.ExpiredDeprecationError'> as of version: 5.0
  imgdata = img.get_data()
```



Note: As the figures shown above, we easily known that because of the difference intensities, some of the t1 image cannot directly transform to tof. Thus, we came out an idea that a threshold should be applied to the tof to average the tof image.

In [25]:
```python
# threshold the sub-02 tof image
path2 = r"/Volumes/myDisk/final/sub-02"
thresh = fsl.Threshold(direction = 'above',out_file = str(path2)+'/tof_thresh.nii.gz')
thresh.inputs.in_file = str(path2)+'/tof.nii.gz'
thresh.inputs.thresh = 130
thresh.run()
```

Out[25]: `<nipype.interfaces.base.support.InterfaceResult at 0x7fb15f2468e0>`

In [26]:
```python
# transform the sub-02 t1_to_t2_to_tof image
flt2f = fsl.FLIRT(out_matrix_file = str(path2)+'/t1_to_t2_to_tof.mat',out_file = str(path2)+'/t1_to_t2_to_tof.n
flt2f.inputs.in_file = str(path2)+'/t1_to_t2.nii.gz'
flt2f.inputs.reference = str(path2)+'/tof_thresh.nii.gz'
flt2f.inputs.output_type = "NIFTI_GZ"
flt2f.run()
```

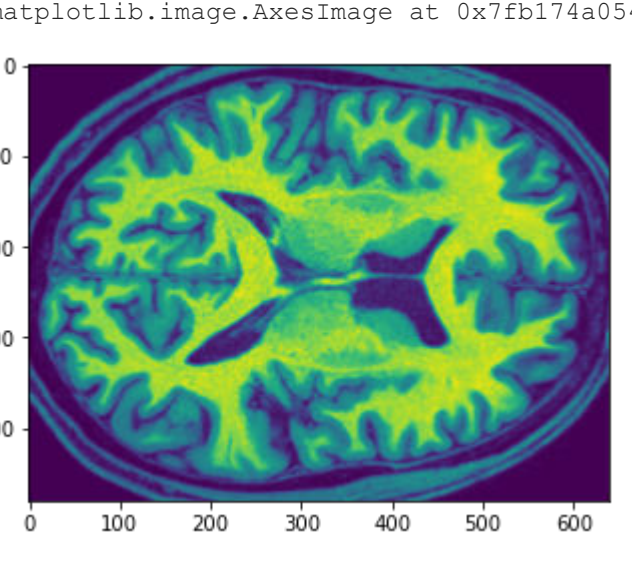
Out[26]: `<nipype.interfaces.base.support.InterfaceResult at 0x7fb15f2465e0>`

In [28]:
```python
# load the image and show
img = nib.load(str(path2)+'/t1_to_t2_to_tof.nii.gz').get_data()
plt.imshow(img[:,:,100])
```

```
<ipython-input-28-b6023e29c890>:1: DeprecationWarning: get_data() is deprecated in favor of get_fdata(), which
has a more predictable return type. To obtain get_data() behavior going forward, use numpy.asanyarray(img.datao
bj).

* deprecated from version: 3.0
* Will raise <class 'nibabel.deprecator.ExpiredDeprecationError'> as of version: 5.0
  img = nib.load(str(path2)+'/t1_to_t2_to_tof.nii.gz').get_data()
```

Out[28]: `<matplotlib.image.AxesImage at 0x7fb174a05490>`



Note: We built the tof image's histogram bins to analyze its intensity. We found that the correct threshold value was always beween the 24th and 40th bins in the histogram. Such that, we guess each tof image's threshold and guess, which cost us a lot of time.

In [31]:
```python
# same as sub-02
path3 = r"/Volumes/myDisk/final/sub-03"
thresh = fsl.Threshold(direction = 'above',out_file = str(path3)+'/tof_thresh.nii.gz')
thresh.inputs.in_file = str(path3)+'/tof.nii.gz'
thresh.inputs.thresh = 202.5
thresh.run()
```

Out[31]: `<nipype.interfaces.base.support.InterfaceResult at 0x7fb15e344700>`

In [32]:
```python
flt2f = fsl.FLIRT(out_matrix_file = str(path3)+'/t1_to_t2_to_tof.mat',out_file = str(path3)+'/t1_to_t2_to_tof.n
flt2f.inputs.in_file = str(path3)+'/t1_to_t2.nii.gz'
flt2f.inputs.reference = str(path3)+'/tof_thresh.nii.gz'
flt2f.inputs.output_type = "NIFTI_GZ"
flt2f.run()
```

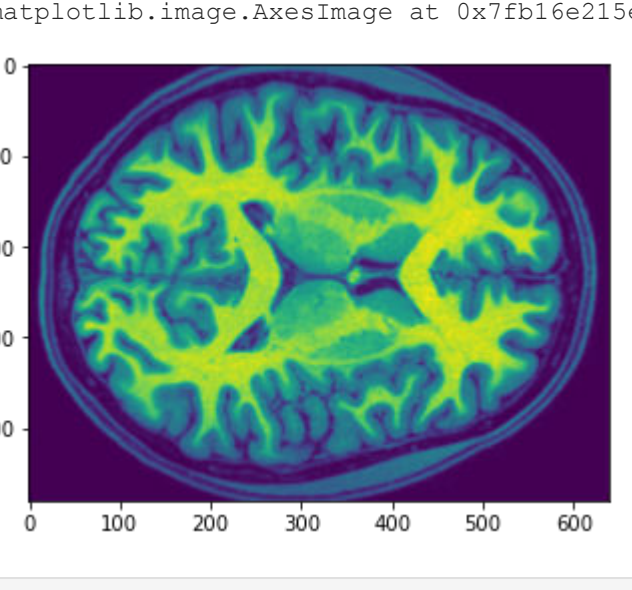
Out[32]: `<nipype.interfaces.base.support.InterfaceResult at 0x7fb15e344190>`

In [34]:
```python
img = nib.load(str(path3)+'/t1_to_t2_to_tof.nii.gz').get_data()
plt.imshow(img[:,:,100])
```

```
<ipython-input-34-99aa9fa7a799>:1: DeprecationWarning: get_data() is deprecated in favor of get_fdata(), which
has a more predictable return type. To obtain get_data() behavior going forward, use numpy.asanyarray(img.datao
bj).

* deprecated from version: 3.0
* Will raise <class 'nibabel.deprecator.ExpiredDeprecationError'> as of version: 5.0
  img = nib.load(str(path3)+'/t1_to_t2_to_tof.nii.gz').get_data()
```

Out[34]: `<matplotlib.image.AxesImage at 0x7fb16e215e80>`



In [35]:
```python
# same as sub-02
path4 = r"/Volumes/myDisk/final/sub-04"
thresh = fsl.Threshold(direction = 'above',out_file = str(path4)+'/tof_thresh.nii.gz')
thresh.inputs.in_file = str(path4)+'/tof.nii.gz'
thresh.inputs.thresh = 202
thresh.run()
```

Out[35]: `<nipype.interfaces.base.support.InterfaceResult at 0x7fb16ea5ea60>`

In [36]:
```python
flt2f = fsl.FLIRT(out_matrix_file = str(path4)+'/t1_to_t2_to_tof.mat',out_file = str(path4)+'/t1_to_t2_to_tof.n
flt2f.inputs.in_file = str(path4)+'/t1_to_t2.nii.gz'
flt2f.inputs.reference = str(path4)+'/tof_thresh.nii.gz'
flt2f.inputs.output_type = "NIFTI_GZ"
flt2f.run()
```

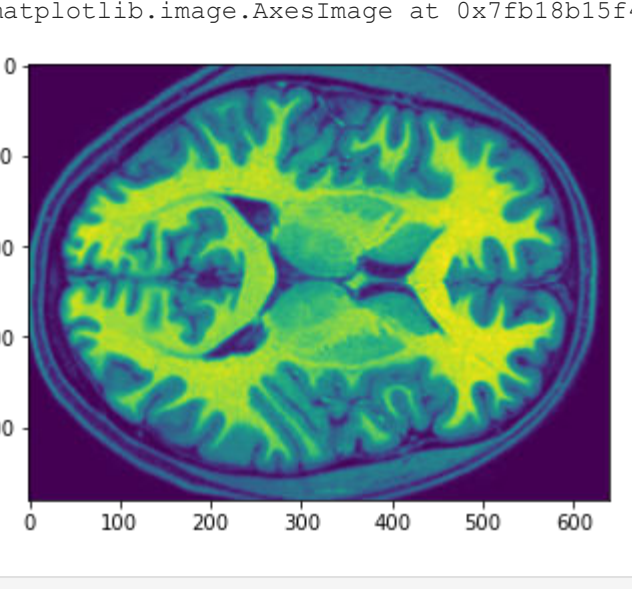
Out[36]: `<nipype.interfaces.base.support.InterfaceResult at 0x7fb15f54eb80>`

In [37]:
```python
img = nib.load(str(path4)+'/t1_to_t2_to_tof.nii.gz').get_data()
plt.imshow(img[:,:,100])
```

```
<ipython-input-37-048fdeeaf531>:1: DeprecationWarning: get_data() is deprecated in favor of get_fdata(), which
has a more predictable return type. To obtain get_data() behavior going forward, use numpy.asanyarray(img.datao
bj).

* deprecated from version: 3.0
* Will raise <class 'nibabel.deprecator.ExpiredDeprecationError'> as of version: 5.0
  img = nib.load(str(path4)+'/t1_to_t2_to_tof.nii.gz').get_data()
```

Out[37]: `<matplotlib.image.AxesImage at 0x7fb18b15f430>`



In [38]:
```python
# same as sub-02
path5 = r"/Volumes/myDisk/final/sub-05"
thresh = fsl.Threshold(direction = 'above',out_file = str(path5)+'/tof_thresh.nii.gz')
thresh.inputs.in_file = str(path5)+'/tof.nii.gz'
thresh.inputs.thresh = 180
thresh.run()
```

Out[38]: `<nipype.interfaces.base.support.InterfaceResult at 0x7fb17577cf40>`

In [39]:
```python
flt2f = fsl.FLIRT(out_matrix_file = str(path5)+'/t1_to_t2_to_tof.mat',out_file = str(path5)+'/t1_to_t2_to_tof.n
flt2f.inputs.in_file = str(path5)+'/t1_to_t2.nii.gz'
flt2f.inputs.reference = str(path5)+'/tof_thresh.nii.gz'
flt2f.inputs.output_type = "NIFTI_GZ"
flt2f.run()
```

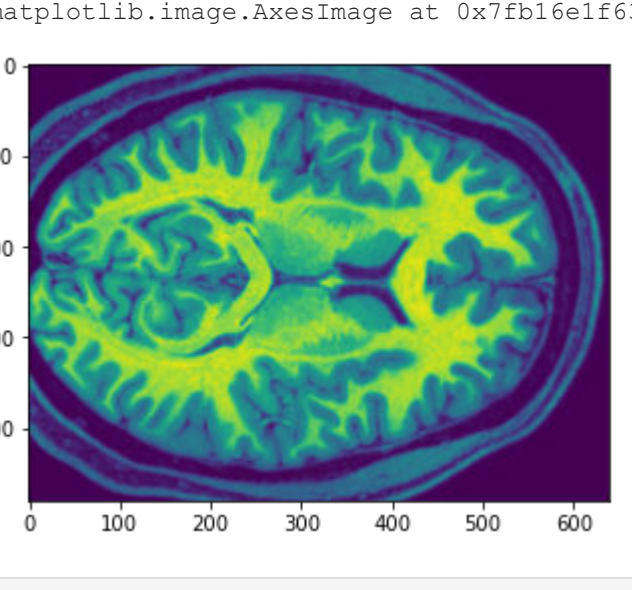
Out[39]: `<nipype.interfaces.base.support.InterfaceResult at 0x7fb15e7e0e50>`

In [40]:
```python
img = nib.load(str(path5)+'/t1_to_t2_to_tof.nii.gz').get_data()
plt.imshow(img[:,:,100])
```

```
<ipython-input-40-89f7a810b28e>:1: DeprecationWarning: get_data() is deprecated in favor of get_fdata(), which
has a more predictable return type. To obtain get_data() behavior going forward, use numpy.asanyarray(img.datao
bj).

* deprecated from version: 3.0
* Will raise <class 'nibabel.deprecator.ExpiredDeprecationError'> as of version: 5.0
  img = nib.load(str(path5)+'/t1_to_t2_to_tof.nii.gz').get_data()
```

Out[40]: `<matplotlib.image.AxesImage at 0x7fb16e1f63a0>`



In [43]:
```python
# same as sub-02
path6 = r"/Volumes/myDisk/final/sub-06"
thresh = fsl.Threshold(direction = 'above',out_file = str(path6)+'/tof_thresh.nii.gz')
thresh.inputs.in_file = str(path6)+'/tof.nii.gz'
thresh.inputs.thresh = 220
thresh.run()
```

Out[43]: `<nipype.interfaces.base.support.InterfaceResult at 0x7fb16ea5e2e0>`

In [44]:
```python
flt2f = fsl.FLIRT(out_matrix_file = str(path6)+'/t1_to_t2_to_tof.mat',out_file = str(path6)+'/t1_to_t2_to_tof.n
flt2f.inputs.in_file = str(path6)+'/t1_to_t2.nii.gz'
flt2f.inputs.reference = str(path6)+'/tof_thresh.nii.gz'
flt2f.inputs.output_type = "NIFTI_GZ"
flt2f.run()
```

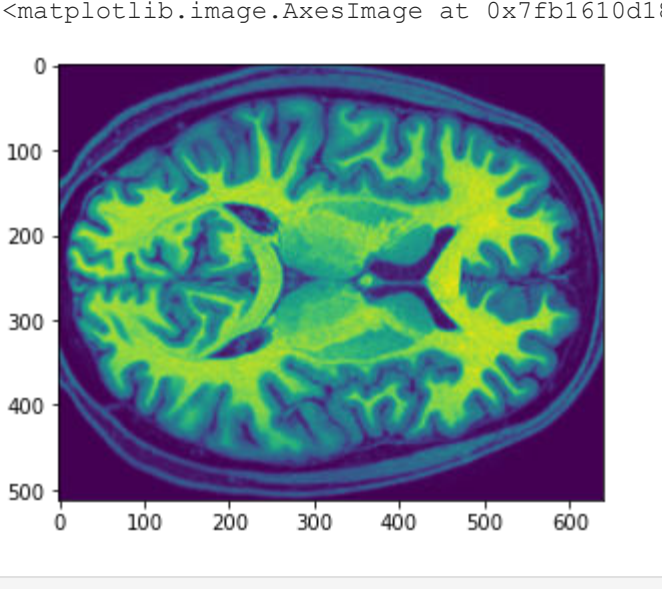
Out[44]: `<nipype.interfaces.base.support.InterfaceResult at 0x7fb15ef106a0>`

In [45]:
```python
img = nib.load(str(path6)+'/t1_to_t2_to_tof.nii.gz').get_data()
plt.imshow(img[:,:,100])
```

```
<ipython-input-45-362b15155a25>:1: DeprecationWarning: get_data() is deprecated in favor of get_fdata(), which
has a more predictable return type. To obtain get_data() behavior going forward, use numpy.asanyarray(img.datao
bj).

* deprecated from version: 3.0
* Will raise <class 'nibabel.deprecator.ExpiredDeprecationError'> as of version: 5.0
  img = nib.load(str(path6)+'/t1_to_t2_to_tof.nii.gz').get_data()
```

Out[45]: `<matplotlib.image.AxesImage at 0x7fb1610d18b0>`



In [51]:
```python
# same as sub-02
path7 = r"/Volumes/myDisk/final/sub-07"
thresh = fsl.Threshold(direction = 'above',out_file = str(path7)+'/tof_thresh.nii.gz')
thresh.inputs.in_file = str(path7)+'/tof.nii.gz'
thresh.inputs.thresh = 130
thresh.run()
```

Out[51]: `<nipype.interfaces.base.support.InterfaceResult at 0x7fb160a81eb0>`

In [52]:
```python
flt2f = fsl.FLIRT(out_matrix_file = str(path7)+'/t1_to_t2_to_tof.mat',out_file = str(path7)+'/t1_to_t2_to_tof.n
flt2f.inputs.in_file = str(path7)+'/t1_to_t2.nii.gz'
flt2f.inputs.reference = str(path7)+'/tof_thresh.nii.gz'
flt2f.inputs.output_type = "NIFTI_GZ"
flt2f.run()
```

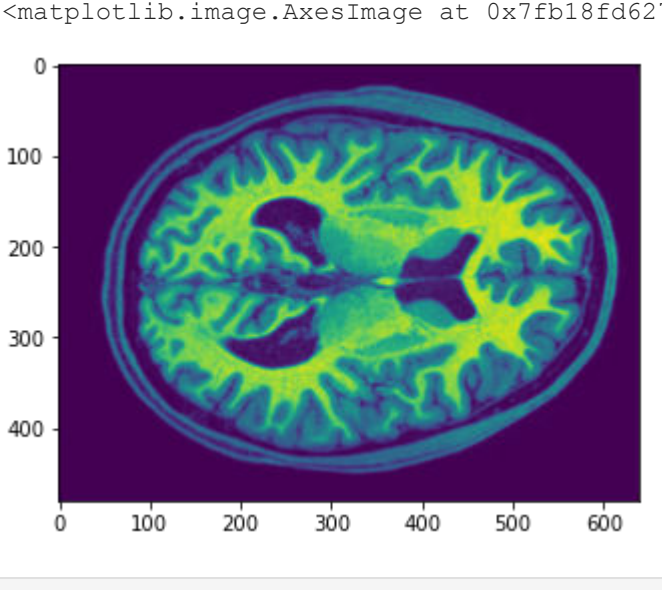
Out[52]: `<nipype.interfaces.base.support.InterfaceResult at 0x7fb16e273550>`

In [53]:
```python
img = nib.load(str(path7)+'/t1_to_t2_to_tof.nii.gz').get_data()
plt.imshow(img[:,:,100])
```

```
<ipython-input-53-cf5855e9b805>:1: DeprecationWarning: get_data() is deprecated in favor of get_fdata(), which
has a more predictable return type. To obtain get_data() behavior going forward, use numpy.asanyarray(img.datao
bj).

* deprecated from version: 3.0
* Will raise <class 'nibabel.deprecator.ExpiredDeprecationError'> as of version: 5.0
  img = nib.load(str(path7)+'/t1_to_t2_to_tof.nii.gz').get_data()
```

Out[53]: `<matplotlib.image.AxesImage at 0x7fb18fd62700>`



In [54]:
```python
# same as sub-02
path8 = r"/Volumes/myDisk/final/sub-08"
thresh = fsl.Threshold(direction = 'above',out_file = str(path8)+'/tof_thresh.nii.gz')
thresh.inputs.in_file = str(path8)+'/tof.nii.gz'
thresh.inputs.thresh = 170
thresh.run()
```

Out[54]: `<nipype.interfaces.base.support.InterfaceResult at 0x7fb16083b970>`

In [55]:
```python
flt2f = fsl.FLIRT(out_matrix_file = str(path8)+'/t1_to_t2_to_tof.mat',out_file = str(path8)+'/t1_to_t2_to_tof.n
flt2f.inputs.in_file = str(path8)+'/t1_to_t2.nii.gz'
flt2f.inputs.reference = str(path8)+'/tof_thresh.nii.gz'
flt2f.inputs.output_type = "NIFTI_GZ"
flt2f.run()
```

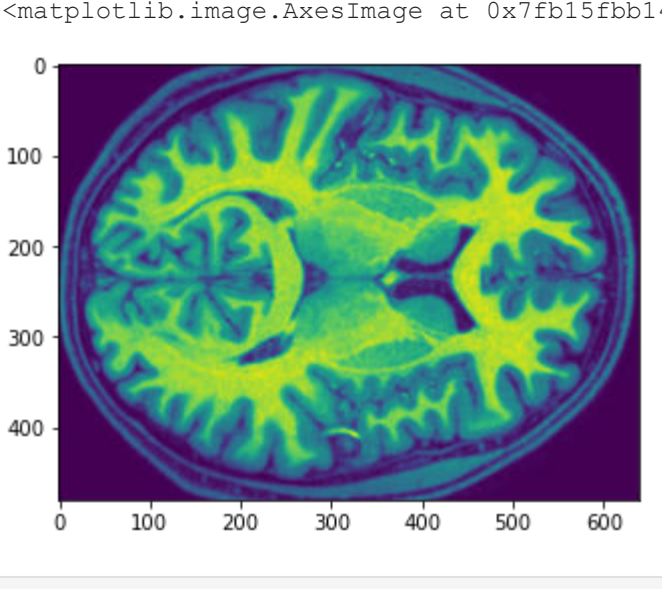
Out[55]: `<nipype.interfaces.base.support.InterfaceResult at 0x7fb16e1fe790>`

In [56]:
```python
img = nib.load(str(path8)+'/t1_to_t2_to_tof.nii.gz').get_data()
plt.imshow(img[:,:,100])
```

```
<ipython-input-56-ac2c1e7b5825>:1: DeprecationWarning: get_data() is deprecated in favor of get_fdata(), which
has a more predictable return type. To obtain get_data() behavior going forward, use numpy.asanyarray(img.datao
bj).

* deprecated from version: 3.0
* Will raise <class 'nibabel.deprecator.ExpiredDeprecationError'> as of version: 5.0
  img = nib.load(str(path8)+'/t1_to_t2_to_tof.nii.gz').get_data()
```

Out[56]: `<matplotlib.image.AxesImage at 0x7fb15fbb1460>`



In [57]:
```python
# same as sub-02
path9 = r"/Volumes/myDisk/final/sub-09"
thresh = fsl.Threshold(direction = 'above',out_file = str(path9)+'/tof_thresh.nii.gz')
thresh.inputs.in_file = str(path9)+'/tof.nii.gz'
thresh.inputs.thresh = 200
thresh.run()
```

Out[57]: `<nipype.interfaces.base.support.InterfaceResult at 0x7fb18b16e100>`

In [58]:
```python
flt2f = fsl.FLIRT(out_matrix_file = str(path9)+'/t1_to_t2_to_tof.mat',out_file = str(path9)+'/t1_to_t2_to_tof.n
flt2f.inputs.in_file = str(path9)+'/t1_to_t2.nii.gz'
flt2f.inputs.reference = str(path9)+'/tof_thresh.nii.gz'
flt2f.inputs.output_type = "NIFTI_GZ"
flt2f.run()
```

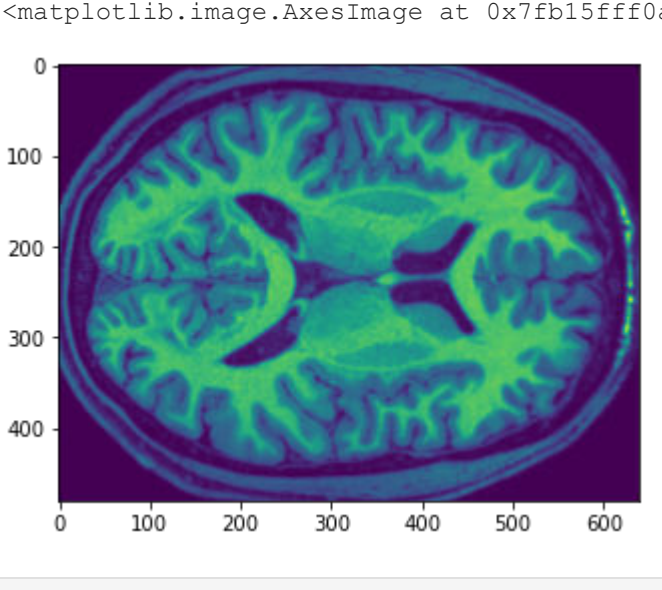
Out[58]: `<nipype.interfaces.base.support.InterfaceResult at 0x7fb175575b80>`

In [59]:
```python
img = nib.load(str(path9)+'/t1_to_t2_to_tof.nii.gz').get_data()
plt.imshow(img[:,:,100])
```

```
<ipython-input-59-045266dfa90b>:1: DeprecationWarning: get_data() is deprecated in favor of get_fdata(), which
has a more predictable return type. To obtain get_data() behavior going forward, use numpy.asanyarray(img.datao
bj).

* deprecated from version: 3.0
* Will raise <class 'nibabel.deprecator.ExpiredDeprecationError'> as of version: 5.0
  img = nib.load(str(path9)+'/t1_to_t2_to_tof.nii.gz').get_data()
```

Out[59]: `<matplotlib.image.AxesImage at 0x7fb15fff0a30>`



In [60]:
```python
# same as sub-02
path11 = '/Volumes/myDisk/final/sub-11'
thresh = fsl.Threshold(direction = 'above',out_file = str(path11)+'/tof_thresh.nii.gz')
thresh.inputs.in_file = str(path11)+'/tof.nii.gz'
thresh.inputs.thresh = 220
thresh.run()
```

Out[60]: `<nipype.interfaces.base.support.InterfaceResult at 0x7fb16083b0d0>`

In [61]:
```python
flt2f = fsl.FLIRT(out_matrix_file = str(path11)+'/t1_to_t2_to_tof.mat',out_file = str(path11)+'/t1_to_t2_to_tof
flt2f.inputs.in_file = str(path11)+'/t1_to_t2.nii.gz'
flt2f.inputs.reference = str(path11)+'/tof_thresh.nii.gz'
flt2f.inputs.output_type = "NIFTI_GZ"
flt2f.run()
```

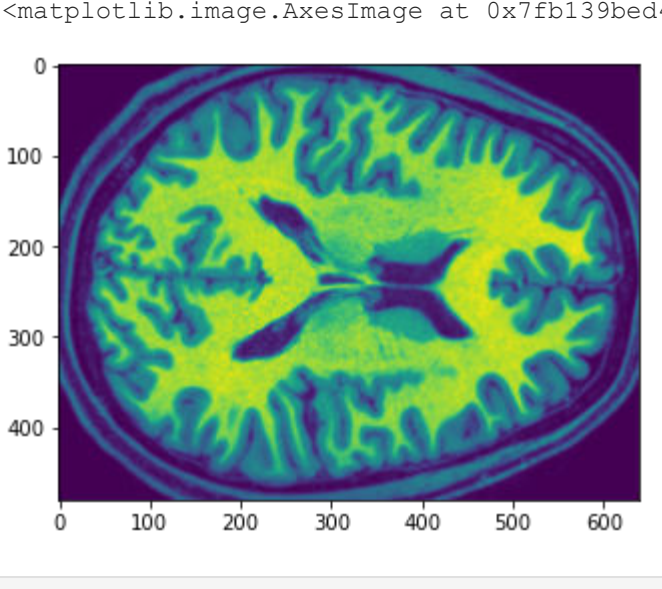
Out[61]: `<nipype.interfaces.base.support.InterfaceResult at 0x7fb17554ea60>`

In [62]:
```python
img = nib.load(str(path11)+'/t1_to_t2_to_tof.nii.gz').get_data()
plt.imshow(img[:,:,100])
```

```
<ipython-input-62-bda13e3e26fd>:1: DeprecationWarning: get_data() is deprecated in favor of get_fdata(), which
has a more predictable return type. To obtain get_data() behavior going forward, use numpy.asanyarray(img.datao
bj).

* deprecated from version: 3.0
* Will raise <class 'nibabel.deprecator.ExpiredDeprecationError'> as of version: 5.0
  img = nib.load(str(path11)+'/t1_to_t2_to_tof.nii.gz').get_data()
```

Out[62]: `<matplotlib.image.AxesImage at 0x7fb139bed490>`



In [65]:
```python
# same as sub-02
path15 = '/Volumes/myDisk/final/sub-15'
thresh = fsl.Threshold(direction = 'above',out_file = str(path15)+'/tof_thresh.nii.gz')
thresh.inputs.in_file = str(path15)+'/tof.nii.gz'
thresh.inputs.thresh = 160
thresh.run()
```

Out[65]: `<nipype.interfaces.base.support.InterfaceResult at 0x7fb139163e50>`

In [67]:
```python
flt2f = fsl.FLIRT(out_matrix_file = str(path15)+'/t1_to_t2_to_tof.mat',out_file = str(path15)+'/t1_to_t2_to_tof
flt2f.inputs.in_file = str(path15)+'/t1_to_t2.nii.gz'
flt2f.inputs.reference = str(path15)+'/tof_thresh.nii.gz'
flt2f.inputs.output_type = "NIFTI_GZ"
flt2f.run()
```

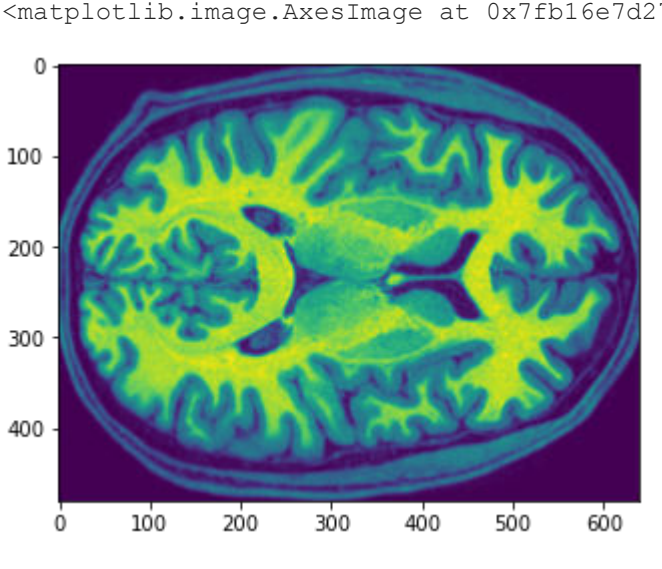
Out[67]: `<nipype.interfaces.base.support.InterfaceResult at 0x7fb15f4cb730>`

In [68]:
```python
img = nib.load(str(path15)+'/t1_to_t2_to_tof.nii.gz').get_data()
plt.imshow(img[:,:,100])
```

```
<ipython-input-68-38cdd18db510>:1: DeprecationWarning: get_data() is deprecated in favor of get_fdata(), which
has a more predictable return type. To obtain get_data() behavior going forward, use numpy.asanyarray(img.datao
bj).

* deprecated from version: 3.0
* Will raise <class 'nibabel.deprecator.ExpiredDeprecationError'> as of version: 5.0
  img = nib.load(str(path15)+'/t1_to_t2_to_tof.nii.gz').get_data()
```

Out[68]: `<matplotlib.image.AxesImage at 0x7fb16e7d27f0>`



In [69]:
```python
# same as sub-02
path16 = '/Volumes/myDisk/final/sub-16'
thresh = fsl.Threshold(direction = 'above',out_file = str(path16)+'/tof_thresh.nii.gz')
thresh.inputs.in_file = str(path16)+'/tof.nii.gz'
thresh.inputs.thresh = 165
thresh.run()
```

Out[69]: `<nipype.interfaces.base.support.InterfaceResult at 0x7fb15e4a9f10>`

In [70]:
```python
flt2f = fsl.FLIRT(out_matrix_file = str(path16)+'/t1_to_t2_to_tof.mat',out_file = str(path16)+'/t1_to_t2_to_tof
flt2f.inputs.in_file = str(path16)+'/t1_to_t2.nii.gz'
flt2f.inputs.reference = str(path16)+'/tof_thresh.nii.gz'
flt2f.inputs.output_type = "NIFTI_GZ"
flt2f.run()
```

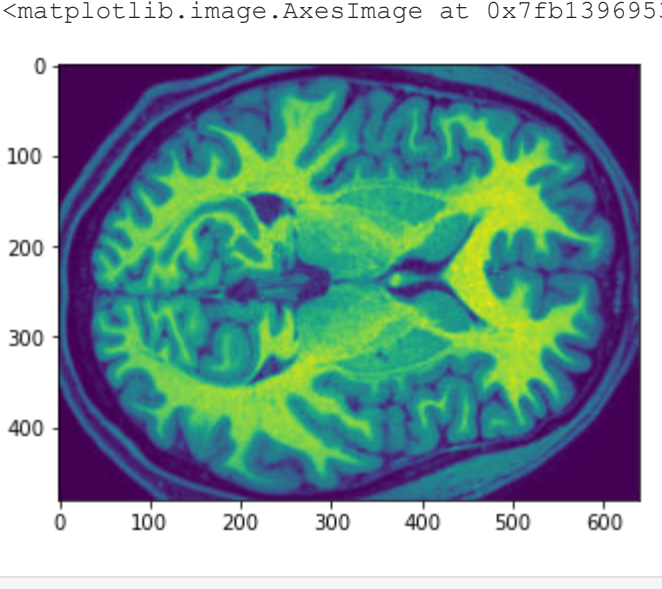
Out[70]: `<nipype.interfaces.base.support.InterfaceResult at 0x7fb15f8db4c0>`

In [71]:
```python
img = nib.load(str(path16)+'/t1_to_t2_to_tof.nii.gz').get_data()
plt.imshow(img[:,:,100])
```

```
<ipython-input-71-d65b07620a4f>:1: DeprecationWarning: get_data() is deprecated in favor of get_fdata(), which
has a more predictable return type. To obtain get_data() behavior going forward, use numpy.asanyarray(img.datao
bj).

* deprecated from version: 3.0
* Will raise <class 'nibabel.deprecator.ExpiredDeprecationError'> as of version: 5.0
  img = nib.load(str(path16)+'/t1_to_t2_to_tof.nii.gz').get_data()
```

Out[71]: `<matplotlib.image.AxesImage at 0x7fb1396953a0>`



In [72]:
```python
# same as sub-02
path18 = '/Volumes/myDisk/final/sub-18'
thresh = fsl.Threshold(direction = 'above',out_file = str(path18)+'/tof_thresh.nii.gz')
thresh.inputs.in_file = str(path18)+'/tof.nii.gz'
thresh.inputs.thresh = 60
thresh.run()
```

Out[72]: `<nipype.interfaces.base.support.InterfaceResult at 0x7fb16e251a90>`

In [73]:
```python
flt2f = fsl.FLIRT(out_matrix_file = str(path18)+'/t1_to_t2_to_tof.mat',out_file = str(path18)+'/t1_to_t2_to_tof
flt2f.inputs.in_file = str(path18)+'/t1_to_t2.nii.gz'
flt2f.inputs.reference = str(path18)+'/tof_thresh.nii.gz'
flt2f.inputs.output_type = "NIFTI_GZ"
flt2f.run()
```

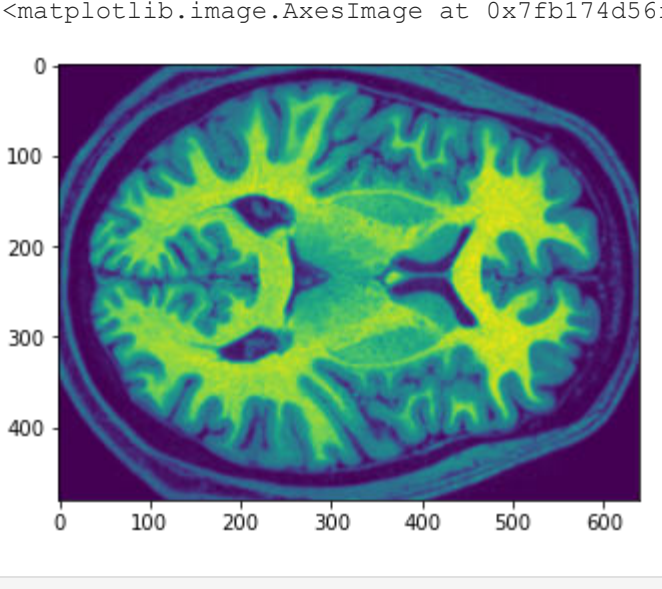
Out[73]: `<nipype.interfaces.base.support.InterfaceResult at 0x7fb1395cd640>`

In [74]:
```python
img = nib.load(str(path18)+'/t1_to_t2_to_tof.nii.gz').get_data()
plt.imshow(img[:,:,100])
```

```
<ipython-input-74-d3b6a2bdee62>:1: DeprecationWarning: get_data() is deprecated in favor of get_fdata(), which
has a more predictable return type. To obtain get_data() behavior going forward, use numpy.asanyarray(img.datao
bj).

* deprecated from version: 3.0
* Will raise <class 'nibabel.deprecator.ExpiredDeprecationError'> as of version: 5.0
  img = nib.load(str(path18)+'/t1_to_t2_to_tof.nii.gz').get_data()
```

Out[74]: `<matplotlib.image.AxesImage at 0x7fb174d56f70>`



In [75]:
```python
# same as sub-02
path19 = '/Volumes/myDisk/final/sub-19'
thresh = fsl.Threshold(direction = 'above',out_file = str(path19)+'/tof_thresh.nii.gz')
thresh.inputs.in_file = str(path19)+'/tof.nii.gz'
thresh.inputs.thresh = 60
thresh.run()
```

Out[75]: `<nipype.interfaces.base.support.InterfaceResult at 0x7fb16e251e80>`

In [76]:
```python
flt2f = fsl.FLIRT(out_matrix_file = str(path19)+'/t1_to_t2_to_tof.mat',out_file = str(path19)+'/t1_to_t2_to_tof
flt2f.inputs.in_file = str(path19)+'/t1_to_t2.nii.gz'
flt2f.inputs.reference = str(path19)+'/tof_thresh.nii.gz'
flt2f.inputs.output_type = "NIFTI_GZ"
flt2f.run()
```

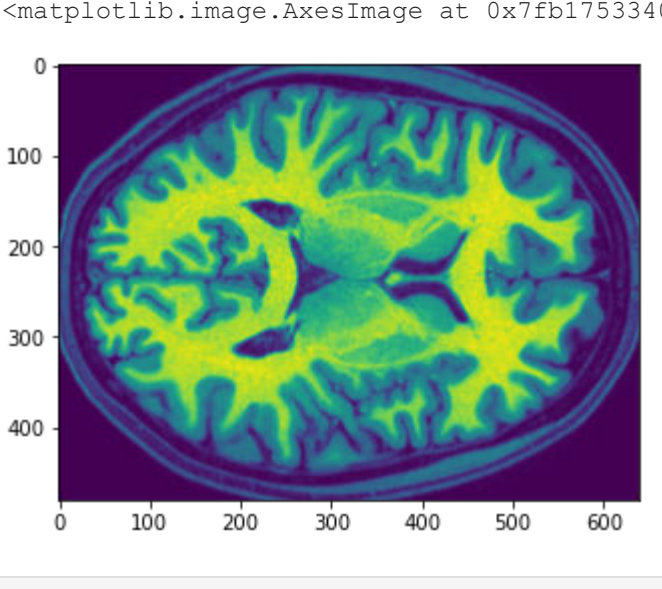
Out[76]: `<nipype.interfaces.base.support.InterfaceResult at 0x7fb16e2516d0>`

In [77]:
```python
img = nib.load(str(path19)+'/t1_to_t2_to_tof.nii.gz').get_data()
plt.imshow(img[:,:,100])
```

```
<ipython-input-77-f15e79b62e4e>:1: DeprecationWarning: get_data() is deprecated in favor of get_fdata(), which
has a more predictable return type. To obtain get_data() behavior going forward, use numpy.asanyarray(img.datao
bj).

* deprecated from version: 3.0
* Will raise <class 'nibabel.deprecator.ExpiredDeprecationError'> as of version: 5.0
  img = nib.load(str(path19)+'/t1_to_t2_to_tof.nii.gz').get_data()
```

Out[77]: `<matplotlib.image.AxesImage at 0x7fb1753340a0>`

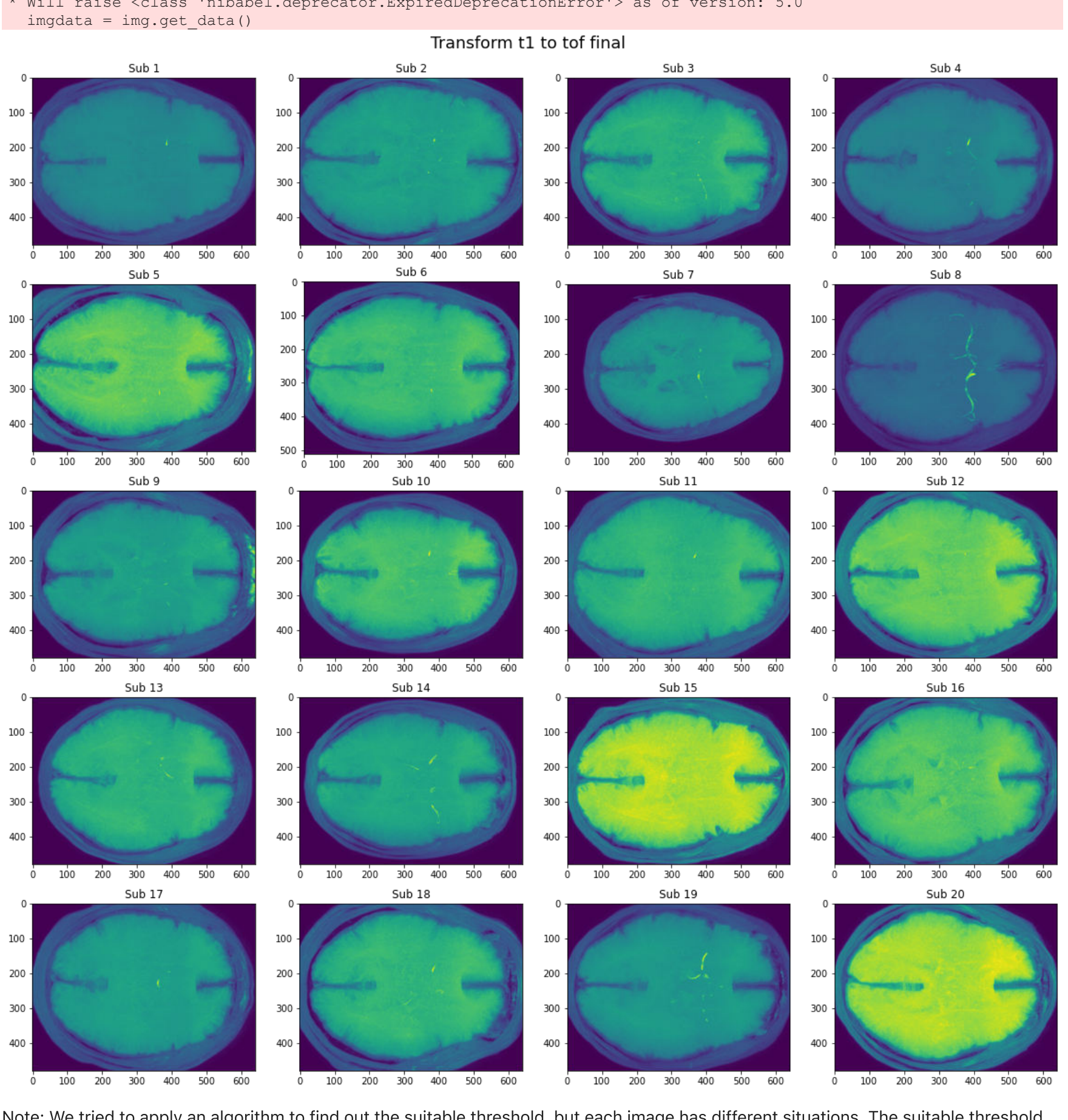


In [78]:
```python
# define a function to plot the transformed t1
def plott1t2totof(path):
    img = nib.load(str(path)+'/t1_to_t2_to_tof.nii.gz')
    imgdata = img.get_data()
    i,j,k = imgdata.shape
    plt.imshow(np.max(imgdata[:,:,:],axis = 2))
```

In [79]:
```python
# plot the images
import os
go = os.walk(r"/Volumes/myDisk/final")
plt.figure(figsize = (20,20))
plt.suptitle('Transform t1 to tof final',fontsize = 18, y=0.91)
plt.tight_layout()
m = 1
for path,dir_list,file_list in go:
    for dir_name in dir_list:
        plt.subplot(5,4,m)
        plott1t2totof(os.path.join(path,dir_name))
        plt.title("Sub {}".format(m))
        m = m+1
```

```
<ipython-input-78-4b4ad628eaaf>:3: DeprecationWarning: get_data() is deprecated in favor of get_fdata(), which
has a more predictable return type. To obtain get_data() behavior going forward, use numpy.asanyarray(img.datao
bj).

* deprecated from version: 3.0
* Will raise <class 'nibabel.deprecator.ExpiredDeprecationError'> as of version: 5.0
  imgdata = img.get_data()
```



Note: We tried to apply an algorithm to find out the suitable threshold, but each image has different situations. The suitable threshold value was between the 24th and the 40th image. A wrong theshold will lead to mistake transformation. Thus, we chose to analyze the image piece by piece.

In [4]:
```python
# brain extraction function
def t1_to_t2_brain(path):    # brain extraction
    btr = fsl.BET(out_file = str(path)+'/t1_to_t2_brain.nii',mask = True)
    btr.inputs.in_file = str(path)+'/t1_to_t2.nii.gz'
    res = btr.run()
```

In [5]:
```python
# extract the brain transformed from t1
go = os.walk(r"/Volumes/myDisk/final")
for path,dir_list,file_list in go:
    for dir_name in dir_list:
        t1_to_t2_brain(os.path.join(path,dir_name))
```

In [5]:
```python
# transform the t1 brain and the t1 brain mask to tof
def t1_mask_to_tof(path):
    fslm = fsl.FLIRT(apply_xfm = True,in_matrix_file = str(path)+'/t1_to_t2_to_tof.mat',out_file = str(path)+'/
    fslm.inputs.in_file = str(path)+'/t1_to_t2_brain_mask.nii.gz'
    fslm.inputs.reference = str(path)+'/tof.nii.gz'
    fslm.inputs.output_type = "NIFTI_GZ"
    fslm.run()

def t1_brain_to_tof(path):
    fslb = fsl.FLIRT(apply_xfm = True,in_matrix_file = str(path)+'/t1_to_t2_to_tof.mat',out_file = str(path)+'/
    fslb.inputs.in_file = str(path)+'/t1_to_t2_brain.nii.gz'
    fslb.inputs.reference = str(path)+'/tof.nii.gz'
    fslb.inputs.output_type = "NIFTI_GZ"
    fslb.run()
```

In [6]:
```python
go = os.walk(r"/Volumes/myDisk/final")
for path,dir_list,file_list in go:
    for dir_name in dir_list:
        t1_mask_to_tof(os.path.join(path,dir_name))
        t1_brain_to_tof(os.path.join(path,dir_name))
```

In [7]:
```python
# apply the brain mask to tof
def appplymask(path):    # brain extraction
    am=fsl.ApplyMask(in_file = str(path)+'/tof.nii.gz',mask_file = str(path)+'/t1_mask_to_tof.nii.gz')
    am.inputs.out_file = str(path)+'/tof_masked.nii.gz'
    am.run()
```

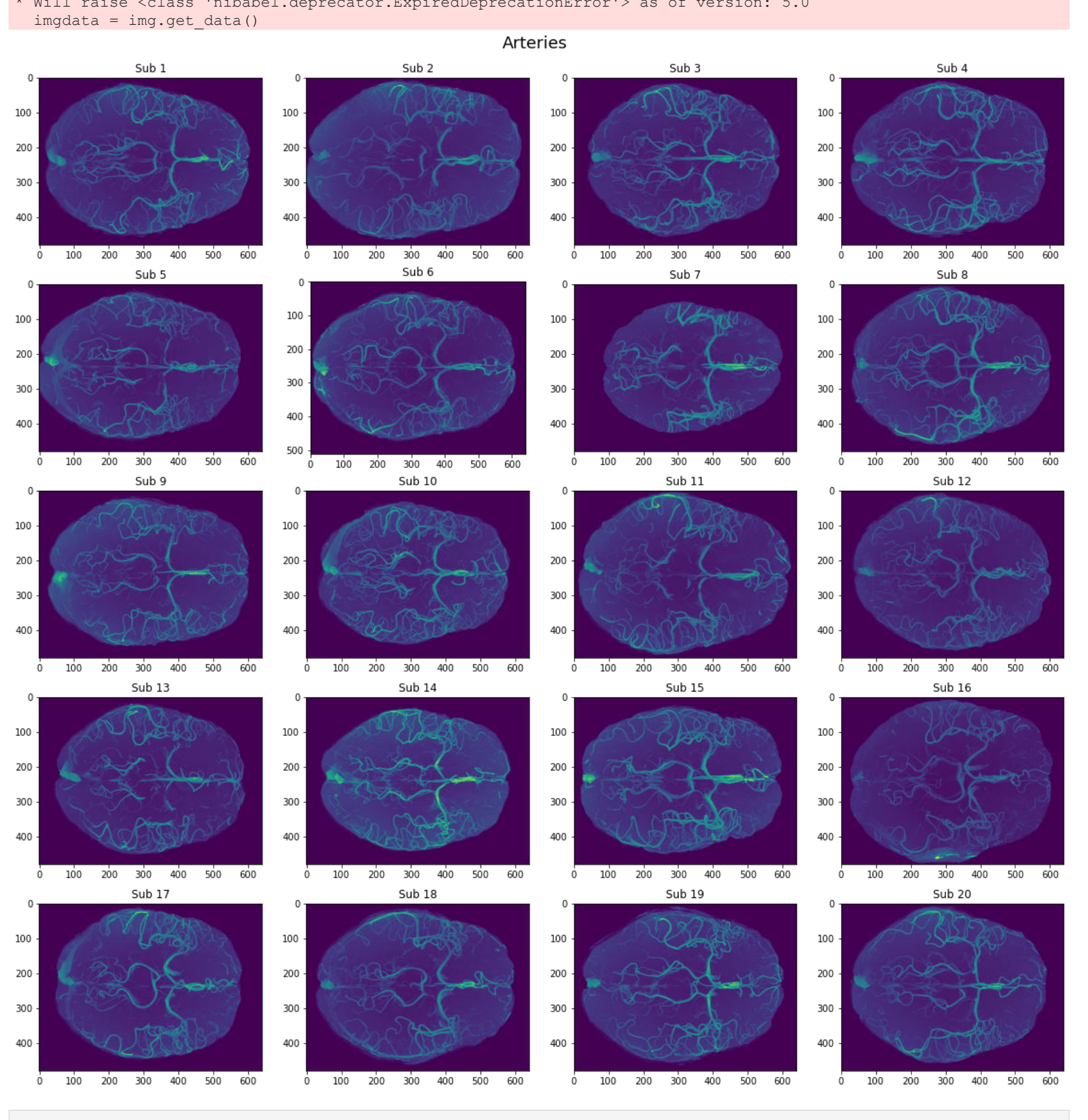
In [8]:
```python
go = os.walk(r"/Volumes/myDisk/final")
for path,dir_list,file_list in go:
    for dir_name in dir_list:
        appplymask(os.path.join(path,dir_name))
```

In [23]:
```python
# plot the masked tof
def plotmasked(path):
    img = nib.load(str(path)+'/tof_masked.nii.gz')
    imgdata = img.get_data()
    i,j,k = imgdata.shape
    plt.imshow(np.max(imgdata[:,:,:],axis = 2))
```

In [24]:
```python
# tof masked results
import os
go = os.walk(r"/Volumes/myDisk/final")
plt.figure(figsize = (20,20))
plt.suptitle('Arteries',fontsize = 18, y=0.91)
plt.tight_layout()
m = 1
for path,dir_list,file_list in go:
    for dir_name in dir_list:
        plt.subplot(5,4,m)
        plotmasked(os.path.join(path,dir_name))
        plt.title("Sub {}".format(m))
        m = m+1
```

```
<ipython-input-23-db87639f2851>:3: DeprecationWarning: get_data() is deprecated in favor of get_fdata(), which has a more predictable return type. To obtain get_data() behavior going forward, use numpy.asanyarray(img.dataobj).

* deprecated from version: 3.0
* Will raise <class 'nibabel.deprecator.ExpiredDeprecationError'> as of version: 5.0
  imgdata = img.get_data()
```



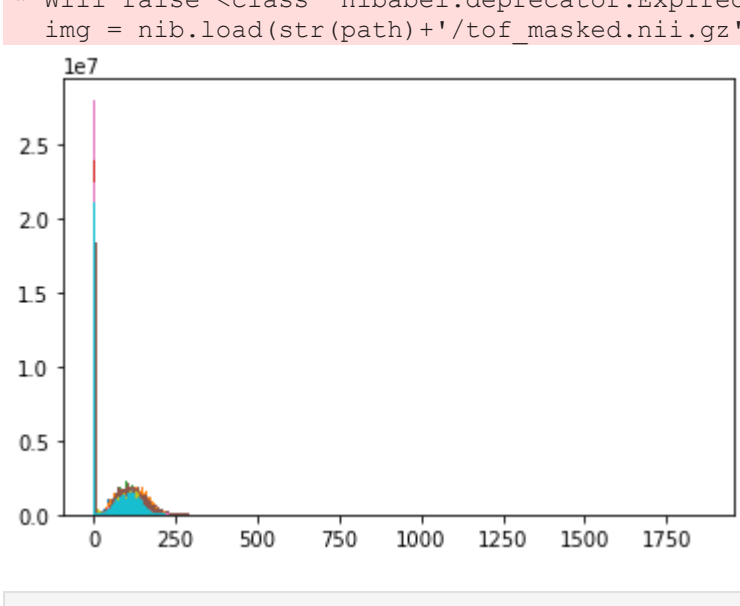
In [13]:
```python
# find general threshold for each subject
def threshtof(path):
    img = nib.load(str(path)+'/tof_masked.nii.gz').get_data()
    thresh = plt.hist(img.ravel(),bins = 256)[1][54]
    thres = fsl.Threshold(direction = 'below',out_file = str(path)+'/tof_arteries.nii.gz')
    thres.inputs.in_file = str(path)+'/tof_masked.nii.gz'
    thres.inputs.thresh = thresh
    thres.run()
```

Note: While extracting the arteries, we found the threshold was also important to perform the arteries. Unlike the threshold transformation, a general threshold value filtered low intensity and highlight the arteries, which has no affects to the transformation. Thus,we chose the 55th bin as the threshold value.

In [14]:
```python
go = os.walk(r"/Volumes/myDisk/final")
for path,dir_list,file_list in go:
    for dir_name in dir_list:
        threshtof(os.path.join(path,dir_name))
```

```
<ipython-input-13-c4cbcb555fd6>:2: DeprecationWarning: get_data() is deprecated in favor of get_fdata(), which has a more predictable return type. To obtain get_data() behavior going forward, use numpy.asanyarray(img.dataobj).

* deprecated from version: 3.0
* Will raise <class 'nibabel.deprecator.ExpiredDeprecationError'> as of version: 5.0
  img = nib.load(str(path)+'/tof_masked.nii.gz').get_data()
```

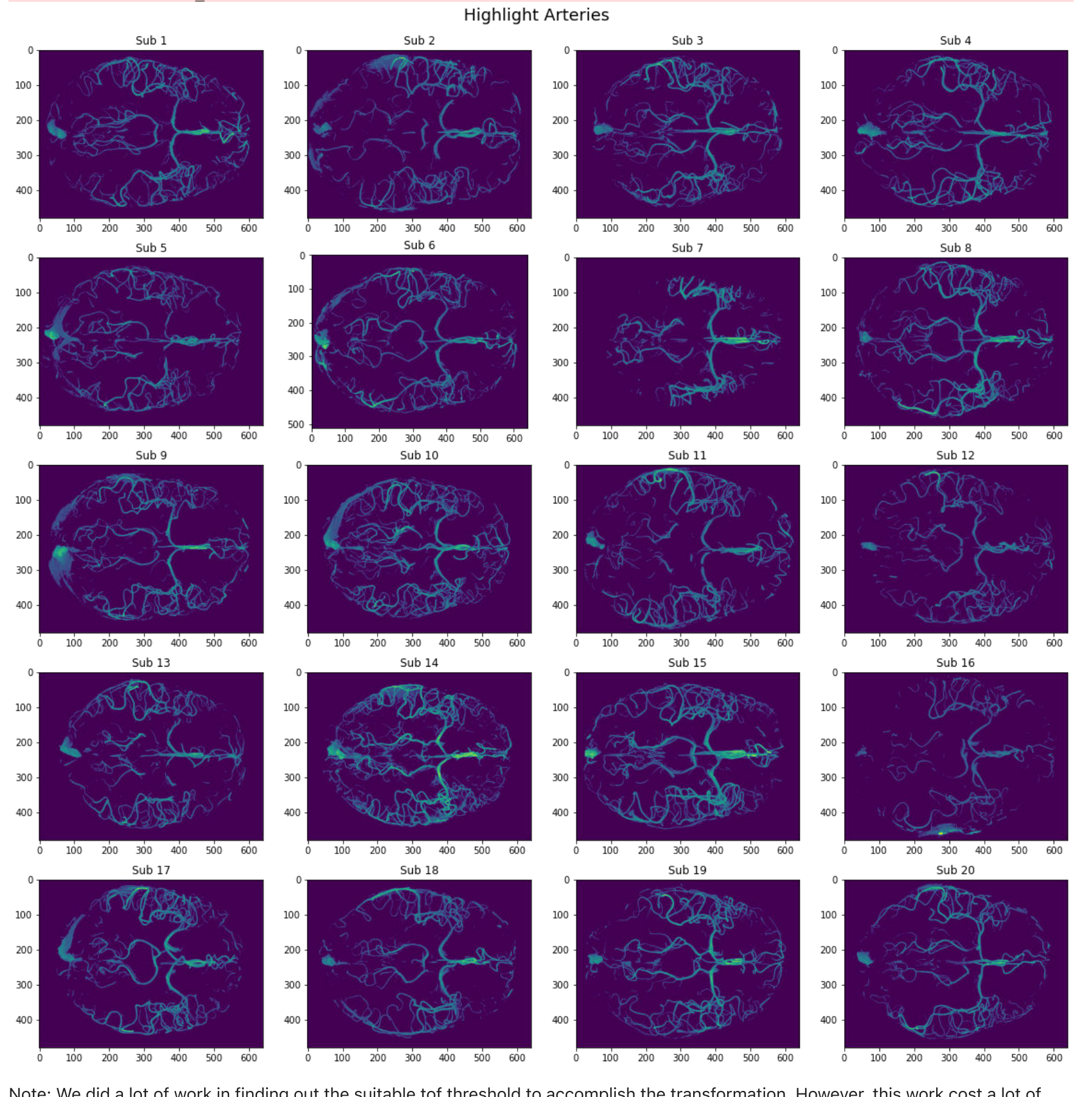


In [15]:
```python
def plotarteries(path):
    img = nib.load(str(path)+'/tof_arteries.nii.gz')
    imgdata = img.get_data()
    plt.imshow(np.max(imgdata[:,:,:],axis = 2))
```

In [16]:
```python
import os
go = os.walk(r"/Volumes/myDisk/final")
plt.figure(figsize = (20,20))
plt.suptitle('Highlight Arteries',fontsize = 18, y=0.91)
plt.tight_layout()
m = 1
for path,dir_list,file_list in go:
    for dir_name in dir_list:
        plt.subplot(5,4,m)
        plotarteries(os.path.join(path,dir_name))
        plt.title("Sub {}".format(m))
        m = m+1
```

```
<ipython-input-15-1faa93cb1184>:3: DeprecationWarning: get_data() is deprecated in favor of get_fdata(), which has a more predictable return type. To obtain get_data() behavior going forward, use numpy.asanyarray(img.dataobj).

* deprecated from version: 3.0
* Will raise <class 'nibabel.deprecator.ExpiredDeprecationError'> as of version: 5.0
  imgdata = img.get_data()
```



Note: We did a lot of work in finding out the suitable tof threshold to accomplish the transformation. However, this work cost a lot of energy and time.

In [23]:
```python
def overlaytof(path):
    combine = fsl.Overlay(out_file = str(path)+'/tof_arteries_overlay.nii.gz')
    combine.inputs.background_image = str(path)+'/t1_to_t2_to_tof.nii.gz'
    combine.inputs.auto_thresh_bg = True
    combine.inputs.stat_image = str(path)+'/tof_arteries.nii.gz'
    combine.inputs.stat_thresh = (100, 2000)
    combine.inputs.show_negative_stats = True
    res = combine.run()
```

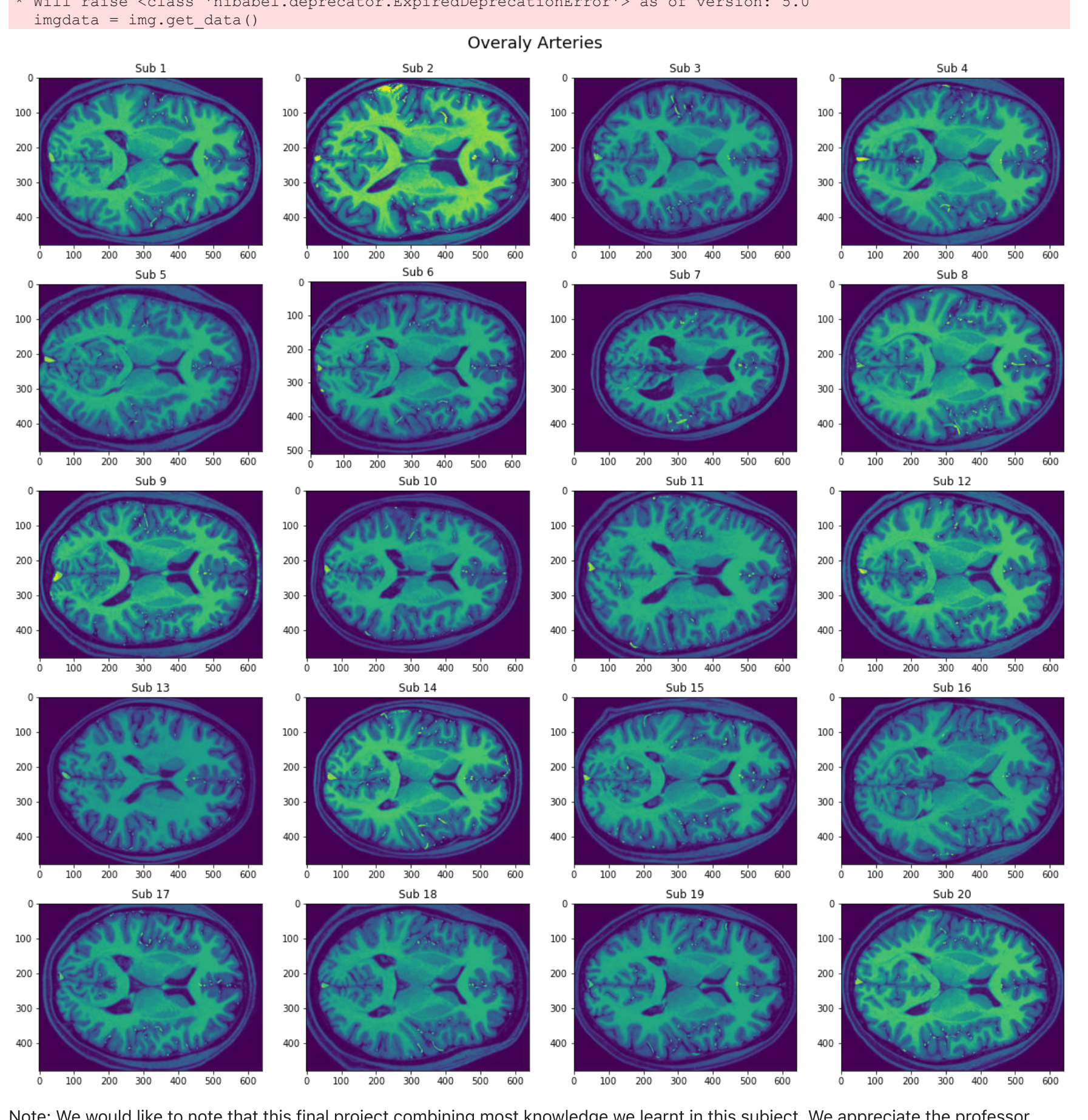
In [24]:
```python
go = os.walk(r"/Volumes/myDisk/final")
for path,dir_list,file_list in go:
    for dir_name in dir_list:
        overlaytof(os.path.join(path,dir_name))
```

In [27]:
```python
# define a function overlaying the t1_to_tof image
def plotoverlay(path):
    img = nib.load(str(path)+'/tof_arteries_overlay.nii.gz')
    imgdata = img.get_data()
    plt.imshow(imgdata[:,:,100])
```

In [28]:
```python
# plot the overlay image
import os
go = os.walk(r"/Volumes/myDisk/final")
plt.figure(figsize = (20,20))
plt.suptitle('Overaly Arteries',fontsize = 18, y=0.91)
plt.tight_layout()
m = 1
for path,dir_list,file_list in go:
    for dir_name in dir_list:
        plt.subplot(5,4,m)
        plotoverlay(os.path.join(path,dir_name))
        plt.title("Sub {}".format(m))
        m = m+1
```

```
<ipython-input-27-3bdbe33ff14f>:3: DeprecationWarning: get_data() is deprecated in favor of get_fdata(), which has a more predictable return type. To obtain get_data() behavior going forward, use numpy.asanyarray(img.dataobj).

* deprecated from version: 3.0
* Will raise <class 'nibabel.deprecator.ExpiredDeprecationError'> as of version: 5.0
  imgdata = img.get_data()
```



Note: We would like to note that this final project combining most knowledge we learnt in this subject. We appreciate the professor introducing us these useful libraries and methods.

In [ ]:

CS516 Medical Imaging

Final Project Part 2

Luyun Nie 002268087

Junjia Lin 002268506

In [1]:

```
# import some essencial packages
from google.colab import drive
import gc
gc.collect()
drive.mount('/content/drive')
import os
import numpy as np
import nibabel as nib
import matplotlib.pyplot as plt
from keras.optimizers import SGD, Adagrad, Adadelta, RMSprop, Adam
import tensorflow as tf
%env SM_FRAMEWORK=tf.keras
!pip install segmentation_models
import segmentation_models as sm
```

```
Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", for
ce_remount=True).
env: SM_FRAMEWORK=tf.keras
Requirement already satisfied: segmentation_models in /usr/local/lib/python3.7/dist-packages (1.0.1)
Requirement already satisfied: image-classifiers==1.0.0 in /usr/local/lib/python3.7/dist-packages (from segment
ation_models) (1.0.0)
Requirement already satisfied: keras-applications<=1.0.8,>=1.0.7 in /usr/local/lib/python3.7/dist-packages (fro
m segmentation_models) (1.0.8)
Requirement already satisfied: efficientnet==1.0.0 in /usr/local/lib/python3.7/dist-packages (from segmentation
_models) (1.0.0)
Requirement already satisfied: scikit-image in /usr/local/lib/python3.7/dist-packages (from efficientnet==1.0.0
->segmentation_models) (0.16.2)
Requirement already satisfied: numpy>=1.9.1 in /usr/local/lib/python3.7/dist-packages (from keras-applications<
=1.0.8,>=1.0.7->segmentation_models) (1.19.5)
Requirement already satisfied: h5py in /usr/local/lib/python3.7/dist-packages (from keras-applications<=1.0.8,>
=1.0.7->segmentation_models) (3.1.0)
Requirement already satisfied: cached-property in /usr/local/lib/python3.7/dist-packages (from h5py->keras-appl
ications<=1.0.8,>=1.0.7->segmentation_models) (1.5.2)
Requirement already satisfied: networkx>=2.0 in /usr/local/lib/python3.7/dist-packages (from scikit-image->effi
cientnet==1.0.0->segmentation_models) (2.5.1)
Requirement already satisfied: matplotlib!=3.0.0,>=2.0.0 in /usr/local/lib/python3.7/dist-packages (from scikit
-image->efficientnet==1.0.0->segmentation_models) (3.2.2)
Requirement already satisfied: PyWavelets>=0.4.0 in /usr/local/lib/python3.7/dist-packages (from scikit-image->
efficientnet==1.0.0->segmentation_models) (1.1.1)
Requirement already satisfied: scipy>=0.19.0 in /usr/local/lib/python3.7/dist-packages (from scikit-image->effi
cientnet==1.0.0->segmentation_models) (1.4.1)
Requirement already satisfied: imageio>=2.3.0 in /usr/local/lib/python3.7/dist-packages (from scikit-image->eff
icientnet==1.0.0->segmentation_models) (2.4.1)
Requirement already satisfied: pillow>=4.3.0 in /usr/local/lib/python3.7/dist-packages (from scikit-image->effi
cientnet==1.0.0->segmentation_models) (7.1.2)
Requirement already satisfied: kiwisolver>=1.0.1 in /usr/local/lib/python3.7/dist-packages (from matplotlib!=3.
0.0,>=2.0.0->scikit-image->efficientnet==1.0.0->segmentation_models) (1.3.1)
Requirement already satisfied: pyparsing!=2.0.4,!=2.1.2,!=2.1.6,>=2.0.1 in /usr/local/lib/python3.7/dist-packag
es (from matplotlib!=3.0.0,>=2.0.0->scikit-image->efficientnet==1.0.0->segmentation_models) (2.4.7)
Requirement already satisfied: python-dateutil>=2.1 in /usr/local/lib/python3.7/dist-packages (from matplotlib!
=3.0.0,>=2.0.0->scikit-image->efficientnet==1.0.0->segmentation_models) (2.8.1)
Requirement already satisfied: cycler>=0.10 in /usr/local/lib/python3.7/dist-packages (from matplotlib!=3.0.0,>
=2.0.0->scikit-image->efficientnet==1.0.0->segmentation_models) (0.10.0)
Requirement already satisfied: six in /usr/local/lib/python3.7/dist-packages (from cycler>=0.10->matplotlib!=3.
0.0,>=2.0.0->scikit-image->efficientnet==1.0.0->segmentation_models) (1.15.0)
Requirement already satisfied: decorator<5,>=4.3 in /usr/local/lib/python3.7/dist-packages (from networkx>=2.0-
>scikit-image->efficientnet==1.0.0->segmentation_models) (4.4.2)
Segmentation Models: using `tf.keras` framework.
```

In [2]:

```
# scan all files from the root
go = os.listdir('/content/drive/My Drive/final example/')
go.sort(key = str.lower)
trainfile = go[0:15] # set the train files
testfile = go[15:20] # set the test files
```

In [3]:

```
trainfile = go[0:15]
testfile = go[15:20]
print(trainfile,testfile)
```

```
['sub-01', 'sub-02', 'sub-03', 'sub-04', 'sub-05', 'sub-06', 'sub-07', 'sub-08', 'sub-09', 'sub-10', 'sub-11',
'sub-12', 'sub-13', 'sub-14', 'sub-15'] ['sub-16', 'sub-17', 'sub-18', 'sub-19', 'sub-20']
```

Note: We utilize the groundseg data provided from Dr.Butler to confirm we would not mess up by image data made from part 1.

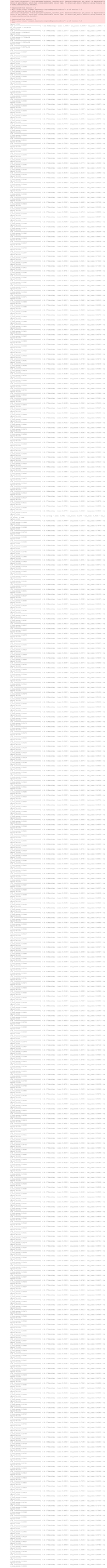
In [4]:

```
# set the model
BACKBONE = 'resnet34'
#preprocess_input = sm.get_preprocessing(BACKBONE)

model = sm.Unet(BACKBONE, input_shape=(None,None,3),encoder_weights='imagenet')
model.compile(
    'Adam',
    loss=sm.losses.bce_jaccard_loss,
    metrics=[sm.metrics.iou_score],
)
```

In [5]:

```
# set a loop to run fitting model to train itself
val_iou_scorelist = [[]]
for train in trainfile:
  groundseg = nib.load(os.path.join('/content/drive/My Drive/final example/',train)+'/tof_arteries.nii.gz').get
  t1 = nib.load(os.path.join('/content/drive/My Drive/final example/',train)+'/t1_in_tof.nii.gz').get_data()
  groundseg = np.swapaxes(groundseg,0,2)
  t1 = np.swapaxes(t1,0,2)
  groundseg = np.expand_dims(groundseg,axis=3)
  t1 = np.expand_dims(t1,axis=3)
  t1 = np.tile(t1,[1,1,1,3])
  x_train = t1[0::2,:,:,:]
  y_train = groundseg[0::2,:,:,:].astype(np.float32)
  x_val = t1[1::2,:,:,:]
  y_val = groundseg[1::2,:,:,:].astype(np.float32)
  model.fit(x=x_train,y=y_train,batch_size=16,epochs=100,validation_data=(x_val, y_val))
  # add the history result to a list
  val_iou_scorelist.append(model.history.history['val_iou_score'])
```



```
Epoch 56/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3328 - iou_score: 0.6891 - val_loss: 0.5091 - va
l_iou_score: 0.4838
Epoch 57/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3309 - iou_score: 0.6876 - val_loss: 0.5050 - va
l_iou_score: 0.4901
Epoch 58/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3381 - iou_score: 0.6648 - val_loss: 0.5022 - va
l_iou_score: 0.4944
Epoch 59/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3321 - iou_score: 0.6655 - val_loss: 0.5143 - va
l_iou_score: 0.4825
Epoch 60/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3298 - iou_score: 0.6825 - val_loss: 0.5106 - va
l_iou_score: 0.4833
Epoch 61/100
6/6 [==============================] - 2s 371ms/step - loss: 0.3275 - iou_score: 0.6797 - val_loss: 0.5027 - va
l_iou_score: 0.4911
Epoch 62/100
6/6 [==============================] - 2s 369ms/step - loss: 0.3260 - iou_score: 0.6841 - val_loss: 0.5080 - va
l_iou_score: 0.4874
Epoch 63/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3299 - iou_score: 0.6732 - val_loss: 0.4948 - va
l_iou_score: 0.4997
Epoch 64/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3215 - iou_score: 0.6925 - val_loss: 0.4989 - va
l_iou_score: 0.4963
Epoch 65/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3249 - iou_score: 0.6522 - val_loss: 0.4956 - va
l_iou_score: 0.5025
Epoch 66/100
6/6 [==============================] - 2s 371ms/step - loss: 0.3237 - iou_score: 0.6948 - val_loss: 0.4981 - va
l_iou_score: 0.4985
Epoch 67/100
6/6 [==============================] - 2s 375ms/step - loss: 0.3150 - iou_score: 0.7065 - val_loss: 0.4899 - va
l_iou_score: 0.5042
Epoch 68/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3165 - iou_score: 0.7009 - val_loss: 0.4919 - va
l_iou_score: 0.5023
Epoch 69/100
6/6 [==============================] - 2s 370ms/step - loss: 0.3113 - iou_score: 0.6944 - val_loss: 0.4880 - va
l_iou_score: 0.5051
Epoch 70/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3071 - iou_score: 0.7011 - val_loss: 0.4846 - va
l_iou_score: 0.5117
Epoch 71/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3049 - iou_score: 0.7070 - val_loss: 0.4851 - va
l_iou_score: 0.5119
Epoch 72/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3075 - iou_score: 0.6865 - val_loss: 0.4831 - va
l_iou_score: 0.5153
Epoch 73/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3034 - iou_score: 0.7019 - val_loss: 0.4878 - va
l_iou_score: 0.5101
Epoch 74/100
6/6 [==============================] - 2s 371ms/step - loss: 0.3039 - iou_score: 0.7114 - val_loss: 0.4874 - va
l_iou_score: 0.5097
Epoch 75/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3051 - iou_score: 0.6927 - val_loss: 0.4940 - va
l_iou_score: 0.5063
Epoch 76/100
6/6 [==============================] - 2s 371ms/step - loss: 0.3053 - iou_score: 0.7157 - val_loss: 0.4872 - va
l_iou_score: 0.5121
Epoch 77/100
6/6 [==============================] - 2s 371ms/step - loss: 0.2990 - iou_score: 0.7035 - val_loss: 0.4890 - va
l_iou_score: 0.5082
Epoch 78/100
6/6 [==============================] - 2s 370ms/step - loss: 0.2993 - iou_score: 0.7168 - val_loss: 0.4860 - va
l_iou_score: 0.5103
Epoch 79/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3006 - iou_score: 0.7040 - val_loss: 0.4994 - va
l_iou_score: 0.4975
Epoch 80/100
6/6 [==============================] - 2s 371ms/step - loss: 0.2943 - iou_score: 0.7089 - val_loss: 0.4868 - va
l_iou_score: 0.5092
Epoch 81/100
6/6 [==============================] - 2s 371ms/step - loss: 0.2939 - iou_score: 0.7064 - val_loss: 0.4862 - va
l_iou_score: 0.5118
Epoch 82/100
6/6 [==============================] - 2s 373ms/step - loss: 0.2933 - iou_score: 0.7143 - val_loss: 0.4807 - va
l_iou_score: 0.5159
Epoch 83/100
6/6 [==============================] - 2s 372ms/step - loss: 0.2923 - iou_score: 0.7209 - val_loss: 0.4856 - va
l_iou_score: 0.5117
Epoch 84/100
6/6 [==============================] - 2s 373ms/step - loss: 0.2947 - iou_score: 0.7012 - val_loss: 0.4763 - va
l_iou_score: 0.5213
Epoch 85/100
6/6 [==============================] - 2s 374ms/step - loss: 0.2869 - iou_score: 0.7255 - val_loss: 0.4849 - va
l_iou_score: 0.5148
Epoch 86/100
6/6 [==============================] - 2s 372ms/step - loss: 0.2879 - iou_score: 0.7087 - val_loss: 0.4817 - va
l_iou_score: 0.5161
Epoch 87/100
6/6 [==============================] - 2s 374ms/step - loss: 0.2852 - iou_score: 0.7101 - val_loss: 0.4852 - va
l_iou_score: 0.5129
Epoch 88/100
6/6 [==============================] - 2s 371ms/step - loss: 0.2868 - iou_score: 0.7396 - val_loss: 0.4768 - va
l_iou_score: 0.5215
Epoch 89/100
6/6 [==============================] - 2s 373ms/step - loss: 0.2878 - iou_score: 0.7088 - val_loss: 0.4825 - va
l_iou_score: 0.5152
Epoch 90/100
6/6 [==============================] - 2s 370ms/step - loss: 0.2881 - iou_score: 0.7197 - val_loss: 0.4757 - va
l_iou_score: 0.5247
Epoch 91/100
6/6 [==============================] - 2s 373ms/step - loss: 0.2931 - iou_score: 0.7099 - val_loss: 0.4794 - va
l_iou_score: 0.5204
Epoch 92/100
6/6 [==============================] - 2s 374ms/step - loss: 0.2905 - iou_score: 0.7118 - val_loss: 0.4820 - va
l_iou_score: 0.5194
Epoch 93/100
6/6 [==============================] - 2s 373ms/step - loss: 0.2986 - iou_score: 0.7154 - val_loss: 0.4783 - va
l_iou_score: 0.5217
Epoch 94/100
6/6 [==============================] - 2s 374ms/step - loss: 0.2961 - iou_score: 0.7257 - val_loss: 0.4793 - va
l_iou_score: 0.5245
Epoch 95/100
6/6 [==============================] - 2s 371ms/step - loss: 0.2999 - iou_score: 0.7131 - val_loss: 0.4782 - va
l_iou_score: 0.5233
Epoch 96/100
6/6 [==============================] - 2s 372ms/step - loss: 0.2937 - iou_score: 0.7070 - val_loss: 0.4823 - va
l_iou_score: 0.5182
Epoch 97/100
6/6 [==============================] - 2s 373ms/step - loss: 0.2887 - iou_score: 0.7229 - val_loss: 0.4727 - va
l_iou_score: 0.5281
Epoch 98/100
6/6 [==============================] - 2s 372ms/step - loss: 0.2947 - iou_score: 0.7256 - val_loss: 0.4777 - va
l_iou_score: 0.5246
Epoch 99/100
6/6 [==============================] - 2s 372ms/step - loss: 0.2905 - iou_score: 0.7196 - val_loss: 0.4748 - va
l_iou_score: 0.5265
Epoch 100/100
6/6 [==============================] - 2s 367ms/step - loss: 0.2890 - iou_score: 0.6941 - val_loss: 0.4775 - va
l_iou_score: 0.5249
Epoch 1/100
6/6 [==============================] - 2s 385ms/step - loss: 0.9074 - iou_score: 0.1676 - val_loss: 0.8677 - va
l_iou_score: 0.1904
Epoch 2/100
6/6 [==============================] - 2s 370ms/step - loss: 0.8070 - iou_score: 0.2500 - val_loss: 0.8075 - va
l_iou_score: 0.2344
Epoch 3/100
6/6 [==============================] - 2s 374ms/step - loss: 0.7391 - iou_score: 0.3038 - val_loss: 0.7883 - va
l_iou_score: 0.2558
Epoch 4/100
6/6 [==============================] - 2s 370ms/step - loss: 0.6874 - iou_score: 0.3394 - val_loss: 0.7340 - va
l_iou_score: 0.2983
Epoch 5/100
6/6 [==============================] - 2s 372ms/step - loss: 0.6505 - iou_score: 0.3763 - val_loss: 0.6947 - va
l_iou_score: 0.3265
Epoch 6/100
6/6 [==============================] - 2s 372ms/step - loss: 0.6164 - iou_score: 0.4025 - val_loss: 0.6589 - va
l_iou_score: 0.3565
Epoch 7/100
6/6 [==============================] - 2s 374ms/step - loss: 0.5946 - iou_score: 0.4279 - val_loss: 0.6668 - va
l_iou_score: 0.3395
Epoch 8/100
6/6 [==============================] - 2s 373ms/step - loss: 0.5704 - iou_score: 0.4439 - val_loss: 0.6350 - va
l_iou_score: 0.3791
Epoch 9/100
6/6 [==============================] - 2s 373ms/step - loss: 0.5439 - iou_score: 0.4725 - val_loss: 0.6182 - va
l_iou_score: 0.3925
Epoch 10/100
6/6 [==============================] - 2s 374ms/step - loss: 0.5198 - iou_score: 0.4996 - val_loss: 0.5950 - va
l_iou_score: 0.4136
Epoch 11/100
6/6 [==============================] - 2s 371ms/step - loss: 0.5002 - iou_score: 0.5153 - val_loss: 0.5918 - va
l_iou_score: 0.4141
Epoch 12/100
6/6 [==============================] - 2s 373ms/step - loss: 0.4865 - iou_score: 0.5280 - val_loss: 0.5706 - va
l_iou_score: 0.4354
Epoch 13/100
6/6 [==============================] - 2s 374ms/step - loss: 0.4704 - iou_score: 0.5461 - val_loss: 0.5707 - va
l_iou_score: 0.4346
Epoch 14/100
6/6 [==============================] - 2s 371ms/step - loss: 0.4561 - iou_score: 0.5596 - val_loss: 0.5500 - va
l_iou_score: 0.4524
Epoch 15/100
6/6 [==============================] - 2s 373ms/step - loss: 0.4448 - iou_score: 0.5761 - val_loss: 0.5543 - va
l_iou_score: 0.4494
Epoch 16/100
6/6 [==============================] - 2s 372ms/step - loss: 0.4352 - iou_score: 0.5872 - val_loss: 0.5437 - va
l_iou_score: 0.4573
Epoch 17/100
6/6 [==============================] - 2s 372ms/step - loss: 0.4278 - iou_score: 0.5901 - val_loss: 0.5548 - va
l_iou_score: 0.4458
Epoch 18/100
6/6 [==============================] - 2s 371ms/step - loss: 0.4181 - iou_score: 0.6006 - val_loss: 0.5335 - va
l_iou_score: 0.4688
Epoch 19/100
6/6 [==============================] - 2s 372ms/step - loss: 0.4113 - iou_score: 0.6028 - val_loss: 0.5330 - va
l_iou_score: 0.4690
Epoch 20/100
6/6 [==============================] - 2s 372ms/step - loss: 0.4059 - iou_score: 0.6133 - val_loss: 0.5363 - va
l_iou_score: 0.4663
Epoch 21/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3961 - iou_score: 0.6151 - val_loss: 0.5214 - va
l_iou_score: 0.4804
Epoch 22/100
6/6 [==============================] - 2s 370ms/step - loss: 0.3943 - iou_score: 0.6145 - val_loss: 0.5224 - va
l_iou_score: 0.4802
Epoch 23/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3884 - iou_score: 0.6226 - val_loss: 0.5206 - va
l_iou_score: 0.4805
Epoch 24/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3859 - iou_score: 0.6444 - val_loss: 0.5147 - va
l_iou_score: 0.4887
Epoch 25/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3826 - iou_score: 0.6408 - val_loss: 0.5084 - va
l_iou_score: 0.4937
Epoch 26/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3823 - iou_score: 0.6263 - val_loss: 0.4954 - va
l_iou_score: 0.5101
Epoch 27/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3844 - iou_score: 0.6341 - val_loss: 0.4958 - va
l_iou_score: 0.5096
Epoch 28/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3773 - iou_score: 0.6396 - val_loss: 0.5040 - va
l_iou_score: 0.4985
Epoch 29/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3807 - iou_score: 0.6311 - val_loss: 0.4956 - va
l_iou_score: 0.5056
Epoch 30/100
6/6 [==============================] - 2s 371ms/step - loss: 0.3782 - iou_score: 0.6340 - val_loss: 0.4904 - va
l_iou_score: 0.5142
Epoch 31/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3727 - iou_score: 0.6446 - val_loss: 0.4878 - va
l_iou_score: 0.5141
Epoch 32/100
6/6 [==============================] - 2s 371ms/step - loss: 0.3685 - iou_score: 0.6397 - val_loss: 0.4844 - va
l_iou_score: 0.5167
Epoch 33/100
6/6 [==============================] - 2s 371ms/step - loss: 0.3671 - iou_score: 0.6445 - val_loss: 0.4816 - va
l_iou_score: 0.5270
Epoch 34/100
6/6 [==============================] - 2s 371ms/step - loss: 0.3613 - iou_score: 0.6636 - val_loss: 0.4721 - va
l_iou_score: 0.5378
Epoch 35/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3623 - iou_score: 0.6520 - val_loss: 0.4819 - va
l_iou_score: 0.5249
Epoch 36/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3586 - iou_score: 0.6538 - val_loss: 0.4763 - va
l_iou_score: 0.5309
Epoch 37/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3554 - iou_score: 0.6714 - val_loss: 0.4684 - va
l_iou_score: 0.5404
Epoch 38/100
6/6 [==============================] - 2s 371ms/step - loss: 0.3530 - iou_score: 0.6632 - val_loss: 0.4710 - va
l_iou_score: 0.5348
Epoch 39/100
6/6 [==============================] - 2s 376ms/step - loss: 0.3482 - iou_score: 0.6715 - val_loss: 0.4604 - va
l_iou_score: 0.5438
Epoch 40/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3437 - iou_score: 0.6691 - val_loss: 0.4597 - va
l_iou_score: 0.5357
Epoch 41/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3391 - iou_score: 0.6802 - val_loss: 0.4553 - va
l_iou_score: 0.5409
Epoch 42/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3418 - iou_score: 0.6724 - val_loss: 0.4514 - va
l_iou_score: 0.5464
Epoch 43/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3378 - iou_score: 0.6752 - val_loss: 0.4548 - va
l_iou_score: 0.5426
Epoch 44/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3306 - iou_score: 0.6837 - val_loss: 0.4501 - va
l_iou_score: 0.5460
Epoch 45/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3293 - iou_score: 0.6937 - val_loss: 0.4535 - va
l_iou_score: 0.5461
Epoch 46/100
6/6 [==============================] - 2s 375ms/step - loss: 0.3281 - iou_score: 0.6780 - val_loss: 0.4551 - va
l_iou_score: 0.5442
Epoch 47/100
6/6 [==============================] - 2s 371ms/step - loss: 0.3298 - iou_score: 0.6991 - val_loss: 0.4461 - va
l_iou_score: 0.5567
Epoch 48/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3283 - iou_score: 0.6893 - val_loss: 0.4461 - va
l_iou_score: 0.5591
Epoch 49/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3268 - iou_score: 0.6811 - val_loss: 0.4541 - va
l_iou_score: 0.5499
Epoch 50/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3239 - iou_score: 0.6869 - val_loss: 0.4441 - va
l_iou_score: 0.5620
Epoch 51/100
6/6 [==============================] - 2s 375ms/step - loss: 0.3205 - iou_score: 0.7013 - val_loss: 0.4436 - va
l_iou_score: 0.5619
Epoch 52/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3143 - iou_score: 0.6973 - val_loss: 0.4397 - va
l_iou_score: 0.5655
Epoch 53/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3130 - iou_score: 0.7009 - val_loss: 0.4451 - va
l_iou_score: 0.5587
Epoch 54/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3123 - iou_score: 0.6957 - val_loss: 0.4344 - va
l_iou_score: 0.5735
Epoch 55/100
6/6 [==============================] - 2s 369ms/step - loss: 0.3103 - iou_score: 0.7046 - val_loss: 0.4399 - va
l_iou_score: 0.5666
Epoch 56/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3082 - iou_score: 0.6989 - val_loss: 0.4318 - va
l_iou_score: 0.5722
Epoch 57/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3044 - iou_score: 0.7013 - val_loss: 0.4348 - va
l_iou_score: 0.5699
Epoch 58/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3031 - iou_score: 0.7061 - val_loss: 0.4339 - va
l_iou_score: 0.5739
Epoch 59/100
6/6 [==============================] - 2s 371ms/step - loss: 0.3060 - iou_score: 0.7129 - val_loss: 0.4387 - va
l_iou_score: 0.5654
Epoch 60/100
6/6 [==============================] - 2s 375ms/step - loss: 0.3001 - iou_score: 0.7096 - val_loss: 0.4347 - va
l_iou_score: 0.5719
Epoch 61/100
6/6 [==============================] - 2s 373ms/step - loss: 0.2962 - iou_score: 0.7179 - val_loss: 0.4299 - va
l_iou_score: 0.5778
Epoch 62/100
6/6 [==============================] - 2s 375ms/step - loss: 0.2903 - iou_score: 0.7189 - val_loss: 0.4234 - va
l_iou_score: 0.5864
Epoch 63/100
6/6 [==============================] - 2s 373ms/step - loss: 0.2872 - iou_score: 0.7299 - val_loss: 0.4225 - va
l_iou_score: 0.5870
Epoch 64/100
6/6 [==============================] - 2s 372ms/step - loss: 0.2867 - iou_score: 0.7297 - val_loss: 0.4207 - va
l_iou_score: 0.5879
Epoch 65/100
6/6 [==============================] - 2s 374ms/step - loss: 0.2887 - iou_score: 0.7252 - val_loss: 0.4229 - va
l_iou_score: 0.5833
Epoch 66/100
6/6 [==============================] - 2s 373ms/step - loss: 0.2870 - iou_score: 0.7282 - val_loss: 0.4213 - va
l_iou_score: 0.5853
Epoch 67/100
6/6 [==============================] - 2s 374ms/step - loss: 0.2831 - iou_score: 0.7294 - val_loss: 0.4190 - va
l_iou_score: 0.5878
Epoch 68/100
6/6 [==============================] - 2s 375ms/step - loss: 0.2783 - iou_score: 0.7329 - val_loss: 0.4186 - va
l_iou_score: 0.5874
Epoch 69/100
6/6 [==============================] - 2s 375ms/step - loss: 0.2783 - iou_score: 0.7369 - val_loss: 0.4272 - va
l_iou_score: 0.5789
Epoch 70/100
6/6 [==============================] - 2s 381ms/step - loss: 0.2837 - iou_score: 0.7245 - val_loss: 0.4237 - va
l_iou_score: 0.5843
Epoch 71/100
6/6 [==============================] - 2s 373ms/step - loss: 0.2815 - iou_score: 0.7360 - val_loss: 0.4184 - va
l_iou_score: 0.5880
Epoch 72/100
6/6 [==============================] - 2s 372ms/step - loss: 0.2826 - iou_score: 0.7255 - val_loss: 0.4238 - va
l_iou_score: 0.5832
Epoch 73/100
6/6 [==============================] - 2s 373ms/step - loss: 0.2796 - iou_score: 0.7430 - val_loss: 0.4159 - va
l_iou_score: 0.5926
Epoch 74/100
6/6 [==============================] - 2s 372ms/step - loss: 0.2766 - iou_score: 0.7415 - val_loss: 0.4217 - va
l_iou_score: 0.5833
Epoch 75/100
6/6 [==============================] - 2s 372ms/step - loss: 0.2760 - iou_score: 0.7342 - val_loss: 0.4180 - va
l_iou_score: 0.5868
Epoch 76/100
6/6 [==============================] - 2s 374ms/step - loss: 0.2701 - iou_score: 0.7424 - val_loss: 0.4144 - va
l_iou_score: 0.5899
Epoch 77/100
6/6 [==============================] - 2s 373ms/step - loss: 0.2704 - iou_score: 0.7469 - val_loss: 0.4166 - va
l_iou_score: 0.5890
Epoch 78/100
6/6 [==============================] - 2s 375ms/step - loss: 0.2709 - iou_score: 0.7430 - val_loss: 0.4218 - va
l_iou_score: 0.5846
Epoch 79/100
6/6 [==============================] - 2s 373ms/step - loss: 0.2662 - iou_score: 0.7335 - val_loss: 0.4225 - va
l_iou_score: 0.5858
Epoch 80/100
6/6 [==============================] - 2s 375ms/step - loss: 0.2643 - iou_score: 0.7441 - val_loss: 0.4163 - va
l_iou_score: 0.5906
Epoch 81/100
6/6 [==============================] - 2s 374ms/step - loss: 0.2636 - iou_score: 0.7413 - val_loss: 0.4098 - va
l_iou_score: 0.5968
Epoch 82/100
6/6 [==============================] - 2s 373ms/step - loss: 0.2657 - iou_score: 0.7451 - val_loss: 0.4118 - va
l_iou_score: 0.5930
Epoch 83/100
6/6 [==============================] - 2s 373ms/step - loss: 0.2696 - iou_score: 0.7397 - val_loss: 0.4148 - va
l_iou_score: 0.5901
Epoch 84/100
6/6 [==============================] - 2s 374ms/step - loss: 0.2680 - iou_score: 0.7453 - val_loss: 0.4180 - va
l_iou_score: 0.5858
Epoch 85/100
6/6 [==============================] - 2s 375ms/step - loss: 0.2695 - iou_score: 0.7427 - val_loss: 0.4224 - va
l_iou_score: 0.5817
Epoch 86/100
6/6 [==============================] - 2s 371ms/step - loss: 0.2719 - iou_score: 0.7401 - val_loss: 0.4184 - va
l_iou_score: 0.5899
Epoch 87/100
6/6 [==============================] - 2s 373ms/step - loss: 0.2679 - iou_score: 0.7437 - val_loss: 0.4173 - va
l_iou_score: 0.5882
Epoch 88/100
6/6 [==============================] - 2s 373ms/step - loss: 0.2665 - iou_score: 0.7421 - val_loss: 0.4222 - va
l_iou_score: 0.5843
Epoch 89/100
6/6 [==============================] - 2s 373ms/step - loss: 0.2702 - iou_score: 0.7410 - val_loss: 0.4154 - va
l_iou_score: 0.5889
Epoch 90/100
6/6 [==============================] - 2s 373ms/step - loss: 0.2701 - iou_score: 0.7422 - val_loss: 0.4152 - va
l_iou_score: 0.5894
Epoch 91/100
6/6 [==============================] - 2s 374ms/step - loss: 0.2702 - iou_score: 0.7265 - val_loss: 0.4222 - va
l_iou_score: 0.5816
Epoch 92/100
6/6 [==============================] - 2s 373ms/step - loss: 0.2680 - iou_score: 0.7496 - val_loss: 0.4172 - va
l_iou_score: 0.5861
Epoch 93/100
6/6 [==============================] - 2s 371ms/step - loss: 0.2653 - iou_score: 0.7523 - val_loss: 0.4164 - va
l_iou_score: 0.5873
Epoch 94/100
6/6 [==============================] - 2s 371ms/step - loss: 0.2636 - iou_score: 0.7449 - val_loss: 0.4131 - va
l_iou_score: 0.5904
Epoch 95/100
6/6 [==============================] - 2s 372ms/step - loss: 0.2595 - iou_score: 0.7538 - val_loss: 0.4113 - va
l_iou_score: 0.5942
Epoch 96/100
6/6 [==============================] - 2s 372ms/step - loss: 0.2575 - iou_score: 0.7534 - val_loss: 0.4048 - va
l_iou_score: 0.6012
Epoch 97/100
6/6 [==============================] - 2s 374ms/step - loss: 0.2544 - iou_score: 0.7557 - val_loss: 0.4068 - va
l_iou_score: 0.5991
Epoch 98/100
6/6 [==============================] - 2s 373ms/step - loss: 0.2548 - iou_score: 0.7532 - val_loss: 0.4123 - va
l_iou_score: 0.5923
Epoch 99/100
6/6 [==============================] - 2s 373ms/step - loss: 0.2562 - iou_score: 0.7566 - val_loss: 0.4072 - va
l_iou_score: 0.6002
Epoch 100/100
6/6 [==============================] - 2s 369ms/step - loss: 0.2550 - iou_score: 0.7606 - val_loss: 0.4052 - va
l_iou_score: 0.5982
Epoch 1/100
6/6 [==============================] - 2s 384ms/step - loss: 0.9014 - iou_score: 0.1599 - val_loss: 0.8541 - va
l_iou_score: 0.1838
Epoch 2/100
6/6 [==============================] - 2s 373ms/step - loss: 0.8240 - iou_score: 0.2100 - val_loss: 0.8299 - va
l_iou_score: 0.2112
Epoch 3/100
6/6 [==============================] - 2s 373ms/step - loss: 0.7761 - iou_score: 0.2588 - val_loss: 0.7974 - va
l_iou_score: 0.2364
Epoch 4/100
6/6 [==============================] - 2s 371ms/step - loss: 0.7450 - iou_score: 0.2791 - val_loss: 0.7714 - va
l_iou_score: 0.2536
Epoch 5/100
6/6 [==============================] - 2s 373ms/step - loss: 0.7043 - iou_score: 0.3243 - val_loss: 0.7425 - va
l_iou_score: 0.2787
Epoch 6/100
6/6 [==============================] - 2s 371ms/step - loss: 0.6757 - iou_score: 0.3415 - val_loss: 0.7117 - va
l_iou_score: 0.3029
Epoch 7/100
6/6 [==============================] - 2s 373ms/step - loss: 0.6506 - iou_score: 0.3669 - val_loss: 0.6839 - va
l_iou_score: 0.3256
Epoch 8/100
6/6 [==============================] - 2s 373ms/step - loss: 0.6229 - iou_score: 0.4075 - val_loss: 0.6701 - va
l_iou_score: 0.3378
Epoch 9/100
6/6 [==============================] - 2s 376ms/step - loss: 0.6031 - iou_score: 0.4139 - val_loss: 0.6560 - va
l_iou_score: 0.3510
Epoch 10/100
6/6 [==============================] - 2s 374ms/step - loss: 0.5812 - iou_score: 0.4315 - val_loss: 0.6450 - va
l_iou_score: 0.3595
Epoch 11/100
6/6 [==============================] - 2s 373ms/step - loss: 0.5681 - iou_score: 0.4405 - val_loss: 0.6375 - va
l_iou_score: 0.3637
Epoch 12/100
6/6 [==============================] - 2s 372ms/step - loss: 0.5589 - iou_score: 0.4639 - val_loss: 0.6321 - va
l_iou_score: 0.3681
Epoch 13/100
6/6 [==============================] - 2s 374ms/step - loss: 0.5425 - iou_score: 0.4872 - val_loss: 0.6356 - va
l_iou_score: 0.3633
Epoch 14/100
6/6 [==============================] - 2s 372ms/step - loss: 0.5328 - iou_score: 0.4910 - val_loss: 0.6240 - va
l_iou_score: 0.3762
Epoch 15/100
6/6 [==============================] - 2s 373ms/step - loss: 0.5177 - iou_score: 0.4962 - val_loss: 0.6113 - va
l_iou_score: 0.3893
Epoch 16/100
6/6 [==============================] - 2s 371ms/step - loss: 0.5107 - iou_score: 0.5072 - val_loss: 0.6087 - va
l_iou_score: 0.3927
Epoch 17/100
6/6 [==============================] - 2s 373ms/step - loss: 0.5020 - iou_score: 0.5130 - val_loss: 0.5953 - va
l_iou_score: 0.4072
Epoch 18/100
6/6 [==============================] - 2s 373ms/step - loss: 0.4976 - iou_score: 0.5398 - val_loss: 0.5918 - va
l_iou_score: 0.4121
Epoch 19/100
6/6 [==============================] - 2s 374ms/step - loss: 0.4931 - iou_score: 0.5107 - val_loss: 0.5941 - va
l_iou_score: 0.4090
Epoch 20/100
6/6 [==============================] - 2s 371ms/step - loss: 0.4817 - iou_score: 0.5448 - val_loss: 0.5871 - va
l_iou_score: 0.4143
Epoch 21/100
6/6 [==============================] - 2s 374ms/step - loss: 0.4804 - iou_score: 0.5299 - val_loss: 0.5879 - va
l_iou_score: 0.4150
Epoch 22/100
6/6 [==============================] - 2s 373ms/step - loss: 0.4762 - iou_score: 0.5394 - val_loss: 0.5814 - va
l_iou_score: 0.4194
Epoch 23/100
6/6 [==============================] - 2s 373ms/step - loss: 0.4742 - iou_score: 0.5399 - val_loss: 0.5953 - va
l_iou_score: 0.4066
Epoch 24/100
6/6 [==============================] - 2s 374ms/step - loss: 0.4764 - iou_score: 0.5311 - val_loss: 0.5857 - va
l_iou_score: 0.4162
Epoch 25/100
6/6 [==============================] - 2s 371ms/step - loss: 0.4701 - iou_score: 0.5436 - val_loss: 0.5728 - va
l_iou_score: 0.4258
Epoch 26/100
6/6 [==============================] - 2s 374ms/step - loss: 0.4650 - iou_score: 0.5607 - val_loss: 0.5713 - va
l_iou_score: 0.4297
Epoch 27/100
6/6 [==============================] - 2s 374ms/step - loss: 0.4613 - iou_score: 0.5564 - val_loss: 0.5582 - va
l_iou_score: 0.4445
Epoch 28/100
6/6 [==============================] - 2s 374ms/step - loss: 0.4589 - iou_score: 0.5693 - val_loss: 0.5642 - va
l_iou_score: 0.4358
Epoch 29/100
6/6 [==============================] - 2s 372ms/step - loss: 0.4525 - iou_score: 0.5522 - val_loss: 0.5779 - va
l_iou_score: 0.4223
Epoch 30/100
6/6 [==============================] - 2s 375ms/step - loss: 0.4493 - iou_score: 0.5649 - val_loss: 0.5508 - va
l_iou_score: 0.4508
Epoch 31/100
6/6 [==============================] - 2s 372ms/step - loss: 0.4466 - iou_score: 0.5796 - val_loss: 0.5419 - va
l_iou_score: 0.4593
Epoch 32/100
6/6 [==============================] - 2s 374ms/step - loss: 0.4442 - iou_score: 0.5865 - val_loss: 0.5528 - va
l_iou_score: 0.4468
Epoch 33/100
6/6 [==============================] - 2s 376ms/step - loss: 0.4430 - iou_score: 0.5634 - val_loss: 0.5442 - va
l_iou_score: 0.4566
Epoch 34/100
6/6 [==============================] - 2s 372ms/step - loss: 0.4337 - iou_score: 0.5875 - val_loss: 0.5373 - va
l_iou_score: 0.4631
Epoch 35/100
6/6 [==============================] - 2s 371ms/step - loss: 0.4310 - iou_score: 0.5797 - val_loss: 0.5435 - va
l_iou_score: 0.4576
Epoch 36/100
6/6 [==============================] - 2s 373ms/step - loss: 0.4261 - iou_score: 0.5822 - val_loss: 0.5321 - va
l_iou_score: 0.4678
Epoch 37/100
6/6 [==============================] - 2s 372ms/step - loss: 0.4220 - iou_score: 0.6022 - val_loss: 0.5288 - va
l_iou_score: 0.4733
Epoch 38/100
6/6 [==============================] - 2s 373ms/step - loss: 0.4222 - iou_score: 0.5918 - val_loss: 0.5330 - va
l_iou_score: 0.4671
Epoch 39/100
6/6 [==============================] - 2s 372ms/step - loss: 0.4161 - iou_score: 0.6000 - val_loss: 0.5239 - va
l_iou_score: 0.4752
Epoch 40/100
6/6 [==============================] - 2s 370ms/step - loss: 0.4126 - iou_score: 0.5929 - val_loss: 0.5248 - va
l_iou_score: 0.4752
Epoch 41/100
6/6 [==============================] - 2s 373ms/step - loss: 0.4083 - iou_score: 0.5953 - val_loss: 0.5229 - va
l_iou_score: 0.4762
Epoch 42/100
6/6 [==============================] - 2s 373ms/step - loss: 0.4039 - iou_score: 0.6212 - val_loss: 0.5262 - va
l_iou_score: 0.4731
Epoch 43/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3998 - iou_score: 0.6142 - val_loss: 0.5218 - va
l_iou_score: 0.4782
Epoch 44/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3995 - iou_score: 0.6098 - val_loss: 0.5166 - va
l_iou_score: 0.4829
Epoch 45/100
6/6 [==============================] - 2s 373ms/step - loss: 0.4001 - iou_score: 0.6274 - val_loss: 0.5202 - va
l_iou_score: 0.4805
Epoch 46/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3980 - iou_score: 0.6125 - val_loss: 0.5236 - va
l_iou_score: 0.4743
Epoch 47/100
6/6 [==============================] - 2s 375ms/step - loss: 0.3946 - iou_score: 0.6170 - val_loss: 0.5135 - va
l_iou_score: 0.4875
Epoch 48/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3930 - iou_score: 0.6136 - val_loss: 0.5137 - va
l_iou_score: 0.4884
Epoch 49/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3905 - iou_score: 0.6324 - val_loss: 0.5139 - va
l_iou_score: 0.4823
Epoch 50/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3861 - iou_score: 0.6232 - val_loss: 0.5088 - va
l_iou_score: 0.4888
Epoch 51/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3833 - iou_score: 0.6165 - val_loss: 0.5083 - va
l_iou_score: 0.4880
Epoch 52/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3771 - iou_score: 0.6377 - val_loss: 0.5092 - va
l_iou_score: 0.4865
Epoch 53/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3766 - iou_score: 0.6336 - val_loss: 0.5007 - va
l_iou_score: 0.4966
Epoch 54/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3721 - iou_score: 0.6438 - val_loss: 0.5012 - va
l_iou_score: 0.4970
Epoch 55/100
6/6 [==============================] - 2s 371ms/step - loss: 0.3727 - iou_score: 0.6481 - val_loss: 0.5005 - va
l_iou_score: 0.4972
Epoch 56/100
6/6 [==============================] - 2s 371ms/step - loss: 0.3714 - iou_score: 0.6368 - val_loss: 0.4993 - va
l_iou_score: 0.5000
Epoch 57/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3672 - iou_score: 0.6462 - val_loss: 0.4953 - va
l_iou_score: 0.5039
Epoch 58/100
6/6 [==============================] - 2s 375ms/step - loss: 0.3653 - iou_score: 0.6495 - val_loss: 0.4939 - va
l_iou_score: 0.5074
Epoch 59/100
6/6 [==============================] - 2s 371ms/step - loss: 0.3609 - iou_score: 0.6567 - val_loss: 0.4945 - va
l_iou_score: 0.5057
Epoch 60/100
6/6 [==============================] - 2s 370ms/step - loss: 0.3584 - iou_score: 0.6488 - val_loss: 0.4922 - va
l_iou_score: 0.5092
Epoch 61/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3599 - iou_score: 0.6527 - val_loss: 0.4903 - va
l_iou_score: 0.5095
Epoch 62/100
6/6 [==============================] - 2s 371ms/step - loss: 0.3578 - iou_score: 0.6500 - val_loss: 0.4974 - va
l_iou_score: 0.5014
Epoch 63/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3592 - iou_score: 0.6325 - val_loss: 0.4924 - va
l_iou_score: 0.5070
Epoch 64/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3547 - iou_score: 0.6655 - val_loss: 0.4921 - va
l_iou_score: 0.5087
Epoch 65/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3552 - iou_score: 0.6427 - val_loss: 0.4961 - va
l_iou_score: 0.5040
Epoch 66/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3541 - iou_score: 0.6623 - val_loss: 0.4913 - va
l_iou_score: 0.5090
Epoch 67/100
6/6 [==============================] - 2s 375ms/step - loss: 0.3525 - iou_score: 0.6700 - val_loss: 0.4922 - va
l_iou_score: 0.5094
Epoch 68/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3559 - iou_score: 0.6528 - val_loss: 0.4912 - va
l_iou_score: 0.5108
Epoch 69/100
6/6 [==============================] - 2s 375ms/step - loss: 0.3555 - iou_score: 0.6587 - val_loss: 0.4936 - va
l_iou_score: 0.5099
Epoch 70/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3513 - iou_score: 0.6619 - val_loss: 0.4893 - va
l_iou_score: 0.5094
Epoch 71/100
6/6 [==============================] - 2s 375ms/step - loss: 0.3511 - iou_score: 0.6698 - val_loss: 0.4872 - va
l_iou_score: 0.5137
Epoch 72/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3469 - iou_score: 0.6558 - val_loss: 0.4816 - va
l_iou_score: 0.5200
Epoch 73/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3412 - iou_score: 0.6849 - val_loss: 0.4835 - va
l_iou_score: 0.5195
Epoch 74/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3408 - iou_score: 0.6744 - val_loss: 0.4848 - va
l_iou_score: 0.5189
Epoch 75/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3404 - iou_score: 0.6730 - val_loss: 0.4814 - va
l_iou_score: 0.5220
Epoch 76/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3383 - iou_score: 0.6705 - val_loss: 0.4828 - va
l_iou_score: 0.5205
Epoch 77/100
6/6 [==============================] - 2s 376ms/step - loss: 0.3379 - iou_score: 0.6763 - val_loss: 0.4801 - va
l_iou_score: 0.5241
Epoch 78/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3448 - iou_score: 0.6795 - val_loss: 0.4847 - va
l_iou_score: 0.5184
Epoch 79/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3458 - iou_score: 0.6452 - val_loss: 0.4940 - va
l_iou_score: 0.5102
Epoch 80/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3424 - iou_score: 0.6751 - val_loss: 0.4827 - va
l_iou_score: 0.5210
Epoch 81/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3387 - iou_score: 0.6758 - val_loss: 0.4827 - va
l_iou_score: 0.5218
Epoch 82/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3379 - iou_score: 0.6773 - val_loss: 0.4866 - va
l_iou_score: 0.5181
Epoch 83/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3373 - iou_score: 0.6661 - val_loss: 0.4895 - va
l_iou_score: 0.5144
Epoch 84/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3369 - iou_score: 0.6683 - val_loss: 0.4819 - va
l_iou_score: 0.5236
Epoch 85/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3334 - iou_score: 0.6739 - val_loss: 0.4796 - va
l_iou_score: 0.5242
Epoch 86/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3335 - iou_score: 0.6880 - val_loss: 0.4843 - va
l_iou_score: 0.5206
Epoch 87/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3325 - iou_score: 0.6811 - val_loss: 0.4847 - va
l_iou_score: 0.5214
Epoch 88/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3327 - iou_score: 0.6838 - val_loss: 0.4855 - va
l_iou_score: 0.5205
Epoch 89/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3339 - iou_score: 0.6677 - val_loss: 0.4788 - va
l_iou_score: 0.5269
Epoch 90/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3304 - iou_score: 0.6870 - val_loss: 0.4768 - va
l_iou_score: 0.5291
Epoch 91/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3287 - iou_score: 0.6779 - val_loss: 0.4762 - va
l_iou_score: 0.5293
Epoch 92/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3255 - iou_score: 0.6740 - val_loss: 0.4761 - va
l_iou_score: 0.5295
Epoch 93/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3250 - iou_score: 0.6851 - val_loss: 0.4735 - va
l_iou_score: 0.5289
Epoch 94/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3239 - iou_score: 0.6902 - val_loss: 0.4720 - va
l_iou_score: 0.5283
Epoch 95/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3210 - iou_score: 0.6760 - val_loss: 0.4728 - va
l_iou_score: 0.5281
Epoch 96/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3214 - iou_score: 0.6931 - val_loss: 0.4720 - va
l_iou_score: 0.5293
Epoch 97/100
6/6 [==============================] - 2s 377ms/step - loss: 0.3197 - iou_score: 0.6804 - val_loss: 0.4729 - va
l_iou_score: 0.5288
Epoch 98/100
6/6 [==============================] - 2s 375ms/step - loss: 0.3223 - iou_score: 0.6869 - val_loss: 0.4698 - va
l_iou_score: 0.5308
Epoch 99/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3216 - iou_score: 0.6679 - val_loss: 0.4742 - va
l_iou_score: 0.5272
Epoch 100/100
6/6 [==============================] - 2s 371ms/step - loss: 0.3200 - iou_score: 0.6980 - val_loss: 0.4728 - va
l_iou_score: 0.5286
Epoch 1/100
6/6 [==============================] - 2s 387ms/step - loss: 0.8507 - iou_score: 0.2005 - val_loss: 0.8158 - va
l_iou_score: 0.2252
Epoch 2/100
6/6 [==============================] - 2s 372ms/step - loss: 0.7778 - iou_score: 0.2684 - val_loss: 0.7913 - va
l_iou_score: 0.2479
Epoch 3/100
6/6 [==============================] - 2s 376ms/step - loss: 0.7338 - iou_score: 0.2999 - val_loss: 0.7562 - va
l_iou_score: 0.2767
Epoch 4/100
6/6 [==============================] - 2s 375ms/step - loss: 0.6981 - iou_score: 0.3253 - val_loss: 0.7259 - va
l_iou_score: 0.2959
Epoch 5/100
6/6 [==============================] - 2s 373ms/step - loss: 0.6670 - iou_score: 0.3550 - val_loss: 0.6980 - va
l_iou_score: 0.3181
Epoch 6/100
6/6 [==============================] - 2s 374ms/step - loss: 0.6374 - iou_score: 0.3814 - val_loss: 0.6742 - va
l_iou_score: 0.3398
Epoch 7/100
6/6 [==============================] - 2s 375ms/step - loss: 0.6111 - iou_score: 0.4079 - val_loss: 0.6553 - va
l_iou_score: 0.3550
Epoch 8/100
6/6 [==============================] - 2s 374ms/step - loss: 0.5861 - iou_score: 0.4279 - val_loss: 0.6384 - va
l_iou_score: 0.3700
Epoch 9/100
6/6 [==============================] - 2s 370ms/step - loss: 0.5640 - iou_score: 0.4551 - val_loss: 0.6271 - va
l_iou_score: 0.3796
Epoch 10/100
6/6 [==============================] - 2s 374ms/step - loss: 0.5433 - iou_score: 0.4760 - val_loss: 0.6195 - va
l_iou_score: 0.3870
Epoch 11/100
6/6 [==============================] - 2s 375ms/step - loss: 0.5282 - iou_score: 0.4945 - val_loss: 0.6088 - va
l_iou_score: 0.3945
Epoch 12/100
6/6 [==============================] - 2s 372ms/step - loss: 0.5133 - iou_score: 0.4957 - val_loss: 0.6056 - va
l_iou_score: 0.3993
Epoch 13/100
6/6 [==============================] - 2s 372ms/step - loss: 0.5055 - iou_score: 0.5092 - val_loss: 0.5925 - va
l_iou_score: 0.4123
Epoch 14/100
6/6 [==============================] - 2s 374ms/step - loss: 0.4977 - iou_score: 0.5121 - val_loss: 0.5947 - va
l_iou_score: 0.4109
Epoch 15/100
6/6 [==============================] - 2s 372ms/step - loss: 0.4864 - iou_score: 0.5363 - val_loss: 0.5901 - va
l_iou_score: 0.4149
Epoch 16/100
6/6 [==============================] - 2s 373ms/step - loss: 0.4764 - iou_score: 0.5482 - val_loss: 0.5751 - va
l_iou_score: 0.4280
Epoch 17/100
6/6 [==============================] - 2s 374ms/step - loss: 0.4665 - iou_score: 0.5531 - val_loss: 0.5769 - va
l_iou_score: 0.4224
Epoch 18/100
6/6 [==============================] - 2s 376ms/step - loss: 0.4589 - iou_score: 0.5611 - val_loss: 0.5675 - va
l_iou_score: 0.4327
Epoch 19/100
6/6 [==============================] - 2s 374ms/step - loss: 0.4567 - iou_score: 0.5477 - val_loss: 0.5698 - va
l_iou_score: 0.4302
Epoch 20/100
6/6 [==============================] - 2s 371ms/step - loss: 0.4478 - iou_score: 0.5640 - val_loss: 0.5618 - va
l_iou_score: 0.4366
Epoch 21/100
6/6 [==============================] - 2s 371ms/step - loss: 0.4415 - iou_score: 0.5715 - val_loss: 0.5618 - va
l_iou_score: 0.4395
Epoch 22/100
6/6 [==============================] - 2s 375ms/step - loss: 0.4388 - iou_score: 0.5862 - val_loss: 0.5653 - va
l_iou_score: 0.4357
Epoch 23/100
6/6 [==============================] - 2s 374ms/step - loss: 0.4333 - iou_score: 0.5771 - val_loss: 0.5563 - va
l_iou_score: 0.4464
Epoch 24/100
6/6 [==============================] - 2s 374ms/step - loss: 0.4287 - iou_score: 0.5936 - val_loss: 0.5459 - va
l_iou_score: 0.4543
Epoch 25/100
6/6 [==============================] - 2s 372ms/step - loss: 0.4268 - iou_score: 0.5911 - val_loss: 0.5419 - va
l_iou_score: 0.4592
Epoch 26/100
6/6 [==============================] - 2s 374ms/step - loss: 0.4213 - iou_score: 0.6085 - val_loss: 0.5479 - va
l_iou_score: 0.4529
Epoch 27/100
6/6 [==============================] - 2s 373ms/step - loss: 0.4165 - iou_score: 0.5971 - val_loss: 0.5375 - va
l_iou_score: 0.4632
Epoch 28/100
6/6 [==============================] - 2s 372ms/step - loss: 0.4102 - iou_score: 0.6040 - val_loss: 0.5411 - va
l_iou_score: 0.4598
Epoch 29/100
6/6 [==============================] - 2s 373ms/step - loss: 0.4078 - iou_score: 0.6119 - val_loss: 0.5288 - va
l_iou_score: 0.4731
Epoch 30/100
6/6 [==============================] - 2s 373ms/step - loss: 0.4070 - iou_score: 0.6054 - val_loss: 0.5330 - va
l_iou_score: 0.4686
Epoch 31/100
6/6 [==============================] - 2s 374ms/step - loss: 0.4070 - iou_score: 0.5874 - val_loss: 0.5313 - va
l_iou_score: 0.4724
Epoch 32/100
6/6 [==============================] - 2s 374ms/step - loss: 0.4082 - iou_score: 0.6063 - val_loss: 0.5276 - va
l_iou_score: 0.4665
Epoch 33/100
6/6 [==============================] - 2s 374ms/step - loss: 0.4114 - iou_score: 0.6098 - val_loss: 0.5262 - va
l_iou_score: 0.4704
Epoch 34/100
6/6 [==============================] - 2s 373ms/step - loss: 0.4061 - iou_score: 0.6032 - val_loss: 0.5222 - va
l_iou_score: 0.4759
Epoch 35/100
6/6 [==============================] - 2s 374ms/step - loss: 0.4011 - iou_score: 0.6144 - val_loss: 0.5209 - va
l_iou_score: 0.4798
Epoch 36/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3978 - iou_score: 0.6152 - val_loss: 0.5308 - va
l_iou_score: 0.4707
Epoch 37/100
6/6 [==============================] - 2s 375ms/step - loss: 0.3971 - iou_score: 0.6221 - val_loss: 0.5200 - va
l_iou_score: 0.4807
Epoch 38/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3914 - iou_score: 0.6283 - val_loss: 0.5171 - va
l_iou_score: 0.4842
Epoch 39/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3863 - iou_score: 0.6246 - val_loss: 0.5149 - va
l_iou_score: 0.4886
Epoch 40/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3822 - iou_score: 0.6322 - val_loss: 0.5124 - va
l_iou_score: 0.4880
Epoch 41/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3771 - iou_score: 0.6347 - val_loss: 0.5070 - va
l_iou_score: 0.4952
Epoch 42/100
6/6 [==============================] - 2s 371ms/step - loss: 0.3767 - iou_score: 0.6449 - val_loss: 0.5041 - va
l_iou_score: 0.4945
Epoch 43/100
6/6 [==============================] - 2s 376ms/step - loss: 0.3719 - iou_score: 0.6349 - val_loss: 0.5060 - va
l_iou_score: 0.4950
Epoch 44/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3688 - iou_score: 0.6533 - val_loss: 0.5032 - va
l_iou_score: 0.4955
Epoch 45/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3681 - iou_score: 0.6478 - val_loss: 0.4972 - va
l_iou_score: 0.5028
Epoch 46/100
6/6 [==============================] - 2s 371ms/step - loss: 0.3622 - iou_score: 0.6401 - val_loss: 0.4959 - va
l_iou_score: 0.5038
Epoch 47/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3587 - iou_score: 0.6643 - val_loss: 0.4947 - va
l_iou_score: 0.5061
Epoch 48/100
6/6 [==============================] - 2s 375ms/step - loss: 0.3627 - iou_score: 0.6465 - val_loss: 0.4938 - va
l_iou_score: 0.5060
Epoch 49/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3618 - iou_score: 0.6524 - val_loss: 0.4908 - va
l_iou_score: 0.5097
Epoch 50/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3604 - iou_score: 0.6585 - val_loss: 0.4913 - va
l_iou_score: 0.5087
Epoch 51/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3569 - iou_score: 0.6653 - val_loss: 0.4911 - va
l_iou_score: 0.5081
Epoch 52/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3550 - iou_score: 0.6674 - val_loss: 0.4870 - va
l_iou_score: 0.5125
Epoch 53/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3531 - iou_score: 0.6512 - val_loss: 0.4844 - va
l_iou_score: 0.5159
Epoch 54/100
6/6 [==============================] - 2s 375ms/step - loss: 0.3529 - iou_score: 0.6617 - val_loss: 0.4858 - va
l_iou_score: 0.5143
Epoch 55/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3457 - iou_score: 0.6647 - val_loss: 0.4801 - va
l_iou_score: 0.5202
Epoch 56/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3403 - iou_score: 0.6696 - val_loss: 0.4807 - va
l_iou_score: 0.5197
Epoch 57/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3396 - iou_score: 0.6661 - val_loss: 0.4811 - va
l_iou_score: 0.5184
Epoch 58/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3410 - iou_score: 0.6719 - val_loss: 0.4812 - va
l_iou_score: 0.5212
Epoch 59/100
6/6 [==============================] - 2s 371ms/step - loss: 0.3379 - iou_score: 0.6734 - val_loss: 0.4780 - va
l_iou_score: 0.5241
Epoch 60/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3339 - iou_score: 0.6641 - val_loss: 0.4812 - va
l_iou_score: 0.5201
Epoch 61/100
6/6 [==============================] - 2s 375ms/step - loss: 0.3319 - iou_score: 0.6707 - val_loss: 0.4782 - va
l_iou_score: 0.5218
Epoch 62/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3314 - iou_score: 0.6831 - val_loss: 0.4764 - va
l_iou_score: 0.5190
Epoch 63/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3308 - iou_score: 0.6883 - val_loss: 0.4776 - va
l_iou_score: 0.5175
Epoch 64/100
6/6 [==============================] - 2s 376ms/step - loss: 0.3294 - iou_score: 0.6876 - val_loss: 0.4737 - va
l_iou_score: 0.5225
Epoch 65/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3313 - iou_score: 0.6752 - val_loss: 0.4770 - va
l_iou_score: 0.5197
Epoch 66/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3327 - iou_score: 0.6876 - val_loss: 0.4783 - va
l_iou_score: 0.5188
Epoch 67/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3308 - iou_score: 0.6742 - val_loss: 0.4815 - va
l_iou_score: 0.5173
Epoch 68/100
6/6 [==============================] - 2s 375ms/step - loss: 0.3289 - iou_score: 0.6847 - val_loss: 0.4774 - va
l_iou_score: 0.5219
Epoch 69/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3264 - iou_score: 0.6885 - val_loss: 0.4743 - va
l_iou_score: 0.5250
Epoch 70/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3242 - iou_score: 0.6881 - val_loss: 0.4757 - va
l_iou_score: 0.5250
Epoch 71/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3188 - iou_score: 0.6994 - val_loss: 0.4887 - va
l_iou_score: 0.5129
Epoch 72/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3236 - iou_score: 0.6944 - val_loss: 0.4742 - va
l_iou_score: 0.5260
Epoch 73/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3240 - iou_score: 0.6748 - val_loss: 0.4779 - va
l_iou_score: 0.5231
Epoch 74/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3248 - iou_score: 0.6851 - val_loss: 0.4766 - va
l_iou_score: 0.5236
Epoch 75/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3208 - iou_score: 0.6954 - val_loss: 0.4803 - va
l_iou_score: 0.5202
```

```
Epoch 76/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3138 - iou_score: 0.6961 - val_loss: 0.4697 - va
l_iou_score: 0.5304
Epoch 77/100
6/6 [==============================] - 2s 377ms/step - loss: 0.3131 - iou_score: 0.6966 - val_loss: 0.4675 - va
l_iou_score: 0.5323
Epoch 78/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3115 - iou_score: 0.6992 - val_loss: 0.4674 - va
l_iou_score: 0.5304
Epoch 79/100
6/6 [==============================] - 2s 376ms/step - loss: 0.3085 - iou_score: 0.6972 - val_loss: 0.4681 - va
l_iou_score: 0.5316
Epoch 80/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3051 - iou_score: 0.7023 - val_loss: 0.4662 - va
l_iou_score: 0.5333
Epoch 81/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3043 - iou_score: 0.6916 - val_loss: 0.4662 - va
l_iou_score: 0.5328
Epoch 82/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3018 - iou_score: 0.7120 - val_loss: 0.4704 - va
l_iou_score: 0.5294
Epoch 83/100
6/6 [==============================] - 2s 374ms/step - loss: 0.2992 - iou_score: 0.7205 - val_loss: 0.4627 - va
l_iou_score: 0.5371
Epoch 84/100
6/6 [==============================] - 2s 372ms/step - loss: 0.2959 - iou_score: 0.7216 - val_loss: 0.4624 - va
l_iou_score: 0.5383
Epoch 85/100
6/6 [==============================] - 2s 373ms/step - loss: 0.2955 - iou_score: 0.7185 - val_loss: 0.4590 - va
l_iou_score: 0.5397
Epoch 86/100
6/6 [==============================] - 2s 373ms/step - loss: 0.2942 - iou_score: 0.7149 - val_loss: 0.4634 - va
l_iou_score: 0.5356
Epoch 87/100
6/6 [==============================] - 2s 371ms/step - loss: 0.2971 - iou_score: 0.7016 - val_loss: 0.4706 - va
l_iou_score: 0.5278
Epoch 88/100
6/6 [==============================] - 2s 373ms/step - loss: 0.2968 - iou_score: 0.7135 - val_loss: 0.4591 - va
l_iou_score: 0.5404
Epoch 89/100
6/6 [==============================] - 2s 373ms/step - loss: 0.2921 - iou_score: 0.7212 - val_loss: 0.4571 - va
l_iou_score: 0.5417
Epoch 90/100
6/6 [==============================] - 2s 376ms/step - loss: 0.2925 - iou_score: 0.7110 - val_loss: 0.4598 - va
l_iou_score: 0.5391
Epoch 91/100
6/6 [==============================] - 2s 371ms/step - loss: 0.2913 - iou_score: 0.7140 - val_loss: 0.4548 - va
l_iou_score: 0.5423
Epoch 92/100
6/6 [==============================] - 2s 374ms/step - loss: 0.2891 - iou_score: 0.7171 - val_loss: 0.4556 - va
l_iou_score: 0.5408
Epoch 93/100
6/6 [==============================] - 2s 372ms/step - loss: 0.2902 - iou_score: 0.7175 - val_loss: 0.4528 - va
l_iou_score: 0.5395
Epoch 94/100
6/6 [==============================] - 2s 372ms/step - loss: 0.2914 - iou_score: 0.7177 - val_loss: 0.4545 - va
l_iou_score: 0.5384
Epoch 95/100
6/6 [==============================] - 2s 371ms/step - loss: 0.2915 - iou_score: 0.7127 - val_loss: 0.4546 - va
l_iou_score: 0.5379
Epoch 96/100
6/6 [==============================] - 2s 374ms/step - loss: 0.2887 - iou_score: 0.7203 - val_loss: 0.4531 - va
l_iou_score: 0.5391
Epoch 97/100
6/6 [==============================] - 2s 371ms/step - loss: 0.2888 - iou_score: 0.7225 - val_loss: 0.4497 - va
l_iou_score: 0.5447
Epoch 98/100
6/6 [==============================] - 2s 373ms/step - loss: 0.2858 - iou_score: 0.7346 - val_loss: 0.4488 - va
l_iou_score: 0.5458
Epoch 99/100
6/6 [==============================] - 2s 372ms/step - loss: 0.2838 - iou_score: 0.7359 - val_loss: 0.4511 - va
l_iou_score: 0.5425
Epoch 100/100
6/6 [==============================] - 2s 369ms/step - loss: 0.2853 - iou_score: 0.7322 - val_loss: 0.4515 - va
l_iou_score: 0.5444
Epoch 1/100
6/6 [==============================] - 2s 383ms/step - loss: 0.8350 - iou_score: 0.2257 - val_loss: 0.7857 - va
l_iou_score: 0.2397
Epoch 2/100
6/6 [==============================] - 2s 372ms/step - loss: 0.7501 - iou_score: 0.2983 - val_loss: 0.7425 - va
l_iou_score: 0.2721
Epoch 3/100
6/6 [==============================] - 2s 372ms/step - loss: 0.7014 - iou_score: 0.3240 - val_loss: 0.7056 - va
l_iou_score: 0.3107
Epoch 4/100
6/6 [==============================] - 2s 372ms/step - loss: 0.6655 - iou_score: 0.3514 - val_loss: 0.6831 - va
l_iou_score: 0.3252
Epoch 5/100
6/6 [==============================] - 2s 370ms/step - loss: 0.6333 - iou_score: 0.3861 - val_loss: 0.6729 - va
l_iou_score: 0.3339
Epoch 6/100
6/6 [==============================] - 2s 373ms/step - loss: 0.6039 - iou_score: 0.4096 - val_loss: 0.6539 - va
l_iou_score: 0.3547
Epoch 7/100
6/6 [==============================] - 2s 370ms/step - loss: 0.5785 - iou_score: 0.4358 - val_loss: 0.6499 - va
l_iou_score: 0.3557
Epoch 8/100
6/6 [==============================] - 2s 372ms/step - loss: 0.5516 - iou_score: 0.4614 - val_loss: 0.6332 - va
l_iou_score: 0.3727
Epoch 9/100
6/6 [==============================] - 2s 371ms/step - loss: 0.5301 - iou_score: 0.4871 - val_loss: 0.6347 - va
l_iou_score: 0.3671
Epoch 10/100
6/6 [==============================] - 2s 374ms/step - loss: 0.5140 - iou_score: 0.4991 - val_loss: 0.6208 - va
l_iou_score: 0.3837
Epoch 11/100
6/6 [==============================] - 2s 373ms/step - loss: 0.5000 - iou_score: 0.5080 - val_loss: 0.6296 - va
l_iou_score: 0.3719
Epoch 12/100
6/6 [==============================] - 2s 373ms/step - loss: 0.4873 - iou_score: 0.5218 - val_loss: 0.6228 - va
l_iou_score: 0.3776
Epoch 13/100
6/6 [==============================] - 2s 373ms/step - loss: 0.4774 - iou_score: 0.5396 - val_loss: 0.6101 - va
l_iou_score: 0.3910
Epoch 14/100
6/6 [==============================] - 2s 373ms/step - loss: 0.4684 - iou_score: 0.5432 - val_loss: 0.6119 - va
l_iou_score: 0.3890
Epoch 15/100
6/6 [==============================] - 2s 371ms/step - loss: 0.4611 - iou_score: 0.5637 - val_loss: 0.5958 - va
l_iou_score: 0.4027
Epoch 16/100
6/6 [==============================] - 2s 372ms/step - loss: 0.4507 - iou_score: 0.5655 - val_loss: 0.5936 - va
l_iou_score: 0.4044
Epoch 17/100
6/6 [==============================] - 2s 372ms/step - loss: 0.4453 - iou_score: 0.5665 - val_loss: 0.5808 - va
l_iou_score: 0.4142
Epoch 18/100
6/6 [==============================] - 2s 374ms/step - loss: 0.4411 - iou_score: 0.5704 - val_loss: 0.5750 - va
l_iou_score: 0.4215
Epoch 19/100
6/6 [==============================] - 2s 370ms/step - loss: 0.4365 - iou_score: 0.5779 - val_loss: 0.5770 - va
l_iou_score: 0.4171
Epoch 20/100
6/6 [==============================] - 2s 375ms/step - loss: 0.4340 - iou_score: 0.5757 - val_loss: 0.5754 - va
l_iou_score: 0.4197
Epoch 21/100
6/6 [==============================] - 2s 374ms/step - loss: 0.4250 - iou_score: 0.5903 - val_loss: 0.5767 - va
l_iou_score: 0.4180
Epoch 22/100
6/6 [==============================] - 2s 372ms/step - loss: 0.4157 - iou_score: 0.5873 - val_loss: 0.5497 - va
l_iou_score: 0.4466
Epoch 23/100
6/6 [==============================] - 2s 371ms/step - loss: 0.4114 - iou_score: 0.6173 - val_loss: 0.5425 - va
l_iou_score: 0.4556
Epoch 24/100
6/6 [==============================] - 2s 370ms/step - loss: 0.4046 - iou_score: 0.5994 - val_loss: 0.5408 - va
l_iou_score: 0.4590
Epoch 25/100
6/6 [==============================] - 2s 373ms/step - loss: 0.4001 - iou_score: 0.6166 - val_loss: 0.5489 - va
l_iou_score: 0.4486
Epoch 26/100
6/6 [==============================] - 2s 372ms/step - loss: 0.4041 - iou_score: 0.6035 - val_loss: 0.5357 - va
l_iou_score: 0.4602
Epoch 27/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3989 - iou_score: 0.6079 - val_loss: 0.5236 - va
l_iou_score: 0.4753
Epoch 28/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3934 - iou_score: 0.6181 - val_loss: 0.5158 - va
l_iou_score: 0.4852
Epoch 29/100
6/6 [==============================] - 2s 370ms/step - loss: 0.3899 - iou_score: 0.6212 - val_loss: 0.5124 - va
l_iou_score: 0.4892
Epoch 30/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3931 - iou_score: 0.6114 - val_loss: 0.5276 - va
l_iou_score: 0.4737
Epoch 31/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3881 - iou_score: 0.6234 - val_loss: 0.5283 - va
l_iou_score: 0.4730
Epoch 32/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3899 - iou_score: 0.6295 - val_loss: 0.5204 - va
l_iou_score: 0.4815
Epoch 33/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3874 - iou_score: 0.6326 - val_loss: 0.5160 - va
l_iou_score: 0.4871
Epoch 34/100
6/6 [==============================] - 2s 371ms/step - loss: 0.3795 - iou_score: 0.6259 - val_loss: 0.5218 - va
l_iou_score: 0.4794
Epoch 35/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3777 - iou_score: 0.6221 - val_loss: 0.5096 - va
l_iou_score: 0.4926
Epoch 36/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3718 - iou_score: 0.6460 - val_loss: 0.4937 - va
l_iou_score: 0.5084
Epoch 37/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3669 - iou_score: 0.6356 - val_loss: 0.4956 - va
l_iou_score: 0.5075
Epoch 38/100
6/6 [==============================] - 2s 376ms/step - loss: 0.3642 - iou_score: 0.6447 - val_loss: 0.5072 - va
l_iou_score: 0.4935
Epoch 39/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3655 - iou_score: 0.6315 - val_loss: 0.4875 - va
l_iou_score: 0.5168
Epoch 40/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3602 - iou_score: 0.6577 - val_loss: 0.4891 - va
l_iou_score: 0.5132
Epoch 41/100
6/6 [==============================] - 2s 376ms/step - loss: 0.3559 - iou_score: 0.6578 - val_loss: 0.4943 - va
l_iou_score: 0.5084
Epoch 42/100
6/6 [==============================] - 2s 371ms/step - loss: 0.3523 - iou_score: 0.6580 - val_loss: 0.4860 - va
l_iou_score: 0.5133
Epoch 43/100
6/6 [==============================] - 2s 370ms/step - loss: 0.3467 - iou_score: 0.6560 - val_loss: 0.4895 - va
l_iou_score: 0.5107
Epoch 44/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3415 - iou_score: 0.6547 - val_loss: 0.4866 - va
l_iou_score: 0.5132
Epoch 45/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3381 - iou_score: 0.6593 - val_loss: 0.4847 - va
l_iou_score: 0.5147
Epoch 46/100
6/6 [==============================] - 2s 371ms/step - loss: 0.3337 - iou_score: 0.6714 - val_loss: 0.4852 - va
l_iou_score: 0.5140
Epoch 47/100
6/6 [==============================] - 2s 370ms/step - loss: 0.3310 - iou_score: 0.6811 - val_loss: 0.4895 - va
l_iou_score: 0.5113
Epoch 48/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3314 - iou_score: 0.6826 - val_loss: 0.4806 - va
l_iou_score: 0.5202
Epoch 49/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3287 - iou_score: 0.6831 - val_loss: 0.4820 - va
l_iou_score: 0.5201
Epoch 50/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3263 - iou_score: 0.6734 - val_loss: 0.4821 - va
l_iou_score: 0.5212
Epoch 51/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3250 - iou_score: 0.6813 - val_loss: 0.4803 - va
l_iou_score: 0.5233
Epoch 52/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3263 - iou_score: 0.6859 - val_loss: 0.4894 - va
l_iou_score: 0.5145
Epoch 53/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3268 - iou_score: 0.6843 - val_loss: 0.4881 - va
l_iou_score: 0.5153
Epoch 54/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3264 - iou_score: 0.6902 - val_loss: 0.4811 - va
l_iou_score: 0.5245
Epoch 55/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3250 - iou_score: 0.6847 - val_loss: 0.4863 - va
l_iou_score: 0.5187
Epoch 56/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3239 - iou_score: 0.6759 - val_loss: 0.4856 - va
l_iou_score: 0.5172
Epoch 57/100
6/6 [==============================] - 2s 370ms/step - loss: 0.3197 - iou_score: 0.6829 - val_loss: 0.4835 - va
l_iou_score: 0.5192
Epoch 58/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3192 - iou_score: 0.6730 - val_loss: 0.4734 - va
l_iou_score: 0.5292
Epoch 59/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3163 - iou_score: 0.6969 - val_loss: 0.4699 - va
l_iou_score: 0.5334
Epoch 60/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3184 - iou_score: 0.6857 - val_loss: 0.4745 - va
l_iou_score: 0.5297
Epoch 61/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3218 - iou_score: 0.6817 - val_loss: 0.4682 - va
l_iou_score: 0.5361
Epoch 62/100
6/6 [==============================] - 2s 371ms/step - loss: 0.3202 - iou_score: 0.6840 - val_loss: 0.4678 - va
l_iou_score: 0.5360
Epoch 63/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3181 - iou_score: 0.7020 - val_loss: 0.4666 - va
l_iou_score: 0.5362
Epoch 64/100
6/6 [==============================] - 2s 371ms/step - loss: 0.3162 - iou_score: 0.6890 - val_loss: 0.4664 - va
l_iou_score: 0.5357
Epoch 65/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3152 - iou_score: 0.6984 - val_loss: 0.4645 - va
l_iou_score: 0.5378
Epoch 66/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3131 - iou_score: 0.6921 - val_loss: 0.4669 - va
l_iou_score: 0.5358
Epoch 67/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3109 - iou_score: 0.6993 - val_loss: 0.4678 - va
l_iou_score: 0.5355
Epoch 68/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3118 - iou_score: 0.6919 - val_loss: 0.4690 - va
l_iou_score: 0.5342
Epoch 69/100
6/6 [==============================] - 2s 375ms/step - loss: 0.3110 - iou_score: 0.6865 - val_loss: 0.4672 - va
l_iou_score: 0.5359
Epoch 70/100
6/6 [==============================] - 2s 375ms/step - loss: 0.3073 - iou_score: 0.7032 - val_loss: 0.4653 - va
l_iou_score: 0.5370
Epoch 71/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3054 - iou_score: 0.6890 - val_loss: 0.4654 - va
l_iou_score: 0.5370
Epoch 72/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3073 - iou_score: 0.7074 - val_loss: 0.4664 - va
l_iou_score: 0.5348
Epoch 73/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3030 - iou_score: 0.7063 - val_loss: 0.4656 - va
l_iou_score: 0.5376
Epoch 74/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3048 - iou_score: 0.6962 - val_loss: 0.4624 - va
l_iou_score: 0.5414
Epoch 75/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3035 - iou_score: 0.7167 - val_loss: 0.4606 - va
l_iou_score: 0.5427
Epoch 76/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3020 - iou_score: 0.7153 - val_loss: 0.4617 - va
l_iou_score: 0.5405
Epoch 77/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3014 - iou_score: 0.7068 - val_loss: 0.4637 - va
l_iou_score: 0.5379
Epoch 78/100
6/6 [==============================] - 2s 374ms/step - loss: 0.2990 - iou_score: 0.7166 - val_loss: 0.4601 - va
l_iou_score: 0.5412
Epoch 79/100
6/6 [==============================] - 2s 373ms/step - loss: 0.2963 - iou_score: 0.7078 - val_loss: 0.4586 - va
l_iou_score: 0.5420
Epoch 80/100
6/6 [==============================] - 2s 374ms/step - loss: 0.2961 - iou_score: 0.7043 - val_loss: 0.4591 - va
l_iou_score: 0.5424
Epoch 81/100
6/6 [==============================] - 2s 374ms/step - loss: 0.2957 - iou_score: 0.7043 - val_loss: 0.4588 - va
l_iou_score: 0.5440
Epoch 82/100
6/6 [==============================] - 2s 370ms/step - loss: 0.2970 - iou_score: 0.7198 - val_loss: 0.4589 - va
l_iou_score: 0.5436
Epoch 83/100
6/6 [==============================] - 2s 373ms/step - loss: 0.2948 - iou_score: 0.7147 - val_loss: 0.4564 - va
l_iou_score: 0.5460
Epoch 84/100
6/6 [==============================] - 2s 373ms/step - loss: 0.2935 - iou_score: 0.7257 - val_loss: 0.4570 - va
l_iou_score: 0.5447
Epoch 85/100
6/6 [==============================] - 2s 372ms/step - loss: 0.2959 - iou_score: 0.7113 - val_loss: 0.4576 - va
l_iou_score: 0.5430
Epoch 86/100
6/6 [==============================] - 2s 373ms/step - loss: 0.2968 - iou_score: 0.7089 - val_loss: 0.4577 - va
l_iou_score: 0.5422
Epoch 87/100
6/6 [==============================] - 2s 373ms/step - loss: 0.2958 - iou_score: 0.7085 - val_loss: 0.4570 - va
l_iou_score: 0.5445
Epoch 88/100
6/6 [==============================] - 2s 373ms/step - loss: 0.2934 - iou_score: 0.7244 - val_loss: 0.4593 - va
l_iou_score: 0.5419
Epoch 89/100
6/6 [==============================] - 2s 371ms/step - loss: 0.2917 - iou_score: 0.7212 - val_loss: 0.4546 - va
l_iou_score: 0.5457
Epoch 90/100
6/6 [==============================] - 2s 371ms/step - loss: 0.2917 - iou_score: 0.7083 - val_loss: 0.4577 - va
l_iou_score: 0.5418
Epoch 91/100
6/6 [==============================] - 2s 372ms/step - loss: 0.2916 - iou_score: 0.7054 - val_loss: 0.4525 - va
l_iou_score: 0.5507
Epoch 92/100
6/6 [==============================] - 2s 372ms/step - loss: 0.2932 - iou_score: 0.7185 - val_loss: 0.4559 - va
l_iou_score: 0.5470
Epoch 93/100
6/6 [==============================] - 2s 372ms/step - loss: 0.2912 - iou_score: 0.7106 - val_loss: 0.4542 - va
l_iou_score: 0.5480
Epoch 94/100
6/6 [==============================] - 2s 371ms/step - loss: 0.2896 - iou_score: 0.7208 - val_loss: 0.4523 - va
l_iou_score: 0.5497
Epoch 95/100
6/6 [==============================] - 2s 372ms/step - loss: 0.2881 - iou_score: 0.7167 - val_loss: 0.4553 - va
l_iou_score: 0.5458
Epoch 96/100
6/6 [==============================] - 2s 373ms/step - loss: 0.2827 - iou_score: 0.7274 - val_loss: 0.4537 - va
l_iou_score: 0.5469
Epoch 97/100
6/6 [==============================] - 2s 372ms/step - loss: 0.2824 - iou_score: 0.7181 - val_loss: 0.4502 - va
l_iou_score: 0.5526
Epoch 98/100
6/6 [==============================] - 2s 372ms/step - loss: 0.2817 - iou_score: 0.7341 - val_loss: 0.4480 - va
l_iou_score: 0.5556
Epoch 99/100
6/6 [==============================] - 2s 372ms/step - loss: 0.2805 - iou_score: 0.7242 - val_loss: 0.4521 - va
l_iou_score: 0.5490
Epoch 100/100
6/6 [==============================] - 2s 369ms/step - loss: 0.2836 - iou_score: 0.7211 - val_loss: 0.4483 - va
l_iou_score: 0.5533
Epoch 1/100
6/6 [==============================] - 2s 381ms/step - loss: 0.8931 - iou_score: 0.2012 - val_loss: 0.8275 - va
l_iou_score: 0.2133
Epoch 2/100
6/6 [==============================] - 2s 375ms/step - loss: 0.7696 - iou_score: 0.2588 - val_loss: 0.7808 - va
l_iou_score: 0.2643
Epoch 3/100
6/6 [==============================] - 2s 374ms/step - loss: 0.7397 - iou_score: 0.2862 - val_loss: 0.7381 - va
l_iou_score: 0.2937
Epoch 4/100
6/6 [==============================] - 2s 373ms/step - loss: 0.6973 - iou_score: 0.3339 - val_loss: 0.7111 - va
l_iou_score: 0.3114
Epoch 5/100
6/6 [==============================] - 2s 374ms/step - loss: 0.6603 - iou_score: 0.3587 - val_loss: 0.6852 - va
l_iou_score: 0.3348
Epoch 6/100
6/6 [==============================] - 2s 374ms/step - loss: 0.6277 - iou_score: 0.4026 - val_loss: 0.6718 - va
l_iou_score: 0.3452
Epoch 7/100
6/6 [==============================] - 2s 372ms/step - loss: 0.5991 - iou_score: 0.4236 - val_loss: 0.6522 - va
l_iou_score: 0.3580
Epoch 8/100
6/6 [==============================] - 2s 373ms/step - loss: 0.5717 - iou_score: 0.4493 - val_loss: 0.6390 - va
l_iou_score: 0.3689
Epoch 9/100
6/6 [==============================] - 2s 375ms/step - loss: 0.5458 - iou_score: 0.4690 - val_loss: 0.6271 - va
l_iou_score: 0.3769
Epoch 10/100
6/6 [==============================] - 2s 372ms/step - loss: 0.5244 - iou_score: 0.4949 - val_loss: 0.6178 - va
l_iou_score: 0.3859
Epoch 11/100
6/6 [==============================] - 2s 374ms/step - loss: 0.5037 - iou_score: 0.5202 - val_loss: 0.6067 - va
l_iou_score: 0.3950
Epoch 12/100
6/6 [==============================] - 2s 374ms/step - loss: 0.4851 - iou_score: 0.5283 - val_loss: 0.6128 - va
l_iou_score: 0.3890
Epoch 13/100
6/6 [==============================] - 2s 372ms/step - loss: 0.4735 - iou_score: 0.5415 - val_loss: 0.5911 - va
l_iou_score: 0.4060
Epoch 14/100
6/6 [==============================] - 2s 374ms/step - loss: 0.4602 - iou_score: 0.5505 - val_loss: 0.5950 - va
l_iou_score: 0.4035
Epoch 15/100
6/6 [==============================] - 2s 373ms/step - loss: 0.4479 - iou_score: 0.5733 - val_loss: 0.5835 - va
l_iou_score: 0.4136
Epoch 16/100
6/6 [==============================] - 2s 373ms/step - loss: 0.4416 - iou_score: 0.5721 - val_loss: 0.5731 - va
l_iou_score: 0.4292
Epoch 17/100
6/6 [==============================] - 2s 374ms/step - loss: 0.4351 - iou_score: 0.5777 - val_loss: 0.5798 - va
l_iou_score: 0.4159
Epoch 18/100
6/6 [==============================] - 2s 370ms/step - loss: 0.4273 - iou_score: 0.5987 - val_loss: 0.5511 - va
l_iou_score: 0.4482
Epoch 19/100
6/6 [==============================] - 2s 372ms/step - loss: 0.4181 - iou_score: 0.6059 - val_loss: 0.5423 - va
l_iou_score: 0.4534
Epoch 20/100
6/6 [==============================] - 2s 372ms/step - loss: 0.4152 - iou_score: 0.6076 - val_loss: 0.5449 - va
l_iou_score: 0.4550
Epoch 21/100
6/6 [==============================] - 2s 374ms/step - loss: 0.4099 - iou_score: 0.6075 - val_loss: 0.5391 - va
l_iou_score: 0.4598
Epoch 22/100
6/6 [==============================] - 2s 374ms/step - loss: 0.4026 - iou_score: 0.6283 - val_loss: 0.5373 - va
l_iou_score: 0.4607
Epoch 23/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3998 - iou_score: 0.6153 - val_loss: 0.5367 - va
l_iou_score: 0.4611
Epoch 24/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3962 - iou_score: 0.6237 - val_loss: 0.5411 - va
l_iou_score: 0.4546
Epoch 25/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3905 - iou_score: 0.6361 - val_loss: 0.5262 - va
l_iou_score: 0.4720
Epoch 26/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3853 - iou_score: 0.6299 - val_loss: 0.5348 - va
l_iou_score: 0.4640
Epoch 27/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3775 - iou_score: 0.6326 - val_loss: 0.5316 - va
l_iou_score: 0.4693
Epoch 28/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3753 - iou_score: 0.6546 - val_loss: 0.5294 - va
l_iou_score: 0.4707
Epoch 29/100
6/6 [==============================] - 2s 370ms/step - loss: 0.3740 - iou_score: 0.6439 - val_loss: 0.5139 - va
l_iou_score: 0.4858
Epoch 30/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3664 - iou_score: 0.6479 - val_loss: 0.5070 - va
l_iou_score: 0.4949
Epoch 31/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3647 - iou_score: 0.6552 - val_loss: 0.5000 - va
l_iou_score: 0.5020
Epoch 32/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3639 - iou_score: 0.6568 - val_loss: 0.5037 - va
l_iou_score: 0.4969
Epoch 33/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3616 - iou_score: 0.6518 - val_loss: 0.5087 - va
l_iou_score: 0.4908
Epoch 34/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3591 - iou_score: 0.6443 - val_loss: 0.5029 - va
l_iou_score: 0.4935
Epoch 35/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3539 - iou_score: 0.6658 - val_loss: 0.4931 - va
l_iou_score: 0.5037
Epoch 36/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3526 - iou_score: 0.6594 - val_loss: 0.4903 - va
l_iou_score: 0.5134
Epoch 37/100
6/6 [==============================] - 2s 375ms/step - loss: 0.3546 - iou_score: 0.6519 - val_loss: 0.4884 - va
l_iou_score: 0.5141
Epoch 38/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3509 - iou_score: 0.6720 - val_loss: 0.4925 - va
l_iou_score: 0.5111
Epoch 39/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3466 - iou_score: 0.6747 - val_loss: 0.4828 - va
l_iou_score: 0.5231
Epoch 40/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3426 - iou_score: 0.6752 - val_loss: 0.4876 - va
l_iou_score: 0.5153
Epoch 41/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3448 - iou_score: 0.6796 - val_loss: 0.4915 - va
l_iou_score: 0.5122
Epoch 42/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3426 - iou_score: 0.6661 - val_loss: 0.4849 - va
l_iou_score: 0.5196
Epoch 43/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3386 - iou_score: 0.6732 - val_loss: 0.4849 - va
l_iou_score: 0.5193
Epoch 44/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3370 - iou_score: 0.6879 - val_loss: 0.4862 - va
l_iou_score: 0.5208
Epoch 45/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3348 - iou_score: 0.6726 - val_loss: 0.4896 - va
l_iou_score: 0.5166
Epoch 46/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3321 - iou_score: 0.6879 - val_loss: 0.4806 - va
l_iou_score: 0.5225
Epoch 47/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3309 - iou_score: 0.6839 - val_loss: 0.4752 - va
l_iou_score: 0.5236
Epoch 48/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3288 - iou_score: 0.6965 - val_loss: 0.4787 - va
l_iou_score: 0.5299
Epoch 49/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3273 - iou_score: 0.6880 - val_loss: 0.4757 - va
l_iou_score: 0.5286
Epoch 50/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3236 - iou_score: 0.6931 - val_loss: 0.4828 - va
l_iou_score: 0.5169
Epoch 51/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3193 - iou_score: 0.6848 - val_loss: 0.4776 - va
l_iou_score: 0.5242
Epoch 52/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3146 - iou_score: 0.7021 - val_loss: 0.4694 - va
l_iou_score: 0.5336
Epoch 53/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3159 - iou_score: 0.6961 - val_loss: 0.4721 - va
l_iou_score: 0.5314
Epoch 54/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3137 - iou_score: 0.6952 - val_loss: 0.4696 - va
l_iou_score: 0.5370
Epoch 55/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3156 - iou_score: 0.7130 - val_loss: 0.4658 - va
l_iou_score: 0.5385
Epoch 56/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3173 - iou_score: 0.7046 - val_loss: 0.4715 - va
l_iou_score: 0.5362
Epoch 57/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3153 - iou_score: 0.6943 - val_loss: 0.4733 - va
l_iou_score: 0.5323
Epoch 58/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3111 - iou_score: 0.6955 - val_loss: 0.4665 - va
l_iou_score: 0.5399
Epoch 59/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3076 - iou_score: 0.7037 - val_loss: 0.4665 - va
l_iou_score: 0.5410
Epoch 60/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3076 - iou_score: 0.7048 - val_loss: 0.4664 - va
l_iou_score: 0.5395
Epoch 61/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3057 - iou_score: 0.7079 - val_loss: 0.4588 - va
l_iou_score: 0.5465
Epoch 62/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3058 - iou_score: 0.6927 - val_loss: 0.4586 - va
l_iou_score: 0.5479
Epoch 63/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3066 - iou_score: 0.6910 - val_loss: 0.4669 - va
l_iou_score: 0.5381
Epoch 64/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3046 - iou_score: 0.7005 - val_loss: 0.4601 - va
l_iou_score: 0.5457
Epoch 65/100
6/6 [==============================] - 2s 375ms/step - loss: 0.3036 - iou_score: 0.6989 - val_loss: 0.4577 - va
l_iou_score: 0.5479
Epoch 66/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3036 - iou_score: 0.7059 - val_loss: 0.4599 - va
l_iou_score: 0.5461
Epoch 67/100
6/6 [==============================] - 2s 372ms/step - loss: 0.3027 - iou_score: 0.7121 - val_loss: 0.4581 - va
l_iou_score: 0.5472
Epoch 68/100
6/6 [==============================] - 2s 374ms/step - loss: 0.3007 - iou_score: 0.7041 - val_loss: 0.4557 - va
l_iou_score: 0.5494
Epoch 69/100
6/6 [==============================] - 2s 373ms/step - loss: 0.3020 - iou_score: 0.7179 - val_loss: 0.4566 - va
l_iou_score: 0.5489
Epoch 70/100
6/6 [==============================] - 2s 375ms/step - loss: 0.2990 - iou_score: 0.7184 - val_loss: 0.4603 - va
l_iou_score: 0.5446
Epoch 71/100
6/6 [==============================] - 2s 373ms/step - loss: 0.2968 - iou_score: 0.7225 - val_loss: 0.4513 - va
l_iou_score: 0.5545
Epoch 72/100
6/6 [==============================] - 2s 374ms/step - loss: 0.2941 - iou_score: 0.7221 - val_loss: 0.4497 - va
l_iou_score: 0.5559
Epoch 73/100
6/6 [==============================] - 2s 373ms/step - loss: 0.2879 - iou_score: 0.7108 - val_loss: 0.4511 - va
l_iou_score: 0.5529
Epoch 74/100
6/6 [==============================] - 2s 374ms/step - loss: 0.2872 - iou_score: 0.7347 - val_loss: 0.4517 - va
l_iou_score: 0.5512
Epoch 75/100
6/6 [==============================] - 2s 373ms/step - loss: 0.2851 - iou_score: 0.7252 - val_loss: 0.4518 - va
l_iou_score: 0.5493
Epoch 76/100
6/6 [==============================] - 2s 374ms/step - loss: 0.2875 - iou_score: 0.7234 - val_loss: 0.4466 - va
l_iou_score: 0.5551
Epoch 77/100
6/6 [==============================] - 2s 373ms/step - loss: 0.2876 - iou_score: 0.7061 - val_loss: 0.4456 - va
l_iou_score: 0.5567
Epoch 78/100
6/6 [==============================] - 2s 375ms/step - loss: 0.2903 - iou_score: 0.7303 - val_loss: 0.4483 - va
l_iou_score: 0.5542
Epoch 79/100
6/6 [==============================] - 2s 372ms/step - loss: 0.2878 - iou_score: 0.7338 - val_loss: 0.4639 - va
l_iou_score: 0.5379
Epoch 80/100
6/6 [==============================] - 2s 373ms/step - loss: 0.2879 - iou_score: 0.7327 - val_loss: 0.4448 - va
l_iou_score: 0.5608
Epoch 81/100
6/6 [==============================] - 2s 373ms/step - loss: 0.2859 - iou_score: 0.7208 - val_loss: 0.4433 - va
l_iou_score: 0.5610
Epoch 82/100
6/6 [==============================] - 2s 372ms/step - loss: 0.2833 - iou_score: 0.7348 - val_loss: 0.4435 - va
l_iou_score: 0.5611
Epoch 83/100
6/6 [==============================] - 2s 374ms/step - loss: 0.2825 - iou_score: 0.7198 - val_loss: 0.4458 - va
l_iou_score: 0.5588
Epoch 84/100
6/6 [==============================] - 2s 373ms/step - loss: 0.2800 - iou_score: 0.7239 - val_loss: 0.4451 - va
l_iou_score: 0.5578
Epoch 85/100
6/6 [==============================] - 2s 373ms/step - loss: 0.2799 - iou_score: 0.7269 - val_loss: 0.4423 - va
l_iou_score: 0.5586
Epoch 86/100
6/6 [==============================] - 2s 373ms/step - loss: 0.2795 - iou_score: 0.7350 - val_loss: 0.4505 - va
l_iou_score: 0.5492
Epoch 87/100
6/6 [==============================] - 2s 373ms/step - loss: 0.2786 - iou_score: 0.7426 - val_loss: 0.4388 - va
l_iou_score: 0.5650
Epoch 88/100
6/6 [==============================] - 2s 373ms/step - loss: 0.2772 - iou_score: 0.7466 - val_loss: 0.4385 - va
l_iou_score: 0.5636
Epoch 89/100
6/6 [==============================] - 2s 373ms/step - loss: 0.2745 - iou_score: 0.7361 - val_loss: 0.4408 - va
l_iou_score: 0.5616
Epoch 90/100
6/6 [==============================] - 2s 374ms/step - loss: 0.2719 - iou_score: 0.7458 - val_loss: 0.4415 - va
l_iou_score: 0.5594
Epoch 91/100
6/6 [==============================] - 2s 374ms/step - loss: 0.2727 - iou_score: 0.7379 - val_loss: 0.4384 - va
l_iou_score: 0.5665
Epoch 92/100
6/6 [==============================] - 2s 372ms/step - loss: 0.2711 - iou_score: 0.7434 - val_loss: 0.4368 - va
l_iou_score: 0.5660
Epoch 93/100
6/6 [==============================] - 2s 373ms/step - loss: 0.2686 - iou_score: 0.7381 - val_loss: 0.4352 - va
l_iou_score: 0.5699
Epoch 94/100
6/6 [==============================] - 2s 375ms/step - loss: 0.2683 - iou_score: 0.7516 - val_loss: 0.4333 - va
l_iou_score: 0.5715
Epoch 95/100
6/6 [==============================] - 2s 369ms/step - loss: 0.2692 - iou_score: 0.7423 - val_loss: 0.4307 - va
l_iou_score: 0.5756
Epoch 96/100
6/6 [==============================] - 2s 372ms/step - loss: 0.2689 - iou_score: 0.7429 - val_loss: 0.4375 - va
l_iou_score: 0.5655
Epoch 97/100
6/6 [==============================] - 2s 373ms/step - loss: 0.2680 - iou_score: 0.7549 - val_loss: 0.4316 - va
l_iou_score: 0.5735
Epoch 98/100
6/6 [==============================] - 2s 374ms/step - loss: 0.2692 - iou_score: 0.7486 - val_loss: 0.4360 - va
l_iou_score: 0.5673
Epoch 99/100
6/6 [==============================] - 2s 374ms/step - loss: 0.2676 - iou_score: 0.7520 - val_loss: 0.4331 - va
l_iou_score: 0.5708
Epoch 100/100
6/6 [==============================] - 2s 371ms/step - loss: 0.2661 - iou_score: 0.7446 - val_loss: 0.4326 - va
l_iou_score: 0.5713
```

In [4]:

```
# define a function to plot val_iou_score
def plotval_iou_score(i,data):
  plt.subplot(5,3,i)
  plt.plot(data)
  plt.title('Sub {}'.format(i))
  plt.xlabel('Epoch')
  plt.ylabel('val_iou_score')
```

In [8]:

```
# plot the val_iou_scorelist
plt.subplots(5,3,constrained_layout = True,figsize = (15,10))
plt.subplots_adjust(left=None, bottom=None, right=None, top=None, wspace=None, hspace=1)
plt.suptitle('Train Set Scores',y = 1)
for i in range(1,16):
  plotval_iou_score(i,val_iou_scorelist[i])
```

```
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:3: UserWarning: This figure was using constrained_
layout==True, but that is incompatible with subplots_adjust and or tight_layout: setting constrained_layout==Fa
lse.
  This is separate from the ipykernel package so we can avoid doing imports until
```

Note: From the figures above, we found the val_iou_core becomes generally high and converges to a specific value.

In [9]:

```
# test the model with the test data
t1prelist = [[]]
groundseglist = [[]]
t1list = [[]]
for test in testfile:
  groundseg = nib.load(os.path.join('/content/drive/My Drive/final example/',test)+'/tof_arteries.nii.gz').get_
  t1 = nib.load(os.path.join('/content/drive/My Drive/final example/',test)+'/t1_in_tof.nii.gz').get_data()
  groundseg = np.swapaxes(groundseg,0,2)
  t1 = np.swapaxes(t1,0,2)
  t1 = np.expand_dims(t1,axis=3)
  t1 = np.tile(t1,[1,1,1,3])
  t1list.append(t1)
  t1prelist.append(model.predict(t1))
  groundseglist.append(groundseg)
```

```
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:6: DeprecationWarning: get_data() is deprecated in
favor of get_fdata(), which has a more predictable return type. To obtain get_data() behavior going forward, us
e numpy.asanyarray(img.dataobj).

* deprecated from version: 3.0
* Will raise <class 'nibabel.deprecator.ExpiredDeprecationError'> as of version: 5.0

/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:7: DeprecationWarning: get_data() is deprecated in
favor of get_fdata(), which has a more predictable return type. To obtain get_data() behavior going forward, us
e numpy.asanyarray(img.dataobj).

* deprecated from version: 3.0
* Will raise <class 'nibabel.deprecator.ExpiredDeprecationError'> as of version: 5.0
  import sys
```

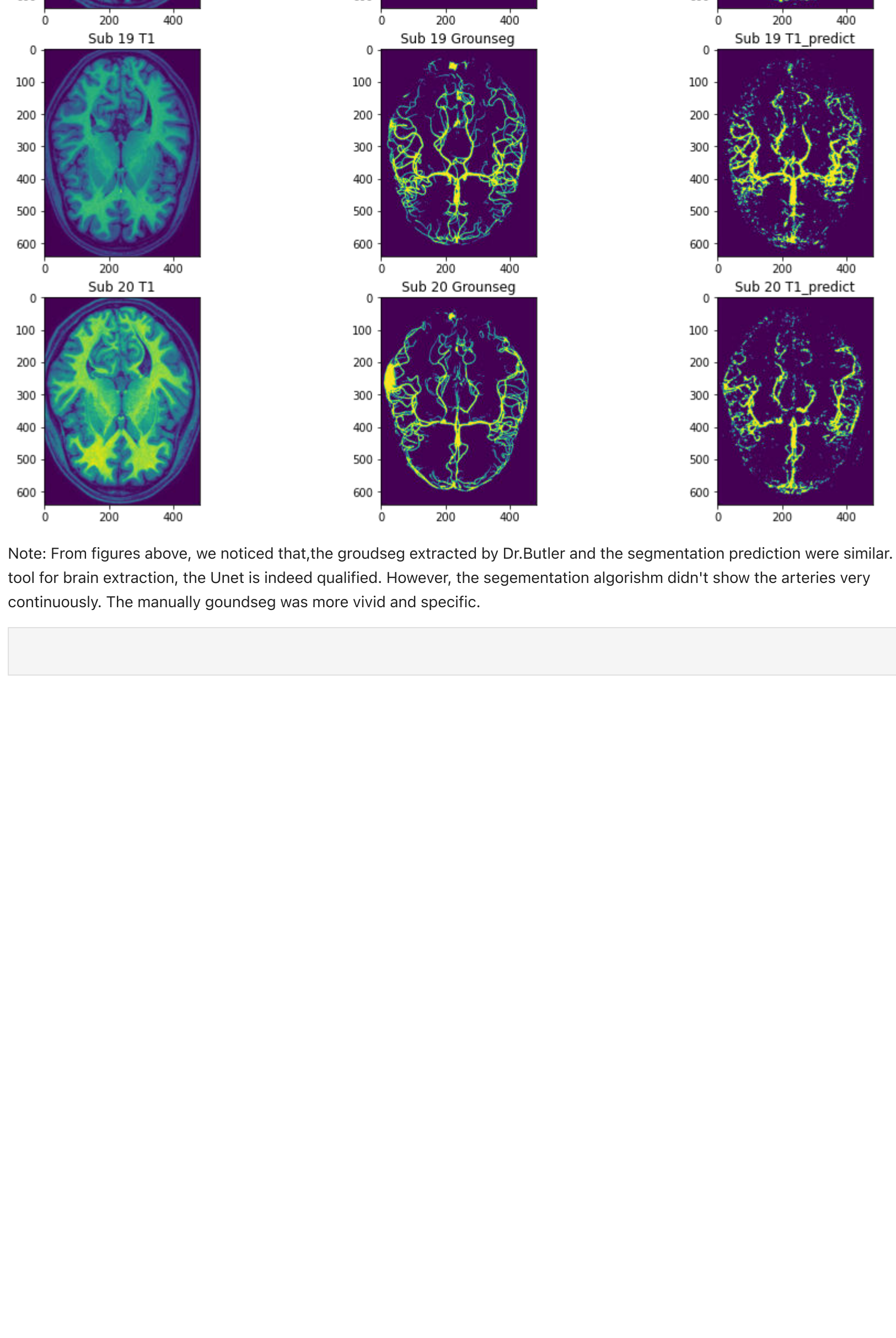
Note: We created three lists to store the test image sets.

In [10]:

```
# define a function to plot the comparison between groundseg and the predict t1
def plottestresult(i,t1,groundseg,t1_predict):
  plt.subplot(5,3,i*3+1)
  plt.imshow(t1[75,:,:,0])
  plt.title('Sub {} T1'.format(16+i))
  plt.subplot(5,3,i*3+2)
  plt.imshow(np.max(groundseg[:,:,:],axis = 0))
  plt.title('Sub {} Grounseg'.format(16+i))
  plt.subplot(5,3,i*3+3)
  plt.imshow(np.max(t1_predict[:,:,:,0],axis = 0))
  plt.title('Sub {} T1_predict'.format(16+i))
```

In [11]:

```
plt.subplots(5,3,figsize = (15,20))
plt.subplots_adjust(left=0.1, bottom=None, right=None, top=None, wspace=None, hspace=None)
plt.suptitle('Test Set Results',y = 0.9)
for i in range(0,5):
  plottestresult(i,t1list[i+1],groundseglist[i+1],t1prelist[i+1])
```



Note: From figures above, we noticed that,the groudseg extracted by Dr.Butler and the segmentation prediction were similar. As a tool for brain extraction, the Unet is indeed qualified. However, the segementation algorishm didn't show the arteries very continuously. The manually goundseg was more vivid and specific.

In [ ]: