HUB4045F - Liam Breytenbach - BRYLIA002 - Assignment 4

August 28, 2025

#

HUB4045F - Liam Breytenbach - BRYLIA002 - Assignment 4:

13 May 2023

#

Necessary Packages:

- Scipy
- Numpy
- OpenCV
- Pydicom
- Matplotlib
- Trimesh

#

Viewing Installed Packages

[1]:
```
pip list
```

```
Package                            Version
---------------------------------- -------------------
alabaster                          0.7.12
anaconda-client                    1.7.2
anaconda-navigator                 1.9.12
anaconda-project                   0.8.3
argh                               0.26.2
asn1crypto                         1.3.0
astroid                            2.3.3
astropy                            4.0
atomicwrites                       1.3.0
attrs                              19.3.0
autopep8                           1.4.4
Babel                              2.8.0
backcall                           0.1.0
backports.functools-lru-cache      1.6.1
backports.shutil-get-terminal-size 1.0.0
backports.tempfile                 1.0
backports.weakref                  1.0.post1
```

bcrypt	3.1.7
beautifulsoup4	4.8.2
bitarray	1.2.1
bkcharts	0.2
bleach	3.1.0
bokeh	1.4.0
boto	2.49.0
Bottleneck	1.3.2
certifi	2019.11.28
cffi	1.14.0
chardet	3.0.4
Click	7.0
cloudpickle	1.3.0
clyent	1.2.2
colorama	0.4.3
comtypes	1.1.7
conda	4.8.2
conda-build	3.18.11
conda-package-handling	1.6.0
conda-verify	3.4.2
contextlib2	0.6.0.post1
cryptography	2.8
cycler	0.10.0
Cython	0.29.15
cytoolz	0.10.1
dask	2.11.0
decorator	4.4.1
defusedxml	0.6.0
diff-match-patch	20181111
distributed	2.11.0
docutils	0.16
entrypoints	0.3
et-xmlfile	1.0.1
fastcache	1.1.0
filelock	3.0.12
flake8	3.7.9
Flask	1.1.1
freetype-py	2.4.0
fsspec	0.6.2
future	0.18.2
gevent	1.4.0
glob2	0.7
greenlet	0.4.15
h5py	2.10.0
HeapDict	1.0.1
html5lib	1.0.1
hypothesis	5.5.4
idna	2.8

imageio	2.6.1
imagesize	1.2.0
importlib-metadata	1.5.0
importlib-resources	5.12.0
intervaltree	3.0.2
ipydatawidgets	4.3.3
ipykernel	5.1.4
ipython	7.12.0
ipython-genutils	0.2.0
ipywidgets	7.5.1
isort	4.3.21
itsdangerous	1.1.0
jdcal	1.4.1
jedi	0.14.1
Jinja2	2.11.1
joblib	0.14.1
json5	0.9.1
jsonschema	3.2.0
jupyter	1.0.0
jupyter-client	5.3.4
jupyter-console	6.1.0
jupyter-core	4.6.1
jupyterlab	1.2.6
jupyterlab-server	1.0.6
keyring	21.1.0
kiwisolver	1.1.0
lazy-object-proxy	1.4.3
libarchive-c	2.8
llvmlite	0.31.0
locket	0.2.0
lxml	4.5.0
MarkupSafe	1.1.1
matplotlib	3.1.3
mccabe	0.6.1
menuinst	1.4.16
mistune	0.8.4
mkl-fft	1.0.15
mkl-random	1.1.0
mkl-service	2.3.0
mock	4.0.1
more-itertools	8.2.0
mpmath	1.1.0
msgpack	0.6.1
multipledispatch	0.6.0
navigator-updater	0.2.1
nbconvert	5.6.1
nbformat	5.0.4
networkx	2.4

nltk	3.4.5
nose	1.3.7
notebook	6.0.3
numba	0.48.0
numexpr	2.7.1
numpy	1.18.1
numpydoc	0.9.2
olefile	0.46
opencv-python	4.7.0.72
openpyxl	3.0.3
packaging	20.1
pandas	1.0.1
pandocfilters	1.4.2
paramiko	2.7.1
parso	0.5.2
partd	1.1.0
path	13.1.0
pathlib2	2.3.5
pathtools	0.1.2
patsy	0.5.1
pep8	1.7.1
pexpect	4.8.0
pickleshare	0.7.5
Pillow	7.0.0
pip	20.0.2
pkginfo	1.5.0.1
platformdirs	3.5.1
pluggy	0.13.1
ply	3.11
pooch	1.7.0
prometheus-client	0.7.1
prompt-toolkit	3.0.3
psutil	5.6.7
py	1.8.1
pycodestyle	2.5.0
pycosat	0.6.3
pycparser	2.19
pycrypto	2.6.1
pycurl	7.43.0.5
pydicom	2.3.1
pydocstyle	4.0.1
pyflakes	2.1.1
pyglet	2.0.7
Pygments	2.5.2
pylint	2.4.4
PyNaCl	1.3.0
pyodbc	4.0.0-unsupported
PyOpenGL	3.1.0

pyOpenSSL	19.1.0
pyparsing	2.4.6
pyreadline	2.1
pyrender	0.1.45
pyrsistent	0.15.7
PySocks	1.7.1
pytest	5.3.5
pytest-arraydiff	0.3
pytest-astropy	0.8.0
pytest-astropy-header	0.1.2
pytest-doctestplus	0.5.0
pytest-openfiles	0.4.0
pytest-remotedata	0.3.2
python-dateutil	2.8.1
python-jsonrpc-server	0.3.4
python-language-server	0.31.7
pythreejs	2.4.2
pytz	2019.3
pyvista	0.38.6
pyvistaqt	0.10.0
PyWavelets	1.1.1
pywin32	227
pywin32-ctypes	0.2.0
pywinpty	0.5.7
PyYAML	5.3
pyzmq	18.1.1
QDarkStyle	2.8
QtAwesome	0.6.1
qtconsole	4.6.0
QtPy	1.9.0
requests	2.22.0
rope	0.16.0
Rtree	0.9.3
ruamel-yaml	0.15.87
scikit-image	0.16.2
scikit-learn	0.22.1
scipy	1.4.1
scooby	0.7.2
seaborn	0.10.0
Send2Trash	1.5.0
setuptools	45.2.0.post20200210
simplegeneric	0.8.1
singledispatch	3.4.0.3
six	1.14.0
snowballstemmer	2.0.0
sortedcollections	1.1.2
sortedcontainers	2.1.0
soupsieve	1.9.5

```
Sphinx                             2.4.0
sphinxcontrib-applehelp            1.0.1
sphinxcontrib-devhelp              1.0.1
sphinxcontrib-htmlhelp             1.0.2
sphinxcontrib-jsmath               1.0.1
sphinxcontrib-qthelp               1.0.2
sphinxcontrib-serializinghtml      1.1.3
sphinxcontrib-websupport           1.2.0
spyder                             4.0.1
spyder-kernels                     1.8.1
SQLAlchemy                         1.3.13
statsmodels                        0.11.0
sympy                              1.5.1
tables                             3.6.1
tblib                              1.6.0
terminado                          0.8.3
testpath                           0.4.4
toolz                              0.10.0
tornado                            6.0.3
tqdm                               4.42.1
traitlets                          4.3.3
traittypes                         0.2.1
trimesh                            3.21.6
typing-extensions                  4.5.0
ujson                              1.35
unicodecsv                         0.14.1
urllib3                            1.25.8
vtk                                9.2.6
watchdog                           0.10.2
wcwidth                            0.1.8
webencodings                       0.5.1
Werkzeug                           1.0.0
wheel                              0.34.2
widgetsnbextension                 3.5.1
win-inet-pton                      1.1.0
win-unicode-console                0.5
wincertstore                       0.2
wrapt                              1.11.2
xlrd                               1.2.0
XlsxWriter                         1.2.7
xlwings                            0.17.1
xlwt                               1.3.0
xmltodict                          0.12.0
yapf                               0.28.0
zict                               1.0.0
zipp                               3.15.0
Note: you may need to restart the kernel to use updated packages.
```

1 Question 1 [20 marks]: Segmentation

2 A)

[2]:
```
pip install opencv-python
```

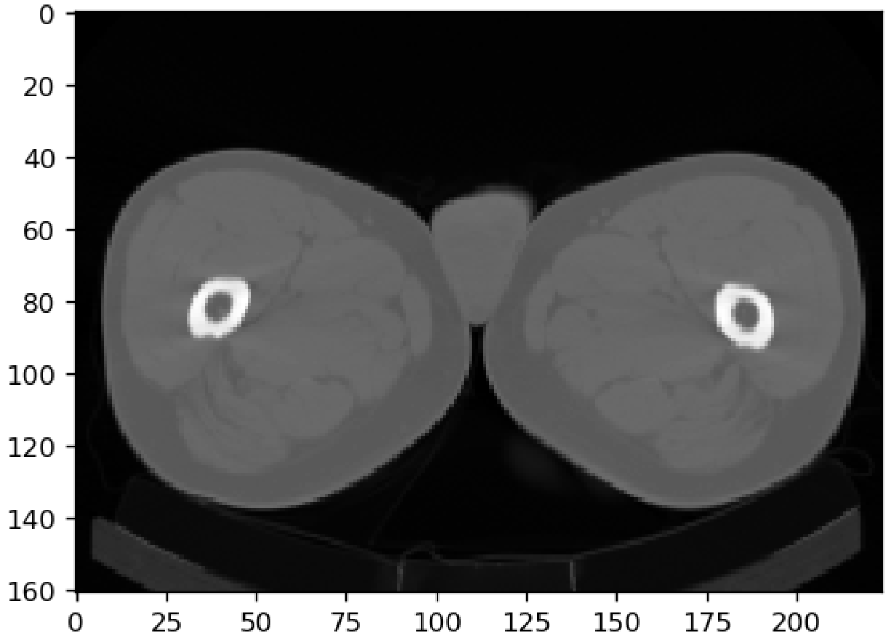
```
Requirement already satisfied: opencv-python in
c:\users\starl\anaconda3\lib\site-packages (4.7.0.72)
Requirement already satisfied: numpy>=1.17.0; python_version >= "3.7" in
c:\users\starl\anaconda3\lib\site-packages (from opencv-python) (1.18.1)
Note: you may need to restart the kernel to use updated packages.
```

[3]:
```
# Import necessary packages
from __future__ import division
import cv2, numpy as np
from matplotlib import pyplot as plt, cm


## Install dicom reader package if necessary

import sys

!{sys.executable} -m pip install pydicom

import pydicom as DCM
import matplotlib.pyplot as plt

# read image file into program
dicom_image_file_0 = DCM.read_file('CTimage.dcm')
```

```
Requirement already satisfied: pydicom in c:\users\starl\anaconda3\lib\site-
packages (2.3.1)
```

[4]:
```
# CT image
I_ct = dicom_image_file_0.pixel_array

#Display the MR Image
plt.figure(dpi=120)
plt.imshow(I_ct,'gray')
```

[4]:
```
<matplotlib.image.AxesImage at 0x248d5d7f288>
```



[5]:
```
########   For future Use   ########
I_ct_n = cv2.normalize(I_ct,I_ct,0,255,cv2.NORM_MINMAX) # rescale dynamic range:
 ↪ 0-255
I_ct_he = cv2.equalizeHist(cv2.convertScaleAbs(I_ct_n)) # int16 -> uint8
hist_obj = cv2.createCLAHE(clipLimit=2, tileGridSize=(18,18))
I_ct_hea = hist_obj.apply(cv2.convertScaleAbs(I_ct_n))

#noise reduction
I_ct_hea_avg = cv2.blur(I_ct_hea,(5,5))
I_ct_hea_med = cv2.medianBlur(I_ct_hea,5)
I_ct_hea_nlm = cv2.fastNlMeansDenoising(I_ct_hea,2,7,35)
I_ct_hea_test = cv2.fastNlMeansDenoising(I_ct_hea,100,1,80)

#erosion
kernel = np.ones((5,5),np.uint8)
I_ct_hea_nlm_erosion = cv2.erode(I_ct_hea_nlm, kernel,iterations=1)

#dilation
I_ct_hea_nlm_dilation = cv2.dilate(I_ct_hea_nlm, kernel,iterations=1)


#opening
opening = cv2.morphologyEx(I_ct_hea_nlm, cv2.MORPH_OPEN, kernel)

#closing
```

```
closing = cv2.morphologyEx(I_ct_hea_nlm, cv2.MORPH_CLOSE, kernel)

#gradient
gradient = cv2.morphologyEx(I_ct_hea_nlm, cv2.MORPH_GRADIENT, kernel)

#tophat
tophat = cv2.morphologyEx(I_ct_hea_nlm, cv2.MORPH_TOPHAT, kernel)

#blackhat
blackhat = cv2.morphologyEx(I_ct_hea_nlm, cv2.MORPH_BLACKHAT, kernel)
```

3 B)

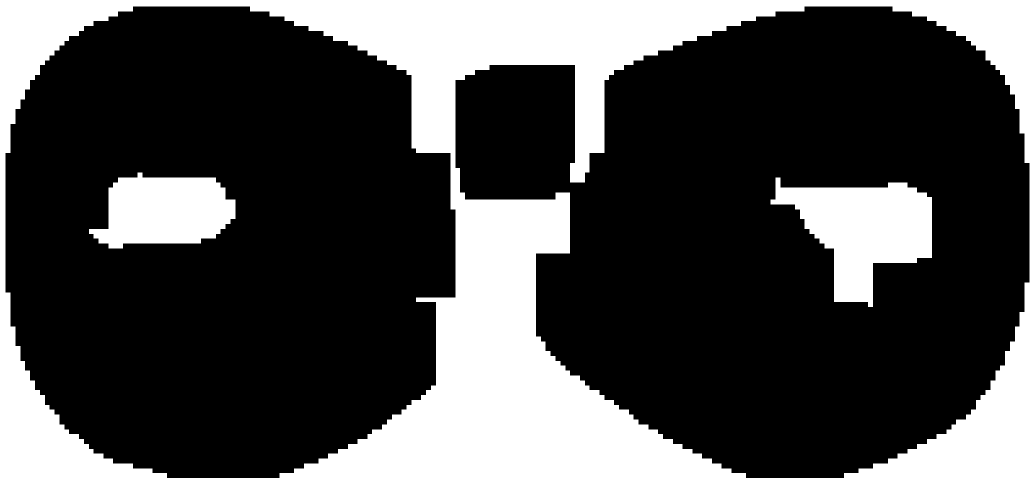
[6]:

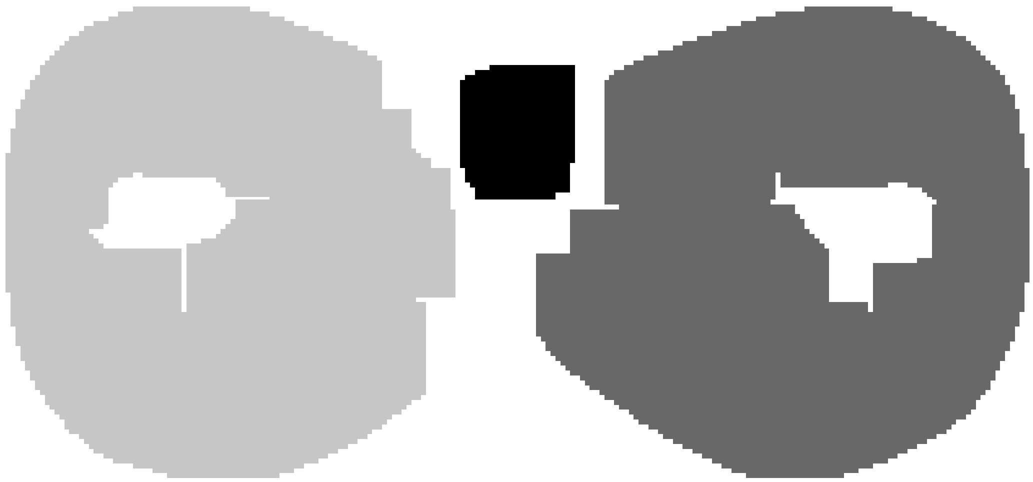
```
# Function used to show relevant images in graw scale
# This function is taken from ipython-books (Refferenced at the end)
def show(img, cmap=None):
    cmap = cmap or plt.cm.gray
    fig, ax = plt.subplots(1, 1, figsize=(8, 6))
    ax.imshow(img, cmap=cmap)
    ax.set_axis_off()
    plt.show()
```

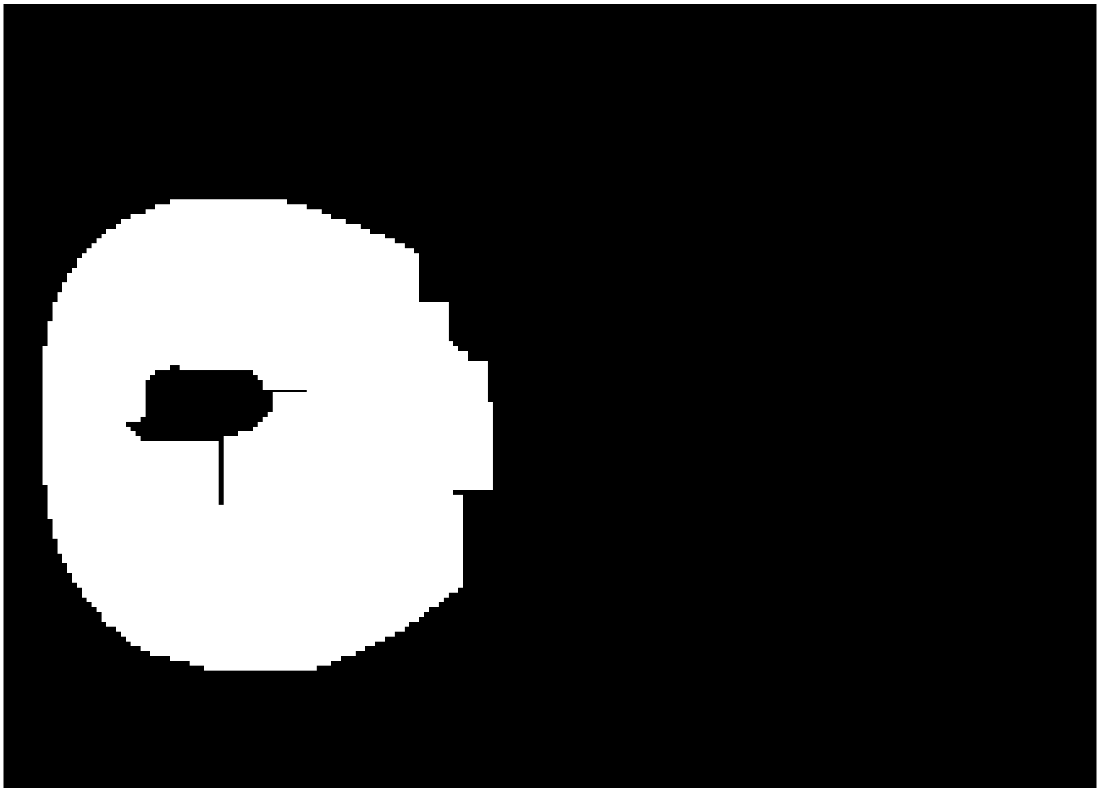
[7]:

```
from skimage.feature import canny
from skimage import data
import matplotlib.pyplot as plt
import numpy as np
from skimage.filters import threshold_otsu
from skimage.segmentation import clear_border
from skimage.morphology import label, closing, square
from skimage.measure import label, regionprops
import trimesh
import pydicom as dicom

%matplotlib inline


#Fething the Image
leg_segment = I_ct

# Image enhancement (adjust the image through erosion and dilation)
I_ct_ntt = cv2.normalize(leg_segment, leg_segment,0,255,cv2.NORM_MINMAX) #␣
 ↪rescale dynamic range: 0-255
I_ct_heatt = hist_obj.apply(cv2.convertScaleAbs(I_ct_ntt))
I_ct_hea_nlmtt = cv2.fastNlMeansDenoising(I_ct_heatt,2,5,90)
I_ct_hea_nlm_erosiontt = cv2.erode(I_ct_hea_nlmtt, kernel,iterations=4)
```

```
I_ct_hea_nlm_dilationtt = cv2.dilate(I_ct_hea_nlm_erosiontt,␣
↪kernel,iterations=4)

#Displaying the Image Enhancement
show(I_ct_hea_nlm_dilationtt < 99)

#Creating a variable to store the desired appearance
x = I_ct_hea_nlm_dilationtt > 99

#Dividing the Segmented Regions into Individual Parts
labels = label(x)
img_bin = clear_border(labels)
show(img_bin, 'Greys')

#Creating an array of the Segmented Parts
regions = regionprops(labels)
boxes = np.array([label['BoundingBox']
                  for label in regions
                  if label['Area'] > 1])


# Getting the first item in the Segmented Parts array
first_box = boxes[0]

# Creating a binary mask with the same shape as the labeled regions
mask = np.zeros((labels.shape[0], labels.shape[1], 3), dtype=np.uint8)

#Storing portion of mask for future calculations
l = labels == 1

# Making the mask white
mask[labels == 1] = (255,255,255)

# Setting all other pixels as black
mask[mask != 255] = 0

#Displaying The Isolated leg on the left in white
show(mask)



left_leg_segment = mask
```



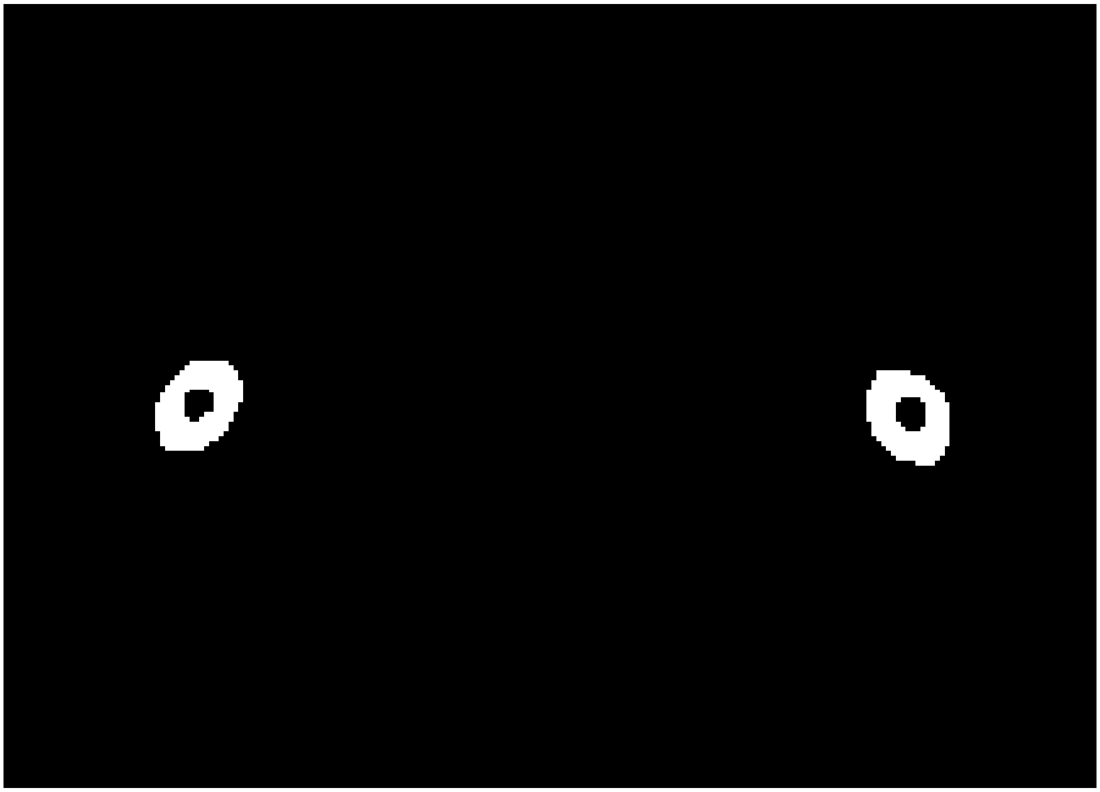


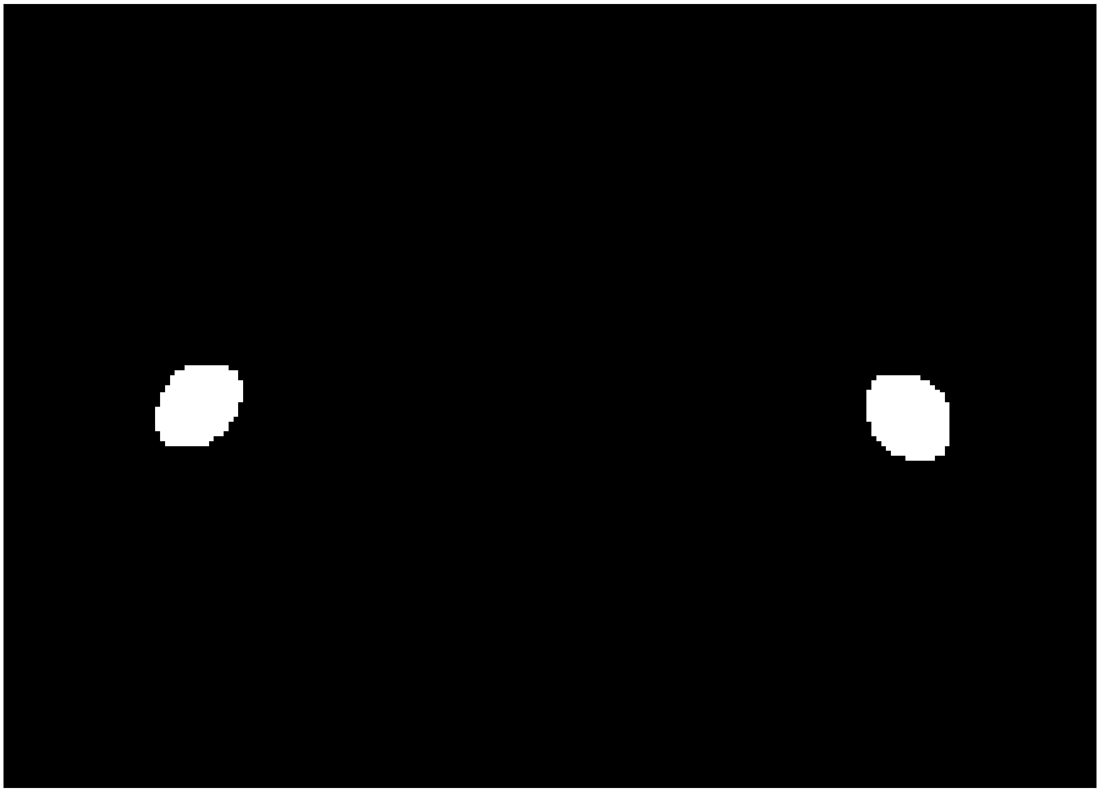


4 C)

[8]:
```
from skimage.feature import canny
from skimage import data
import matplotlib.pyplot as plt
import numpy as np
from skimage.filters import threshold_otsu
from skimage.segmentation import clear_border
from skimage.morphology import label, closing, square
%matplotlib inline

#Fetching Original Image
bone = I_ct

#Displaying Bone after Threshold is applied
threshold_otsu(bone)
show(bone > 115)

#Finding the Edges and Filling in the Holes
edges = canny(bone/255.)
```

```python
from scipy import ndimage as ndi
fill_bone = ndi.binary_fill_holes(edges)

#Processing the Image and clearing the borders
img_bone = clear_border(closing(fill_bone > 0, square(3)))
show(img_bone)



#Creating a variable to store the desired appearance
x = img_bone

#Dividing the Segmented Regions into Individual Parts
labels = label(x)
img_bin = clear_border(labels)
show(img_bin, 'Greys')

#Creating an array of the Segmented Parts
regions = regionprops(labels)
boxes = np.array([label['BoundingBox']
                  for label in regions
                  if label['Area'] > 1])


# Getting the first item in the Segmented Parts array
first_box = boxes[0]

# Creating a binary mask with the same shape as the labeled regions
mask = np.zeros((labels.shape[0], labels.shape[1], 3), dtype=np.uint8)


#Storing portion of mask for future calculations
b = labels == 1

# Making the mask yellow
mask[labels == 1] = (220, 200, 1)

# Setting all other pixels as black
mask[mask != (220, 200, 1)] = 0

#Displaying The Isolated leg on the left in white
show(mask)

left_bone_segment = mask
```





5 D)

[9]:
```
# Apply threshold to obtain binary mask
segmented_bone_mask = b

# Calculate the Area of the segmented region
segmented_bone_area = np.sum(segmented_bone_mask)

# Print the calculated Area
print("Segmented Left Bone area:", segmented_bone_area)


# Apply threshold to obtain binary mask of Leg Area
segmented_leg_mask = l

# Calculate the Total Area
segmented_leg_area = np.sum(segmented_leg_mask)

# Print the calculated Total Area
print("Total Left Leg area:", segmented_leg_area)
```

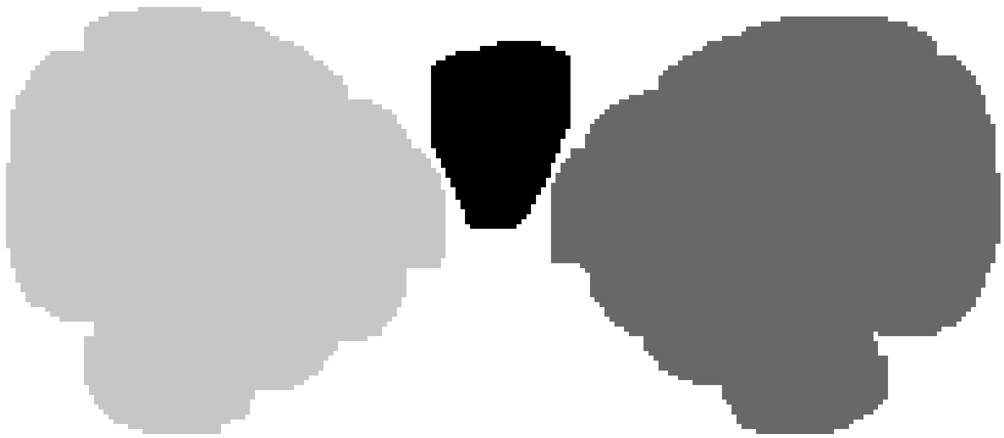
```python
#Percentage of Leg Area
percentage = round((segmented_bone_area/segmented_leg_area)*100, 2)
print(f"The left bone area is {percentage} % of the left leg area")
```

```
Segmented Left Bone area: 246
Total Left Leg area: 6944
The left bone area is 3.54 % of the left leg area
```

6 E)

[10]:
```python
#Fething the Image
muscle_segment = I_ct

# Image enhancement (adjust the image through erosion and dilation)
I_ct_ntt = cv2.normalize(muscle_segment, muscle_segment,0,255,cv2.NORM_MINMAX)␣
↪# rescale dynamic range: 0-255
I_ct_heatt = hist_obj.apply(cv2.convertScaleAbs(I_ct_ntt))
I_ct_hea_nlmtt = cv2.fastNlMeansDenoising(I_ct_heatt,2,7,35)
I_ct_hea_nlm_erosiontt = cv2.erode(I_ct_ntt, kernel,iterations=1)
I_ct_hea_nlm_dilationtt = cv2.dilate(I_ct_hea_nlm_erosiontt,␣
↪kernel,iterations=2)

#Displaying the Image Enhancement
show(I_ct_hea_nlm_dilationtt <94)

#Creating a variable to store the desired appearance
x = I_ct_hea_nlm_dilationtt > 94

#Dividing the Segmented Regions into Individual Parts
labels = label(x)
img_bin = clear_border(labels)
show(img_bin, 'Greys')

#Creating an array of the Segmented Parts
regions = regionprops(labels)
boxes = np.array([label['BoundingBox']
                  for label in regions
                  if label['Area'] > 1])


# Getting the first item in the Segmented Parts array
first_box = boxes[0]

# Creating a binary mask with the same shape as the labeled regions
```

```
mask = np.zeros((labels.shape[0], labels.shape[1], 3), dtype=np.uint8)


#Storing portion of mask for future calculations
m = labels == 1

# Making the mask red
mask[labels == 1] = (170,0,0)

# Setting all other pixels as black
mask[mask != (170,0,0)] = 0

#Displaying The Isolated Muscle on the left in red
show(mask)



left_muscle_segment = mask
```



7 F)

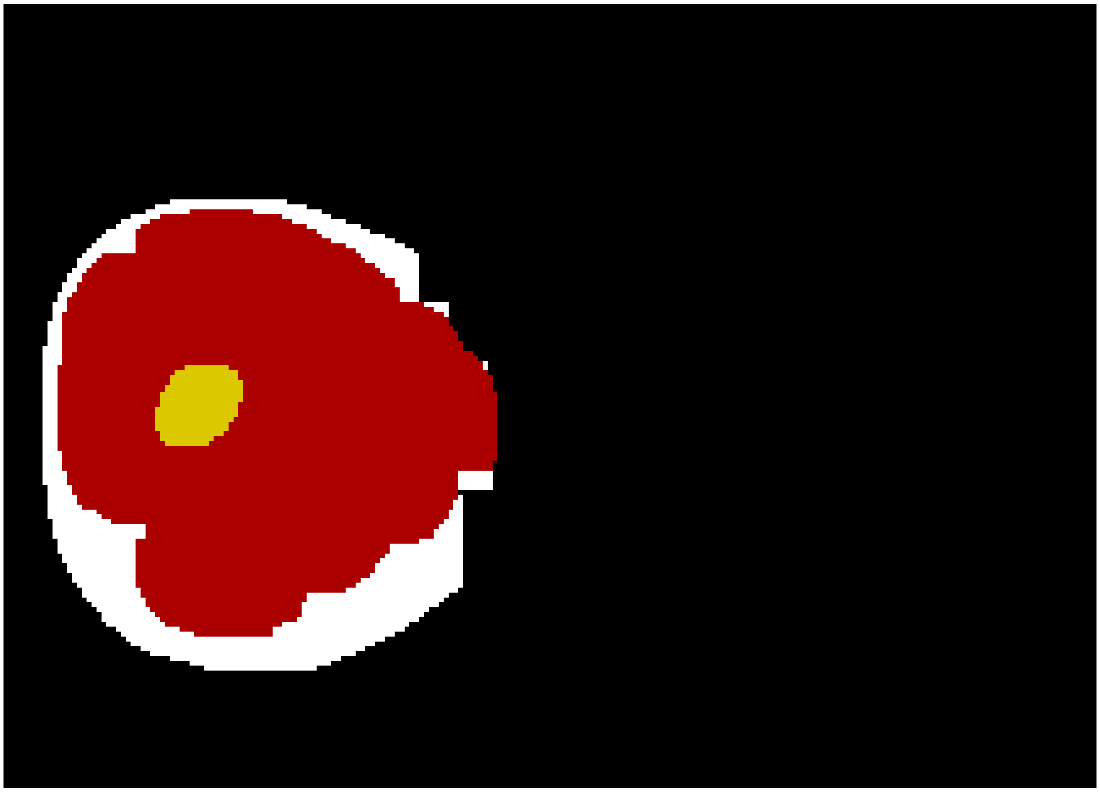
[11]:
```
# Apply threshold to obtain binary mask
segmented_muscle_mask = m

# Calculate the Area of the segmented region
segmented_muscle_area = np.sum(segmented_muscle_mask)

# Print the calculated Area
print("Segmented Left Muscle area:", segmented_muscle_area)




# Apply threshold to obtain binary mask of Leg Area
segmented_leg_mask = l

# Calculate the Total Area
segmented_leg_area = np.sum(segmented_leg_mask)
```

```python
# Print the calculated Total Area
print("Total Left Leg area:", segmented_leg_area)


#Percentage of Leg Area
percentage = round((segmented_muscle_area/segmented_leg_area)*100, 2)
print(f"The left muscle area is {percentage} % of the left leg area")
```

```
Segmented Left Muscle area: 5739
Total Left Leg area: 6944
The left muscle area is 82.65 % of the left leg area
```

8 Segemented Image

[12]:
```python
import numpy as np
import matplotlib.pyplot as plt
from skimage.measure import label, regionprops


# Create an empty array for the overlay
overlay = np.zeros((labels.shape[0], labels.shape[1], 3), dtype=np.uint8)

# Overlay the leg mask
mask1 = l
overlay[mask1] = (255, 255, 255)

# Overlay the muscle
mask2 = m
overlay[mask2] = (170,0,0)

# Overlay the bone mask
mask3 = b
overlay[mask3] = (220, 200, 1)

# Display the overlay
show(overlay)
```



9 Question 2 [10 marks]: Import surfaces

[13]:
```
pip install trimesh
```

```
Requirement already satisfied: trimesh in c:\users\starl\anaconda3\lib\site-
packages (3.21.6)
Requirement already satisfied: numpy in c:\users\starl\anaconda3\lib\site-
packages (from trimesh) (1.18.1)
Note: you may need to restart the kernel to use updated packages.
```

10 A)

[14]:
```
import trimesh

# Load the surface meshes
femur1 = 'femur1.stl'
mesh1 = trimesh.load_mesh(femur1)
mesh1.show()
```

[14]: <IPython.core.display.HTML object>

[15]:
```
# Load the surface meshes
femur2 = 'femur2.stl'
mesh2 = trimesh.load_mesh(femur2)
mesh2.show()
```

[15]: <IPython.core.display.HTML object>

11 B)

In order to compare the shapes of the two femurs the requirements would be to find the correspondence between corresponding points or regions on the surfaces. Correspondence means establishing a mapping or alignment between corresponding points, edges, or regions on different surfaces. This involves image registration, where similar attributes in the two images line up perfectly to identify corresponding landmarks or regions on the surfaces. By establishing correspondence, we can then analyze the differences and similarities between the shapes of the two femurs.

12 C)

12.0.1 i)

[16]:
```
# Finding the number of vertices on each femur
vertices_femur1 = mesh1.vertices.shape[0]
vertices_femur2 = mesh2.vertices.shape[0]

#Printing Result
print(f"Femur 1 has {vertices_femur1} vertices")
print(f"Femur 2 has {vertices_femur2} vertices")
```

```
Femur 1 has 1000 vertices
Femur 2 has 1000 vertices
```

12.0.2 ii)

[17]:
```
# Finding the number of vertices on each femur
faces_femur1 = mesh1.faces.shape[0]
faces_femur2 = mesh2.faces.shape[0]

#Printing Result
print(f"Femur 1 has {faces_femur1} faces")
print(f"Femur 2 has {faces_femur2} faces")
```

```
Femur 1 has 1996 faces
Femur 2 has 1996 faces
```

12.0.3 iii)

If the number of vertices, faces and edges between the 2 femurs are equal, and the connectivity between the surfaces are also equal, we can say the the 2 femurs are Iso-Topological.

[18]:
```
# Counting the number edges for each femur
edges_femur1 = len(mesh1.edges) // 2
edges_femur2 = len(mesh2.edges) // 2

iso_topological = False

# Comparing the counts
if vertices_femur1 == vertices_femur2 and faces_femur1 == faces_femur2 and␣
↪edges_femur1 == edges_femur2:
    # Comparing the connectivity of the surfaces
    for vertex in range(len(mesh1.vertices)):
        neighbors1 = set(mesh1.vertex_neighbors[vertex])
        neighbors2 = set(mesh2.vertex_neighbors[vertex])

        if neighbors1 != neighbors2:
            iso_topological = False
            break
        else:
            iso_topological = True
else:
    iso_topological = False

if iso_topological == True:
    print ("Femur 1 and Femur 2 are Iso-Topological")
else:
    print("Femur 1 and Femur 2 are not Iso-Topological")
```

```
Femur 1 and Femur 2 are Iso-Topological
```

13 Question 3 [20 marks]: Shape matching

13.0.1 A) Direct Method

13.0.2 i)

[19]:
```
#Creating array to hold the average positions
average_positions = []


#Iterating through the femur1 and femur2 vertices to find the average of each
for i in range(len(mesh1.vertices)):
    vertex1 = mesh1.vertices[i]
    vertex2 = mesh2.vertices[i]
```

```python
    average_position = (vertex1 + vertex2) / 2
    average_positions.append(average_position)

# Array of the average positions
average_positions = np.array(average_positions)


# Printing the average positions
for position in average_positions:
    print(position)
```

```
[ 135.29148483  201.92897034 1131.66790771]
[ 139.18535233  199.23165131 1139.9921875 ]
[ 141.71643448  204.19429779 1132.84173584]
[ 131.72693253  191.77107239 1140.70501709]
[ 136.92876434  189.07749939 1146.42553711]
[ 143.47039032  196.79219818 1144.1607666 ]
[ 149.77606964  192.51206207 1147.0748291 ]
[ 150.79366684  198.87026215 1142.10424805]
[ 142.62000656  187.70692444 1148.70819092]
[ 148.96775436  187.04063416 1149.10510254]
[ 157.40297699  185.2342453  1147.25244141]
[ 158.93347931  191.40287018 1144.28192139]
[ 156.415802    196.00708008 1142.66394043]
[ 149.53856659  203.40113068 1136.1786499 ]
[ 144.91290283  201.51969147 1139.45281982]
[ 156.30406952  200.35668182 1137.64477539]
[ 161.88209915  197.62207031 1136.01495361]
[ 158.0819931   202.03447723 1132.20916748]
[ 165.52928543  192.25504303 1137.65557861]
[ 164.75840378  186.29777527 1142.00915527]
[ 165.6787262   196.57714081 1130.80291748]
[ 169.27865601  189.63582611 1133.40930176]
[ 168.46751785  180.63088989 1138.2442627 ]
[ 170.39466095  184.10462952 1133.46948242]
[ 172.34200287  181.53400421 1128.06634521]
[ 143.01004028  205.49716187 1129.14593506]
[ 128.61538315  192.67599487 1134.72503662]
[ 128.01547623  195.78668213 1125.30255127]
[ 135.34986496  202.52082062 1122.05578613]
[ 129.0580368   196.65929413 1120.30804443]
[ 133.23275757  198.51348114 1115.25109863]
[ 133.44950867  195.65006256 1111.05310059]
[ 127.62747955  187.46440887 1112.55126953]
[ 150.93768311  205.18654633 1123.19348145]
[ 156.58013535  203.5477829  1127.05426025]
[ 158.14173508  201.1444397  1116.27331543]
```

```
[ 150.64350128  205.24967957 1129.39056396]
[ 142.1414299   205.09474182 1121.15002441]
[ 148.08415222  204.02180481 1116.62573242]
[ 140.49691772  203.11643982 1115.96594238]
[ 150.37567139  201.31150818 1112.36486816]
[ 162.81422806  199.93721771 1124.49200439]
[ 171.43768692  189.8817215  1122.79431152]
[ 168.97583008  192.19433594 1117.39538574]
[ 160.58551407  199.67058563 1116.91992188]
[ 163.29859161  194.56475067 1113.02722168]
[ 173.19528198  186.36820984 1113.41351318]
[ 165.61315155  191.13243103 1108.61889648]
[ 171.17079544  187.60597992 1108.78100586]
[ 143.24571228  201.03710175 1111.88360596]
[ 140.70074463  194.81642914 1107.73901367]
[ 148.37866592  196.96315765 1108.17871094]
[ 152.03536224  190.84962463 1104.11773682]
[ 133.56797409  190.3349762  1107.98510742]
[ 141.06111145  185.58670044 1104.6270752 ]
[ 132.11168671  185.93548584 1107.79858398]
[ 145.78430557  189.06475067 1103.98242188]
[ 148.31116104  186.25163269 1100.72930908]
[ 140.43366241  180.68615723 1103.76141357]
[ 147.03698349  181.75478363 1096.41113281]
[ 155.27521133  195.9845047  1108.91571045]
[ 156.04324722  189.20785522 1102.68780518]
[ 164.19318008  187.77449799 1103.25537109]
[ 152.00082397  185.85277557 1098.09210205]
[ 158.204216    186.18669891 1097.36987305]
[ 153.78809738  184.03675079 1092.09429932]
[ 172.07041931  186.28326416 1105.1383667 ]
[ 161.22383118  184.40826416 1092.07250977]
[ 150.85789108  179.63842773 1082.8883667 ]
[ 155.2687645   183.11257172 1082.66369629]
[ 159.40541077  183.37223816 1077.55895996]
[ 167.39088058  183.2648468  1092.80358887]
[ 152.50120544  181.49168396 1073.04986572]
[ 148.6878624   176.65296936 1073.0425415 ]
[ 150.35102081  180.57460785 1068.38848877]
[ 152.88166428  182.09085846 1064.0526123 ]
[ 156.17086411  182.68041992 1062.18988037]
[ 158.23742294  182.35295105 1058.27142334]
[ 149.46226501  179.1827774  1062.37652588]
[ 149.52835464  180.91107178 1051.8480835 ]
[ 152.94241714  182.72396088 1052.43145752]
[ 146.66231918  176.35383606 1057.00195312]
[ 145.64957809  175.59397888 1050.38415527]
[ 147.54018021  182.85336304 1039.00732422]
```

```
[ 149.60968018  185.09661865 1035.51556396]
[ 141.18680954  176.7950058  1031.47070312]
[ 142.95576859  180.14800262 1032.04223633]
[ 143.39858627  184.93618774 1023.00891113]
[ 145.79123688  188.7226181  1015.18447876]
[ 138.92017365  189.54076385 1001.27737427]
[ 135.16374969  181.39331055 1008.16043091]
[134.87233353 186.61878967 999.33087158]
[ 139.11554337  174.15866089 1025.76959229]
[ 134.38715363  178.56787872 1007.68237305]
[130.48715973 184.40630341 991.7515564 ]
[140.51066971 193.0013504  991.98492432]
[128.20372391 179.06629181 986.75802612]
[ 133.60377502  175.17539978 1004.41207886]
[126.08130646 175.25585175 976.86447144]
[131.70925903 194.42273712 974.90643311]
[134.80340958 195.65309143 973.35287476]
[129.25802612 192.37895203 975.75601196]
[126.9563446  182.13434601 983.05291748]
[122.55203247 189.42308044 965.94207764]
[123.8220253  195.94818878 955.56488037]
[127.58356094 197.88249207 950.69857788]
[118.75312042 186.3443985  959.24121094]
[117.45651627 182.9899826  955.82992554]
[119.74984741 177.6581955  959.11004639]
[115.57736969 192.3431778  946.00231934]
[116.12133408 197.62731171 932.69613647]
[115.84931564 196.29779816 938.03665161]
[109.12235641 188.20189667 935.61935425]
[108.14590073 183.44927979 932.97161865]
[112.62189865 177.93675232 939.87948608]
[118.57963943 198.80662537 925.95196533]
[107.19081497 192.08527374 927.61688232]
[100.76260757 195.78433228 906.50033569]
[103.52410126 197.74386597 906.10952759]
[ 93.33386993 185.43644714 898.14068604]
[ 174.24889374  183.50678253 1115.7144165 ]
[ 173.15182495  181.72958374 1120.92602539]
[ 175.16532516  177.79742432 1117.2779541 ]
[ 172.21636963  177.56601715 1121.74621582]
[ 178.91809464  180.41057587 1111.8192749 ]
[ 176.89981079  184.25166321 1107.94610596]
[ 180.02462006  174.89414215 1113.75195312]
[ 185.17195892  177.92144775 1108.59429932]
[ 184.2026825   172.95910645 1112.43457031]
[ 192.53129578  174.44057465 1111.34466553]
[ 193.09431458  179.44221497 1105.15765381]
[ 195.41719055  172.95262909 1111.22796631]
```

```
[ 196.96224213  172.97125244 1105.96746826]
[ 184.6272049   181.30550385 1104.88555908]
[ 195.26539612  177.11315918 1102.95977783]
[ 200.70929718  168.18282318 1105.09448242]
[ 198.05732727  170.45854187 1094.78930664]
[ 200.71829987  168.0150528  1096.54516602]
[ 179.75359344  183.97971344 1102.49786377]
[ 188.17156219  181.40246582 1102.92205811]
[ 185.40586853  183.37724304 1097.84033203]
[ 181.41413879  184.57434082 1096.86395264]
[ 175.18166351  184.73033142 1101.39910889]
[ 175.68440247  184.20896149 1094.45019531]
[ 178.31051636  182.32254028 1089.48577881]
[ 185.2677002   176.75626373 1087.49609375]
[ 191.36483002  179.89404297 1096.27844238]
[ 195.70494843  176.54167938 1095.19775391]
[ 202.51023102  163.16921997 1087.33392334]
[ 187.93943024  177.59940338 1091.18884277]
[ 194.21583557  175.50926208 1089.10186768]
[ 191.68870544  174.75344086 1083.34790039]
[ 196.0273056   173.7252121  1087.40753174]
[ 197.84723663  170.23698425 1084.93914795]
[ 172.01984787  182.76952362 1087.4821167 ]
[ 166.10660172  182.37558746 1079.29510498]
[ 179.81689453  177.41121674 1080.44506836]
[ 174.49622345  178.15935516 1076.48596191]
[ 167.75941849  179.45423126 1067.48272705]
[ 184.38710785  174.84580994 1077.17370605]
[ 196.74281311  170.6075058  1079.19720459]
[ 200.23173523  166.90705872 1080.97192383]
[ 192.51158142  171.78997803 1073.8102417 ]
[ 186.12245941  172.58061981 1070.03607178]
[ 199.49793243  165.23490143 1076.16021729]
[ 188.89790344  168.80522156 1068.28381348]
[ 162.71760178  181.36128998 1062.79852295]
[ 173.97745514  177.66822052 1063.22894287]
[ 164.69298553  181.82012939 1042.90533447]
[ 177.06203461  175.15550232 1056.01239014]
[ 182.13460541  173.72679901 1066.27783203]
[ 181.84963226  171.56621552 1059.3605957 ]
[ 176.79641724  171.95650482 1047.36187744]
[ 155.70667648  184.87049103 1040.18218994]
[ 167.76659393  179.98856354 1037.99780273]
[ 156.31559372  186.30570984 1029.11715698]
[ 168.85762024  178.03604889 1031.55273438]
[ 168.76216888  175.12792969 1023.74740601]
[ 159.75452423  185.19667053 1020.43850708]
[ 163.06890106  182.30879974 1018.16983032]
```

```
[ 169.63291168  169.81598663 1022.15447998]
[ 150.01644516  189.91074371 1010.35748291]
[ 154.19464874  188.81974792 1008.67715454]
[ 157.83433533  186.49556732 1005.31185913]
[157.02256012 185.99877167 994.9440918 ]
[ 163.97093201  174.19065094 1004.33361816]
[ 166.1145401   169.49589539 1010.89492798]
[ 162.97210693  179.55291748 1006.42565918]
[160.26073456 172.03772736 993.30606079]
[143.831604   193.95140839 986.20358276]
[149.58747482 191.7071228  990.63912964]
[150.7227478  190.50845337 983.44351196]
[158.98986816 178.67879486 989.38528442]
[154.76731873 185.10186005 982.40710449]
[157.25327301 175.51019287 983.22659302]
[144.04588699 193.7980957  971.85552979]
[138.17668915 195.94625092 968.53793335]
[147.77989197 189.85459137 966.78094482]
[152.11643982 183.87081909 971.16366577]
[153.96053314 172.10488892 977.58352661]
[158.77318573 168.49238586 992.89016724]
[150.99891663 178.99064636 965.11358643]
[150.15541077 176.48627472 964.31411743]
[145.27902985 172.78086853 959.14355469]
[133.70424271 196.22794342 944.20422363]
[138.18465805 193.69179535 946.40588379]
[132.01237106 197.61586761 951.03317261]
[139.05276489 191.02075958 941.46310425]
[144.97036743 187.61615753 952.66659546]
[144.87201691 184.3380661  949.08959961]
[143.75053406 180.48468781 946.13418579]
[143.13716125 176.88326263 948.24765015]
[122.28354263 198.41474152 925.90802002]
[129.20454407 193.40971375 919.41488647]
[131.9484024  189.52934265 919.77670288]
[130.5459137  175.70787048 923.98477173]
[138.08795166 169.29964447 954.77035522]
[132.21318817 177.99338531 923.34936523]
[126.57033157 173.45814514 921.25338745]
[141.65759277 182.5471344  940.46975708]
[124.00806808 196.79327393 917.54571533]
[128.98480225 181.20539093 911.84481812]
[129.71408081 185.61857605 911.9161377 ]
[ 131.11397934  188.41210938 1142.03265381]
[ 131.99779892  181.31507874 1144.95635986]
[ 137.68723679  179.32421875 1148.33276367]
[ 128.92503738  177.70597839 1141.02801514]
[ 129.90699768  173.61581421 1141.42541504]
```

```
[ 134.10534668  174.21839142 1145.26916504]
[ 146.83811951  181.26109314 1149.99121094]
[ 155.61254501  179.82694244 1148.34484863]
[ 141.83329391  173.2780304  1148.08898926]
[ 150.11211777  175.04340363 1148.90020752]
[ 156.91305542  173.40868378 1146.72119141]
[ 145.95325851  168.8259201  1146.53143311]
[ 149.87245941  166.02692413 1144.82672119]
[ 162.58775711  177.33287811 1144.51708984]
[ 166.0134201   175.54537964 1140.69824219]
[ 159.28009415  169.89199066 1144.00738525]
[ 156.31465912  163.69708252 1140.91094971]
[ 168.90953827  175.55626678 1135.6572876 ]
[ 163.51032257  167.51516724 1138.17211914]
[ 164.1191597   164.73297119 1129.12609863]
[ 169.0184288   172.57498169 1131.59619141]
[ 159.89908981  162.36937714 1135.09197998]
[ 139.76268387  167.09850311 1144.13739014]
[ 133.25041962  167.04704285 1139.93188477]
[ 136.05263901  163.05448914 1137.94360352]
[ 142.27633667  161.61373901 1139.746521  ]
[ 149.74156952  159.93126678 1138.25836182]
[ 145.78149033  157.87284088 1134.50134277]
[ 156.18999481  159.54392242 1133.9432373 ]
[ 128.1444931   182.62423706 1135.84649658]
[ 125.59870911  188.67719269 1131.68444824]
[ 127.05646896  174.11366272 1137.7600708 ]
[ 124.16425323  182.02610016 1131.53271484]
[ 124.14053726  175.14161682 1131.49151611]
[ 124.52480316  188.88012695 1123.21954346]
[ 123.14038467  182.06135559 1127.44677734]
[ 123.88243866  180.59806824 1118.00689697]
[ 124.94828415  186.71108246 1117.47595215]
[ 123.24491882  177.03270721 1123.84454346]
[ 125.17395782  170.7898407  1122.53771973]
[ 127.36292267  170.27864075 1135.51831055]
[ 126.45664215  167.89903259 1127.61340332]
[ 127.8797226   178.67198181 1111.15228271]
[ 124.94929123  173.94312286 1118.3293457 ]
[ 130.50085449  163.56890106 1130.04406738]
[ 134.94683075  159.8256073  1128.57287598]
[ 139.84386444  158.19190216 1131.73352051]
[ 142.98042297  156.49274445 1128.01043701]
[ 128.92069626  166.53111267 1118.37530518]
[ 132.46600723  162.10437012 1121.08630371]
[ 139.79382324  157.87554169 1120.88934326]
[ 147.61949539  156.17710114 1123.77642822]
[ 151.25653839  156.7445755  1130.12139893]
```

```
[ 159.43021011  160.12606049 1122.43365479]
[ 154.02467728  158.09586334 1118.51300049]
[ 148.02380371  158.39606476 1116.07818604]
[ 163.10095596  160.51358032 1113.22442627]
[ 163.15441513  164.38678741 1121.64013672]
[ 159.12182236  159.03136444 1111.20446777]
[ 129.41365814  170.46881104 1113.18707275]
[ 136.95689774  171.31965637 1111.06860352]
[ 133.7322731   165.36164093 1113.74664307]
[ 139.61369705  170.26564026 1109.95239258]
[ 139.61749268  167.79111481 1112.13153076]
[ 136.19518661  182.24751282 1107.0022583 ]
[ 131.17421341  179.25859833 1109.03405762]
[ 136.77253723  177.21582794 1108.37121582]
[ 140.53837204  173.40646362 1105.92767334]
[ 143.01307297  176.98790741 1099.29449463]
[ 144.09823227  171.81482697 1096.14508057]
[ 142.96667099  168.75164795 1106.00927734]
[ 147.87084198  162.26293945 1109.05804443]
[ 145.30177307  167.45310974 1100.402771  ]
[ 136.90388489  160.57003784 1116.87890625]
[ 142.97675705  162.09333038 1114.17877197]
[ 153.99563217  159.19723511 1112.24188232]
[ 153.03158188  160.84671783 1103.75274658]
[ 157.83023834  158.8532486  1101.94647217]
[ 149.87388229  163.1255722  1098.97180176]
[ 158.74450302  157.40355682 1097.597229  ]
[ 146.06158829  176.08651733 1088.9765625 ]
[ 145.26445389  167.84683228 1088.17095947]
[ 148.12701035  163.8168869  1092.93798828]
[ 144.79722214  169.63388062 1081.57885742]
[ 146.31427383  172.77025604 1078.41326904]
[ 145.3351326   163.81932068 1081.70214844]
[ 149.40735626  160.64480591 1086.8973999 ]
[ 147.45118332  159.50099945 1082.23297119]
[ 143.5696373   166.43843842 1079.24871826]
[ 154.82148743  155.35398865 1086.48278809]
[ 163.43534851  153.35552216 1095.16552734]
[ 150.90824127  150.6699295  1078.3973999 ]
[ 147.79951859  153.66883087 1078.51452637]
[ 144.79330826  169.27806091 1071.78833008]
[ 141.03489304  159.28533173 1077.69903564]
[ 139.66804886  162.19540405 1075.12658691]
[ 147.88126755  173.0118103  1065.09863281]
[ 144.1493187   168.0458374  1065.10290527]
[ 143.99360275  167.3183136  1057.52947998]
[ 136.97550583  163.85671997 1069.82623291]
[ 141.75094604  167.78153229 1068.10144043]
```

```
[ 140.7069664   166.71304321 1061.32214355]
[ 139.03902054  154.46115112 1075.3125    ]
[ 136.7227478   158.17790222 1073.25744629]
[ 140.0484581   151.12330627 1072.5055542 ]
[ 149.6757431   147.70608521 1071.80493164]
[ 153.01859283  148.58811188 1078.18591309]
[ 135.93676758  153.06272888 1067.33624268]
[ 135.63319016  161.5747757  1068.25787354]
[ 138.73181915  150.23065948 1065.58441162]
[ 152.94130707  147.56837463 1064.06866455]
[ 143.25460434  148.69281769 1063.70306396]
[ 144.2297287   168.46067047 1054.02203369]
[ 144.19066238  172.21484375 1051.15472412]
[ 138.75679779  164.69321442 1061.05273438]
[ 135.67552948  159.33119965 1064.71173096]
[ 141.87612152  163.69019318 1054.60101318]
[ 138.28798294  157.92153168 1057.5824585 ]
[ 142.02830505  171.69221497 1041.69140625]
[ 142.02660751  158.78369141 1052.25787354]
[ 142.75119019  164.19158936 1049.4052124 ]
[ 143.58314133  159.66825104 1046.02874756]
[ 140.70344162  154.55493927 1056.01831055]
[ 150.78983307  151.58633423 1053.58978271]
[ 143.45021057  155.88175964 1052.4831543 ]
[ 145.62986374  155.13839722 1049.3626709 ]
[ 140.34886169  168.28925323 1033.27874756]
[ 142.03757477  166.33095551 1040.46374512]
[ 143.62962341  161.04864502 1036.25476074]
[ 138.78601837  170.5417099  1023.36978149]
[ 141.08219147  163.91425323 1026.06762695]
[ 139.24307251  165.30122375 1017.23025513]
[ 142.05553818  161.21509552 1020.54165649]
[132.97674942 172.35396576 998.90649414]
[ 136.42176437  165.81278229 1002.07244873]
[134.74889374 165.49798584 995.25152588]
[131.79781723 169.54251099 988.11413574]
[131.61289978 165.81700134 983.28140259]
[125.70998383 167.11795807 963.47232056]
[131.00901413 164.99544525 974.91824341]
[123.76145554 167.04217529 958.11148071]
[119.62867737 169.44722748 950.07928467]
[121.73360062 165.87171936 945.34307861]
[114.5985527  169.38141632 937.46298218]
[110.48486328 173.35748291 929.97799683]
[111.64881897 168.12141418 926.83508301]
[105.61556244 172.44442749 918.59967041]
[105.79059601 169.32446289 916.02798462]
[108.80137253 166.47113037 914.50006104]
```

```
[ 171.57142639  178.23129272 1126.85424805]
[ 166.92551041  170.71766663 1125.95526123]
[ 168.61245728  169.76044464 1121.0244751 ]
[ 171.97970963  172.10407257 1119.30145264]
[ 175.48839569  171.44143677 1116.4430542 ]
[ 174.98570633  163.96192932 1112.08251953]
[ 180.66885757  168.79189301 1113.40332031]
[ 187.05420685  169.38685608 1113.62615967]
[ 182.87297058  165.58759308 1112.22296143]
[ 166.34003067  164.33450317 1117.23321533]
[ 179.39347458  162.32057953 1110.13446045]
[ 182.70026398  159.8824234  1111.44958496]
[ 178.34149933  157.32177734 1105.38031006]
[ 181.9421463   156.26063538 1111.68286133]
[ 193.43141937  168.33586121 1113.90063477]
[ 198.97834778  163.72109985 1112.16339111]
[ 193.33850861  163.11967468 1114.75793457]
[ 196.89719391  157.10910797 1114.65081787]
[ 190.31671143  159.95965576 1116.11291504]
[ 193.59384155  157.53700256 1116.60284424]
[ 186.85110474  155.40468597 1116.77282715]
[ 189.31280518  153.167099   1117.97259521]
[ 187.95858002  164.05937958 1114.36804199]
[ 200.80690002  163.95610809 1109.29675293]
[ 202.5691452   162.29116058 1100.440979  ]
[ 184.70218658  157.18138885 1114.03942871]
[ 199.44126892  154.41932678 1109.32666016]
[ 197.09980011  150.46146393 1110.86993408]
[ 178.85308456  148.54803467 1117.73803711]
[ 182.61760712  147.66886139 1120.06195068]
[ 178.49988937  145.19638824 1118.11901855]
[ 181.7267952   143.25866699 1117.82476807]
[ 175.37988663  147.35414886 1113.168396  ]
[ 179.30124664  153.92152405 1106.00115967]
[ 178.4550209   151.58963013 1105.93475342]
[ 175.03175735  144.86410141 1114.80401611]
[ 175.64545441  141.20614624 1113.59576416]
[ 179.90551758  138.90607452 1113.72857666]
[ 191.1181488   150.83646393 1117.23980713]
[ 186.61530304  145.73321152 1118.30175781]
[ 194.32080841  150.51848602 1114.58862305]
[ 192.20557404  142.8232193  1110.39971924]
[ 199.00572205  147.80044556 1099.35845947]
[ 183.78292084  137.77742386 1110.34912109]
[ 193.39811707  141.42448044 1100.76544189]
[ 190.25801849  139.46928406 1104.87792969]
[ 171.0702095   156.50262451 1102.65612793]
[ 174.54113007  159.93211365 1105.62091064]
```

```
[ 174.32159805  155.95613098 1102.53094482]
[ 166.53316116  156.45046997 1101.78289795]
[ 168.76155853  153.06678009 1098.20672607]
[ 176.58268738  150.47280884 1102.27728271]
[ 173.55343246  144.3102684  1105.26696777]
[ 172.06745148  145.4129982  1101.55194092]
[ 169.63048172  146.78990173 1098.47784424]
[ 174.04730225  141.12882614 1104.1963501 ]
[ 176.94685364  138.08127213 1100.34332275]
[ 177.35676575  138.51090622 1110.25750732]
[ 169.10896301  143.51447296 1096.87158203]
[ 165.27699661  150.05107117 1095.20031738]
[ 162.10201263  144.34209061 1087.3034668 ]
[ 165.41543198  140.85902023 1089.11383057]
[ 172.32357025  138.88093567 1094.48944092]
[ 171.89692688  137.32194138 1087.56365967]
[ 167.91317749  138.61829376 1087.83416748]
[ 180.86158752  137.17292404 1098.55908203]
[ 184.58383942  137.97149658 1096.2086792 ]
[ 186.80361176  137.55033493 1102.42462158]
[ 188.63101196  140.095047   1090.91809082]
[ 191.83283234  143.55945587 1083.27142334]
[ 198.61260223  148.51659393 1087.71557617]
[ 195.20386505  143.96158981 1091.34674072]
[ 176.62662506  137.35296249 1084.44934082]
[ 182.87308502  139.06545258 1083.36547852]
[ 201.88428497  153.76351166 1101.12902832]
[ 203.06720734  163.04036713 1092.16088867]
[ 203.65023041  157.60443115 1088.89050293]
[ 201.23592377  152.66512299 1094.34381104]
[ 203.42293549  160.34860992 1081.79119873]
[ 203.04512024  156.99434662 1082.37280273]
[ 200.93540955  151.78932953 1084.50488281]
[ 196.38719177  147.3963623  1081.33068848]
[ 181.1372757   136.98351288 1107.88446045]
[ 195.19107819  164.48271942 1072.49383545]
[ 189.45175171  162.59315491 1066.95068359]
[ 194.40490723  157.3106308  1072.12042236]
[ 187.77752686  158.0195694  1065.26074219]
[ 189.49053955  152.56406403 1069.88623047]
[ 195.85082245  151.39600372 1074.64550781]
[ 161.33098221  150.65601349 1089.26324463]
[ 159.74423218  144.26114655 1082.90777588]
[ 169.35193253  138.70711517 1081.70709229]
[ 154.88724899  146.35522461 1076.00097656]
[ 159.2902832   144.27494049 1074.13244629]
[ 164.47531509  142.53761292 1073.03887939]
[ 169.51602554  142.02241516 1071.94085693]
```

```
[ 186.23297119  142.09939957 1074.75878906]
[ 181.73401642  141.11071396 1075.46813965]
[ 189.75765228  144.95983124 1074.66583252]
[ 184.36971283  146.78858185 1069.84942627]
[ 180.61558533  144.40615463 1069.72766113]
[ 175.18436432  142.81385803 1069.88604736]
[ 184.00021362  166.06713867 1059.45788574]
[ 181.01889038  161.75437927 1053.9744873 ]
[ 182.68759155  153.50309753 1061.52185059]
[ 177.48471832  166.65211487 1045.51635742]
[ 177.51638031  156.1661377  1051.64044189]
[ 172.20524597  149.06162262 1051.58221436]
[ 165.46471786  145.72167206 1059.68383789]
[ 172.96167755  145.08628082 1061.91772461]
[ 175.8059082   147.19581604 1060.43475342]
[ 166.89904785  146.13560867 1054.48876953]
[ 168.30355835  147.34015656 1048.77581787]
[ 200.18733215  159.8213501  1075.3538208 ]
[ 197.28953552  150.75674438 1076.48681641]
[ 174.75294495  164.69487762 1038.07580566]
[ 173.57313538  159.72659302 1039.63574219]
[ 170.47055817  153.5141983  1043.36584473]
[ 169.71367645  156.46590424 1036.45684814]
[ 154.16730881  153.00202179 1041.03063965]
[ 159.34023285  149.27562714 1050.5927124 ]
[ 164.64494324  146.97578049 1047.82928467]
[ 160.68320847  148.6710434  1044.40002441]
[ 147.57767105  156.19570923 1038.66955566]
[ 160.47941589  150.3155365  1037.99353027]
[ 155.81990051  151.50410461 1036.66357422]
[ 164.59312439  151.11637115 1039.41650391]
[ 145.90277481  158.98616028 1027.85913086]
[ 167.77944946  164.98696136 1018.91650391]
[ 163.58094788  157.95906067 1020.70498657]
[ 163.59770203  162.91335297 1011.12866211]
[ 159.34855652  153.2469101  1027.56542969]
[ 149.43784332  156.31511688 1030.26199341]
[ 157.14655685  150.88548279 1033.4621582 ]
[ 154.96012115  152.87460327 1026.18804932]
[ 151.38734436  154.66639709 1020.45083618]
[ 146.48957443  158.76870728 1012.9510498 ]
[ 153.704216    155.95142365 1011.29696655]
[ 160.26034546  159.22144318 1011.60079956]
[ 140.69473648  162.81978607 1011.63110352]
[153.20632172 161.84429932 994.51971436]
[ 151.07309723  155.4487381  1011.67749023]
[ 147.55289459  157.47203064 1005.69036865]
[137.51880264 163.44351196 998.90460205]
```

```
[141.48014069 161.03379822 997.90356445]
[147.55794144 158.59217834 996.45059204]
[135.44365692 163.63582611 988.11047363]
[144.42440033 158.82862854 994.9317627 ]
[140.29327393 160.92438507 984.30984497]
[143.34954453 160.57946014 985.67687988]
[155.33706665 165.128685   991.171875  ]
[148.16246796 168.56744385 972.22741699]
[142.81196976 162.49549866 978.58816528]
[137.70389175 165.34525299 965.37435913]
[134.74282455 162.76196289 971.23458862]
[137.6722374  162.59120941 972.90597534]
[129.42850494 164.30547333 963.50137329]
[130.09788895 164.14513397 959.48428345]
[127.56439972 165.93493652 946.29333496]
[123.46118164 165.44709015 942.09829712]
[113.60256958 166.86528015 915.11306763]
[109.48309708 166.05057526 909.66482544]
[ 96.70619583 192.26052094 903.63925171]
[108.90918732 199.30750275 903.83395386]
[ 97.3341217  188.35649872 907.65979004]
[ 99.38203812 181.41096497 910.97949219]
[ 92.008564   176.26358795 888.41342163]
[ 94.55287933 198.86407471 876.22848511]
[ 88.5761528  197.2554245  874.45916748]
[ 87.20146561 194.75836945 878.72824097]
[ 90.40121078 181.84875488 889.8710022 ]
[ 80.58897209 188.92947388 869.91937256]
[ 79.7304821  195.67858887 858.61105347]
[ 78.10825729 185.94554138 864.23532104]
[ 85.79130173 175.31108856 873.81838989]
[ 76.08533859 192.18479156 857.40484619]
[ 78.37875366 178.47138977 859.83148193]
[ 82.02462387 197.95477295 848.13278198]
[ 76.96701622 196.42742157 847.61251831]
[ 84.23679352 198.50296021 841.80166626]
[ 71.60941315 183.35910797 848.55078125]
[ 66.77808952 189.44345093 838.16427612]
[ 67.59321022 186.48583984 840.9203186 ]
[ 69.45323372 177.48194885 838.9211731 ]
[ 69.70211411 195.36268616 831.92678833]
[ 63.34544373 193.33428192 821.69030762]
[ 77.63197899 197.76999664 816.77331543]
[ 68.33696938 191.80384827 838.8034668 ]
[ 62.06394958 186.76080322 828.2593689 ]
[ 72.25434494 197.10684967 819.73049927]
[ 62.66715813 194.99775696 806.4564209 ]
[ 56.19191265 190.03435516 810.19293213]
```

```
[ 56.59657669 180.68000031 814.37252808]
[ 61.01344681 177.43063354 820.3972168 ]
[ 53.67620659 186.11552429 808.90240479]
[ 50.71938419 176.32131958 798.57089233]
[ 67.69195461 196.15848541 792.32476807]
[ 59.59288406 194.98957062 790.09384155]
[ 52.67229557 192.98153687 786.81921387]
[ 45.94195843 182.82517242 792.22814941]
[ 50.65774918 181.05764008 801.92019653]
[ 66.64433861 170.28063965 820.11166382]
[ 50.0632658  191.02021027 790.55252075]
[ 51.12824345 193.62542725 765.10778809]
[ 62.97166538 194.1038208  768.32345581]
[ 43.40774381 185.48236084 784.34240723]
[ 40.04293743 181.68535614 780.61819458]
[ 42.46343386 175.30174255 783.8996582 ]
[ 38.68847096 176.82654572 778.58258057]
[ 44.09666419 171.33957672 783.16412354]
[ 41.27530849 189.73097992 769.17630005]
[ 44.46763103 191.90799713 765.28598022]
[ 55.14947414 192.05604553 752.51278687]
[ 46.52701402 192.69448853 750.05944824]
[ 39.1415472  187.20553589 771.70831299]
[ 30.78597641 180.84233093 765.19378662]
[ 32.93816185 178.79473114 770.28884888]
[ 31.26996565 172.12274933 769.62207031]
[ 33.24203968 194.12561798 743.73828125]
[ 28.60364056 189.30213928 751.25601196]
[ 34.18127203 188.1913681  760.31274414]
[ 41.57214355 192.84217072 753.85192871]
[ 40.44572687 193.4074707  748.61846924]
[ 48.03808105 191.44317627 742.09887695]
[ 22.52106762 180.7984848  756.35778809]
[ 25.02807903 174.17160797 762.90130615]
[ 22.2898674  187.06807709 748.51989746]
[ 18.32178688 174.14661407 757.78674316]
[  9.38621521 180.00022125 746.08184814]
[  7.4329052  174.87001038 750.85595703]
[ 16.86597633 179.85503387 752.70388794]
[ 18.03721428 187.17684937 741.84002686]
[ 13.91053581 186.03035736 735.40539551]
[  5.59982491 178.32744598 736.89355469]
[  2.12681007 174.59490204 738.50097656]
[ 10.78956604 182.45593262 721.5043335 ]
[3.01368713e-01 1.71087196e+02 7.32808197e+02]
[  3.65624428 172.74987793 721.56332397]
[ 26.90328884 193.27987671 741.61584473]
[ 30.90189743 196.06130981 733.09326172]
```

```
[ 23.5485487  193.03436279 730.87231445]
[ 35.13473558 194.67235565 740.75897217]
[ 43.43209887 192.40405273 735.37045288]
[ 53.64030695 192.03620148 736.28994751]
[ 32.21282959 196.58972931 725.01757812]
[ 37.54105425 195.66731262 730.93939209]
[ 44.42128944 192.30633545 730.42773438]
[ 27.42270088 195.79354095 722.42230225]
[ 16.27226067 187.77633667 720.77542114]
[ 24.07485008 192.55721283 717.17575073]
[ 19.22808075 188.65205383 716.60794067]
[ 43.39631653 190.24427795 722.70425415]
[ 46.27741361 190.97670746 725.49938965]
[ 37.98929787 191.93200684 721.46154785]
[ 38.85950947 185.91604614 716.99658203]
[ 27.43633461 193.41887665 717.4173584 ]
[ 31.05289745 189.5327301  715.15841675]
[ 36.40331554 181.28199768 713.68966675]
[ 19.98805046 186.05422974 712.56588745]
[ 24.682024   187.76364899 712.76843262]
[ 11.61537743 183.70407104 715.65658569]
[ 15.60447311 184.99741364 713.49395752]
[ 12.66144943 180.451828   711.52816772]
[  8.50260162 179.17464447 713.2623291 ]
[ 13.51849937 176.07566071 709.92214966]
[  3.2338562  173.42536926 716.51690674]
[  3.04412079 172.55899811 714.41448975]
[  5.67522812 172.35197449 711.79415894]
[ 10.93650246 168.88297272 709.50244141]
[ 30.23887157 179.89911652 711.29101562]
[ 21.31562233 173.22200775 708.6590271 ]
[ 35.3953619  175.51872253 711.88980103]
[ 23.4935627  165.8639679  709.17440796]
[ 35.00395679 168.14386749 711.60479736]
[ 30.67083836 165.73227692 711.53088379]
[117.71427536 194.81039429 894.39746094]
[121.37039566 191.29076385 896.34063721]
[109.3960228  197.46855164 884.92489624]
[105.1795845  199.11228943 887.03137207]
[112.08042526 173.80294037 887.49456787]
[118.15594864 188.92797089 886.98901367]
[116.24298477 184.83251953 881.36849976]
[112.59149933 180.42503357 875.56204224]
[116.29751205 178.43435669 886.89550781]
[104.57847977 195.32846069 864.9359436 ]
[102.86961746 193.2633667  855.54421997]
[ 91.92583466 198.77471924 851.52865601]
[105.58828735 190.44947815 858.15170288]
```

```
[100.58901978 186.93921661 843.35400391]
[ 96.53897476 197.58101654 854.76342773]
[102.08407974 174.44290161 858.89349365]
[ 97.59155655 170.15426636 860.52770996]
[105.43948364 183.69207764 855.29534912]
[102.37390518 179.34024811 850.26986694]
[ 87.88959312 195.27431488 823.34951782]
[ 93.07738876 188.73141479 823.68850708]
[ 95.10572815 174.27189636 837.01296997]
[ 93.79866409 171.21578979 842.06524658]
[ 99.13741684 180.76824951 839.9420166 ]
[ 92.50770378 175.35554504 825.11355591]
[ 90.47908783 191.06175995 818.81716919]
[ 90.8326664  183.85860443 814.54574585]
[ 83.4243145  197.24013519 820.65994263]
[ 85.44424057 168.856987   816.08862305]
[ 90.71983147 179.13137817 814.75137329]
[ 77.53133774 196.33751678 802.72421265]
[ 82.3146534  193.42720795 803.25204468]
[ 84.14326859 188.52190399 796.57046509]
[ 83.85861969 172.60700989 796.11053467]
[ 83.98747444 170.5663681  801.51379395]
[ 84.89530182 182.37627411 794.70031738]
[ 71.3751812  195.41438293 785.82107544]
[ 78.0255928  190.22892761 781.75805664]
[ 78.13213921 182.2370224  769.12069702]
[ 77.03991508 186.93083191 770.22753906]
[ 83.90042877 177.28977203 791.55514526]
[ 79.21682739 166.52546692 793.45404053]
[ 71.94264889 193.34066772 775.3560791 ]
[ 74.63944244 189.95323944 769.96920776]
[ 67.80037212 193.7353363  767.8682251 ]
[ 78.27341843 170.06495667 774.81344604]
[ 74.61605453 161.95574951 778.33563232]
[ 77.08469009 174.5400238  763.77740479]
[ 67.29985523 191.66226196 752.99417114]
[ 60.27020979 192.4371109  752.35168457]
[ 71.72628975 188.03821564 752.11938477]
[ 73.93159676 183.8957901  752.52230835]
[ 75.09613228 175.92485809 751.8230896 ]
[ 75.42550564 168.19687653 759.38552856]
[ 73.85873795 161.29862213 762.490448  ]
[ 74.04899025 162.0767746  754.51373291]
[ 65.85019207 191.90161133 743.80148315]
[ 68.78995228 190.91062164 742.53210449]
[ 60.8149991  192.33509064 741.34576416]
[ 66.65299177 193.59191132 731.06832886]
[ 72.13871098 186.29616547 743.9671936 ]
```

```
[ 74.15958405 179.09942627 741.1065979 ]
[ 72.6723814  188.03793335 733.17297363]
[ 75.36859798 178.24359131 732.83682251]
[ 74.8590374  170.29611969 743.84143066]
[ 74.36093712 159.12109375 744.81729126]
[ 76.32697582 166.86468506 733.16622925]
[ 75.68126297 155.15002441 739.42739868]
[ 79.07827759 154.81208801 729.15914917]
[ 67.20994425 194.34043121 722.3102417 ]
[ 69.55982542 193.51714325 723.90942383]
[ 68.55678511 192.59938049 717.72592163]
[ 71.19846725 191.10154724 720.2171936 ]
[ 65.61412573 191.5763855  714.29251099]
[ 68.72970104 188.3143158  712.1491394 ]
[ 59.72628665 194.30374146 727.94070435]
[ 49.16358101 191.94631195 729.36972046]
[ 53.44588426 192.70007324 724.76528931]
[ 57.9072088  192.63412476 719.7824707 ]
[ 73.53088856 185.6992569  726.74584961]
[ 75.05228138 179.00695038 726.08584595]
[ 73.78336716 177.62302399 719.37780762]
[ 77.26211262 163.77796936 725.16873169]
[ 59.326015   190.31822968 714.23861694]
[ 46.92789602 187.98905182 719.63290405]
[ 53.79488516 186.70451355 712.71496582]
[ 44.45273495 182.66746521 714.87207031]
[ 55.42218411 179.72200012 705.57208252]
[ 69.31176472 184.05751801 709.67990112]
[ 70.43795204 173.34873199 708.88116455]
[ 72.3171773  172.08124542 714.20480347]
[ 62.47988367 184.62391663 706.55151367]
[ 65.93476105 185.36209106 707.68875122]
[ 64.76720667 179.2838974  703.5843811 ]
[ 67.288661   171.81148529 703.4927063 ]
[ 42.0046463  173.00661469 710.65634155]
[ 68.62376213 167.87348938 706.82730103]
[ 64.89259791 168.13386536 701.86203003]
[ 67.48383188 161.41609192 702.42962646]
[ 79.43624592 153.32829285 723.94720459]
[ 78.28122807 153.82489777 719.15811157]
[ 75.14015579 162.57654572 716.95584106]
[ 76.60097885 151.76292419 715.52526855]
[ 70.43690825 152.4298172  707.49868774]
[ 58.77723774 175.49544525 701.83743286]
[ 62.73273504 173.9495163  701.44015503]
[ 53.03497601 172.14720154 703.4982605 ]
[ 39.26581478 166.14147949 711.25665283]
[ 38.20747566 163.12667847 713.82894897]
```

```
[ 40.18239594 161.21381378 712.27734375]
[ 43.4588232  162.62502289 708.53540039]
[ 51.32401896 163.59259033 703.15621948]
[ 50.71789551 157.91194916 704.1555481 ]
[ 60.47247593 166.25831604 700.68121338]
[ 56.9268887  160.93045807 700.93487549]
[ 62.75615948 156.62483215 699.40481567]
[ 65.84906697 159.57052612 700.61355591]
[ 57.72212279 153.61045074 700.10131836]
[ 92.31791306 171.1855545  886.38897705]
[ 95.70965576 166.66760254 883.79855347]
[ 88.93523788 169.6283493  876.07977295]
[ 79.38810539 170.27799225 852.32861328]
[ 74.60197067 170.45583344 840.27078247]
[ 71.15529442 169.85162354 830.49612427]
[ 58.47119713 169.62637329 801.27941895]
[ 64.90729809 167.83695984 809.32687378]
[ 53.49210835 173.07060242 799.21810913]
[ 57.66990185 166.64897919 791.8449707 ]
[ 50.57160902 169.32944489 787.99520874]
[ 48.39861417 167.25683594 780.73751831]
[ 38.41024852 170.60080719 776.41372681]
[ 34.44307947 166.97780609 770.81900024]
[ 29.0935936  167.06893158 767.49591064]
[ 40.16901308 167.77146912 775.64068604]
[ 48.4055922  164.49503326 771.72271729]
[ 37.78484821 165.32202148 770.92883301]
[ 43.2490685  163.52268219 766.72189331]
[ 49.33910728 163.0761795  766.47601318]
[ 35.06997538 160.51245117 763.81674194]
[ 23.61602879 167.50242615 762.78164673]
[ 22.40576649 163.66886902 760.08258057]
[ 29.26478291 160.58748627 761.58428955]
[ 17.23540306 165.70410156 759.62322998]
[ 11.51360893 169.36518097 756.68563843]
[ 15.12453175 162.70887756 758.80221558]
[ 21.95375824 159.229599   754.73083496]
[ 27.12602043 156.9890976  753.04418945]
[ 45.89954573 160.70780945 756.46337891]
[ 41.25451446 159.2449646  756.55435181]
[ 44.22381949 158.43894196 750.63323975]
[ 31.06858063 156.95051575 750.76043701]
[  5.17516708 166.00645447 753.53823853]
[  9.47180748 164.98715973 756.82495117]
[  3.81565857 159.33177948 752.71496582]
[  9.04548073 161.04959869 755.58822632]
[  2.71128273 171.43886566 747.45755005]
[  1.76328278 163.83794403 750.09967041]
```

```
[  1.04659081 154.50252533 747.94244385]
[4.84241486e-01 1.63913834e+02 7.46176361e+02]
[6.89970016e-01 1.69998680e+02 7.42696625e+02]
[  6.18404198 155.12261963 751.86663818]
[7.09119797e-01 1.72080879e+02 7.38354401e+02]
[ -1.26783752 154.33701324 735.73907471]
[ -1.42108536 163.64717865 732.17977905]
[ -1.15036583 165.28806305 728.81399536]
[ -2.23210907 154.81939697 729.82528687]
[  1.84188461 149.96452332 745.90075684]
[ -2.41855431 148.12323761 735.88708496]
[ -0.94512939 145.49269104 742.41973877]
[ -4.18515587 149.88796997 728.27548218]
[ 14.54152107 157.02656555 754.01367188]
[ 22.70428371 155.53286743 752.41567993]
[ 27.31880283 155.47943878 750.70803833]
[ 30.26589108 152.63710022 746.52783203]
[ 35.56581926 155.96503448 748.91439819]
[ 36.67559147 152.71495056 742.21081543]
[ 40.99148369 152.82016754 742.31585693]
[ 31.09922028 151.04975891 740.4190979 ]
[ 35.69600868 152.34558105 736.9564209 ]
[ 32.97186661 152.75287628 734.68096924]
[ 27.48562622 152.02915955 731.78826904]
[ 38.29054689 153.37524414 730.89904785]
[ 13.05927086 152.82341003 752.38092041]
[ 15.04139137 144.41308975 750.63769531]
[ 24.17542267 147.11971283 748.97949219]
[  6.23256302 150.27377319 749.98690796]
[  4.5059185  145.12667465 748.63479614]
[  8.16465187 140.06475449 746.48309326]
[  1.67988968 141.89011765 744.1859436 ]
[ 24.02602863 141.45550919 745.37905884]
[ 14.66973495 140.43765259 747.8263855 ]

[ 18.36849213 137.17248535 741.74942017]
[ 13.41021538 135.5987587  739.87231445]
[ 26.16191578 144.56845856 746.52676392]
[ 28.82014656 148.77848053 742.40246582]
[ 27.44835377 142.91955185 742.60144043]
[ 25.87551498 148.21086884 733.57116699]
[ 20.52499008 136.16862869 737.04232788]
[ 24.01717758 138.44228363 735.06463623]
[ 32.55943298 155.61645508 728.3482666 ]
[ 36.89170265 158.01404572 725.09515381]
[ 27.63196754 155.36282349 723.47915649]
[ 25.81815147 152.28865814 726.26034546]
[ 28.84271908 158.14374542 720.19369507]
```

```
[ 30.60652828 161.25301361 714.76980591]
[ 37.02449894 160.85269928 717.53076172]
[ 22.84788513 146.8998909  719.8236084 ]
[ 25.72241306 156.01771545 719.24404907]
[  6.57712173 137.24137878 741.2520752 ]
[ -2.21334457 141.7527771  734.84362793]
[  0.85341454 138.94599915 736.3190918 ]
[ -2.64750862 143.55128479 726.20495605]
[  4.17612457 136.74931717 733.63406372]
[  3.63080788 138.54562759 727.15106201]
[ 11.04698944 138.21860886 723.16815186]
[ 10.86320686 135.11044312 731.5824585 ]
[ 24.50536633 141.94623184 727.74923706]
[ 20.56431198 136.38599396 729.65924072]
[ 15.43773079 134.98697662 730.38839722]
[ 20.81772232 140.71871948 722.05496216]
[ 17.54544258 138.36164856 722.76455688]
[ -1.12936401 160.52323914 724.29470825]
[ -3.81289673 153.80433655 724.1164856 ]
[ -2.77924728 157.89733887 720.38296509]
[  1.91575241 161.73039246 712.96420288]
[ -1.58664703 159.02574921 716.39208984]
[  7.51931    160.93256378 710.32376099]
[ 16.10461807 161.2755127  709.05923462]
[ -3.39616203 152.28940582 721.15432739]
[  0.81939697 154.12062073 715.13269043]
[  5.76046181 150.30916595 714.2644043 ]
[ 12.33127403 151.40923309 712.24206543]
[ 27.69639778 162.68996429 711.89416504]
[ 26.62653255 158.15205383 714.82247925]
[ 23.22427368 154.52574158 713.72406006]
[ 20.23740578 145.69218063 717.03717041]
[ 18.66929436 152.5324173  712.47756958]
[ -1.81502151 148.41924286 719.88778687]
[  3.42326927 141.30223083 722.36376953]
[  6.71029472 143.28327179 718.67648315]
[ 15.60513687 143.92799377 716.40319824]
[102.50469589 168.48984528 881.83584595]
[ 99.38187027 166.18000031 886.06195068]
[ 99.96652603 166.60227966 883.30136108]
[ 92.27088547 166.08995819 871.76364136]
[ 88.97264862 165.75803375 857.07226562]
[ 85.69336891 165.62904358 853.83343506]
[ 89.47838974 166.94020081 849.84628296]
[ 84.06797409 165.64372253 842.81771851]
[ 85.90221596 166.78794098 833.73080444]
[ 80.05484962 166.05435944 838.89047241]
[ 76.0227356  165.04496002 824.08935547]
```

```
[ 77.55543137 164.08080292 818.42919922]
[ 78.66735649 164.69006348 806.32672119]
[ 74.14816475 163.56394196 812.14199829]
[ 73.83289528 162.16371155 795.97958374]
[ 67.02384758 163.12322235 794.45001221]
[ 69.66226864 161.61525726 791.37384033]
[ 76.18500328 163.4655838  790.62454224]
[ 72.61557007 160.96356964 787.00793457]
[ 56.31692314 163.52962494 777.18280029]
[ 69.31682873 159.02487946 774.51193237]
[ 63.90114307 159.99825287 775.74185181]
[ 70.39425755 158.29599762 767.66363525]
[ 62.65621853 157.56001282 765.20394897]
[ 65.58161926 156.40606689 763.56170654]
[ 62.30141973 154.33560181 756.47814941]
[ 71.87666798 158.5252533  762.72515869]
[ 70.50134373 155.17558289 755.44302368]
[ 72.69120502 156.47795105 751.30038452]
[ 65.74534321 152.93473816 754.52160645]
[ 53.2305274  155.90216827 751.25296021]
[ 62.77094078 148.63761139 746.11578369]
[ 50.43786564 153.53453064 745.82946777]
[ 56.72695494 149.89656067 744.69393921]
[ 70.86539555 152.03884888 747.67715454]
[ 74.49228191 152.13365936 741.29806519]
[ 66.81123352 147.63600922 743.81201172]
[ 62.25826645 144.76030731 742.17184448]
[ 57.41843963 146.18330002 742.70196533]
[ 66.03778362 144.62340546 741.28311157]
[ 72.34152794 148.29658508 740.28500366]
[ 77.51978493 147.76403046 735.05355835]
[ 71.82471657 143.3149147  737.19161987]
[ 67.64367199 136.6129303  738.82391357]
[ 78.56798458 150.97936249 733.77462769]
[ 45.39214969 149.26911926 739.74176025]
[ 50.45025748 147.62355042 741.03131104]
[ 41.32151937 150.73413849 737.52935791]
[ 44.14988804 148.34519196 733.43002319]
[ 41.2286725  151.18772125 732.14379883]
[ 73.53664207 142.29570389 733.80639648]
[ 47.12857604 145.08946228 737.01470947]
[ 55.20667791 139.39967728 741.0892334 ]
[ 50.75638822 140.59823608 739.27441406]
[ 54.28860188 135.12392426 738.03662109]
[ 50.49914503 136.87175751 736.65679932]
[ 42.16964817 151.09621429 726.87106323]
[ 48.64846063 139.69589615 735.73583984]
[ 52.31628758 130.70066452 728.85733032]
```

```
[ 47.44580436 135.72626114 728.06512451]
[ 61.522542   136.34500504 740.34674072]
[ 69.85371876 133.22344971 734.02941895]
[ 60.79264927 131.35160446 736.57254028]
[ 65.23762274 131.03782272 734.86843872]
[ 70.79014015 133.09955215 728.61532593]
[ 74.37959766 138.97298813 727.05651855]
[ 60.04402113 128.86904144 732.82650757]
[ 58.27452588 126.7084198  727.15985107]
[ 65.97004223 127.2448082  725.96783447]
[ 69.98699808 129.76578522 723.77142334]
[ 66.32895041 126.7724762  719.58102417]
[ 39.49061012 156.79349518 723.55267334]
[ 40.60219669 156.4991684  718.91894531]
[ 44.56705046 143.64693069 724.03338623]
[ 42.99651194 149.68418121 720.21810913]
[ 41.31722546 151.74512482 714.63531494]
[ 42.03418875 145.91425705 717.70303345]
[ 42.74724102 143.7806778  715.47814941]
[ 43.9649992  139.66720581 720.39755249]
[ 45.92212439 135.82610703 719.32217407]
[ 50.85769367 130.72709274 720.09793091]
[ 47.48509598 139.77656174 710.57949829]
[ 49.74033427 132.76737213 716.16482544]
[ 56.55567288 126.53596497 722.6918335 ]
[ 61.85185814 127.29525757 714.57650757]
[ 53.26619124 134.13398743 708.95013428]
[ 55.62431347 129.66309357 712.8621521 ]
[ 63.3831954  132.36128998 705.9163208 ]
[ 79.61024284 149.90103149 728.00094604]
[ 76.68206882 144.62998581 728.14120483]
[ 77.05915356 149.80331421 716.69042969]
[ 77.22763157 146.72467804 721.81439209]
[ 73.75627708 139.19095993 724.02716064]
[ 70.88370419 135.8613472  722.91156006]
[ 70.33078003 147.12506104 710.15652466]
[ 70.87876034 138.49549103 717.06332397]
[ 69.88123798 142.22614288 712.56362915]
[ 69.61165953 142.04370499 703.478302  ]
[ 70.13457775 140.19989395 705.84701538]
[ 70.78769159 132.99832535 712.46328735]
[ 68.50203943 128.7194252  714.11334229]
[ 67.18813419 130.41322708 709.34057617]
[ 67.96063137 135.06346893 705.76272583]
[ 41.64627647 157.78787994 711.19451904]
[ 43.35694647 152.31285095 710.69110107]
[ 52.06327105 151.34882355 703.31027222]
[ 54.1102705  144.37306976 702.98809814]
```

```
[ 66.72550607 150.71440125 700.12915039]
[ 63.22886133 146.56729126 699.62786865]
[ 61.55693805 139.05134583 702.1065979 ]
[ 55.42097211 137.3102951  705.33630371]
[ 67.37001514 142.51745605 701.48519897]
```

13.0.3 ii)

[20]:
```python
#Creating a mesh from the average positions
average_meshA = trimesh.Trimesh(vertices = average_positions, faces = mesh1.
 ↪faces)

# Displaying the average surface
average_meshA.show()
```

[20]: <IPython.core.display.HTML object>

13.0.4 iii)

The resulting average surface display above does look like a representative shape that would describe the 2 surfaces in our dataset. Just from inspection of femur1 and femur2 displayed in Q3i, we can see that the overall image looks similar when rotated. We can also see that from the starting orientation (the pose) of the combined image is a combination of the starting orientation of femur1 and femur2. This can be seen bellow.

$$Femur1$$

$$Femur2$$

$$FemurAverage$$

13.0.5 B) Rigid alignment method

13.0.6 i)

[56]:
```python
from scipy.spatial import procrustes

# Getting the vertices of each mesh
vertices1 = mesh1.vertices
vertices2 = mesh2.vertices

# Performing Procrustes analysis
mtx1, mtx2, disparity = procrustes(vertices1, vertices2)

print(disparity)
```

```
#Creating a mesh from the average positions
average_meshB = trimesh.Trimesh(vertices = mtx2, faces = mesh2.faces)

# Displaying the average surface
average_meshB.show()
```

0.0014234619899051635

[56]: <IPython.core.display.HTML object>

13.0.7 C)

[57]:
```
# Creating a scene and adding the meshes obtained from Q3A and B
scene = trimesh.Scene([average_meshA, average_meshB])

# Displaying the scene
scene.show()
```

[57]: <IPython.core.display.HTML object>

13.0.8 D)

To improve the registration process, we can normalize and preprocess the meshes. We can preprocess the meshes by normalizing their scale, orientation, and position. This makes sure that the meshes are in a consistent coordinate system that ensures their initial alignments are the same. We can also apply noise reduction, smoothing, or other preprocessing techniques to enhance the quality of the meshes.

Incorporating additional information such as color or texture information into the registration process can provide further indications for mis-alignment and improve the registration accuracy.

By incorporating these techniques, data characteristics and desired registration accuracy can be obtained for specific medical purposes.

14 Bibliography

Chen, H. (2017, January 28). DICOM Processing and Segmentation in Python. Retrieved from Radiology Data Quest: https://www.raddq.com/dicom-processing-segmentation-visualization-in-python/

Haggerty, D. (2019, December 8). trimesh. Retrieved from trimesh 3.21.6: https://pypi.org/project/trimesh/

Namayega, C. (2023, April 18). Medical Image Analysis. Cape Town, Western Cape, South Africa. (Lectures 1-8, Jupyter NoteBook Slides)

Rossant, C. (2018, January). Segmenting an image. Retrieved from ipython-books: https://ipython-books.github.io/113-segmenting-an-image/

```
[ ]:
```