

In [19]:

```

#Q1
import numpy as np
from plyfile import PlyData, PlyElement #open3d is the standard
import matplotlib.pyplot as plt

pcd = PlyData.read('airplane.ply')
assert pcd is not None
points = np.concatenate((pcd['vertex']['x'].reshape(1, -1), pcd['vertex']['y'].reshape(1, -1), pcd['vertex']['z'].reshape(1, -1)), axis=0)
points = points - np.mean(points, axis=0).reshape(3,1)

ones = np.ones((1, points.shape[1]))
X = np.concatenate((points, ones), axis=0)

R = np.array([[1, 0, 0],[0, 1, 0],[0, 0, 1]])
K = np.array([[1, 0, 0],[0, 1, 0],[0, 0, 1]])
t = np.array([[0], [0], [-4000]])

P1 = K @ np.concatenate((R, t), axis=1)

R = np.array([[0,1,0],[1,0,0],[0,0,1]])
K = np.array([[0.5,0,0],[0,0.5,0],[0,0,1]])
t = np.array([[0],[0],[-4000]])

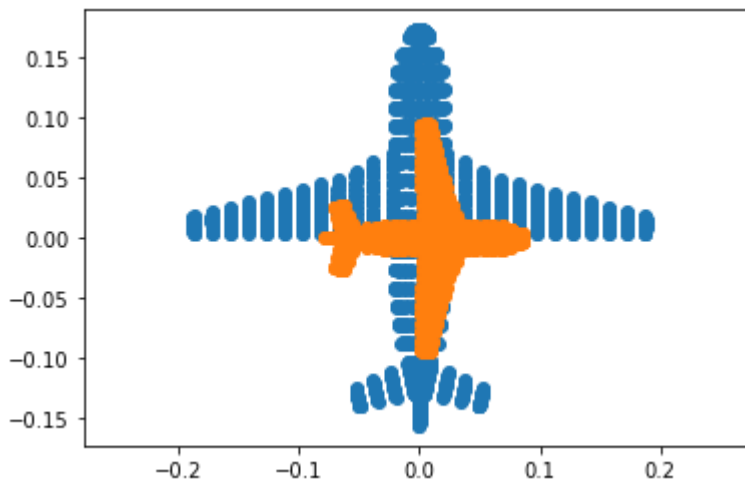
P2 = K @ np.concatenate((R,t),axis=1)

x1 = P1 @ X
x2 = P2 @ X

x1 = x1/x1[2,:]
x2 = x2/x2[2,:]

fig, ax = plt.subplots(1,1, sharex=True, sharey=True)
ax.scatter(x1[0,:], x1[1,:])
ax.scatter(x2[0,:], x2[1,:])
ax.axis('equal')
plt.show()

```



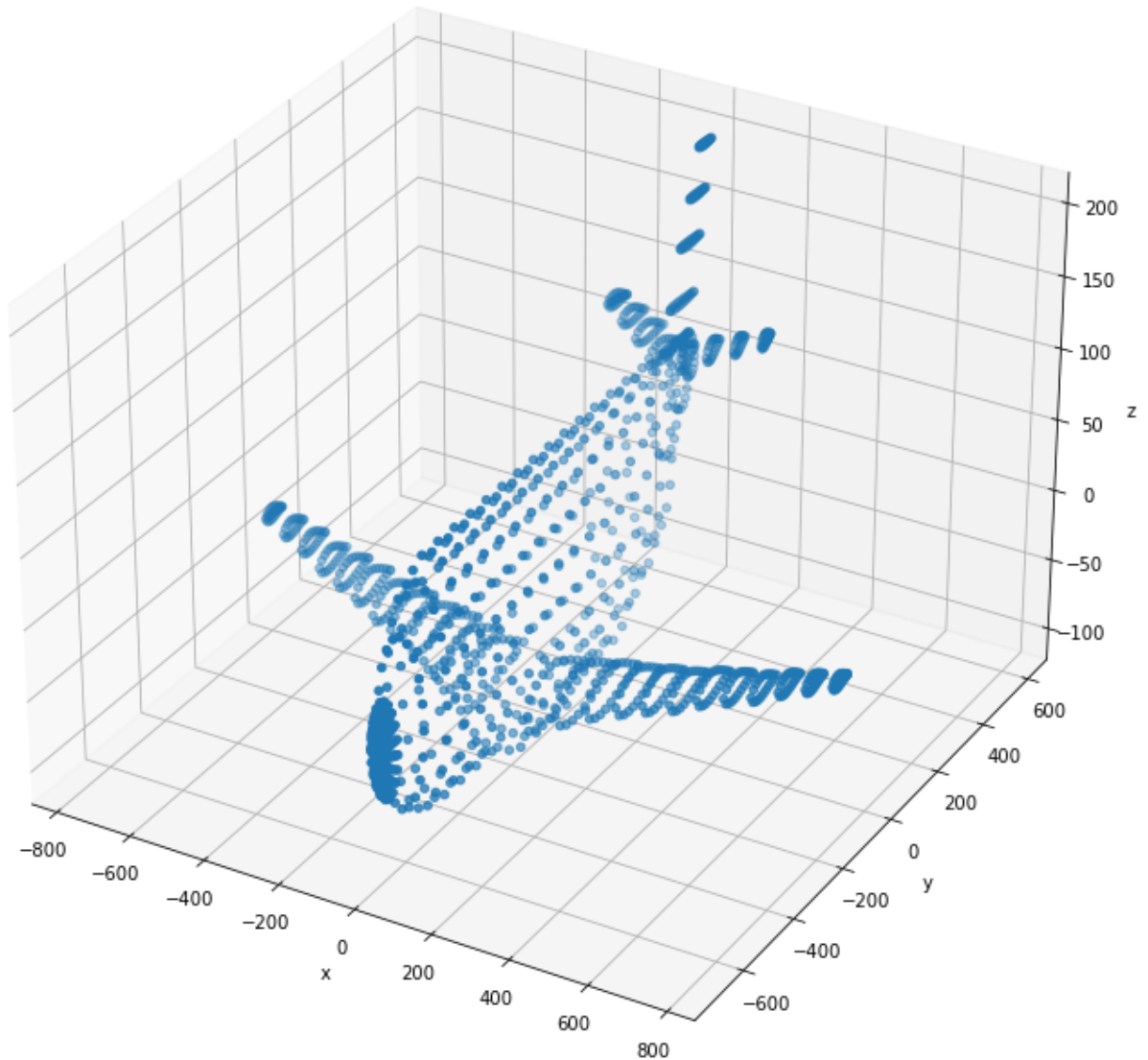
In [20]:

```
fig= plt.figure (figsize=(12,12))
```

```

ax = fig.add_subplot (111, projection='3d')
ax.scatter (points[0,:], points[1,:], points [2,:])
ax.set_xlabel('x')
ax.set_ylabel('y')
ax.set_zlabel('z')
plt.show()

```



In [30]:

```

#Q3
import cv2 as cv
import numpy as np
im = cv.imread('earrings.jpg',cv.IMREAD_COLOR)

hsv = cv.cvtColor(im,cv.COLOR_BGR2HSV)
th , bw = cv.threshold(hsv[:, :,1],0,255,cv.THRESH_BINARY+cv.THRESH_OTSU)

w = 5
kernel = np.ones((w,w),np.uint8)
opened = cv.morphologyEx(bw,cv.MORPH_CLOSE,kernel)

```

```

retval , labels,stats ,centroids = cv.connectedComponentsWithStats(bw)
colormapped = cv.applyColorMap((labels/np.amax(labels)*255).astype('uint8'),cv.COLORMAP_

Z = 720
f = 8
for i,s in enumerate(stats):
    if i!=0:
        print('item',i,',area in pixels =',s[4])
        print('item',i,',area in mm^2 =',s[4]*(2.2e-3)**2*(Z*Z)/(f*f))

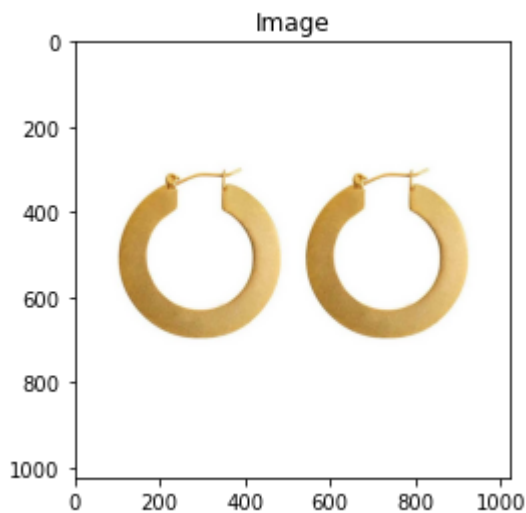
im = cv.cvtColor(im, cv.COLOR_BGR2RGB)
plt.imshow(im)
plt.title('Image')
plt.show()
plt.imshow(hsv[:, :,1])
plt.title('hsv Image')
plt.show()
plt.imshow(bw)
plt.title('bw Image')
plt.show()
plt.imshow(opened)
plt.title('Opened Image')
plt.show()
plt.imshow(colormapped)
plt.title('Colormapped Image')
plt.show()

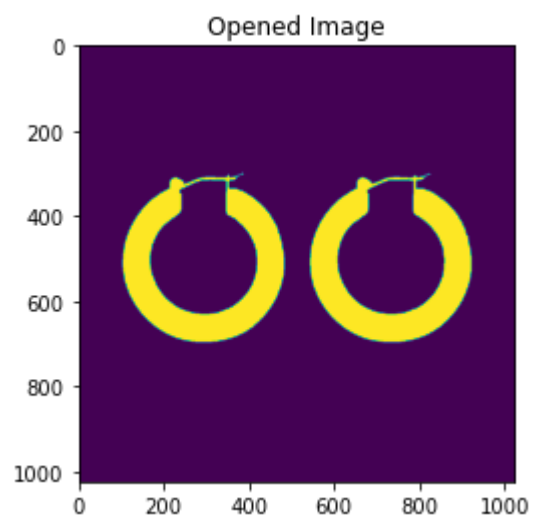
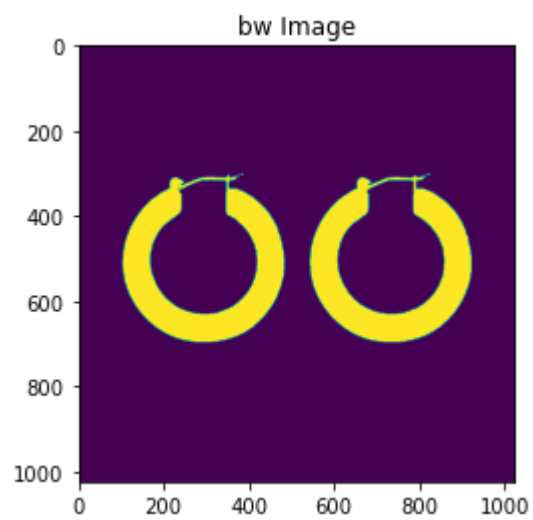
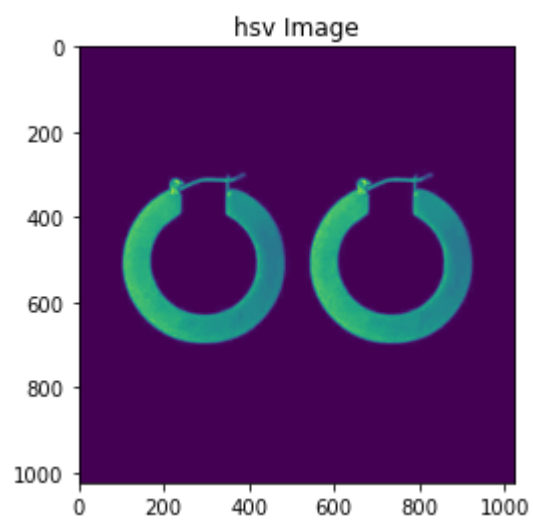
```

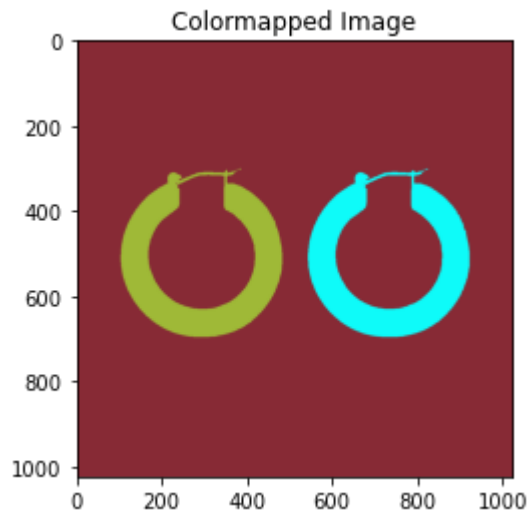
```

item 1 ,area in pixels = 59143
item 1 ,area in mm^2 = 2318.642172
item 2 ,area in pixels = 59211
item 2 ,area in mm^2 = 2321.3080440000003

```







In []:

```
#Q4
import cv2 as cv
import numpy as np
import matplotlib.pyplot as plt

file_name = 'allenkeys.jpg'
im = cv.imread(file_name , cv.IMREAD_REDUCED_GRAYSCALE_2)
canny = cv.Canny(im , 50 , 150)
canny_color = cv.cvtColor(canny , cv.COLOR_GRAY2BGR)
lines = cv.HoughLines(canny , 1 , np.pi/180, 170, None, 0, 0)

if lines is not None :
    for i in range(0,len(lines)):
        rho = lines[i][0][0]
        theta = lines[i][0][1]
        a = np.cos(theta)
        b = np.sin(theta)
        x0 = a*rho
        y0 = b*rho
        pt1 = (int(x0 + 1000*(-b)) , int(y0 + 1000*(a)))
        pt2 = (int(x0 - 1000*(-b)) , int(y0 - 1000*(a)))
        cv.line(canny_color , pt1 , pt2 ,(0,0,255),1,cv.LINE_AA )

cv.namedWindow( 'Image' , cv.WINDOW_AUTOSIZE)
plt.imshow(im)
plt.title('Image')
plt.show()
plt.imshow(canny)
plt.title('canny')
plt.show()
plt.imshow(canny_color)
plt.title('canny color')
plt.show()

r = cv.selectROI( 'Image' , canny_color , showCrosshair = True , fromCenter = False )
cv.waitKey(0)
cv.destroyAllWindows( )

x0,y0 = int(r[0]+r[2]/2), int(r[1]+r[3]/2)
m = b/a
m=np.tan(np.median(lines[:,0,1]))
c=y0-m*x0
```

```

cv.line(canny_color , (0,int(c)),(im.shape[0],int(m*im.shape[0]+c)),(0,255,0),2,cv.LINE_8)

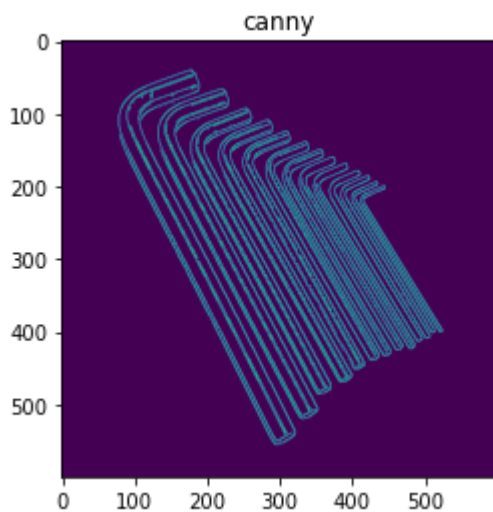
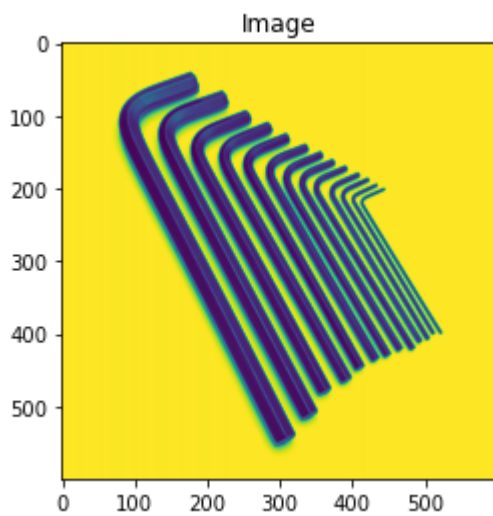
plt.imshow(canny_color)
plt.title('canny_color')
plt.show()

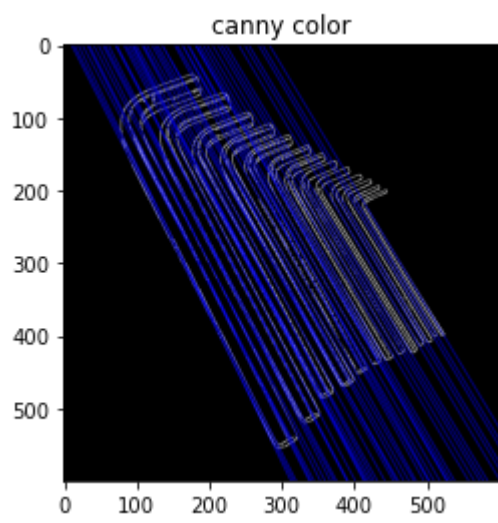
dy=1
y_sub_pixel = np.arange(0,im.shape[0]-1,dy)
f_sub_pixel=np.zeros_like(y_sub_pixel)
f_sub_pixel_nn=np.zeros_like(y_sub_pixel)

for i,y in enumerate(y_sub_pixel):

fig,ax=plt.subplots(figsize=(30,5))
ax.plot(f_sub_pixel_nn)

```





In []: