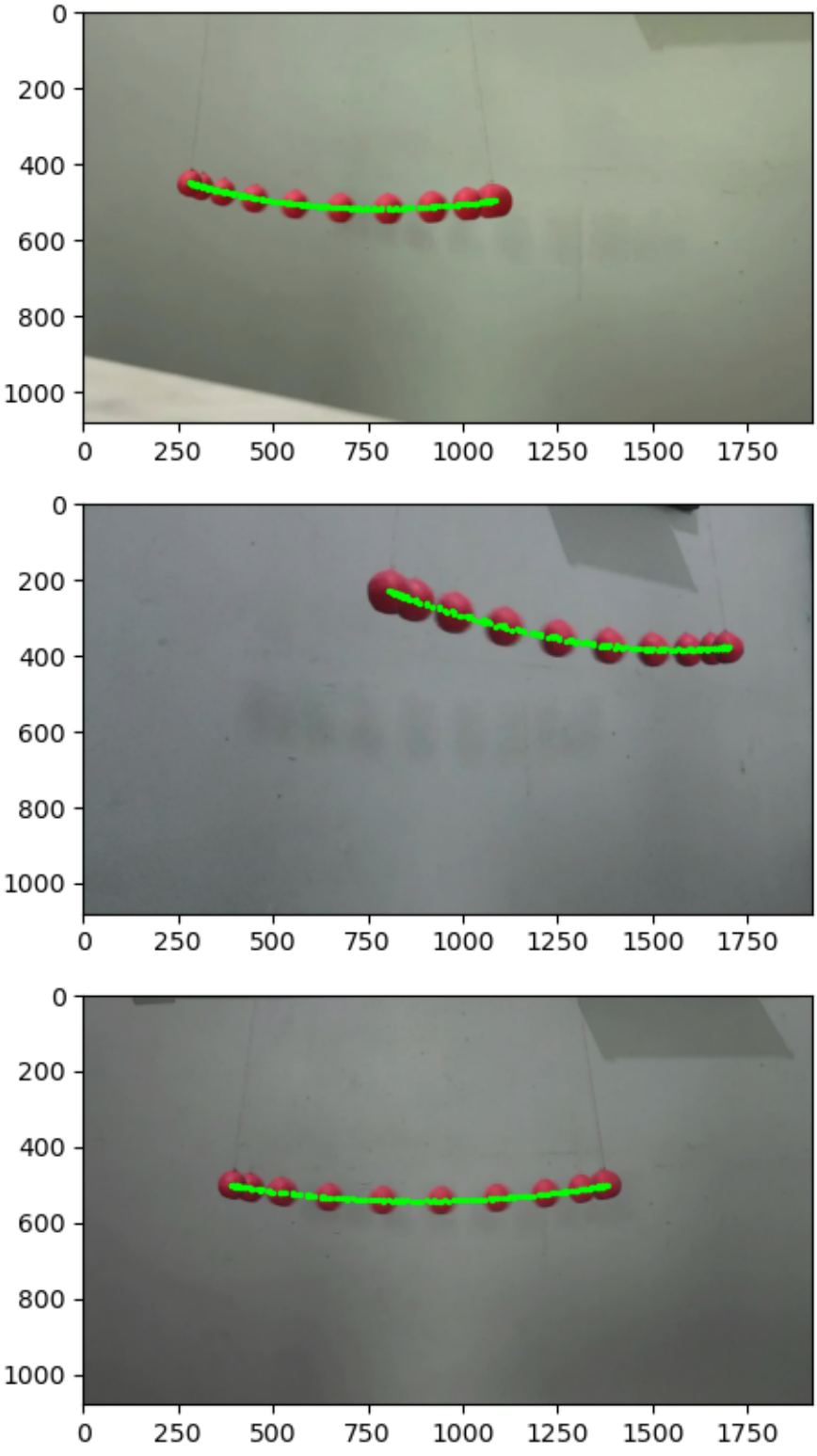
Phys215 Problem Set 2

February 26, 2025

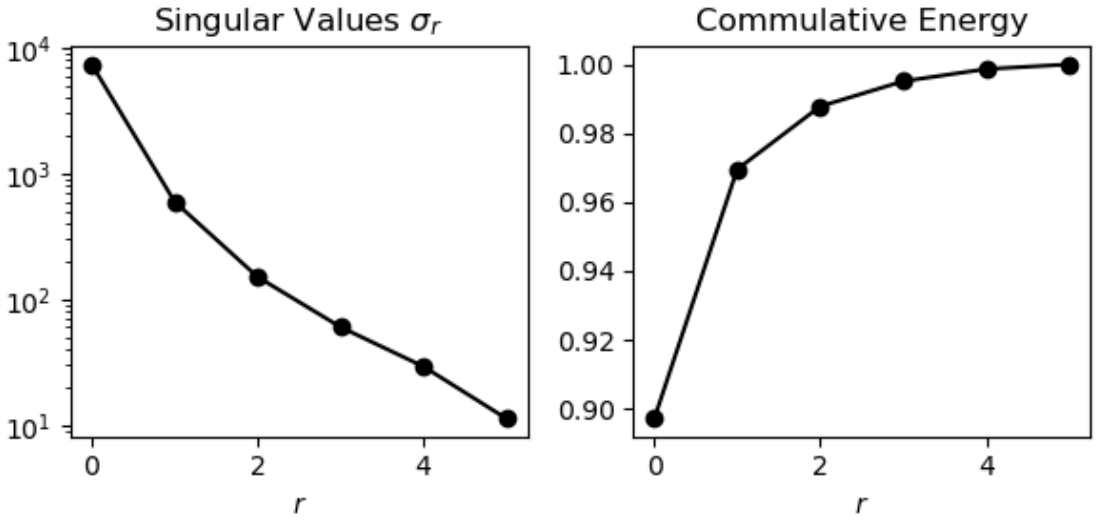
[1]:
```
import matplotlib.pyplot as plt
import numpy as np


# npoints = 185
npoints = 185

X = np.zeros((6,npoints))
X[0:2,:] = np.loadtxt('PendulumData/PendulumData1.txt')[:,:npoints]
X[2:4,:] = np.loadtxt('PendulumData/PendulumData2.txt')[:,:npoints]
X[4:6,:] = np.loadtxt('PendulumData/PendulumData3.txt')[:,:npoints]

Xavg = np.mean(X,axis=1)                # Compute mean
B = X - np.tile(Xavg,(npoints,1)).T     # Mean-subtracted data

obs = B.T
U, S, VT = np.linalg.svd(obs,full_matrices=0) # PCA Analysis
```

[2]:
```
from PIL import Image

image1 = Image.open("PendulumData/Pendulum1.png")
image2 = Image.open("PendulumData/Pendulum2.png")
image3 = Image.open("PendulumData/Pendulum3.png")

plt.figure(figsize=(6,10))
plt.subplot(311)
plt.imshow(image1)
plt.scatter(X[0],-X[1],color="lime",s=2)

# plt.figure(figsize=(8,13.5))
plt.subplot(312)
plt.imshow(image2)
plt.scatter(X[2],-X[3],color="lime",s=2)

# plt.figure(figsize=(8,13.5))
plt.subplot(313)
plt.imshow(image3)
plt.scatter(X[4],-X[5],color="lime",s=2)
```

[2]: <matplotlib.collections.PathCollection at 0x1271274d0>



[3]:
```
# singular values and cumulative sum
fig, ax = plt.subplots(1,2,figsize=(6,3))
ax[0].set_title("Singular Values $\\sigma_r$")
ax[0].semilogy(S,'-o',color='k')
ax[0].set_xlabel("$r$")
ax[1].set_title("Commulative Energy")
ax[1].set_xlabel("$r$")
ax[1].plot(np.cumsum(S)/np.sum(S),'-o',color='k')
plt.tight_layout()
plt.show()
print(np.cumsum(S)/np.sum(S))
```

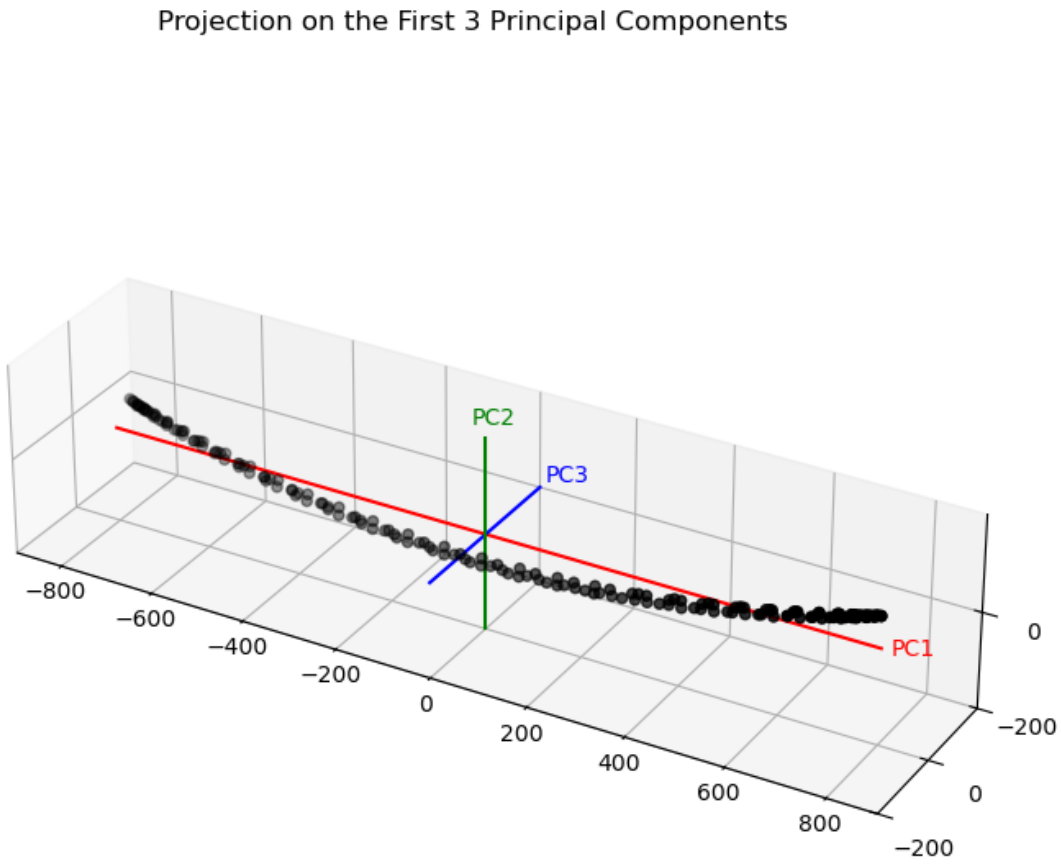



```
[0.89708835 0.96953189 0.98780821 0.9951016  0.99863772 1.        ]
```

[4]:
```
fig1 = plt.figure(figsize=(8,8))
ax = fig1.add_subplot(111, projection='3d')

x = VT[0,:] @ obs.T
y = VT[1,:] @ obs.T
z = VT[2,:] @ obs.T

ax.scatter(x,z,y,color='k')
ax.set_xlim(-900,900)
ax.set_ylim(-200,200)
ax.set_zlim(-200,200)
ax.set_yticks(np.arange(-200,200,200))
ax.set_zticks(np.arange(-200,200,200))
```

```
ax.set_aspect("equal")
ax.plot([-800,800],[0,0],[0,0],color="r")
ax.plot([0,0],[-200,200],[0,0],color="b")
ax.plot([0,0],[0,0],[-200,200],color="g")
ax.text(820,0,-10, "PC1", color="r")
ax.text(0,220,0, "PC3", color="b")
ax.text(-30,0,220, "PC2", color="g")
plt.title("Projection on the First 3 Principal Components")
```

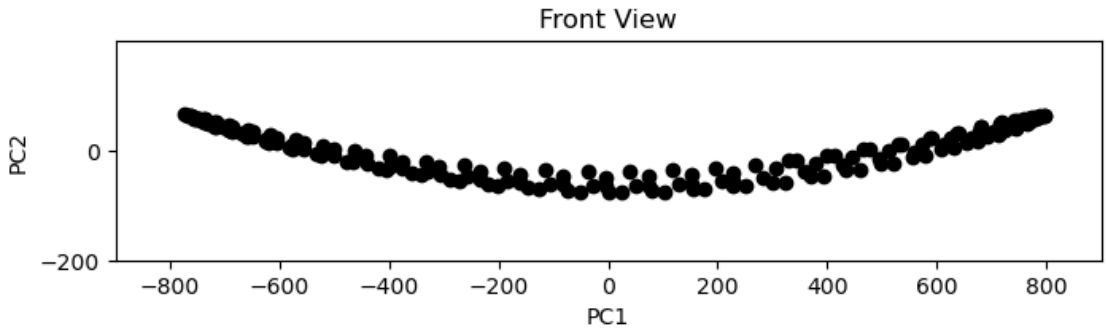
[4]: Text(0.5, 0.92, 'Projection on the First 3 Principal Components')



[5]:
```
fig1 = plt.figure(figsize=(8,8))
ax = fig1.add_subplot(111)
```

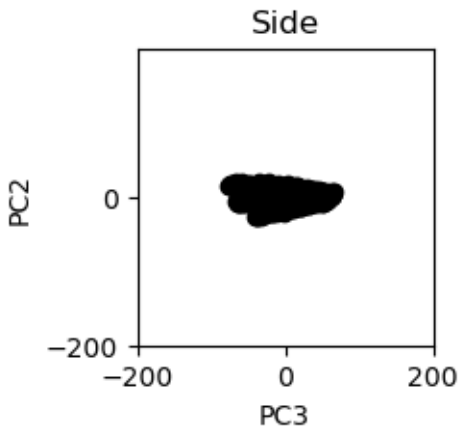
```
x = VT[0,:] @ obs.T
y = VT[1,:] @ obs.T
z = VT[2,:] @ obs.T

ax.scatter(x,y,color='k')
ax.set_xlim(-900,900)
ax.set_ylim(-200,200)
ax.set_yticks(np.arange(-200,200,200))
ax.set_aspect("equal")
ax.set_xlabel("PC1")
ax.set_ylabel("PC2")
plt.title("Front View")
```

[5]: `Text(0.5, 1.0, 'Front View')`



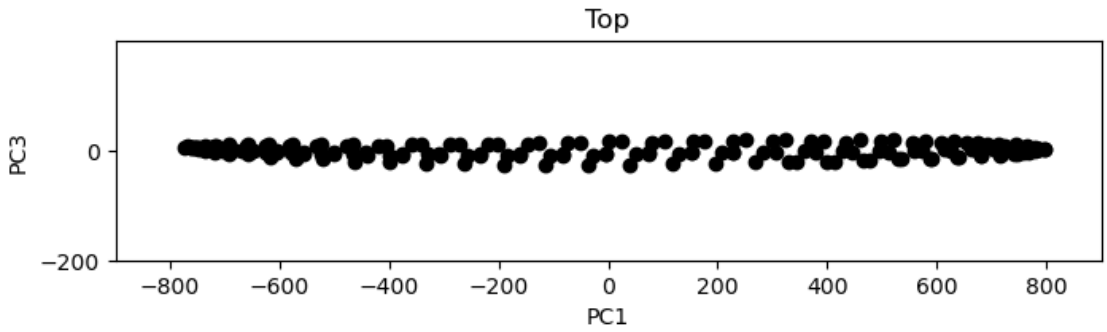
[6]:
```
fig1 = plt.figure(figsize=(2,2))
ax = fig1.add_subplot(111)

x = VT[0,:] @ obs.T
y = VT[1,:] @ obs.T
z = VT[2,:] @ obs.T

ax.scatter(y,z,color='k')
ax.set_xlim(-200,200)
ax.set_ylim(-200,200)
ax.set_yticks(np.arange(-200,200,200))
ax.set_aspect("equal")
ax.set_xlabel("PC3")
ax.set_ylabel("PC2")
plt.title("Side")
```

[6]: `Text(0.5, 1.0, 'Side')`



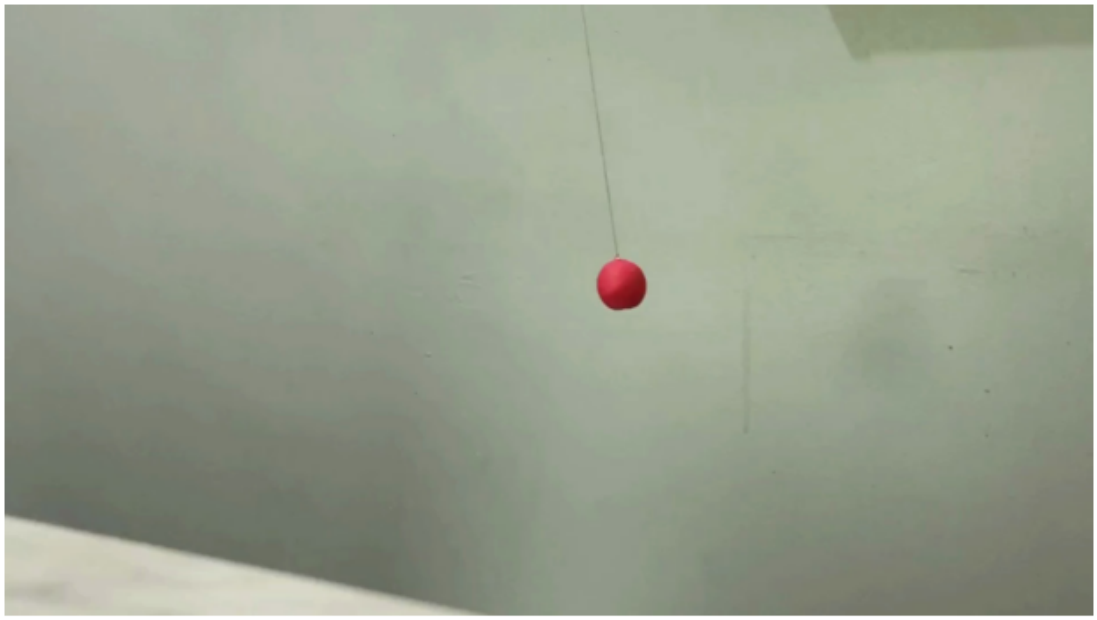
[7]:
```
fig1 = plt.figure(figsize=(8,8))
ax = fig1.add_subplot(111)

x = VT[0,:] @ obs.T
y = VT[1,:] @ obs.T
z = VT[2,:] @ obs.T

ax.scatter(x,z,color='k')
ax.set_xlim(-900,900)
ax.set_ylim(-200,200)
ax.set_yticks(np.arange(-200,200,200))
ax.set_aspect("equal")
ax.set_xlabel("PC1")
ax.set_ylabel("PC3")
plt.title("Top")
```

[7]:
```
Text(0.5, 1.0, 'Top')
```



[8]:
```
from PIL import Image

image1 = Image.open("PendulumData/frame.jpg")
image2 = Image.open("PendulumData/hsv.jpg")
image3 = Image.open("PendulumData/mask.jpg")
image4 = Image.open("PendulumData/result.jpg")

plt.figure(figsize=(8,5))
plt.imshow(image1)
plt.title("(a) Original Image",y=-0.1)
plt.gca().axis('off')
```

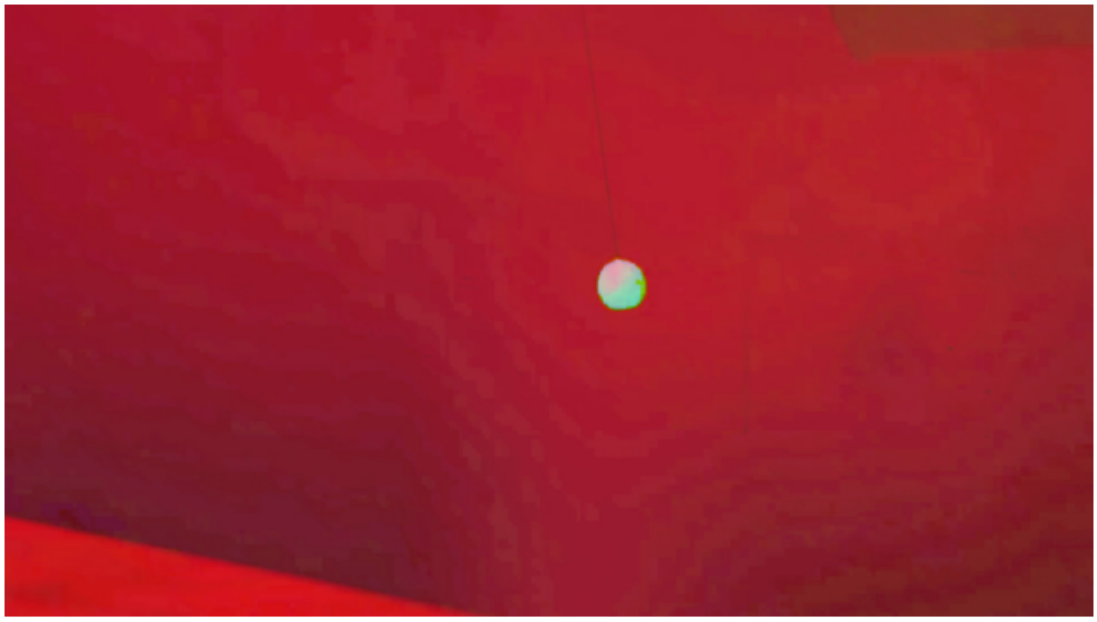
[8]:
```
(np.float64(-0.5), np.float64(1919.5), np.float64(1079.5), np.float64(-0.5))
```



(a) Original Image

[9]:
```
plt.figure(figsize=(8,5))
plt.imshow(image2)
plt.title("(b) HSV Filter",y=-0.1)
plt.gca().axis('off')
```

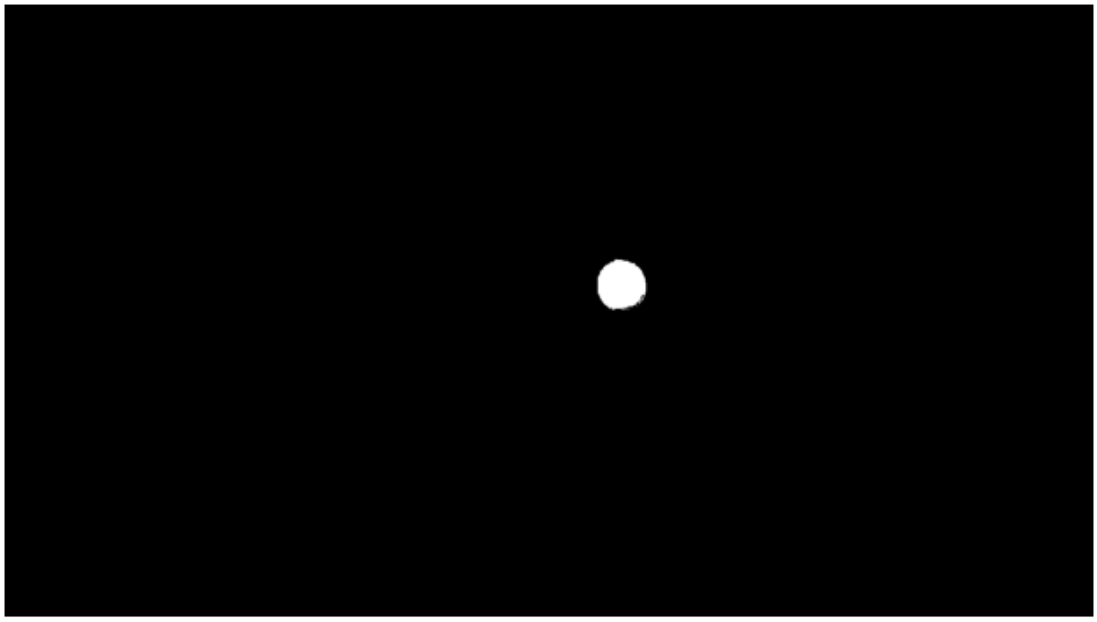
[9]:
```
(np.float64(-0.5), np.float64(1919.5), np.float64(1079.5), np.float64(-0.5))
```



(b) HSV Filter

[10]:
```
plt.figure(figsize=(8,5))
plt.imshow(image3,cmap="grey")
plt.title("(c) Mask",y=-0.1)
plt.gca().axis('off')
```

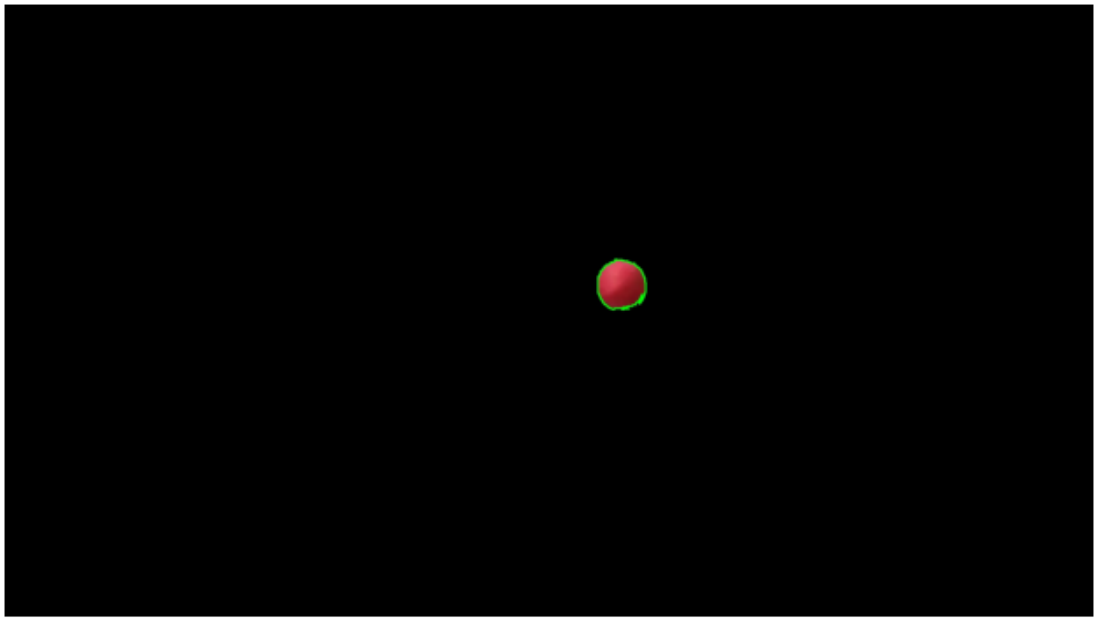
[10]: (np.float64(-0.5), np.float64(1919.5), np.float64(1079.5), np.float64(-0.5))



(c) Mask

[11]:
```
plt.figure(figsize=(8,5))
plt.imshow(image4)
plt.title("(d) Contour Detection",y=-0.1)
plt.gca().axis('off')
```

[11]: (np.float64(-0.5), np.float64(1919.5), np.float64(1079.5), np.float64(-0.5))



(d) Contour Detection