PendulumPosition

April 11, 2025

[2]:
```
import cv2
import numpy as np
import matplotlib.pyplot as plt

cap = cv2.VideoCapture("PendulumVids/Pendulum1.mp4")
xdata = []
ydata = []

for i in range(186): #186 frames to cover three periods
    _, frame = cap.read()
    # It converts the BGR color space of image to HSV color space
    hsv = cv2.cvtColor(frame, cv2.COLOR_BGR2HSV)

    # Threshold of blue in HSV space
    # Play around with the values to isolate the object you want to track
    lower_blue = np.array([0, 100, 100])
    upper_blue = np.array([255, 255, 255])

    # preparing the mask to overlay
    mask = cv2.inRange(hsv, lower_blue, upper_blue)

    # The black region in the mask has the value of 0,
    # so when multiplied with original image removes all non-blue regions
    result = cv2.bitwise_and(frame, frame, mask = mask)

    contours, _ = cv2.findContours(mask, cv2.RETR_TREE, cv2.CHAIN_APPROX_SIMPLE)

    for cnt in contours:
        # Calculate area and remove small elements
        area = cv2.contourArea(cnt)
        if area > 1000:
            cv2.drawContours(result, [cnt], 0, (0,255,0), 2)
            (x,y),radius = cv2.minEnclosingCircle(cnt)
            center = (int(x),int(y))
            radius = int(radius)
            xdata.append(x)
            ydata.append(-y)
```

```
    cv2.imshow('frame', frame)
    cv2.imshow('hsv',hsv)
    cv2.imshow('mask', mask)
    cv2.imshow('result', result)

    cv2.imwrite('frame.jpg', frame)
    cv2.imwrite('hsv.jpg', hsv)
    cv2.imwrite('mask.jpg', mask)
    cv2.imwrite('result.jpg', result)

    cv2.waitKey(0) # wait for key press to proceed to the next frame


cv2.destroyAllWindows()
cap.release()

np.savetxt("PendulumData/PendulumData1.txt",np.vstack((xdata,ydata)),fmt="%f")
```

```
2025-04-11 07:53:34.290 python[63845:2790320] +[IMKClient subclass]: chose
IMKClient_Modern
2025-04-11 07:53:34.290 python[63845:2790320] +[IMKInputSession subclass]: chose
IMKInputSession_Modern
```