DSP Homework 07

Su, Rundong 8

October 17, 2022

Contents

Abstract

In this article, I first write three summaries about the video we watched on class, the first is about how our mouse get its move distance, and the second is about starch based diet is healthier and the last one shows us animals living in the pole areas, and how sea ice influence on them and our planet. Then I write my thoughts about these three videos in chapter1.2. Next I discuss the advantages and dis advantages of Nyquist/Shannon sampling method. Last I write a program to rebuild our phone move path by analyze the video captured by our phone, using method of feature points match, and I analyze the error of path rebuild by our computer.

1 About videos

1.1 Summary of videos we watched

1.1.1 How image sensor in our computer mouse works

The video introduce that there are seven main technology in our mouse: Push Button, Optical Navigation Image Sensor, Scroll Wheel, Wireless Communication, Lithium-Ion Battery, Printed Circuit Boards and System on a chip, this video mainly talks about image sensor. The video first introduce what exactly happens when we move our mouse around on a mouse pad, and next it introduce why game mouse has a higher-performance.

At the bottom of the mouse there is a IAS(Image Acquisition System), which is composed of a image pixel array, a infrared LED and a pairs of lens. The infrared light generated by LED pass through the lens and lighten the surface under the mouse, then the infrared light bounces off the surface get through second lens and directly into image pixel array(also named image sensor composed of 40×40 pixels), a small topographically and texturally complex landscape is captured by our image sensor. Image sensor can capture 1700 images under the mouse, but it doesn't save them at all, it just compare the current image with the last image it captured(59 us time interval between two images). To calculate the x and y direction difference between two images, these two image are send to DSP(Digtial signal processor) and calculate in this part. The DSP confirm the first picture and overlay the second on it , then it moves the second picture until the difference(subtraction of pixel values between the two images) between these image is smallest. Then the amount of shift in position to reach the smallest difference imply how far our mouse move in x axis and y axis. The 57 us later it compared another two pictures, then the change in x axis and in y axis will be send to our computer by Bluetooth or USB transceiver.

The main difference between gaming mouse and non-gaming mouse first is that the former has DPI(Dots Per Inch) much more than the later. In order to get high DPI, gaming mouse a single pixel into many subpixels, and use interpolation to get the value of this subpixels. To make things simple, the high DPI a mouse has, the higher sensitivity it will have. The second difference is that gaming mouse report their movements to computer 1000 times/s, while non-gaming mouse only report 120 times/s.

1.1.2 Living based on starch

This is a speech lead by a doctor, at the beginning of the speech, he give three examples that people start living on starch diet, their health turns better, so he recommend people to live based on starch. And he find starch diet is health for human due to an accident. He worked in Hawaii for three year, and he has observed generations of peoples. The first generation based on starch is more healthy and has less sick, the second generation abandon the starch diet, and they are easier to be sick than the first generation. The third generation who fare the starch has more sicks. After these three years, he went back to Oahu to become a board-certified internist, he spent next two years study in the Medical Library, and he found that in human history, that all large and successful people obtain the bulk of their calories from starch. And the king together with queens who eat meats often had more sick. After introducing it, he said that in current society, children are much easier to be sick so and our solider. In the end he appeals us to live based on starch, but it is not give up meat totally, it is just living on the starch.

1.1.3 Frozen worlds

This video mainly shows us the daily life of animals in the polar regions with the sea ice decreasing, and emphasize the importance of sea ice for our planet. It first shows animals living in the Antarctica, and then shows animals living in the the Arctic Pole, Last, it shows the influence of decreasing of sea ice.

In Antarctica, it first shows penguins breed in Bare rock. And then the camera turn to the world under the sea ice, there are many krill, but when sea ice melt, these krill will lost protection and preyed by penguins. The video then shows the sense that tropics prey krill in teamwork. They create a curtain of bubbles around their prey and as the spiraling net tightens, the krill are driven together, then whales lunge upwards and collect krill. Since the ban of commercial whale-hunt, the number of whales has increased, but their food supply is threaten. With the temperature raise and disappearing sea ice, the number of krill has decline half in the past 50 years. After this the video shows Orca prey on penguins. All these animal the videos shows to us ultimately depend on sea ice.

After showing the life living in the Antarctica, the video show the life around the island of South Georgia. It shows us albatross chick wait for their parents for food, penguins move through leopard seal predation, pass through the sleeping elephant seals at the same time they need look out for being crushed, and finally they get to destination and they find their child by calling, a whole process is how penguins feed their chicks. Also all the creatures living in the the island of South Georgia depend on the krill under the sea ice around the Antarctic peninsula.

In the Arctic, the video shows a sense that adult polar bear lead her pup bear hunting for seals. Now time, the sea freezes so late, that seals cannot but put their children in the open. This is both harmful for seals and polar bear. Polar bear also rely on sea ice, if sea ice disappears, so the polar bear. After showing this, the video shows us narwhal come Arctic for hunting

Sea ice is not just crucial to the lives of those creatures that livw around the poles, it also plays a vital role in determining the climate of the whole planet. It can reflect most energy coming from the sun back into space. In the last of the video, the video shows us the impact of the loss of sea ice on our planet. Lack of sea ice, more 1000000 walrus need to rest on the beach rather than sea ice which is their nature home. But due to the ice has retreated away to the north, that beach is the nearest place to their feeding grounds. Some walrus have no place to rest, so they find a space away form the crowds, they climb to 80-meters cliffs, it is relatively easy for them to climb on, but it is very difficult for them to get done due to their poor eyesight. Most of the walrus fall off from the cliff and died.

But when sea ice reforms, all the order restored, animals rely on ice can live easier.

1.2 Further thoughts about videos

1.2.1 How image sensor in our computer mouse works

In this video, I understand how our mouse works, and as the videos, the DSP confirm the first picture and overlay the second on it , then it moves the second picture until the difference(subtraction of pixel values between the two images) between these image is smallest. This way may cost a lot of hashrate, because we need to move image 2 in no direction and just try it for the minimum difference. Nowadays, machine learning and deep learning is popular, maybe we can use deep learning to find the similar feature district position in two pictures, and directly get the movement distance of image2 compared to image1.

1.2.2 Living based on starch

Once I have a confusion that monk major in martial arts, they don't eat any meat, why they looks so strong and healthier. Now after watching this video, I think I get the answer. In tradition Chinese medicine structure, it says that "five grain to foster, five fruit to help, five livestock to benefit, five greens to supply"(it is not so accurate when transformed from ancient Chinese word). But we can see that in tradition Chinese, it advocates people living on starch diet, too. Also like the speaker said, it don't reject eating meat. After watching, I feel the broad and profound Chinese traditional culture again. And plan to start starch-based diet in the rest of my life.

1.2.3 Frozen world

In this video, it emphasize the importance of sea ice, I think our nature is so magical that it can produce sea ice to reflect the sun light to keep it in a low temperate. And the sea ice also have the relationship with the creatures living in the both polar ares. Sea ice maintain the balance of the ecosphere, and the balance of the ecosphere can strength its immunity. It is a good circle, which implies the wisdom of our nature. It reminds me a say in tao te ching "The tao way follows nature", it is so mysterious for us to know what tao is, maybe it is the rule human should to follow when living.

2 Pros and cons of the Shannon/Nyquist sampling method

2.1 Advantages

Not convincing. Need detailed analysis work.

The circuit the whole process use is basic circuit, don't need integral circuit or multiplying circuit, namely, its equipment is simple, it is easy build a sampling system. And I think it is more easy to understand.

2.2 Disadvantages

1. difficult to determine the bandwidth Time-unlimited signal has infinite bandwidth W, and its energy decays to zero, it's difficult for us to find a way to determine the bandwidth.

2. Gibbs phenomenon

   The reconstruction will generate errors in the place where the signal is discontinuous or derivative discontinuous.

3. lack of local adaptation

   The value of reconstruction signal $\hat{x}(t)$ depends on all samples, even samples that are far away. And the more sample we use, the loser $\hat{x}(t)$ is to $x(t)$.

4. need to know bandwidth of signal before sampling

   Before we need to know the bandwidth of signal, but how to now this bandwidth is also difficult. Though we can use an oscilloscope, but it don't work at all frequency.

3 The trajectory of my phone camera movement

3.1 Way to draw my phone movement through video

In the video about mouse, we see that our mouse compare the two adjacent picture it captured, it keep one picture fixed, and move another picture on it to find the relative movement of picture2 compared with picture1 by finding the minimum difference between two pictures when move. The most important part of this process is to find the relative movement of picture2. In our own program we can use this idea, compare two adjacent frames in our video, and find the relative movement, then we can know how our phone moves.

Supposed, we let left/right move represent y direction, forward/back represent x direction, and up/down represent z direction. Due to the mouse it cast the light on its bottom, so the adjacent pictures it gets are totally different if the mouse moves, so it fix one picture and move another picture to find the minimum difference to get the relative movement. But for our phone, we don't cast any light out, we just receive light reflects by things, and one picture have many features as the Figure 1 shows, so there are some same feature points in adjacent pictures. Thus, we can find the same feature points of two adjacent pictures and calculate the movement of these feature points like the Figure 2 shows, then we can get the movement on y and z axis of our phone. For the movement on x axis, we know that is we phone close to something we then it is more large in the picture, we we get away from some thing, then it is much smaller in the picture, so we can calculate the area of a polygon surrounded by the same feature points in two adjacent frame, but it failed as the Figure 3 shows. These same feature points we get may not just surround a polygon, it may produce cross lines. If the same region is amplified/sharked, the area of it is amplified/sharked, the important thing is that its perimeter also be amplified/sharked. So we calculate the total distance between each adjacent feature points(adjacent in list), and compare them, if it is larger than prior ones, then we know that it is closer(move forward, so in x axis it run to negative direction), else it is further.

OK to use feature points.But it c

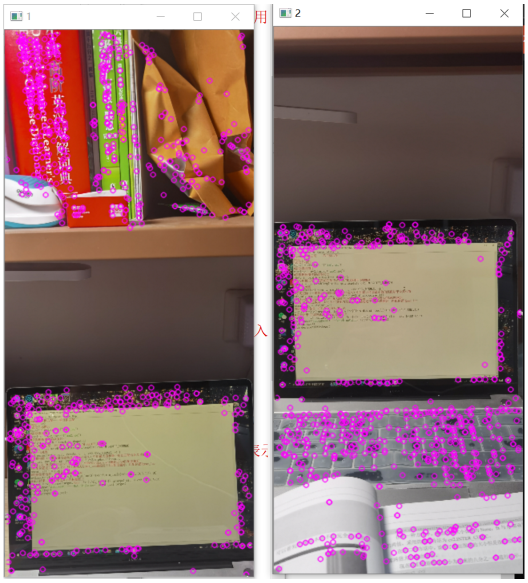

Figure 1: Feature points of two pictures(pink circle shows)

The accurate movement on z and y axis we can calculate the movement of feature points, and get the movement of how many pixels, to calculate the movement on x axis, I use a simple way, the difference between two frame is extremely small, so I calculate the ratio of distance in two frame, and thing it max can move 10 in a frame time, so we use the ratio coefficient to time the max distance, then we will get the movement on x axis.

3.2 Error analysis

First, in y or z axis, we get the movement of the common feature points that two adjacent have, the relative movement L_r is how many pixels, but the physical movement L_s of our phone is $L_r \times l$, where l means how long a pixel represent in real, it is changed constantly because our phone may move forward of back, so the physical length one pixel represents is changed, here we roughly let one pixel represent a stick length(have to do with things in our video is how far from us), but we our phone moves, l will change, so here will produce errors.

Second, in order to make our problem simple, we let the max movement on x axis is uniform between two frame, also the movement on x axis is changed, here we use a fixed max value will produce errors, if our phones move faster or slower than the speed this value corresponding, this error will be more obvious.(**To be honest, I don't really know how one pixel represent physical lengths, I think it has to do with the distance of things from our phone, the further the larger one pixel represent in real length, so in the homework, I first shoot a video and then try many times to find a value that is suitable for my phone path.**)

Good thought. This can be done, but need a bit more information. Y

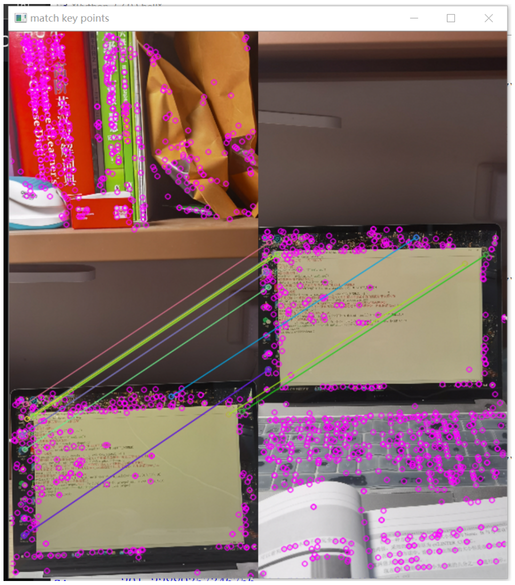

Figure 2: Match the feature points and calculate movement of these points

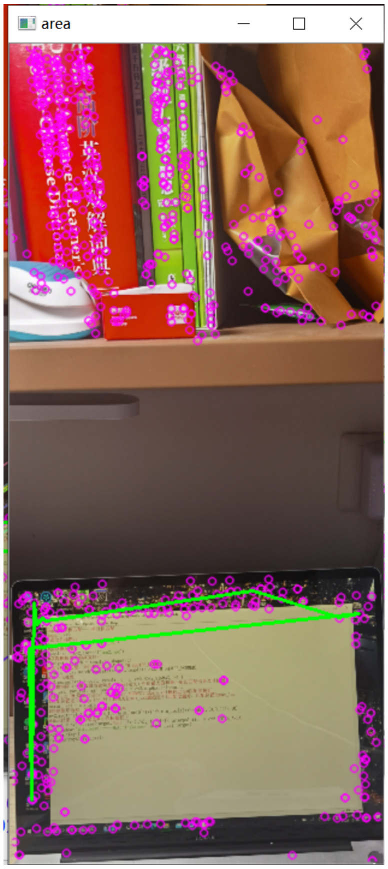

Figure 3: Area surrounded by same feature points

Third, some time it is not easy to mach feature point so it will goes wrong, because some times our phone move fast and it is difficult to get a clear frame, so the common feature points is not exist.

Last, a small rotation of our phone will produce errors. There errors we produced on the processing, so the path our phone moves is relatively right(the path our computer draw is similar to its actual movement). In the video we first move our phone right, and then forward, next left, last then up, the path as the Figure 4 shows. Due to I using a fixed value about a pixel represent how long, so in when I close to bookcase, the distance of feature points is much larger(pixel measured) but one pixel still represent a fixed length, so in the third part, move left, it has much larger error compared with real distance.(Watch the video, can know what I mean more clear)

Good.

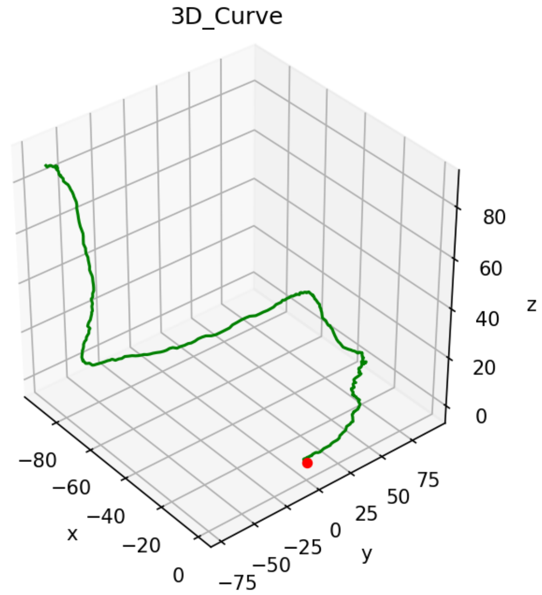

Figure 4: Path of our phone calculated by our computer()

References

Appendix A Code listings

```
import numpy as np
import cv2
from matplotlib import pyplot as plt
from math import *
from mpl_toolkits.mplot3d import axes3d
import matplotlib.pyplot as plt
import numpy as np

def get_move(img1, img2):
    sift = cv2.xfeatures2d.SIFT_create()


    img1 = cv2.resize(img1,None, fx = 1/6, fy = 1/6)
    img2 = cv2.resize(img2,None, fx = 1/6, fy = 1/6)

    gray1 = cv2.cvtColor(img1, cv2.COLOR_BGR2GRAY) #turn to gray
    gray2 = cv2.cvtColor(img2, cv2.COLOR_BGR2GRAY)

    #hmerge = np.hstack((gray1, gray2)) #joint try
    #cv2.imshow("gray", hmerge)
    #cv2.waitKey(0)
    kp1, des1 = sift.detectAndCompute(img1,None) #The key points and sift feature vectors of the image are calculated
                                            #des1 is sift feature vectors
```

```python
    #print("Number of keypoints in Figure 1: "+str(len(kp1)))
    #print(des1.shape)
    kp2, des2 = sift.detectAndCompute(img2,None)
    #print("Number of keypoints in Figure 2: "+str(len(kp2)))

    #img3 = cv2.drawKeypoints(img1,kp1,img1,color=(255,0,255)) # Draw the feature points and display them as red circles
    #img4 = cv2.drawKeypoints(img2,kp2,img2,color=(255,0,255)) #Draw feature points and display them as red circles
    #cv2.imshow('1', img3)
    #cv2.imshow('2', img4)


    # match by use BFMatcher
    bf = cv2.BFMatcher()
    matches = bf.knnMatch(des1,des2, k=2)

    pts2 = []  # key points position in image 2
    pts1 = []
    #good = [] # a key point set that key points strong relationship among two relation
    #well = []
    for m,n in matches:
        if m.distance < 0.7*n.distance:
            #good.append(m)
            #well.append([m])
            pts2.append(kp2[m.trainIdx].pt)
            pts1.append(kp1[m.queryIdx].pt)

    y = 0
    z = 0
    dis1 = 0
    dis2 = 0
    n = 1
    l = len(pts1)
    for i in range(0, l):
        y = y + pts1[i][0] - pts2[i][0] # y axis move
        z = z + pts1[i][1] - pts2[i][1] # z axis move
        if(i):                  # calculate the distance of each key points
            dis1 = dis1 + sqrt((pts1[i][0]- pts1[i-1][0]) ** 2 + (pts1[i][1] - pts1[i-1][1]) ** 2)
            dis2 = dis2 + sqrt((pts2[i][0] - pts2[i-1][0]) ** 2 + (pts2[i][1] - pts2[i-1][1]) ** 2)
    if l:
        dy = y/l
        dz = -z/l
    else:
        dy = y
        dz = -z


    if dis1 and dis2:
        n = dis2/dis1

    if n > 1:
        dx = -(n-1) * 30 #30 have to do with the sence we capture, if things is more far, it must be large
    elif n == 1:
        dx = 0
    else:
        dx = (1/n - 1) * 30

    dy = dy * 0.8       #0.8 have to do with the sence we capture, if things is more far, it must be large
    dz = dz * 0.6       #if things get closer, it need to be small

    #Honesty speaking, 30, 0.8 and 0.6 is not I calculate, I try many times and find a series that close to real length most

    #print(f'dy: {dy}\ndz: {dz}\n')

    '''
    print('dx:\t',dx)   forward or afterward
    print('dy:\t',dy)   -left or +right
    print('dz:\t',dz)   +up -down
    '''
    return dx, dy, dz


def draw_path(X, Y, Z):
    # print to check data\n\nprint first_2000
    # new a figure and set it into 3d
    fig = plt.figure()
    ax = fig.add_subplot(projection='3d')
    # set figure information
    ax.set_title("3D_Curve")
    ax.set_xlabel("x")
    ax.set_ylabel("y")
    ax.set_zlabel("z")
    '''
    ax.set_xlim(-750,750)
    ax.set_ylim(-750,750)
    ax.set_zlim(-750,750)
    '''
    plt.gca().set_box_aspect((1,1,1))
    # draw the figure, the color is r = read
    figure = ax.plot(X, Y, Z, c='g')
    ax.scatter(0,0,0, c='r')
    plt.show()

def RotateClockWise(img):  #rotate the image clockwise
  trans_img = cv2.transpose(img)
  new_img = cv2.flip(trans_img,1)
  return new_img

X = [0]                    #position for each frame
Y = [0]
Z = [0]

x = 0                      #prior or current frame position, just relatively speaking
y = 0
z = 0

cap = cv2.VideoCapture('last.mp4') #read the video we capture

last_ret, last_frame = cap.read()  #first get the prior frame
last_frame = RotateClockWise(last_frame)
if cap.isOpened():
  while True:
    ret, frame = cap.read()
    if not ret:
      break
    frame = RotateClockWise(frame)
    dx, dy, dz = get_move(last_frame, frame) #get relative position move

    last_frame = frame      #!!!!! important

    x = x + dx                              #get current frame position
    y = y + dy
    z = z + dz
    X.append(x)                             #add to position database
    Y.append(y)
    Z.append(z)
    #print(f'dx:{dx}\tdy:{dy}\tdz:{dz}') #forward or afterward


cap.release()                               #end read video
```

```
    draw_path(X, Y, Z)                           # draw path
```