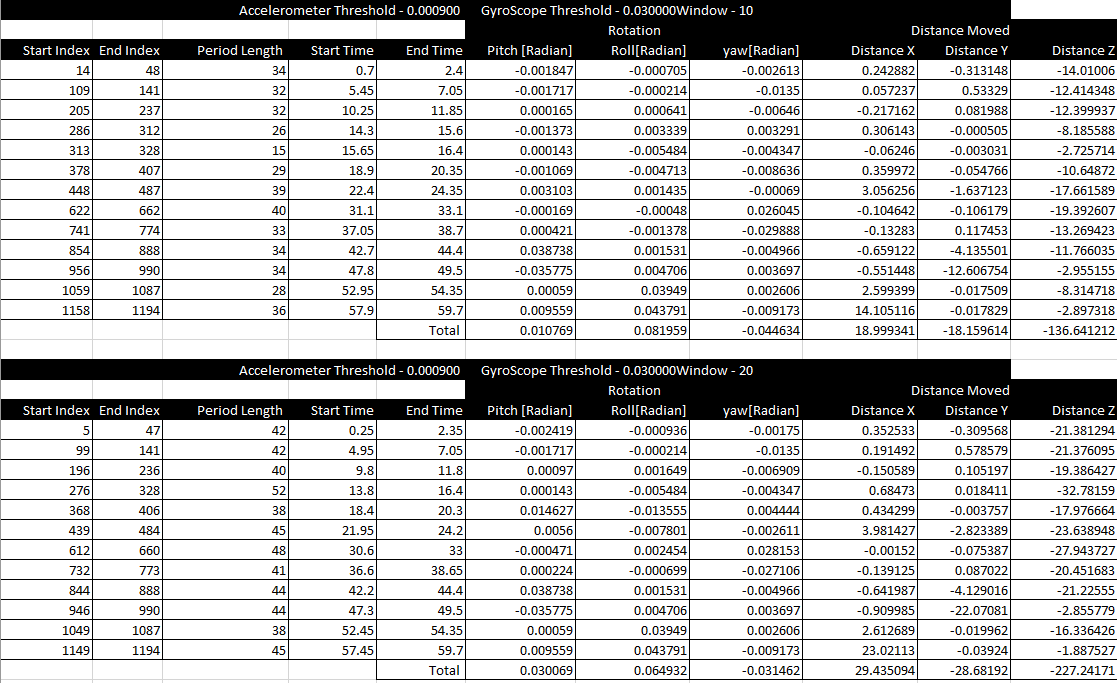
**ECE 4310/6310 Introduction to Computer Vision**

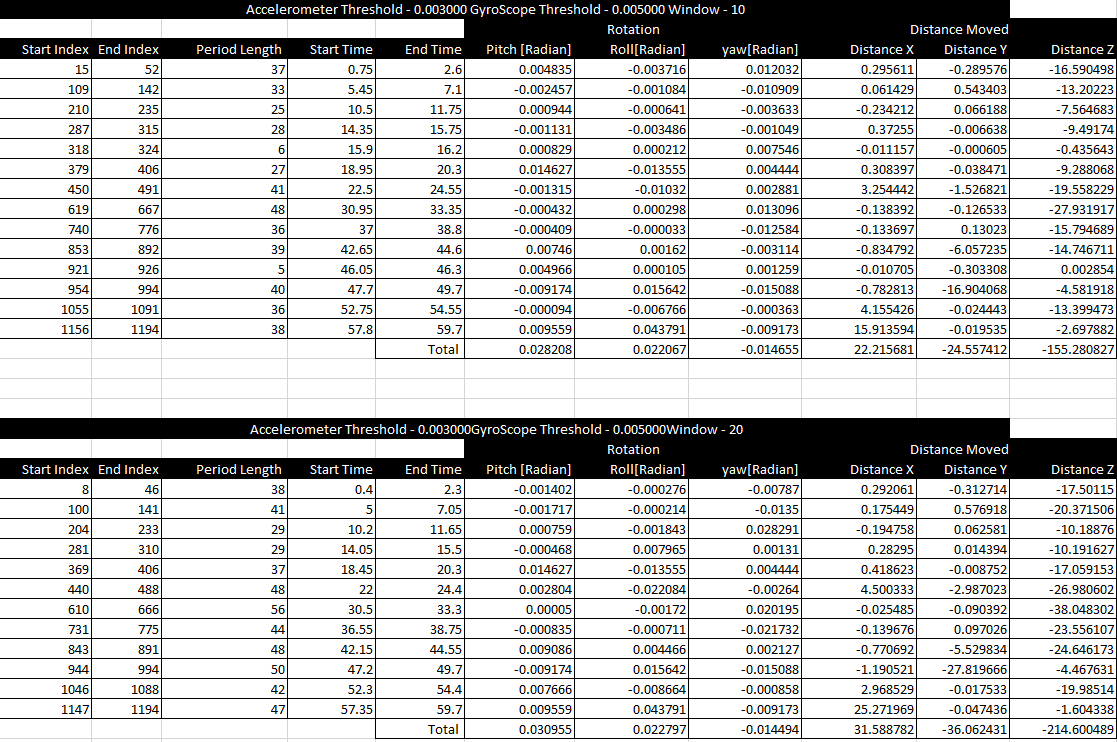
Lab #7 – Motion Tracking

Name -Pinak Kelkar

We are looking for movement in any of the axis with respect to the variance. For this we have a specified window which will give us the variance of the that window and check if it’s above the threshold. If yes, we calculate the gyroscope reading by multiplying it by the time to see the rotation with respect to each axis. For accelerometer I have taken to take the average of the velocity of the previous a new velocity with respect. To read velocity of each axis we multiple it by the gravity and the time and take the average. For some axis it was easy to calculate cause there was no movement in that axis.

The below table shows the threshold of gyro and Accelerometer data with the window used to take the variance. The rotation is in radians and the distance is in meters. We can take an example below for the first period the rotation was minimal as in not that significant. But the most noticeable movement was in Z axis for the acceleration was in -26. which means it’s around 26m in one direction. Each value is the average rotation and velocity in that period. That total distance and the rotation is calculated at the end of it. I have used multiple windows and thresholds and the output is given below.





#include <stdio.h>

#include <stdlib.h>

#include <stdint.h>

#include <string.h>

#include <math.h>

#define AccThreshold 0.003

#define GyroThreshold 0.005

#define Gravity 9.81

#define sample\_windowtime 0.05

void smooth(int Smooth\_window,int size,float \*raw,float \*smooth){

double sum=0;

for (int i = 0; i < sample\_windowtime; i++){

smooth[i] = raw[i];

}

for (int i = sample\_windowtime - 1; i < size; i++){

for (int j = 1; j < sample\_windowtime; j++){

sum += raw[i-j];

}

smooth[i] = (sum+raw[i]) / sample\_windowtime;

sum = 0;

}

}

float Var(float \*smooth, int index,int window){

float variance=0;

float mean=0;

float total=0;

for(int i=index;i<index+window;i++){

total=smooth[i]+total;

}

mean =total/window;

for(int i=index;i<index+window;i++){

variance=variance+pow((smooth[i]-mean),2);

}

variance=(variance/(window-1));

return variance;

}

int main ()

{

FILE \*ftr,\*ftr1;

char \* line = NULL;

size\_t len = 0;

ssize\_t read;

ftr=fopen("acc\_gyro.txt","r");

int size=1250;

float \*time,\*x\_acc,\*y\_acc,\*z\_acc,\*pitch,\*roll,\*yaw;

float \*Smooth\_x\_acc,\*Smooth\_y\_acc,\*Smooth\_z\_acc,\*Smooth\_pitch,\*Smooth\_roll,\*Smooth\_yaw;

int window=10;

time=(float \*)calloc(size,sizeof(float));

x\_acc=(float \*)calloc(size,sizeof(float));

y\_acc=(float \*)calloc(size,sizeof(float));

z\_acc=(float \*)calloc(size,sizeof(float));

pitch=(float \*)calloc(size,sizeof(float));

roll=(float \*)calloc(size,sizeof(float));

yaw=(float \*)calloc(size,sizeof(float));

///Smooth

Smooth\_x\_acc=(float \*)calloc(size,sizeof(float));

Smooth\_y\_acc=(float \*)calloc(size,sizeof(float));

Smooth\_z\_acc=(float \*)calloc(size,sizeof(float));

Smooth\_pitch=(float \*)calloc(size,sizeof(float));

Smooth\_roll=(float \*)calloc(size,sizeof(float));

Smooth\_yaw=(float \*)calloc(size,sizeof(float));

float \*Acc\_movement,\*gyro\_movement;

float \*vel,\*total;

total=(float \*)calloc(6,sizeof(float));

//Read Contour Points

int File\_size=0;

read = getline(&line, &len, ftr);

while ((read = getline(&line, &len, ftr)) != -1){

sscanf( line, "%f %f %f %f %f %f %f\n", &time[File\_size],&x\_acc[File\_size],&y\_acc[File\_size],&z\_acc[File\_size],&pitch[File\_size],&roll[File\_size],&yaw[File\_size]);

File\_size++;

}

///Smoothing

smooth(window,size,x\_acc,Smooth\_x\_acc);

smooth(window,size,y\_acc,Smooth\_y\_acc);

smooth(window,size,z\_acc,Smooth\_z\_acc);

smooth(window,size,pitch,Smooth\_pitch);

smooth(window,size,roll,Smooth\_roll);

smooth(window,size,yaw,Smooth\_yaw);

ftr=fopen("Smooth.txt","w");

for(int i=0;i<File\_size;i++){

fprintf(ftr,"%f %f %f %f %f %f\n",Smooth\_x\_acc[i],Smooth\_y\_acc[i],Smooth\_z\_acc[i],Smooth\_pitch[i],Smooth\_roll[i],Smooth\_yaw[i]);

}

ftr=fopen("TotalData.txt","w");

ftr1=fopen("event.txt","w");

while(window<=50){

fprintf(ftr1,"Accelerometer Threshold - %f GyroScope Threshold - %f Window - %d \n",AccThreshold,GyroThreshold,window);

total=(float \*)calloc(6,sizeof(float));

int moving\_now=0;

int Start=-1;

int end=-1;

float Now\_time;

for(int i=0;i<File\_size;i++){

////reload

vel=(float \*)calloc(3,sizeof(float));

Acc\_movement=(float \*)calloc(3,sizeof(float));

gyro\_movement=(float \*)calloc(3,sizeof(float));

Now\_time=time[i];

float Var\_xacc=Var(x\_acc,i,window);

float Var\_yacc=Var(y\_acc,i,window);

float Var\_zacc=Var(z\_acc,i,window);

float Var\_pitch=Var(pitch,i,window);

float Var\_roll=Var(roll,i,window);

float Var\_yaw=Var(yaw,i,window);

//Check if in Rest

if(Var\_xacc>AccThreshold || Var\_yacc>AccThreshold || Var\_zacc>AccThreshold){

moving\_now=1;

}

if(Var\_pitch>GyroThreshold || Var\_roll>GyroThreshold || Var\_yaw>GyroThreshold){

moving\_now=1;

}

if(Start==-1 && moving\_now==1){

Start=i;

}

if(Start!=-1 && moving\_now==0){

end=i;

}

if((i+1==File\_size) && (Start!=-1 && moving\_now==1))end=i;

if(Start>=0 && end>0 && moving\_now==0){

///acc

for(int i=Start;i<end;i++){

gyro\_movement[0]=pitch[i]\*sample\_windowtime;

gyro\_movement[1]=roll[i]\*sample\_windowtime;

gyro\_movement[2]=yaw[i]\*sample\_windowtime;

float old = vel[0];

vel[0] += (x\_acc[i] \* Gravity \* sample\_windowtime);

Acc\_movement[0] += (((vel[0] + old) / 2) \* sample\_windowtime);

old = vel[1];

vel[1] += (y\_acc[i] \* Gravity \* sample\_windowtime);

Acc\_movement[1] += (((vel[1] + old) / 2) \* sample\_windowtime);

old = vel[2];

vel[2] += (z\_acc[i] \* Gravity \* sample\_windowtime);

Acc\_movement[2] += (((vel[2] + old) / 2) \* sample\_windowtime);

fprintf(ftr,"%d %f %f %f %f %f %f \n",i,gyro\_movement[0],gyro\_movement[1],gyro\_movement[2],Acc\_movement[0],Acc\_movement[1],Acc\_movement[2]);

}

total[0]+=gyro\_movement[0];

total[1]+=gyro\_movement[1];

total[2]+=gyro\_movement[2];

total[3]+=Acc\_movement[0];

total[4]+=Acc\_movement[1];

total[5]+=Acc\_movement[2];

fprintf(ftr1,"%d %d %d %f %f %f %f %f %f %f %f \n",Start,end,(end-Start),time[Start],time[end],gyro\_movement[0],gyro\_movement[1],gyro\_movement[2],Acc\_movement[0],Acc\_movement[1],Acc\_movement[2]);

Start=-1;

end=-1;

moving\_now=0;

}////calculate

else{

fprintf(ftr,"%d %f %f %f %f %f %f \n",i,gyro\_movement[0],gyro\_movement[1],gyro\_movement[2],Acc\_movement[0],Acc\_movement[1],Acc\_movement[2]);

}

moving\_now=0;

}////Main For

fprintf(ftr1,"%d %d %d %f %f %f %f %f %f %f %f \n",Start,end,(end-Start),time[Start],time[end],total[0],total[1],total[2],total[3],total[4],total[5]);

fprintf(ftr1,"---------------------------------------------------------------------------------------------- \n");

fprintf(ftr1,"---------------------------------------------------------------------------------------------- \n");

window+=10;

}///while

}

