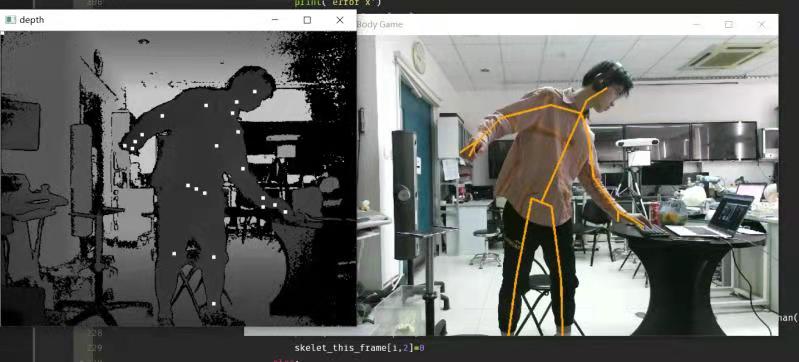
12周进度报告

李浩伟

初版代码用pygame作为界面运行

解决了ctype接口的问题，转换器在输入溢出时会出现不可except的错误，在不报错的情况下强制推出线程，预测骨架点有在rgb点外的情况，已经测试过在深度图像上的追踪点跟随情况，效果可以接受。



准备了多线程运行，目前有多个线程分别进行数据获取，数据处理，数据储存，数据外发，数据清理等不同作用。pygame在使用多线程时在200fps左右位置出现卡顿和无响应，控件反应失效。怀疑其底层线程实现和采用的多线程阻塞。因此去除所有pygame显示，增加cv2线程，用imshow进行显示，原先用pygame的clock控制帧率，现改为time.sleep强制睡眠，测试发现帧率可以超过60，在1/80 sleep的条件下几乎可以跑满80fps，接口速度和效率极高。

通过clean\_up线程处理了内存溢出的问题，总rgb储存在超过2000fps，数据量2000f\*1920pixel\*1080pixel\*4path时发生数据溢出(>16G)，系统崩溃。强制设定最高储存为200f，此时内存占用<1.5G。

经估测，笔记本难以完全支持全套运算，经过系统考虑，将数据上发服务器进行数据处理。发送包为mat，建立两个socket信道，包裹内含发送时间戳，上行数据，下行结果。

联合服务器系统调试，成功运行。有回收输出结果，下一阶段计划进行统计。网络需要针对应用场景进行进一步修改。cv显示中应当添加结果的输出。

统计有两种解决方案，一个是在服务器端统计，一个在本机统计。目前打算在本机上用简单的方法实现一些统计功能，将下发的结果进行储存，而后另开线程进行统计，产生输出，这部分输出同样需要做在CV上。

现有问题：程序在某些情况下在运行一段时间之后（一般大于1分钟）后会发生不可逆的自动弹出，目测是哪个映射器的数据越界了。但是只有在一些奇怪的条件下才会触发，而且在节点图上并不能看出效果。但是崩溃的问题不解决似乎统计也就失去了意义。

这个放在最后修复，现在好歹架子跑起来了。

下周这部分要考虑检测xyz重建算法的正确性，找几个结果数据看一下。

视频心率

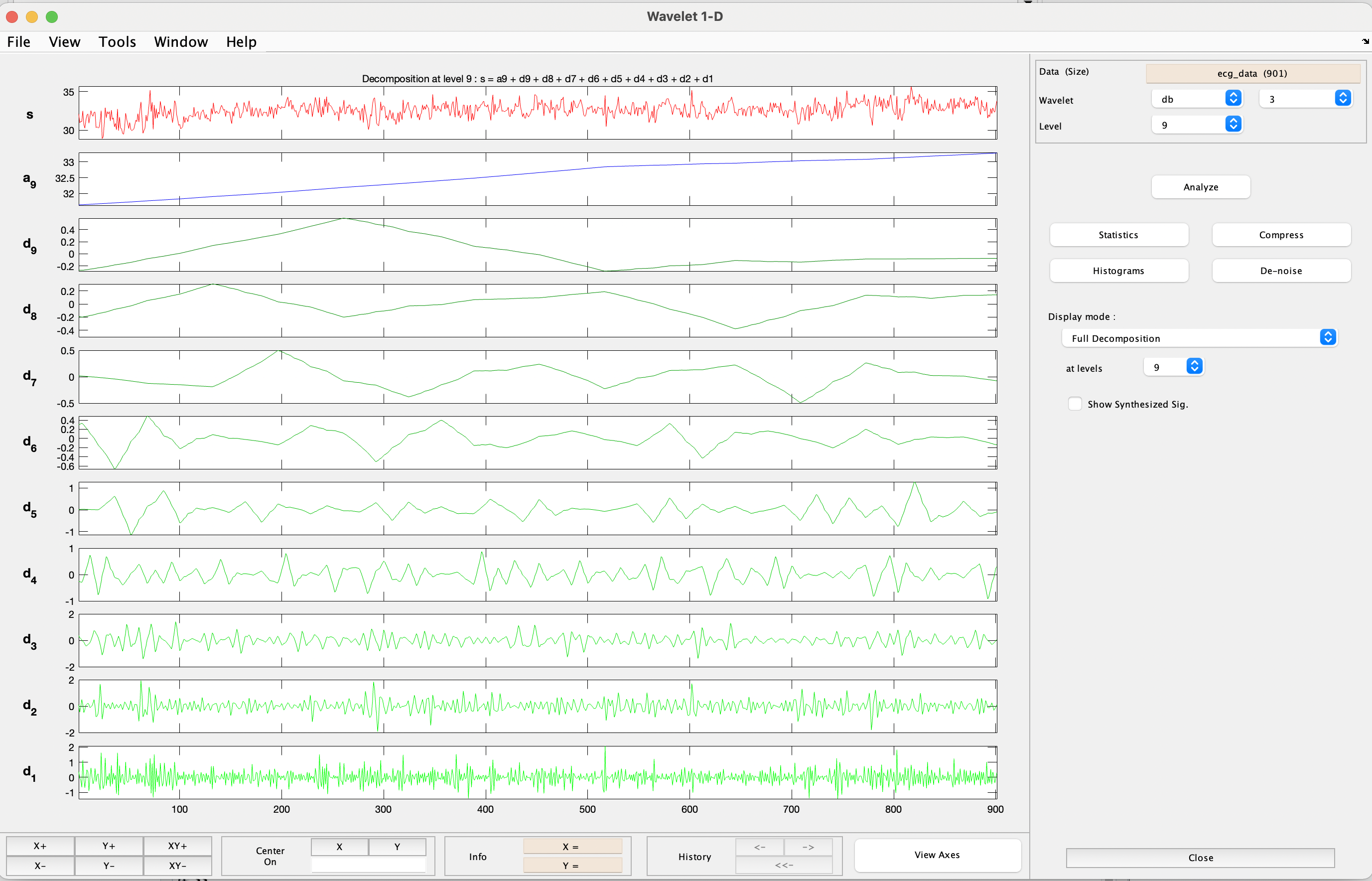
打算现在笔记本上实现再考虑迁移到Kinect上。

用笔记本摄像头编写录像代码，收集rgb信号，采用多线程实时绘图显示。

收集一段1000frame的信号

以下是小波分解的结果，视频帧率为30fps，在d5里有明显心率痕迹。

下周进行实时滤波和心率输出，以及程序迁移。



江柔蓝

1、模型上版方面，上周的MatMul算子报错得到了地平线工程师的回复，需要同步更新板端的两个动态库：libbpu\_predict.so、libhbrt\_bernoulli\_aarch64.so。但是我并不知道如何更新这两个库，因为我看到新发布的工具链已经更新过Linux端的这两个文件了，而且我也检查了开发板系统镜像的版本，发现新发布的文件里使用的版本仍然和之前相同（20210207），目前还在与地平线的工程师沟通。我看到学长的解决方法是用conv2d代替矩阵乘法运算（MatMul），但是这还需要重新修改我们的深度学习模型，较为麻烦。

2、我们与丁辉老师进行了一次沟通，老师提出可以在版上先尝试我在第十周尝试的，不通过深度学习方法、仅通过关键点坐标进行简单的姿态识别。以举手为例，具体方法是检测左臂或右臂的关节点是否高于肩膀的关节点，然后再通过一个滑动窗口对识别结果进行投票，如果连续10帧中有超过5帧都判断为举手就输出“举手”。这一部分目前能够在控制台中输出“检测到举手”的信息，但遇到的问题是，无法将这一信息用网页显示出来。之前李浩伟同学仔细查看了Xproto框架，用于接收Xstream信息→网页显示，但是难度过大（见11周进度报告）。

我们想到的方法是直接在cpp里进行文件读写操作，随后在html里用ifram加载文件并实时刷新进行显示。这周我们尝试了这种方法，但是，首先，通过fstream进行文件读写操作失败，我们猜测很可能是程序运行后，网页所在的文件夹被锁死。随后，我们发现echo命令可以在程序运行的同时进行文件写的操作，并在cpp里使用system(“ ”)进行命令行操作，但是测试后仍然以失败告终。我们询问地平线公司是否支持的开发板是否支持system(“ ”)函数，得到了肯定的答复，并建议我们再尝试一下绝对路径。下周我会继续进行这一尝试。

3、协助李浩伟、李子涵完成了Kinect姿态数据的录制（站，坐，行走，摔倒）。