

# 运动轨迹构建

Move track and spatial distribution are macroscopical features of motion which is valuable for understanding the action of targets.

## 关键步骤一. 数据采集

IMU的运动轨迹构建，数据的采集为其中首要也是重要的环节，此处要确定采集设备，数据采集方案，IMU固定位置，动作定义，及定义好各**空间坐标系**及坐标系间的**关系**。

### 采集设备简介

#### 采集设备的基本信息

本文基于低成本可穿戴式设备JY61（**维特智能科技有限公司出品**）进行数据采集，JY61尺寸为15.24mm15.24mm2mm,重量仅为1克，传输方式为有线传输，输出频率为100HZ，通过串口TTL将输出的数据传输至PC端，并通过上位机进行显示和保存。

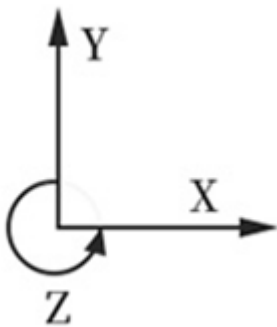
设备输出数据包括设备实时的三轴加速度计数据、三轴陀螺仪数据和欧拉角（俯仰角，翻滚角，航向角）数据。JY61模块内部集成MUP6050及STM8单片机，由于设备不包含磁力计，所以数据的采集不会受到磁场的干扰，但同时，Z轴输出的航向角只能通过陀螺仪积分作为参考值，存在漂移，输出值不是非常准确。

### IMU坐标系及世界坐标系定义

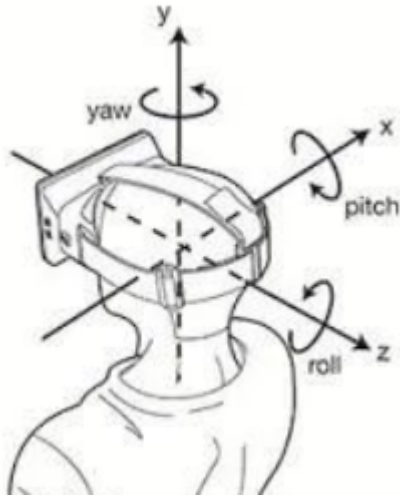
载体运动轨迹的重构中，坐标系的定义尤为关键。

#### IMU坐标系

IMU坐标系也称为载体坐标系，也就是采集设备自身的坐标系，是相对固定的，JY61模块由三轴构成，分别是X，Y，Z轴。



如图所示：若将模块竖直放置在人正前方的桌面，以人的右方向为X轴，Y轴竖直向上，垂直模块向内为Z轴。Z轴的方向定义符合笛卡尔右手系。



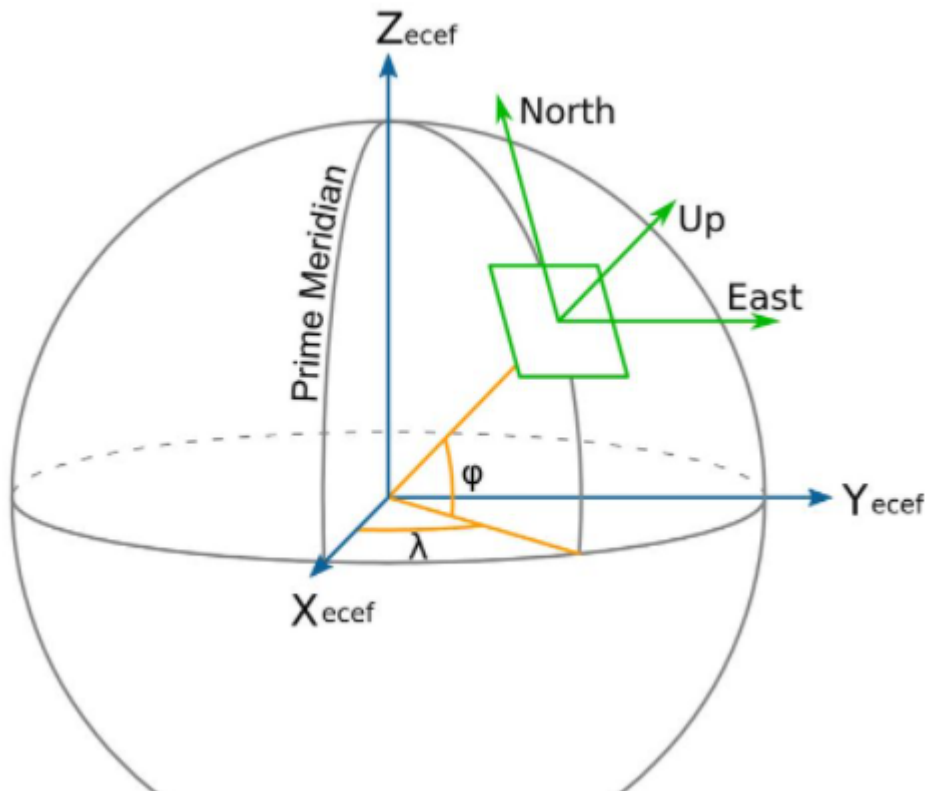
载体坐标系相对载体本身是固定的，但在运动过程中相对于世界坐标系（参考系）是在变换的。其中要注意的是，设备输出的欧拉角数据，也就是姿态角数据，是相对于初始的载体坐标系的旋转角度值，如IMU按如图方式放置，其中俯仰角（pitch）为载体绕载体坐标系X轴旋转的角度偏移值，范围为-90-90，横滚（roll）角为载体绕载体坐标系Z轴旋转的角度偏移值，同理航向角（yaw）。

**注：**不同的放置可产生不同的轴向角与欧拉角的对应，如模块水平放置，其roll和yaw的对于轴则会对调

### 世界坐标系定义（导航系、惯性系）

世界坐标系是独立于载体的坐标系，其效果相当于地图，载体的运动并不会改变世界坐标系的定义，世界坐标系相当于对载体的运动空间进行描述，为更好的建立载体坐标系与世界坐标系的联系，一般世界坐标系与载体坐标系的初始状态是一致的，因为这样可以根据姿态角建立两个坐标系间的相对运动关系。

由于实验环境基于地球环境，本文对世界坐标系相对于地球进行描述。



将地球由站心坐标系进行构建，而运动物体在地球表表面的任意位置，其世界坐标系相对于地球而言，定义如下，由于重力方向始终指向地心，则定义世界坐标系Z轴指向与重力方向完全相反，垂直地面指向天（UP），根据笛卡尔坐标系定义，世界坐标系的其他两轴指向相互垂直，其构成平面与地面平行，其中Y轴指向地球北（North），X轴指向东（East）。也就是东北天（ENU）坐标系。

## 载体坐标系与世界坐标系的相对关系

为建立两坐标系关系（旋转矩阵可对坐标内的空间向量进行相互转换，从而达到轨迹计算的目的），便于后期的轨迹重建，载体坐标系初始状态与世界坐标系保持一致，其实现可通过传感器的手动校准。

所以载体坐标系的初始状态为水平放置于桌面，Z轴朝上（up），X轴朝右，Y轴朝外。

## 传感器手动校准

设备水平放置，Y轴朝正前方，X轴朝右，Z轴垂直地面朝上，通过上位机，将此时的Z轴角度归0，此处的简单校准只是初步校准以对齐坐标系，对于实际的测量误差消除效果不大，同时简单的校准有可能进一步扩大X，Y轴输出角度的误差。

## 传感器放置

。。。自定义

本文选择放置单个传感器于前臂，避免多个传感器数据采集导致数据冗余，同时，不同的上肢动作下，前臂处的运动位移的幅度、速度、差异度都更加明显，具有更好的可区分性。

## 动作及采集

。。。自定义

参考常用运动功能评定量表(*Fugl-Meyer*)设计出四个上肢动作，和一个初始态动作。

1. 坐体位上肢初始态: 自然下垂。
2. 坐体位上肢前展: 坐在椅子上，将偏瘫侧上肢尽量水平抬高。
3. 坐体位上肢侧展: 坐在椅子上，将偏瘫侧上肢尽量向外侧抬高。
4. 坐体位前臂旋前(Forearm pronation)旋后((Forearm supination)):坐在椅子上，将偏瘫侧上肢抬高至水平位置，完成前臂旋前旋后。
5. 站立触腰: 保持站立姿势，将偏瘫侧上肢向后移至腰椎。

## 关键步骤二. 消除干扰和误差

### 基于IMU进行轨迹构建所面临的问题

如何根据IMU输出的数据对载体的运动轨迹进行有效重建？

IMU轨迹重建的基本思路为通过对三轴加速度值进行二次积分，获得三个方向上的位移，从而构建出载体的位移轨迹，其难点在于以下几方面：

1.记录的三轴加速度位移为相对于载体坐标系下的加速度数据，但是在运动过程中，载体的坐标系会发生变化，这就导致不能单纯通过三轴加速度积分来获得载体在世界坐标系下的运动位移。

2.地球存在重力，在载体坐标系变化的同时，重力加速度在不同姿态下会对载体的加速度计产生不同的影响，也就是重力加速度会施加在三轴从而使得载体的真实运动加速度值被累加或被抵消，于是消除重力加速度的影响也是一个必要环节。

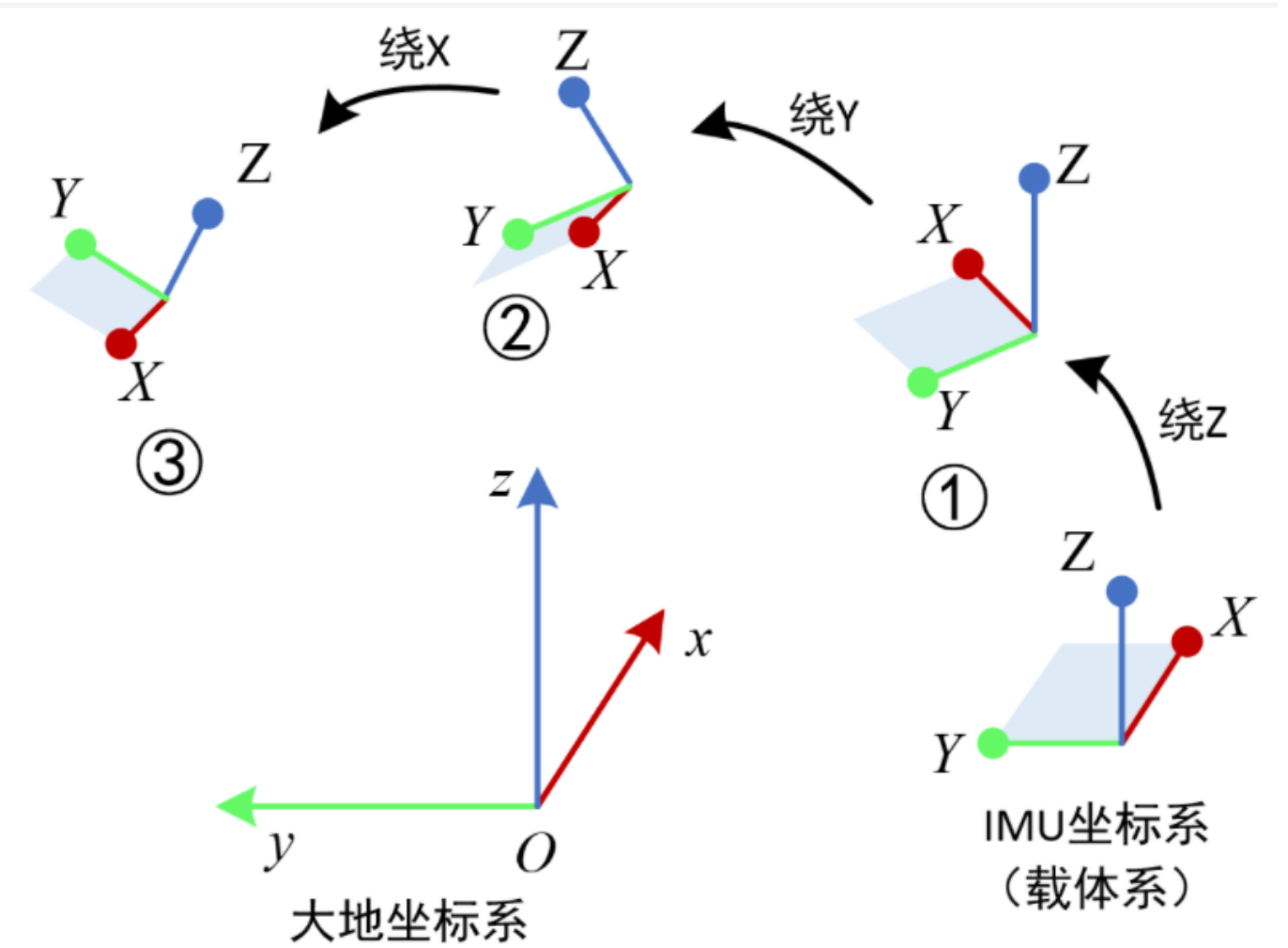
3.传感器采集的数据存在累积误差和漂移问题，如加速度计的测量噪声、陀螺仪受震动计问题影响都会产生漂移误差，且随时间变得越来越大。于是有效的误差消除也是轨迹重构的一大难点。

旋转矩阵

为求得载体在空间的轨迹位移，首先需要获得载体运动的真实加速度值，也就是获得载体在世界坐标系下的三轴加速度数值，载体坐标系和世界坐标系可看作线性空间中的基坐标系，而载体运动的三轴加速度在线性空间中可表示为一根向量（具有方向和值），直接输出三轴加速度值即为该向量在载体坐标系下的坐标（**其中还包含了重力加速度g**），而构建恰当的旋转矩阵和旋转轴，可以将同一个向量在另一个基座标下的坐标计算得到。

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = R * \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

旋转矩阵的构建



由于载体系和世界系在初始时是完全重合，这样，载体系相对于世界系的各轴偏移角度也就通过欧拉角被记录下来，通过欧拉角构建绕各轴旋转的旋转矩阵，并顺序累乘当前的三轴加速度向量，即可得到世界系下的三轴加速度向量（包含g）。<https://www.freesion.com/article/58941204625/>

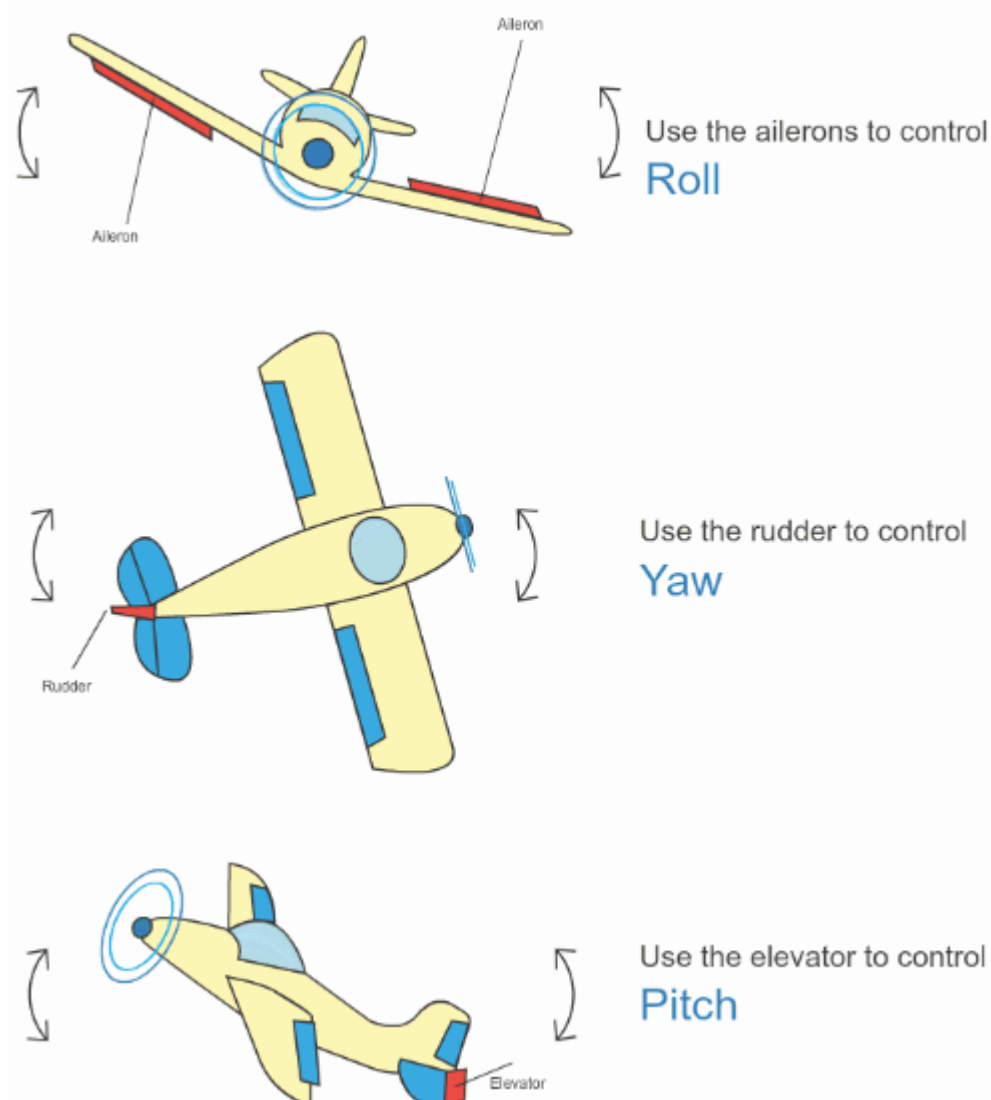
$$M_x = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos r & \sin r \\ 0 & -\sin r & \cos r \end{bmatrix} \quad M_y = \begin{bmatrix} \cos p & 0 & -\sin p \\ 0 & 1 & 0 \\ \sin p & 0 & \cos p \end{bmatrix} \quad M_z = \begin{bmatrix} \cos y & \sin y & 0 \\ -\sin y & \cos y & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

构建代码：

```
'''
    导航坐标系：东北天坐标系（ENU）：X-东， Y-北， Z-垂直平面指向天
    载体坐标系：右前上系：X-右， Y-前， Z-垂直平面指向上
    欧拉角是描述旋转的一种方式，定义姿态角为偏航（Yaw），俯仰（Pitch），横滚（Roll）
    右前上坐标系下，Z对应偏航，Y对应横滚，X对应俯仰。所以在常用的姿态旋转顺序（Yaw-
    Pitch-Roll）中，对应的旋转方式为（Z-X-Y）
    旋转轴分绕固定坐标系旋转和绕动坐标系旋转，前者称为外旋，后者称为内旋。 - Fixed
    Angles 外旋 - Euler Angles 内旋
    旋转矩阵(DCM)又称为方向余弦矩阵
'''

sin_P = math.sin(math.radians(angle_P))
cos_P = math.cos(math.radians(angle_P))
sin_R = math.sin(math.radians(angle_R))
cos_R = math.cos(math.radians(angle_R))
sin_Y = math.sin(math.radians(angle_Y))
cos_Y = math.cos(math.radians(angle_Y))
transX_martrix = np.array(
    [[1, 0, 0],
     [0, cos_P, -1*sin_P],
     [0, sin_P, cos_P]])
transY_martrix = np.array(
    [[cos_R, 0, sin_R],
     [0, 1, 0],
     [-1*sin_R, 0, cos_R]])
transZ_martrix = np.array(
    [[cos_Y, -1*sin_Y, 0],
     [sin_Y, cos_Y, 0],
     [0, 0, 1]])
tensor = np.array([acc_X, acc_Y, acc_Z]).T
result = np.dot(transX_martrix, tensor)
result = np.dot(transY_martrix, result)
result = np.dot(transZ_martrix, result)
```

偏移角是如何能构建旋转矩阵这部分不进行详细赘述，网上有大量资料，这里记录几个注意的要点。

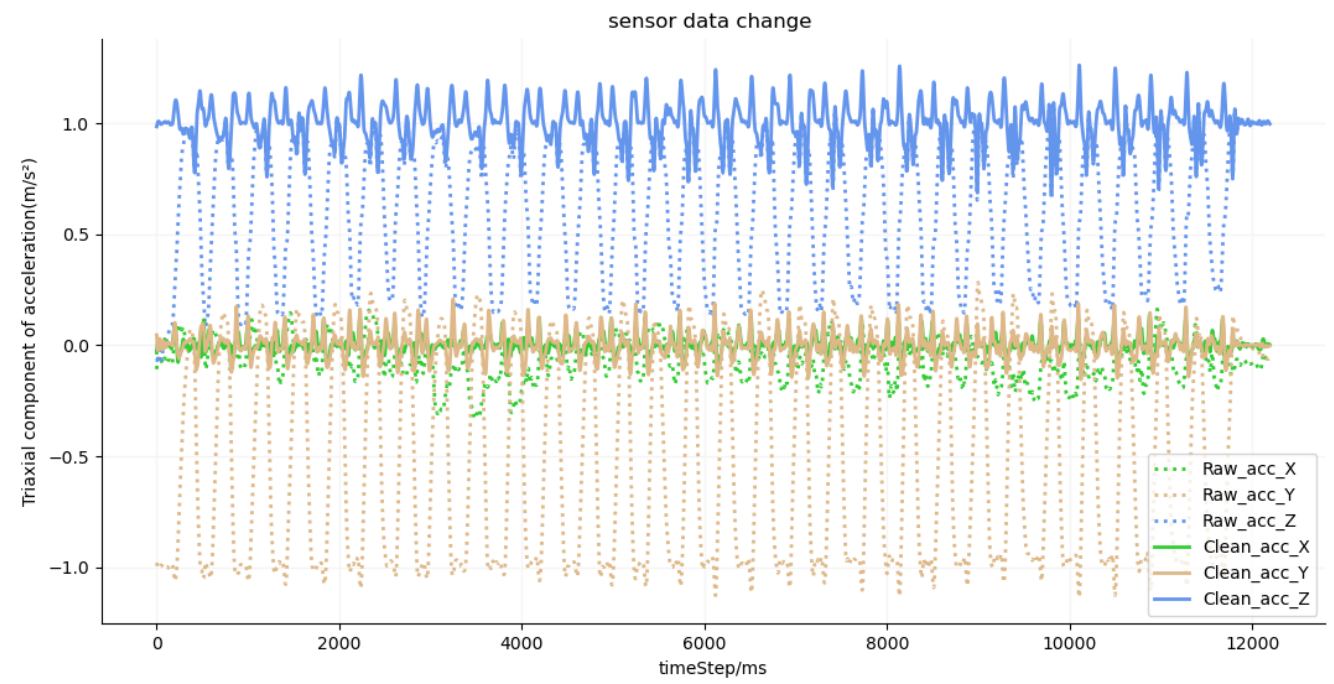


1.分清楚输出的欧拉角与 Roll、Yaw、Pitch的对应关系。分不清楚可以将IMU类比成飞机。

2.旋转顺序问题，在根据欧拉角构建出各轴的旋转矩阵后，旋转顺序即旋转矩阵左乘向量的顺序，会极大的影响最终坐标变换的结果，实验证明Z-X-Y的旋转顺序是最佳。

3.外旋还是内旋，旋转方向（正向或者反向）问题。旋转轴分绕固定坐标系旋转和绕动坐标系旋转，前者称为外旋，后者称为内旋。本文为绕固定坐标系旋转，为外旋。由于是载体系转世界系，为反向旋转。

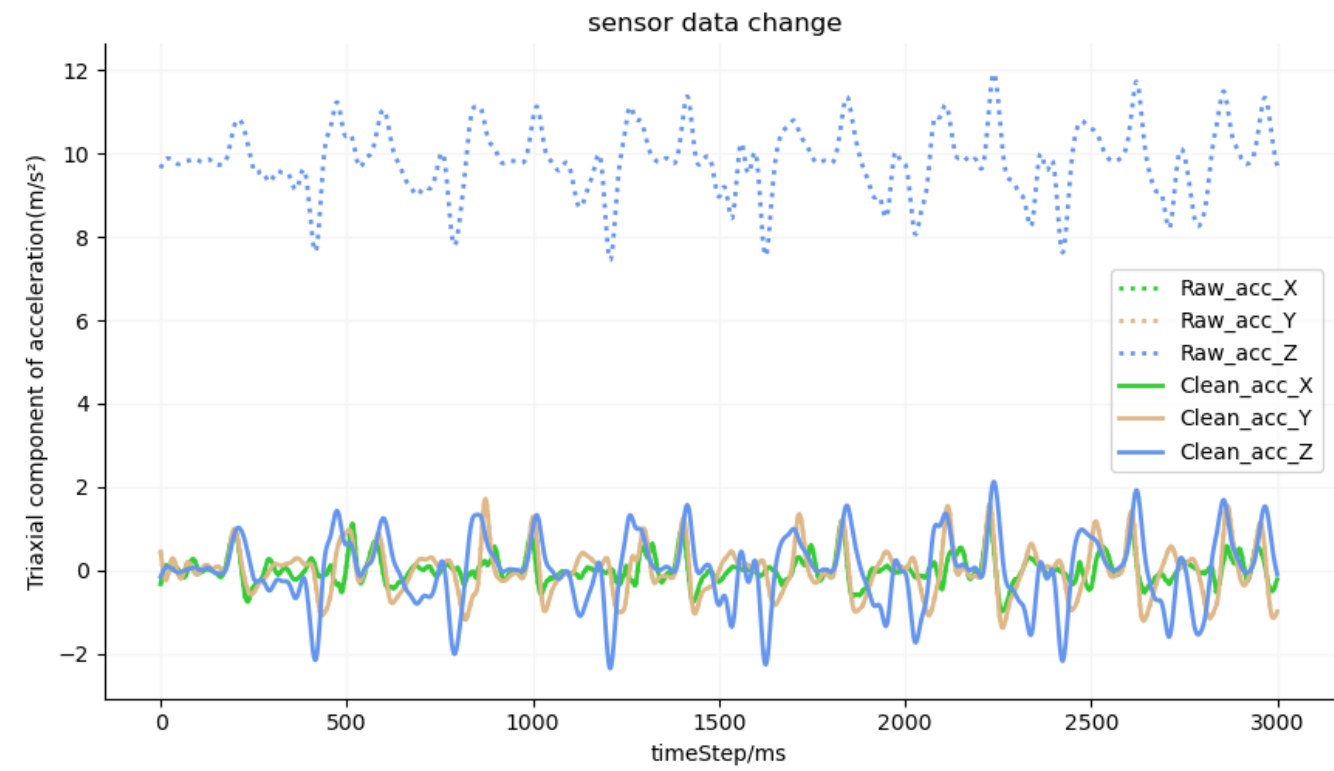
### 加速度旋转后效果对比



上图中可以看到经过旋转矩阵进行坐标变化后，Z轴数据能较为明显的体现出旋转后的效果，数据在1.0附件进行波动，这是由于其中包含了1g的重力加速度和载体在世界坐标系下的垂直方向上（Z轴）的运动加速度。

### 三轴加速度过滤

为获得IMU在世界系下的三轴运动加速度，要对世界系下Z轴方向的重力进行消去，不同纬度的重力加速度值是不同的，纬度越高，重力加速度越大。

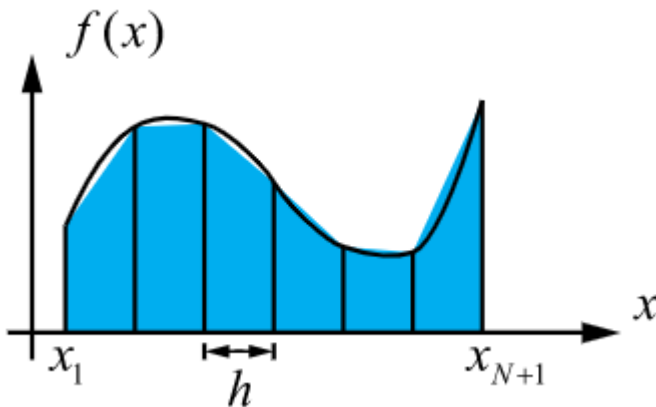


### 位移积分

对计算得到的世界系下的三轴加速度进行积分，即可得到载体的运动轨迹。

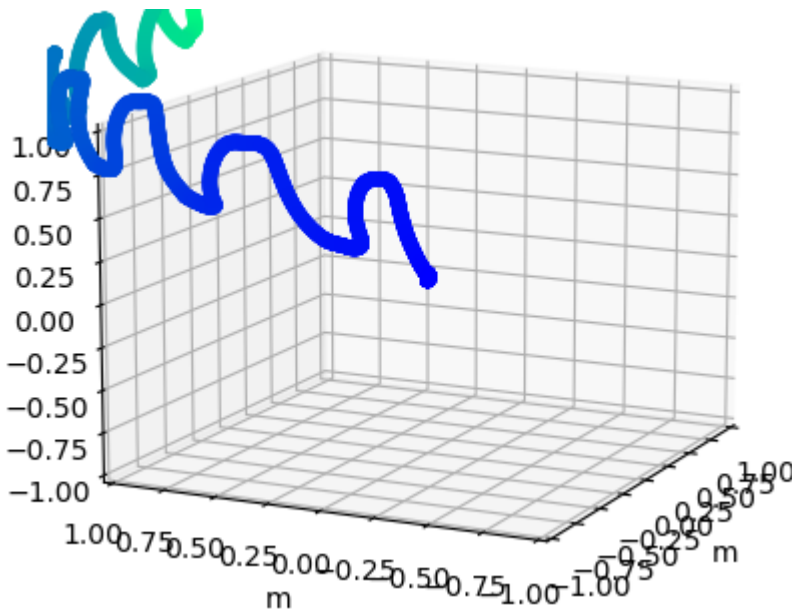
### 梯形积分

由于传感器的数据传输频率为100HZ，这意味着获取到的加速度值并非连续值，实际上是离散的数据点，没有实际的曲线对其进行拟合，于是其积分不能基于通用积分方法。本文采用数值积分对加速度进行二次积分。



公式定义为：
$$\int_a^b f(x) \mathrm{d}x \approx \sum_{i=1}^N \frac{h}{2} [f(x_{i+1}) + f(x_i)]$$

对计算得到的世界系下的三轴加速度进行积分，即可得到载体的运动轨迹？天真了 }:-<



### 零速检测（区间特征统计）

很明显，直接进行积分会导致累积误差，且时间越长，位移偏移真实位移越严重，IMU存在测量误差，纯惯性导航算法无法在较长时间的情况下计算到较为精确的运动轨迹。

人体上肢运动根据运动状态可分为运动态和静止态，通过对IMU数据统计特征，对运动态和静止态进行有效识别，防止静止态时因加速度值测量误差导致位移误差累积。

### 三特征零速检测方法

实时统计当前三轴加速度、三轴角速度，及前后区间的加速度值方差，通过设置合适的阈值判断其是否为静止态、或者说是过渡态，修改其当前状态mode值，以达到速度置零的效果。



特征一: 上肢运动在静止态时刻, 经重力消去后的当前三轴加速度值向量的模理想状态下为0, 考虑到加速度计的测量误差, 设定为在一定阈值以内判断为静止态必备特征之一。

特征二: 同理特征一的设定, 静止态下的三轴陀螺仪输出的角速度理想状态下为0, 但考虑到测量误差, 设定为在一定阈值以内作为为静止态必备特征之一。

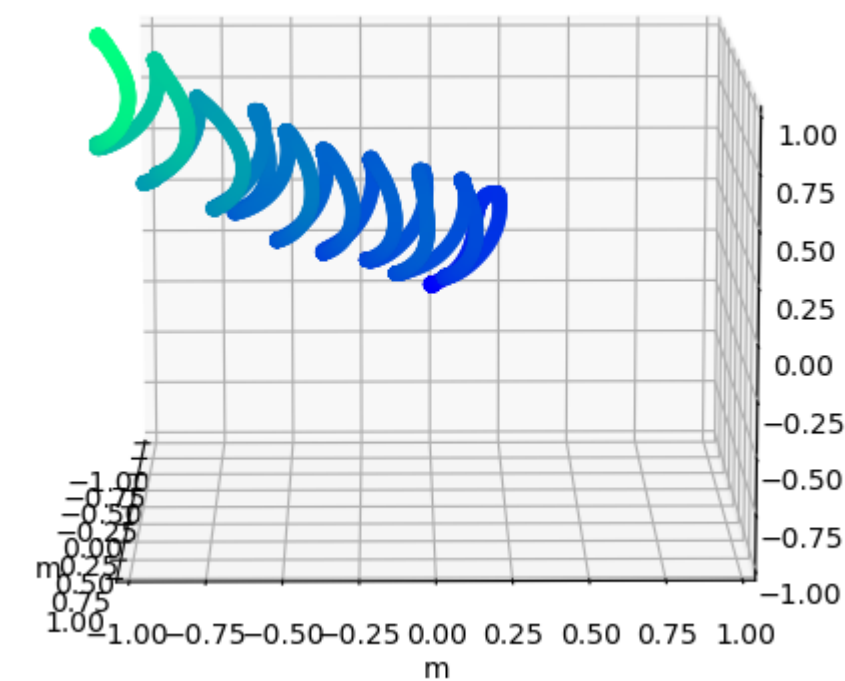
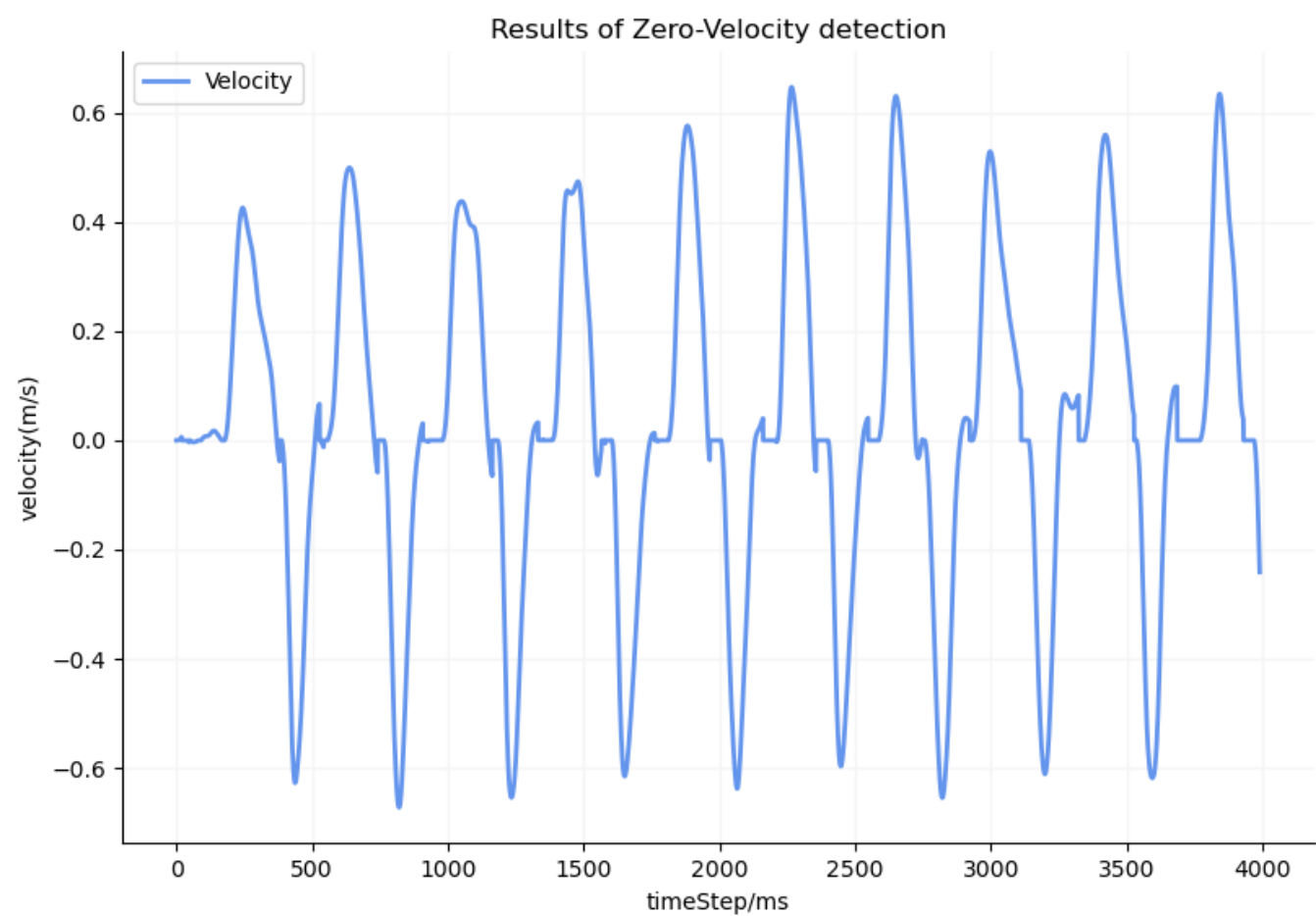
特征三: 前后区间加速度方差, 为进一步放置因测量误差过大导致的静止态时刻误判, 提出基于当前时刻前后固定区间的加速度数据的方差特征, 进一步精确判断运动的状态, 数据统计显示,

代码实现:

```
def detct(listacc_X, listacc_Y, listacc_Z, listgyo_X, listgyo_Y, listgyo_Z,
listangle_R, listangle_P, listangle_Y, time):
    mode = 0 # 初始状态mode=0, 状态为运动态
    before_acc_var = np.var(listacc_X[time-10: time])
    after_acc_var = np.var(listacc_X[time: time+10])
    sum_acc =
math.sqrt(listacc_X[time]*listacc_X[time]+listacc_Y[time]*listacc_Y[time]+listacc_
Z[time]*listacc_Z[time])
    sum_gyo =
math.sqrt(listgyo_X[time]*listgyo_X[time]+listgyo_Y[time]*listgyo_Y[time]+listgyo_
Z[time]*listgyo_Z[time])

    if abs(sum_acc)< 0.3 and sum_gyo<10 and (before_acc_var+after_acc_var) <0.1:
        mode = 1 # # 状态修改为mode=1, 状态为静止态 (过渡态), 0速
    return mode
```

0速检测效果:



零速检测在很大程度上减小了位移轨迹的不规范问题，消除了一定的测量误差，但实际依旧会有系统累积误差。此处添加实验证明。设计对比或计算检测准确率

姿态观测的周期性修正

除了测量误差导致的轨迹偏移外，还有系统的累积误差，同样会导致位移在长时间解算过程中产生累积漂移，本文结合上肢运动动作的周期性，及初始态姿态角特征，在上肢完成单次动作后，对下一次动作的位移初始位

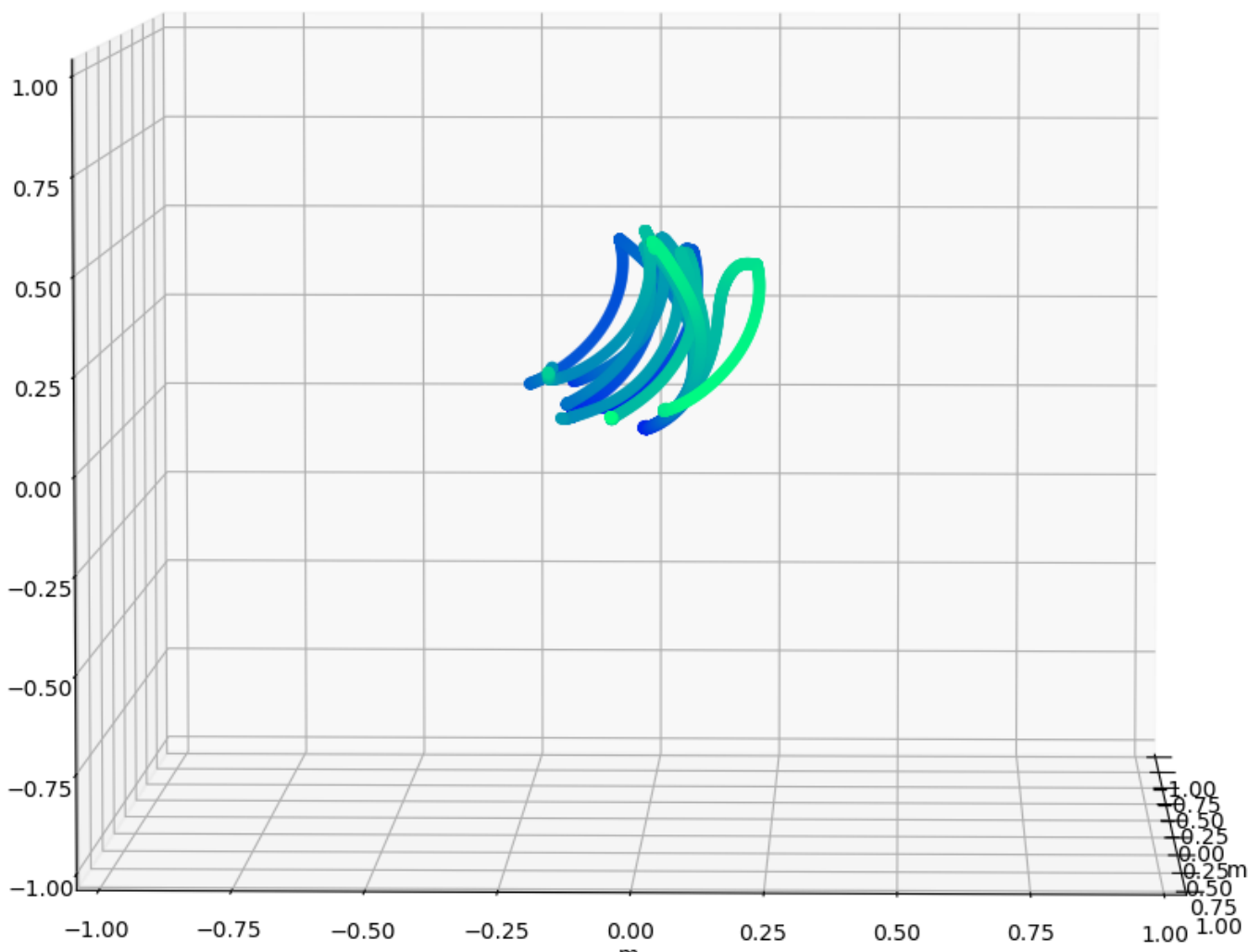
置进行修正，防止产生累积误差，由于初始态固定，即上肢自然下垂，且为静止态(速度=0)，提出在轨迹积分的过程中，实时对运动载体在静止态的姿态进行计算监测，判断其是否为恢复初始态，并对位移积分起始值进行修正，防止累积误差越来越大。

本文选择静止态时载体的Y轴正方向与重力方向夹角重合或接近的载体姿态为判断条件，判断其是否回复周期动作的初始态，具体实现为通过此时的重力加速度在Y轴上的分量，通过反三角函数求夹角值，作为判断依据。当然此处计算的数据均以静止态为前提条件。具体代码实现如下。

```
for k in range(1, len(timestep)-10):
    if V[k]==0 and math.acos(abs(Z[k]))%1 < 0.8: # 零速检测+姿态追踪, reset周期动作初始点,Z为为进行坐标变化前输出的Y轴加速度值列表。
        S[k] = 0
    else:
        S[k] = S[k-1] + ((V[k]+V[k-1])*0.01/2.0)
```

## 轨迹可视化

最终结果如下图，直观上轨迹的偏移和误差被进一步缩小，待进一步量化实验结果。



## 参考

## 博客参考

旋转举证推导: <https://zhuanlan.zhihu.com/p/183973440>

姿态解算: <https://www.freesion.com/article/58941204625/>

欧拉角旋转细节: <https://zhuanlan.zhihu.com/p/85108850>

欧拉角转换旋转矩阵参考: [https://blog.csdn.net/weixin\\_39654751/article/details/111584053](https://blog.csdn.net/weixin_39654751/article/details/111584053)

坐标系定义: <https://zhuanlan.zhihu.com/p/379614638>

梯形积分: <https://wuli.wiki/online/NumInt.html>

## 论文参考

[1] 戴洪德,张笑宇,郑百东,戴邵武,郑伟伟.基于零速修正与姿态自观测的惯性行人导航算法[J/OL].北京航空航天大学学报:1-13[2021-11-05].

[2] 吴斌,刘丽明,薛婷,张晓东.基于惯性传感器的上肢位置跟踪[J].测控技术,2018,37(06):64-67+71.

[3] Millor N , Lecumberri P , Gomez M , et al. Drift-Free Position Estimation for Periodic Movements Using Inertial Units[J]. IEEE Journal of Biomedical & Health Informatics, 2014, 18(4):1131-1137.

[4] Yu L , Xiong D , Guo L , et al. A remote quantitative Fugl-Meyer assessment framework for stroke patients based on wearable sensor networks[J]. Computer Methods and Programs in Biomedicine, 2016, 128(C):100-110.