# 一些统一的定义

* 姿态的定义：

存储顺序：俯仰、横滚、航向

俯仰角（）——运载体纵轴与水平面之间的夹角，在垂直面中测量，抬头为正；(-pi/2~pi/2)

横滚角（）——运载体横轴与水平面之间的夹角，在横截面中测量，左边抬起为正；(-pi/2~pi/2)

航向角（）——运载体纵轴与北向轴之间的夹角，在水平面内测量，顺/逆时针为正；（-pi~pi）（0~2\*pi）

# 关键参数

## visualInputData：视觉输入到导航流程的数据

第一层格式为结构体，成员如下所列。每个成员都是[1\*N ] cell/double 数组， N为时刻数，每个cell为一个时刻前后4副图的匹配成功特征点。

* leftLocCurrent：左相机当前图匹配成功的特征点，[1\*N ] cell。每个cell为一个时刻：[n\*2]（注意一行的顺序为[y,x]，程序中调用时可能需要转置），n为该帧前后4副图匹配成功的特征点个数。

5.16 欲 将数据改成 [x,y]顺序

* rightLocCurrent：右相机当前图匹配成功的特征点，[1\*N ] cell。
* leftLocNext：左相机下时刻图匹配成功的特征点，[1\*N ] cell。
* rightLocNext：左相机下时刻图匹配成功的特征点，[1\*N ] cell。
* matchedNum ：匹配成功特征点个数 ，[1\*N ]double
* aveFeatureNum ：该时刻前后4副图特征点个数的平均值（未做匹配时） ，[1\*N ]double
* frequency：图像采集频率
* cameraSettingAngle：相机安装角，rad，俯仰、横滚、航向
* VisualRT ，[1\*N ]Struct（仅在已知时存在，纯视觉仿真后将RT输出添加到visualInputData中）包含如下两个参数：

visualInputData. VisualRT .Rbb [3,3,:]。Rbb(:,:,1)表示1到2时刻的旋转（注意第一个数据就有效，不是空的），长度正常时比图片少一个，但是有时可能没有严格遵守，使用时注意。

visualInputData. VisualRT .Tbb [3,:]

（1）

visualInputData.leftLocCurrent = leftLocCurrent;

visualInputData.rightLocCurrent = rightLocCurrent;

visualInputData.leftLocNext = leftLocNext;

visualInputData.rightLocNext = rightLocNext;

visualInputData.matchedNum = matchedNum;

visualInputData.aveFeatureNum = aveFeatureNum;

（2）

leftLocCurrent = visualInputData.leftLocCurrent ;

rightLocCurrent = visualInputData.rightLocCurrent ;

leftLocNext = visualInputData.leftLocNext ;

rightLocNext = visualInputData.rightLocNext ;

matchedNum = visualInputData.matchedNum ;

aveFeatureNum = visualInputData.aveFeatureNum ;

* RTError：RTError. TbbErrorMean，RTError. TbbErrorStd，RTError. AngleErrorMean，RTError. AngleErrorStd

## VOResult /VOSimuResult : 视觉里程计的最终导航结果

VOPosition =; 位置 [3\*n]

VOAttitude =; 姿态（直接绘图的）

VOVelocity =; 速度

## trueTrace：真实轨迹参数

trueTrace.planet=’e’/’m’ （地面/月面）

trueTrace.frequency = frequency;以上数据的频率(HZ)

trueTrace. initialPosition\_e：初始绝对大地位置 [3\*1] （经度/°、纬度/°、高度/m）（仅在涉及惯导用）

trueTrace.initialPosition\_r：初始导航系（世界系）位置[3\*1]（m）（惯导和视觉导航均以此作为导航解算的位置初值，轨迹发生器生成的初始位置也是这个）（设为0，不可更改）

trueTrace.initialVelocity\_r：初始导航系（世界系）速度[3\*1]（m/s）（惯导和视觉导航均以此作为导航解算的速度初值，轨迹发生器生成的初始位置也是这个）

trueTrace.initialAttitude\_r：初始姿态（相对初始时刻的地理系/世界系） [3\*1] （俯仰、横滚、航向）（rad）（任何导航均需设置）。取初始时刻的地理系为世界系

已知真实轨迹时还有：

trueTrace.position ：位置（m） [3\*n]。第一列即initialPosition\_r

trueTrace .lat\_lon\_alt：经度;纬度;高度

trueTrace.attitude :姿态（俯仰、横滚、航向）（rad）。第一列即trueTrace.initialAttitude\_r

trueTrace.velocity ：速度

由轨迹发生器生成时还有：

trueTrace.f\_IMU ：轨迹发生器生成的理想 IMU比力数据

trueTrace.wib\_IMU ：轨迹发生器生成的理想陀螺仪数据

trueTrace.trueTraceInput包含如下6个成员（轨迹发生器5个输入和轨迹名）

trueTraceInput.initialPosition\_r：初始导航系（世界系）位置[3\*1]（m）（惯导和视觉导航均以此作为导航解算的位置初值，轨迹发生器生成的初始位置也是这个）

trueTraceInput.initialAttitude：初始姿态 [3\*1] （rad）（任何导航均需设置）

trueTraceInput.initialVb = initialVb;初始速度，载体相对地理系的速度在本体系下的分解（老师的版本好像只能适用于Vb恒定的情况）

trueTraceInput.realTimeAcc = realTimefb; 实时驱动加速度，不平衡的驱动加速度在本体系下的分解（老师的版本只能适用于fb恒定的情况）

trueTraceInput.realTimeW = realTimeWb; 实时姿态角变化率在本体系下的分解（老师的版本只能适用于Wb恒定的情况）

trueTraceInput.frequency = frequency;仿真数据频率(HZ)

trueTraceInput. runTimeSec = runTimeSec; 仿真时间(s)

trueTraceInput.traceName = traceName;

TrueTrace.InitialPositionError ： 初始位置误差（m）

TrueTrace.InitialAttitudeError ：初始姿态误差（rad）

* trueTrace. dataSource =’kitti’：kitti的数据

## cameraCalib ：相机标定的10个参数

T = calibData.T;

alpha\_c\_left = calibData.alpha\_c\_left;

alpha\_c\_right = calibData.alpha\_c\_right;

cc\_left = calibData.cc\_left;

cc\_right = calibData.cc\_right;

fc\_left = calibData.fc\_left; [2\*1] [fc\_left\_x; fc\_left\_y]

fc\_right = calibData.fc\_right;

kc\_left = calibData.kc\_left;

kc\_right = calibData.kc\_right;

om = calibData.om; % rad

## imuInputData ：IMU数据（含噪声）

imuInputData.frequency

imuInputData.dataSource=’kitti’;

imuInputData.runTime; isPicTime .isPicTime(0/1：是否对应图片)

imuInputData.f ： 加计数据 [3\*n]

imuInputData.wib ： 陀螺仪数据

imuInputData.f\_noise ： 加计噪声（只在IMU数据仿真生成时存在）

imuInputData.wib\_noise ： 陀螺噪声（只在IMU数据仿真生成时存在）

imuInputData.pa ： 加计常值偏置（m/s^2）

imuInputData.na ： 加计随机标准差

imuInputData.pg ： 陀螺常值偏置

imuInputData.ng ： 陀螺随机标准差

imuInputData.flag ： 数据类型（’sim’或’exp’）

imuInputData.driftResult :真实噪声绘图格式结果

imuInputData.computerTime ：计算机时间，便于和相机时间匹配

## NavFilterParameter：导航滤波参数

NavFilterParameter.R\_ini\_augment\_dRdT: augment\_dRdT方法的初始R阵

PInitial:初始P阵

QInitial:初始Q阵

RInitial:初始R阵

## 视景仿真参考系 与 导航解算坐标系

导航坐标系：r

视景仿真参考系：s

r（导航系-世界坐标系） 为第一副图时左相机所在的摄像机坐标系。r 的原点在第一幅图处，角度与s相同（注意不是与第一幅图相同），顺序不同：

位置顺序：Xr=-Xs（向右）, Yr=-Zs（向前）, Zr=-Ys（向上）





姿态顺序：Attitude\_r=[俯仰，倾斜，偏航]

Attitude\_s=[偏航，俯仰，倾斜]





## integMethod：组合导航方法

### simple\_dRdT

对应师姐毕业论文（3.4）

简化的状态模型（不扩维），dRdT作为量测

X=[dangleEsm;dVel;dPos;gyroDrift;accDrift]









### 2）augment\_dRdT

对应师姐毕业论文（3.2）







