

# 基于自身坐标系的空间旋转与位移计算

## 问题描述

### Teil 1

现在有一个"无人机系统", 这个系统非常简单, 只需要给定该系统的三个位置坐标  $(px, py, pz)$ , 三个旋转角度(欧拉角) $(rx, ry, rz)$ , 这个无人机就会移动到该位置上, 中间的执行时间为0. 现在, 出于某种特殊的要求, 这六个自由度的变量将会映射到(0,1)上, 既:

$$px : [-1000\text{ m}, 1000\text{ m}] \rightarrow [0, 1]$$

$$py : [-1000\text{ m}, 1000\text{ m}] \rightarrow [0, 1]$$

$$pz : [-1000\text{ m}, 1000\text{ m}] \rightarrow [0, 1]$$

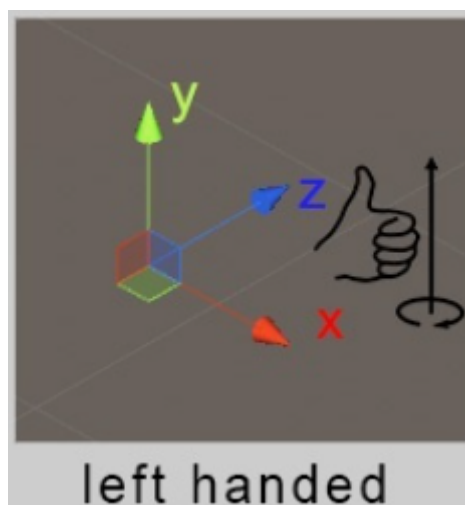
$$rx : [-180^\circ, 180^\circ] \rightarrow [0, 1]$$

$$ry : [-180^\circ, 180^\circ] \rightarrow [0, 1]$$

$$rz : [-180^\circ, 180^\circ] \rightarrow [0, 1]$$

现在, 中间值 0.5 代表着位移和旋转的 0

请注意, 这个系统现在位于左手坐标系下



### Teil 2

对于无人机系统, 现在有一个输入设备(见下图), 这个设备具有六个自由度的输入变量, 正好对应  $(px, py, pz)$  与  $(rx, ry, rz)$ , 不同的是, 这个设备基于右手坐标系(见下图), 并且输入数据的格式为

$$px : [-1, 1]$$

$$py : [-1, 1]$$

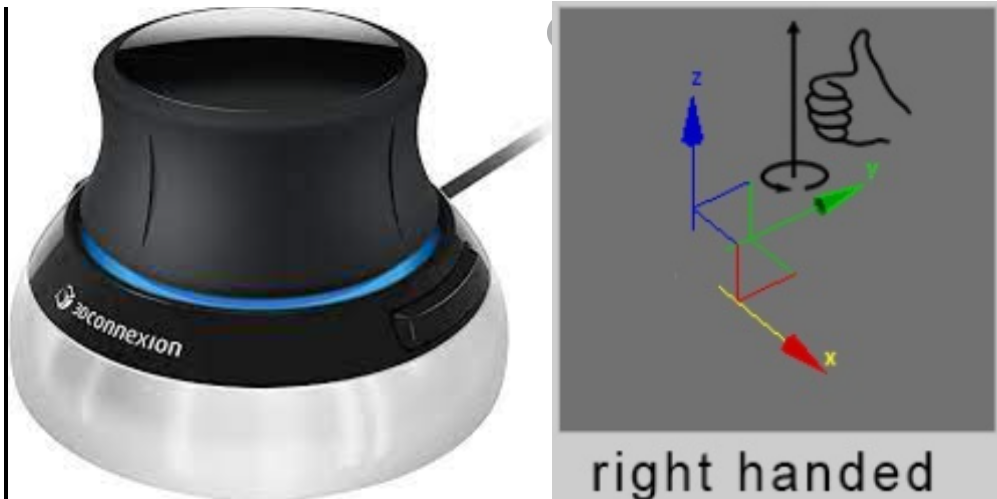
$$pz : [-1, 1]$$

$$rx : [-1, 1]$$

$$ry : [-1, 1]$$

$$rz : [-1, 1]$$

这里的数值只代表传感器最大的位移, 并不映射物理量, 现在, 中间值 0 代表着位移和旋转的 0



### Teil 3

一个纯粹的想法是, 通过读取输入设备的值来控制无人机在三位空间中的导航. 由于输入值一一对应, 很容易想到的是, 以时间间隔  $T_s$  采样输入值, 将输入数值作为无人机当前时刻与上一个时刻之间位移和旋转的两个误差向量  $diff\_pos$  和  $diff\_rot$ , 通过和一个合理的缩放常数  $k_p$  相乘后直接与无人机上一刻的位置相加, 即可得到无人机当前时刻的位置.

既(这里只展示了pos计算过程):

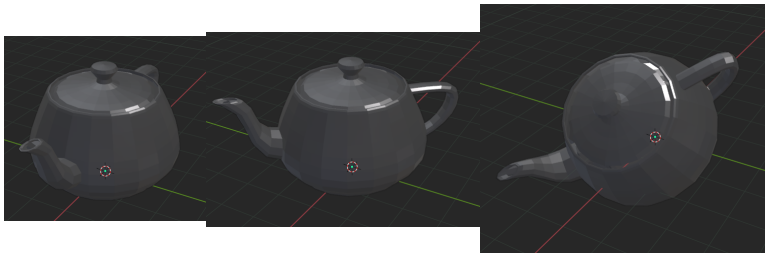
$$pos(\text{当前时刻}) \begin{pmatrix} px \\ py \\ pz \end{pmatrix} = pos(\text{上一时刻}) \begin{pmatrix} px \\ py \\ pz \end{pmatrix} + k_p * diff\_pos(\text{当前时刻}) \begin{pmatrix} px \\ py \\ pz \end{pmatrix}$$

但是这样的系统在实际运用中有很明显的问题, 既输出结果并没有受到机身当前旋转的影响, 也就是说, 输入变量  $diff\_pos$  和  $diff\_rot$  是根据地面参考系给出的, 而并非自身参考系. 对于飞行员来说, 他想让他的操作变得直观, 也就是说, 当他向左旋转摇杆时, 他所看到的画面无论在飞机处于何种旋转状态下永远会向左旋转. 位移也是同理.

下面给出了两张图来阐述根据自身坐标系和根据时间坐标系旋转的不同. 操作同样为:

1. 向左旋转45°
2. 继续向下旋转45°

世界坐标系下:



自身坐标系下:

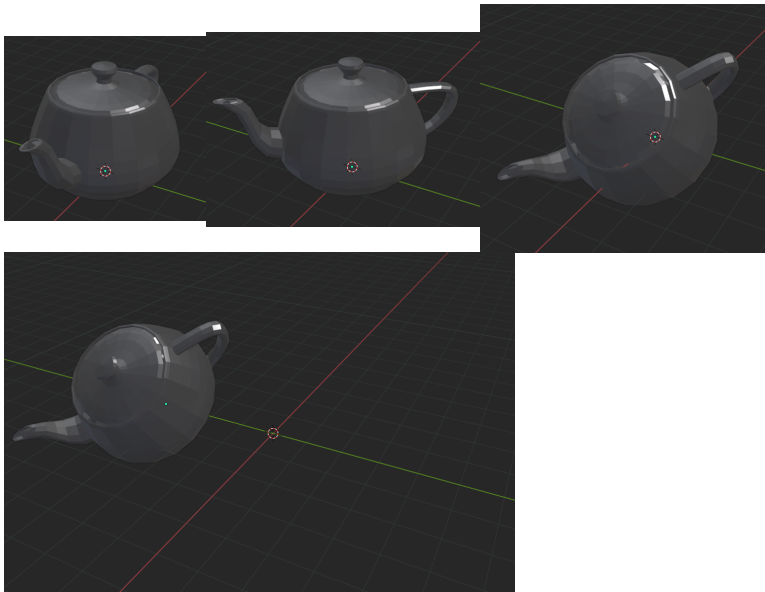


可以看出旋转如何影响了姿态的判断

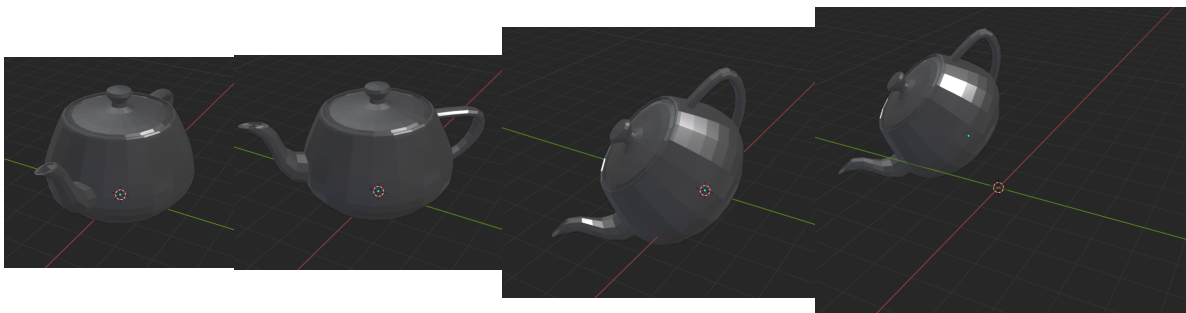
对于位移, 同样的道理

1. 向左旋转45° 2. 继续向下旋转45° 3. 向左位移

世界坐标系下:



自身坐标系下:



可以看出旋转如何影响了姿态的判断

现在, 通过使用无人机当前旋转计算出的旋转矩阵对误差向量进行合理的变换, 应该就可以得出以自身参考系为准的误差.

# 求解

---

构造两个函数  $f(diff\_pos)$  和  $f(diff\_rot)$  他们分别接收误差向量, 输出下一个时刻无人机的对应旋转和位置姿态数据(Teil 1 中的格式)

Tips: 对于旋转的处理, 不需要考虑任何数据之间的转化算法, 如四元数和欧拉角的转换或者欧拉角和旋转矩阵之间的转换, 已经有现成的算法.

感觉还有些地方没说清楚, 没看懂再问我, 难以理解的地方我可以演示