

综述：

使用 dronesim 的 projection 功能会用到两个函数，installcamera 和 projection。installcamera 是在整个程序开头的地方调用来设置 drone 上摄像头方向的。（整个训练程序中只需要调用一次，不需要在每次 reset 中重新调用）。然后每次使用 projection 功能调用 projection 函数。

函数说明：

installcamera 函数：

dronesim.installcamera(installori ,F,H,FOVnear,FOVfar):

输入:

1. installori 一个 list: [roll,pitch,yaw], 代表 drone 上摄像头的方向，单位是角度。摄像头的初始角度在 drone 的屁股后面，如图 1。如果你想让摄像头朝着 drone 的红箭头正方向，也就是 x 轴正向，就把摄像头绕着 z 轴转 180 度，也就是 $\text{yaw} = 180$ ：

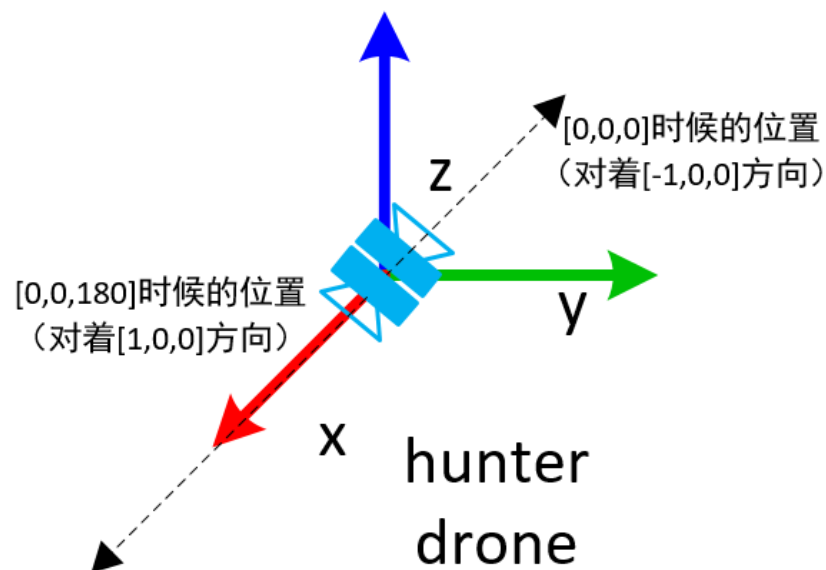


Figure 1 installori 示意图

2. F,H,FOVnear,FOVfar 四个浮点数，分别代表 FOV 的两个张角(in degree)和视锥前界面和视锥后界面的位置，如下图 figure2 所示. Philippe 推荐的参数是 $F = 110^\circ$, $H = 63^\circ$, $\text{FOVnear} = 0.01$, $\text{FOVfar} = 500$

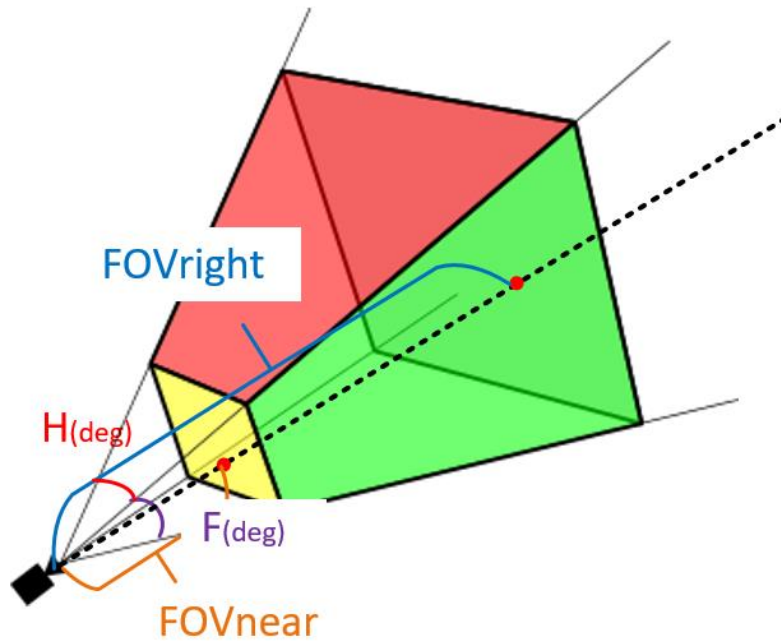


Figure 2 $F, H, FOV_{near}, FOV_{far}$, 示意图

projection 函数

`dronesim.projection(pos_hunter, ori_hunter, pos_target, w, h)`

输入参数：

`pos_hunter`: hunter 的位置 list ($[x, y, z]$)

`ori_hunter`: hunter 的 orientation list ($[roll, pitch, yaw]$)

`pos_target`: target 的位置 list ($[x, y, z]$)

`w, h` : double: 屏幕的 width 和 height

返回值：`u, v, inscreen`:

`u, v` 是 projection 得到的 drone 在图像中的像素坐标，`inscreen` 是一个 bool 用以指明点是否超出范围。这里要注意下，`inscreen` 返回 false 说明 drone 在摄像头背面或者在一些极端位置 (`u, v` 超大甚至是 none) 这个时候对应的是摄像头看不到 target。但是 `inscreen` 返回 true 的时候，`u, v` 返回一些正常的数值，但不代表 camera 一定能看到 target，之前的范围判断该有还是得有

然后放一个空间坐标到像素坐标的对应表，你可以对着做下矫正 (/build/test3.py)：

输入代码: `installcamera([0,0,180],110,63,0.01,500)`

(这个时候 FOV 的视锥前界面的 `up` 和 `right` 坐标分别是，0.035， 0.022)

返回: 0, 800, true

也就是说, hunter 在 $[0,0,0]$ 位置, 方向 $[0,0,0]$,摄像头安装方向 $[0,0,180]$. target 在 $[0.01, 0.022, 0.035]$ (这是视锥前界面的左上角) 得到的投影点在 $[0,800]$,并且实际上是正常的 (inscrean = true)

如下图：

