中级挑战任务1—— MPU6050姿态解算

题目

MPU6050姿态解算

- 利用MPU6050获得的数据，通过姿态计算得到Pitch Row Yaw 姿态角，并将数据展示到OELD显示屏幕上

知识笔记

1. MPU6050姿态解算是将加速度和角速度的数据转换为姿态（如俯仰、横滚和偏航角）的过程，常见的MPU6050姿态解算方法有：

- 欧拉角法
  - 加速度计计算Roll和Pitch
  - 陀螺仪积分计算Yaw（陀螺仪积分存在累积误差，长时间使用会导致Yaw角漂移）
- 四元数法（硬件解算）
  - 获取四元数：利用MPU6050自带的DMP，移植DMP库，调用函数可直接获取四元数。
  - 转换欧拉角：将更新后的四元数转换为欧拉角
- 四元数法（软件解算）
  - 读取原始数据：通过I2C读取MPU6050的加速度和角速度原始数据。
  - AD值转化四元数（欧拉角，旋转矩阵，矢量姿态更新等）
  - 转换欧拉角：将更新后的四元数转换为欧拉角
- 卡尔曼滤波法
  - 利用线性系统状态方程和观测数据，通过递推算法对系统状态进行最优估计，能够有效地抑制噪声和干扰，提高姿态解算的精度和稳定性。

加速度计在静止时刻，根据感受到的重力加速度，只能计算出roll和pitch角，并且具有静态稳定性。而陀螺仪是对时间间隔内的角速度积分，得到每一次的角度变换量，累加到上一次的状态上，得到新的姿态角，可以计算roll、pitch、yaw三个角，并且具有动态稳定性。

实现步骤

四元数法（软件解算）

> 截图（有改动）来自--惯性导航(第三版)(秦永元编著) 9.2姿态更新计算的四元数算法

预备知识点:

1. **四元数是什么?**

顾名思义,四元数是由四个元构成的数:

$$Q(q_0, q_1, q_2, q_3) = q_0 + q_1\boldsymbol{i} + q_2\boldsymbol{j} + q_3\boldsymbol{k} \qquad (9.2.1)$$

式中q0，q1，q2，q3是实数,i，j，k既是互相正交的单位向量,又是虚单位√-1（如i*i=-1）

2. **四元数如何表示?**

（1）矢量式 $\boldsymbol{Q}=q_0+\boldsymbol{q}$ （2）复数式 $\boldsymbol{Q}=q_0+q_1\boldsymbol{i}+q_2\boldsymbol{j}+q_3\boldsymbol{k}$

（3）三角式 $\boldsymbol{Q}=\cos\dfrac{\theta}{2}+\boldsymbol{u}\sin\dfrac{\theta}{2}$ （4）指数式 $\boldsymbol{Q}=\mathrm{e}^{\boldsymbol{u}\frac{\theta}{2}}$

（5）矩阵式 $\boldsymbol{Q}=\begin{bmatrix}q_0\\q_1\\q_2\\q_3\end{bmatrix}$

3. **怎么定量描述某个时间的姿态?**

数学建模：将目标物体抽象为一个三维坐标系，选取一个参考系，计算**姿态矩阵。**

设由运载体的机体轴确定的坐标系（MPU6050或者面包板）为**b**,惯导系统所采用的导航坐标系（地球参考系）为**R**。由于R系和b系均为直角坐标系,各轴之间始终保持直角,所以可将坐标系理解成刚体,当只研究两个坐标系间的角位置关系时,可对一个坐标系做平移,使其原点与另一个坐标系的原点重合。因此,**两坐标系间的空间角位置关系可理解成刚体的定点转动**。然后用姿态矩阵表示相对于参考系的位置。

4. **姿态矩阵是什么?**

由b系到R系的坐标变换矩阵C(b->n)称为运载体（MPU6050）的姿态矩阵。

由刚体等效旋转模型，进行向量，矩阵运算，运用欧拉定理等推导出姿态矩阵：
式子中，角度为坐标系到参考系的等效旋转角.

变换后的向量 $\boldsymbol{r}'^R=\begin{bmatrix}r'_x\\r'_y\\r'_z\end{bmatrix}$， 变换前的向量 $\boldsymbol{r}^R=\begin{bmatrix}r_x\\r_y\\r_z\end{bmatrix}$，旋转轴单位向量 $\boldsymbol{u}^R=\begin{bmatrix}l\\m\\n\end{bmatrix}$

$$\boldsymbol{r}'^R=\boldsymbol{D}\boldsymbol{r}'^b$$

$\boldsymbol{D}$ 为 b 系至 R 系的坐标变换矩阵

$$\boldsymbol{C}_b^R=\begin{bmatrix}1&0&0\\0&1&0\\0&0&1\end{bmatrix}+2\cos\frac{\theta}{2}\begin{bmatrix}0&-n\sin\frac{\theta}{2}&m\sin\frac{\theta}{2}\\n\sin\frac{\theta}{2}&0&-l\sin\frac{\theta}{2}\\-m\sin\frac{\theta}{2}&l\sin\frac{\theta}{2}&0\end{bmatrix}$$

$$+2\begin{bmatrix}-(m^2+n^2)\sin^2\frac{\theta}{2}&lm\sin^2\frac{\theta}{2}&ln\sin^2\frac{\theta}{2}\\lm\sin^2\frac{\theta}{2}&-(l^2+n^2)\sin^2\frac{\theta}{2}&mn\sin^2\frac{\theta}{2}\\ln\sin^2\frac{\theta}{2}&mn\sin^2\frac{\theta}{2}&-(m^2+l^2)\sin^2\frac{\theta}{2}\end{bmatrix}\quad(9.2.30\text{b})$$

5. **姿态更新是什么？**

姿态更新是指根据惯性器件输出实时计算出的C矩阵。

我们的研究目标在空间不停移动，要想实时输出姿态角，就得更新旧数据。姿态更新的关键在于找到新旧状态的联系（状态转移方程），这有点像从初始状态不断递推得到n+1状态的过程。

6. **四元数与姿态矩阵之间的关系是什么？为啥四元数能包含所有姿态信息？**

将姿态矩阵中的角度和旋转轴方向向量（l,m,n）代换为四个新的变量q0,q1,q2,q3便得到四元数。

令

$$\begin{cases} q_0 = \cos\dfrac{\theta}{2} \\ q_1 = l\sin\dfrac{\theta}{2} \\ q_2 = m\sin\dfrac{\theta}{2} \\ q_3 = n\sin\dfrac{\theta}{2} \end{cases} \tag{9.2.31}$$

并以 q_0、q_1、q_2、q_3 构造四元数：

$$\begin{aligned} \boldsymbol{Q} &= q_0 + q_1\boldsymbol{i}_0 + q_2\boldsymbol{j}_0 + q_3\boldsymbol{k}_0 = \cos\frac{\theta}{2} + (l\boldsymbol{i}_0 + m\boldsymbol{j}_0 + n\boldsymbol{k}_0)\sin\frac{\theta}{2} \\ &= \cos\frac{\theta}{2} + \boldsymbol{u}^R\sin\frac{\theta}{2} \end{aligned} \tag{9.2.32}$$

姿态矩阵能表示目标的姿态，所以四元数也包含了所有的姿态信息，并且由上式可知四元数为何定义为一个实数（标量）加一个三维向量（矢量）。

小结:

（1）四元数 $\boldsymbol{Q}=\cos\frac{\theta}{2}+\boldsymbol{u}^R\sin\frac{\theta}{2}$ 描述了刚体的定点转动，即当只关心 b 系相对 R 系的角位置时，可认为 b 系是由 R 系经过无中间过程的一次性等效旋转形成的，$\boldsymbol{Q}$ 包含了这种等效旋转的全部信息：$\boldsymbol{u}^R$ 为旋转瞬轴和旋转方向，θ 为转过的角度。

（2）四元数可确定出 b 系至 R 系的坐标变换矩阵。将式(9.2.31)代入式(9.2.30b)，得

$$\boldsymbol{C}_b^R = \begin{bmatrix} 1-2(q_2^2+q_3^2) & 2(q_1q_2-q_0q_3) & 2(q_1q_3+q_0q_2) \\ 2(q_1q_2+q_0q_3) & 1-2(q_1^2+q_3^2) & 2(q_2q_3-q_0q_1) \\ 2(q_1q_3-q_0q_2) & 2(q_2q_3+q_0q_1) & 1-2(q_1^2+q_2^2) \end{bmatrix} \tag{9.2.33}$$

由于 $\|\boldsymbol{Q}\| = q_0^2+q_1^2+q_2^2+q_3^2 = \cos^2\frac{\theta}{2}+(l^2+m^2+n^2)\sin^2\frac{\theta}{2}=1$，所以可进一步推得如下结论：

（1）描述刚体旋转的四元数是规范化四元数。

（2）式(9.2.33)可写成：

$$\boldsymbol{C}_b^R = \begin{bmatrix} q_0^2+q_1^2-q_2^2-q_3^2 & 2(q_1q_2-q_0q_3) & 2(q_1q_3+q_0q_2) \\ 2(q_1q_2+q_0q_3) & q_0^2-q_1^2+q_2^2-q_3^2 & 2(q_2q_3-q_0q_1) \\ 2(q_1q_3-q_0q_2) & 2(q_2q_3+q_0q_1) & q_0^2-q_1^2-q_2^2+q_3^2 \end{bmatrix} \tag{9.2.34}$$

7. **姿态矩阵的第二种求法**

利用同一矢量在不同坐标系内投影间的变换关系（旋转矩阵）

线性代数中讲过旋转矩阵的概念，这里拓展到了三维：

$$\boldsymbol{r}^1=\begin{bmatrix} r_{X_1} \\ r_{Y_1} \\ r_{Z_1} \end{bmatrix},\qquad \boldsymbol{r}^2=\begin{bmatrix} r_{X_2} \\ r_{Y_2} \\ r_{Z_2} \end{bmatrix},\qquad \boldsymbol{C}_1^2=\begin{bmatrix} \cos\alpha & \sin\alpha & 0 \\ -\sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\boldsymbol{r}^2=\boldsymbol{C}_1^2\boldsymbol{r}^1$$

上述变换关系分析中,坐标系2是经坐标系1仅绕Z轴旋转α角后获得的,即**仅绕一根轴的旋转**为基本旋转。但**两坐标系间任何复杂的角位置关系都可以看作有限次基本旋转的复合**,对应的，变换矩阵等于基本旋转确定的变换矩阵的连乘,连乘顺序依基本旋转的先后次序由右向左排列。
例如MPU6050空间姿态可看做依次绕航向轴（Yaw）、俯仰轴（Pitch）、横滚轴（Roll）做基本旋转后的复合结果。

所以姿态矩阵又可以表示为：

$$\boldsymbol{C}_n^b=\boldsymbol{C}_2^b\boldsymbol{C}_1^2\boldsymbol{C}_n^1=\begin{bmatrix} \cos\gamma & 0 & -\sin\gamma \\ 0 & 1 & 0 \\ \sin\gamma & 0 & \cos\gamma \end{bmatrix}\begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & \sin\theta \\ 0 & -\sin\theta & \cos\theta \end{bmatrix}\begin{bmatrix} \cos\Psi & -\sin\Psi & 0 \\ \sin\Psi & \cos\Psi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$=\begin{bmatrix} \cos\gamma\cos\Psi+\sin\gamma\sin\Psi\sin\theta & -\cos\gamma\sin\Psi+\sin\gamma\cos\Psi\sin\theta & -\sin\gamma\cos\theta \\ \sin\Psi\cos\theta & \cos\Psi\cos\theta & \sin\theta \\ \sin\gamma\cos\Psi-\cos\gamma\sin\Psi\sin\theta & -\sin\gamma\sin\Psi-\cos\gamma\cos\Psi\sin\theta & \cos\gamma\cos\theta \end{bmatrix} \quad (1.2.2)$$

8. **怎么用四元数解算姿态角？**

我们已经使用了两种方法计算出了姿态矩阵，一个的参数为四元数，一个的参数为三个姿态角，则根据矩阵对应相等解方程（两个划星号的式子联立）即可得到等价关系。

为简化表达，令：

$$g1=2(q1q3-q0q2),\quad g2=2(q2q3+q0q1)$$

$$g3=q0^2-q1^2-q2^2+q3^2,\quad g4=2(q1q2+q0q3)$$

$$g5=q0^2+q1^2-q2^2-q3^2$$

则可以解出

$$\gamma=-\arcsin(g1)$$

$$\theta=\arctan(g2/g3)$$

$$\psi=\arctan(g4/g5)$$

9. **已经可以用四元数表示姿态角了，但是要怎么更新四元数进行姿态更新呢？**

更新四元数要用到MPU6050陀螺仪输出的角速度对时间积分，得到单位时间相对于自身坐标系的变化角度，再加上偏移量（旧值），就可以更新四元数了。

四元数微分方程的毕卡求解法（定时采样增量法）

Q是描述从n系至b系等效旋转的四元数，已知Q(t)的导数和Q(t)的关系：

$$\dot{\boldsymbol{Q}}_n^b(t)=\lim_{\Delta t\to 0}\frac{\boldsymbol{Q}_n^b(t+\Delta t)-\boldsymbol{Q}_n^b(t)}{\Delta t}=\boldsymbol{Q}_n^b(t)\otimes\left(0+\frac{1}{2}\boldsymbol{\omega}_{nb}^b\right)=\frac{1}{2}\boldsymbol{Q}_n^b(t)\otimes\boldsymbol{\omega}_{nb}^b \qquad (9.2.47)$$

补充四元数的乘法：

$$\begin{aligned}\boldsymbol{P}\otimes\boldsymbol{Q}&=(p_0+p_1\boldsymbol{i}+p_2\boldsymbol{j}+p_3\boldsymbol{k})\otimes(q_0+q_1\boldsymbol{i}+q_2\boldsymbol{j}+q_3\boldsymbol{k})\\&=(p_0q_0-p_1q_1-p_2q_2-p_3q_3)+(p_0q_1+p_1q_0+p_2q_3-p_3q_2)\boldsymbol{i}\\&\quad+(p_0q_2+p_2q_0+p_3q_1-p_1q_3)\boldsymbol{j}+(p_0q_3+p_3q_0+p_1q_2-p_2q_1)\boldsymbol{k}\\&=r_0+r_1\boldsymbol{i}+r_2\boldsymbol{j}+r_3\boldsymbol{k}\end{aligned} \qquad (9.2.12)$$

上式写成矩阵形式：

$$\begin{bmatrix}r_0\\r_1\\r_2\\r_3\end{bmatrix}=\begin{bmatrix}p_0&-p_1&-p_2&-p_3\\p_1&p_0&-p_3&p_2\\p_2&p_3&p_0&-p_1\\p_3&-p_2&p_1&p_0\end{bmatrix}\begin{bmatrix}q_0\\q_1\\q_2\\q_3\end{bmatrix}=\boldsymbol{M}(\boldsymbol{P})\boldsymbol{Q} \qquad (9.2.13)$$

或

$$\begin{bmatrix}r_0\\r_1\\r_2\\r_3\end{bmatrix}=\begin{bmatrix}q_0&-q_1&-q_2&-q_3\\q_1&q_0&q_3&-q_2\\q_2&-q_3&q_0&q_1\\q_3&q_2&-q_1&q_0\end{bmatrix}\begin{bmatrix}p_0\\p_1\\p_2\\p_3\end{bmatrix}=\boldsymbol{M}'(\boldsymbol{Q})\boldsymbol{P} \qquad (9.2.14)$$

则Q(t)的导数和Q(t)的关系可表示为：

$$\frac{\mathrm{d}\boldsymbol{Q}}{\mathrm{d}t}=\frac{1}{2}\boldsymbol{M}'(\boldsymbol{\omega}_{nb}^b)\boldsymbol{Q}$$

其微分方程的解为：

$$\boldsymbol{Q}(t_{k+1})=\mathrm{e}^{\frac{1}{2}\int_{t_k}^{t_{k+1}}\boldsymbol{M}'(\boldsymbol{\omega}_{nb}^b)\mathrm{d}t}\cdot\boldsymbol{Q}(t_k)$$

令：

$$\Delta\boldsymbol{\Theta}=\int_{t_k}^{t_{k+1}}\boldsymbol{M}'(\boldsymbol{\omega}_{nb}^b)\mathrm{d}t=\int_{t_k}^{t_{k+1}}\begin{bmatrix}0&-\omega_x&-\omega_y&-\omega_z\\\omega_x&0&\omega_z&-\omega_y\\\omega_y&-\omega_z&0&\omega_x\\\omega_z&\omega_y&-\omega_x&0\end{bmatrix}\mathrm{d}t\approx\begin{bmatrix}0&-\Delta\theta_x&-\Delta\theta_y&-\Delta\theta_z\\\Delta\theta_x&0&\Delta\theta_z&-\Delta\theta_y\\\Delta\theta_y&-\Delta\theta_z&0&\Delta\theta_x\\\Delta\theta_z&\Delta\theta_y&-\Delta\theta_x&0\end{bmatrix}$$

则式子可化简，并作泰勒级数展开：

$$\boldsymbol{Q}(t_{k+1})=\mathrm{e}^{\frac{1}{2}\Delta\boldsymbol{\Theta}}\cdot\boldsymbol{Q}(t_k)=\left[\boldsymbol{I}+\frac{\frac{1}{2}\Delta\boldsymbol{\Theta}}{1!}+\frac{\left(\frac{1}{2}\Delta\boldsymbol{\Theta}\right)^2}{2!}+\cdots\right]\boldsymbol{Q}(t_k)$$

由此可得到：

一阶近似算法：

$$Q(t_{k+1}) = \left(I + \frac{\Delta\Theta}{2}\right)Q(t_k)$$

还原变量,将四元数Q用矩阵形式表达，解方程：

$$q0 = q0 + \frac{1}{2}\Delta t(-\omega_x q1 - \omega_y q2 - \omega_z q3)$$

$$q1 = q1 + \frac{1}{2}\Delta t(\omega_x q0 - \omega_y q3 + \omega_z q2)$$

$$q2 = q2 + \frac{1}{2}\Delta t(\omega_x q3 + \omega_y q0 - \omega_z q1)$$

$$q3 = q3 + \frac{1}{2}\Delta t(-\omega_x q2 + \omega_y q1 + \omega_z q0)$$

如此便得到了更新四元数的关系式。

软件解算步骤

1. 初始四元数的值：初始化为q0=1,q1=0,q2=0,q3=0，刚开始目标坐标系和参考坐标系重合。
2. 读取MPU6050加速度计值、角速度值，并将陀螺仪值转为弧度。（传的是地址，用指针接的）

```
//加速度满量程正负16g，MPU6050内置16位AD（一位用来表示正负，所以32768）
*AX_Convert = *AX_Convert / 32768 * 16;
*AY_Convert = *AY_Convert / 32768 * 16;
*AZ_Convert = *AZ_Convert / 32768 * 16;
//角速度满量程正负2000°/s，MPU6050内置16位AD（一位用来表示正负，所以32768）
*GX_Convert = *GX_Convert / 32768 * 2000 * 0.0174532925;//角度化弧度
*GY_Convert = *GY_Convert / 32768 * 2000 * 0.0174532925;
*GZ_Convert = *GZ_Convert / 32768 * 2000 * 0.0174532925;
```

3. 加速度值进行归范化：描述刚体旋转的四元数都是规范化的，也就是单位化。

```
Norm = sqrt(AX_Convert * AX_Convert + AY_Convert * AY_Convert +
AZ_Convert *AZ_Convert);
AX_Convert = AX_Convert / Norm;
AY_Convert = AY_Convert / Norm;
AZ_Convert = AZ_Convert / Norm;
```

4. 求姿态矩阵中的重力分量：本质是用对地重力加速度（0，0，1）左乘姿态矩阵化简而来。

```
vx = 2*(q1*q3 - q0*q2);
vy = 2*(q0*q1 + q2*q3);
vz = q0*q0 - q1*q1 - q2*q2 + q3*q3;
```

5. 求姿态误差：**向量间的误差，可以用向量叉积（也叫向量外积、叉乘）来表示**，对重力分量，加速度向量进行叉乘(ex、ey、ez为三个轴误差元素)，就是陀螺积分后的姿态和加计测出来的姿态之间的误差。

```
ex = (AY_Convert * vz - AZ_Convert * vy);
ey = (AZ_Convert * vx - AX_Convert * vz);
ez = (AX_Convert * vy - AY_Convert * vx);
```

6. **互补滤波**：加速度计对陀螺仪进行修正，将误差与陀螺仪测得的角速度相加，修正角速度值：比例增益kp控制收敛到加速度计速率,积分增益ki控制陀螺仪偏差的收敛速率。
Kp的取值应适中，过大可能导致系统过于敏感，对噪声反应强烈，产生抖动；过小则可能导致系统响应缓慢，无法及时修正误差。
Ki的取值也应适中，过大可能导致系统积分饱和，产生过大的修正量；过小则可能无法有效消除陀螺仪的漂移。

```
exInt = exInt + ex * Ki;
eyInt = eyInt + ey * Ki;
ezInt = ezInt + ez * Ki;

GX_Convert = GX_Convert + Kp*ex + exInt;
GY_Convert = GY_Convert + Kp*ey + eyInt;
GZ_Convert = GZ_Convert + Kp*ez + ezInt;
```

7. 更新四元数：见上四元数微分方程的毕卡求解法（定时采样增量法）

```
q0 = q0 + (-q1*GX_Convert - q2*GY_Convert - q3*GZ_Convert)*halfT;
q1 = q1 + (q0*GX_Convert - q3*GY_Convert + q2*GZ_Convert)*halfT;
q2 = q2 + (q3*GX_Convert + q0*GY_Convert - q1*GZ_Convert)*halfT;
q3 = q3 + (-q2*GX_Convert + q1*GY_Convert + q0*GZ_Convert)*halfT;
```

8. 四元数规范化：更新四元数过程可能导致其失去规范性。和上面一样，就是除以模。

9. 计算姿态角。

```
angle->roll =  atan2f(2*q2*q3 + 2*q0*q1,q0*q0 + q3*q3 - q1*q1 - q2*q2)  *
180 / pi;
angle->pitch =  asinf(-2*q1*q3 + 2*q0*q2)* 180 / pi;
angle->yaw  =  atan2f(2*q1*q2 + 2*q0*q3,q0*q0 + q1*q1 - q2*q2 - q3*q3)  *
180 / pi;
```

10. 将姿态转换函数放入中断函数，间隔时间更新。

```
void TIM2_IRQHandler()
{
```

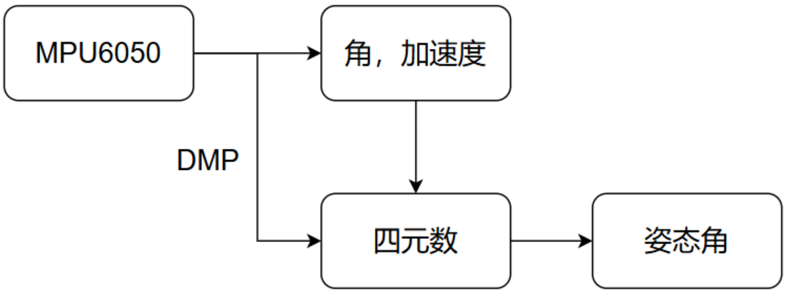
```
    if (TIM_GetITStatus(TIM2, TIM_IT_Update) == SET)
    {
        MPU6050_GetData(&AX, &AY, &AZ, &GX, &GY, &GZ);


MPU6050_DataConvert(&AX_Convert,&AY_Convert,&AZ_Convert,&GX_Convert,&GY_C
onvert,&GZ_Convert,AX, AY, AZ, GX, GY, GZ);

        Pose_Calculating(AX_Convert, AY_Convert, AZ_Convert, GX_Convert,
GY_Convert, GZ_Convert,&angle);

        TIM_ClearITPendingBit(TIM2, TIM_IT_Update);
    }
}
```

框图



照片