

## 旋翼无人飞行器基础传感器融合

**摘要** 旋翼无人飞行器的导航问题备受关注,在获得定位时往往需要利用多种传感器,将传感器数据进行融合,以获得更准确的位姿估计。本文主要使用扩展卡尔曼滤波方法,分别实现IMU、视觉里程计、光流传感器、TFmini的多传感器融合,并用时间同步相关方法解决不同信息获取的频率不同的问题。

**关键词** EKF, 多传感器融合, 旋翼飞行器

**中图分类号**: TP212.9

### Basic sensor fusion of rotorcraft

**Abstract** The navigation problem of rotorcraft has attracted much attention. In order to obtain more accurate position and attitude estimation, it is often necessary to use a variety of sensors to fuse the sensor data. This paper mainly uses the extended Kalman filter method to realize the multi-sensor fusion of IMU, visual odometer, optical flow sensor and tfmini, and uses the time synchronization correlation method to solve the problem of different information acquisition frequency.

**Key words** EKF, Multisensor fusion, Rotorcraft

## 1 引言

经过几十年空中机器人技术的不断发展,自主飞行无人机已在多个领域发挥价值。从动力结构上看无人机主要分为固定翼和旋翼,其中四旋翼飞行器是目前热门的研究领域。针对能够自主飞行的无人机,导航系统是无人机系统重要的一环。导航系统需要考虑问题之一:定位问题,进一步细化即位姿估计问题——空中机器人若要得到位姿信息需要传感器系统。

一个无人机的传感器系统主要有IMU,视觉传感、GPS、雷达、超声波等。而单一传感器存在局限性。因此,需要考虑传感器融合减少位姿估计的误差,平滑位姿估计结果,并可利用高频率的传感器提高位姿估计的频率。

为解决传感器融合问题,获得更优的位姿估计,对旋翼无人机这样的非线性动态系统,本文将使用扩展卡尔曼滤波(EKF)的方法,建立相应的状态模型、观测模型,在ROS中实现融合并通过仿真实验验证方法的有效性 & 优异性。

## 2 方法介绍

### 2.1 多传感器融合

对于旋翼无人机的导航,在动态变化的机体进行位姿估计时,多传感器融合方法利用计算机技术将两个或两个以上的传感器设备从环境中得到的信息输出以一定的规则进行融合。

单一的传感器数据会存在许多问题,不同的传感器在不同环境下的敏感程度不同。如IMU虽然具有高频率但是有零漂和噪声,这些干扰信号叠加在测量信号中会使输出信号无法得出准确的位置和角度,同时其通过积分获取位姿时存在累计误差。通过相机得到的视觉信息虽然无累计误差但是频率低且可能产生跳变。这些噪声虽然可以通过数据平滑滤波方法将其滤除,但这会使系统的实时性变差。因此,进行多传感器融合是极为必要的。

### 2.2 扩展卡尔曼滤波(EKF)原理

卡尔曼滤波是一种在不确定状况下组合多源信息得到所需状态最优估计的一种方法。当系统具有线性动力学模型,且系统与传感器的误差符合高斯白噪声模型,则卡尔曼滤波将为融合数据提供唯一统计意义下的最优估计。对于旋翼无人机的非线性动力学模型,EKF使用一阶泰勒展开将非线性系统线性化,则能够有效地进行估计。

对于一个非线性系统,在连续的时间下,状态 $x$ 存在以下平滑动力学模型

$$\begin{aligned}\dot{x} &= f(x, u, n) \\ z &= g(x, v)\end{aligned}$$

其中

$$\begin{aligned}n_t &\sim N(0, Q_t) \\ v_t &\sim N(0, R_t)\end{aligned}$$

是高斯白噪声。

EKF包含两部分过程,通过状态方程预测下一时刻的状态以及通过观测方程使用观测的传感器信息对状态进行修正更新。

预测模型:

对 $\dot{x}$ 进行线性化处理,

$$\begin{aligned}x &= \mu_{t-1}, u = u_t, n = 0 \\ \dot{x} &\approx f(\mu_{t-1}, u_t, 0) + \left. \frac{\partial f}{\partial x} \right|_{\mu_{t-1}, u_t, 0} (x - \mu_{t-1}) \\ &+ \left. \frac{\partial f}{\partial u} \right|_{\mu_{t-1}, u_t, 0} (u - u_t) + \left. \frac{\partial f}{\partial n} \right|_{\mu_{t-1}, u_t, 0} (n - 0) \\ &\text{令}\end{aligned}$$

1. 浙江大学控制科学与工程学院 杭州 310012

1. College of Control Science and Engineering, Zhejiang University, Hangzhou 310012

$$\begin{aligned} A_t &= \frac{\partial f}{\partial x} \Big|_{\mu_{t-1}, u_t, 0} \\ B_t &= \frac{\partial f}{\partial u} \Big|_{\mu_{t-1}, u_t, 0} \\ U_t &= \frac{\partial f}{\partial n} \Big|_{\mu_{t-1}, u_t, 0} \end{aligned}$$

得到

$$\begin{aligned} \dot{x} &\approx f(\mu_{t-1}, u_t, 0) + A_t(x - \mu_{t-1}) + B_t(u - u_t) \\ &\quad + U_t(n - 0) \end{aligned}$$

在离散时间系统下，一步欧拉积分得

$$x_{\bar{t}} \approx x_{t-1} + f(x_{t-1}, u_t, n_t) \delta t$$

代入上述公式有

$$\begin{aligned} x_{\bar{t}} &\approx (I + \delta t A_t) x_{t-1} + \delta t U_t n_t + \delta t (f(\mu_{t-1}, u_t, 0) \\ &\quad - A_t \mu_{t-1}) \end{aligned}$$

令

$$\begin{aligned} F_t &= (I + \delta t A_t) \\ V_t &= \delta t U_t \end{aligned}$$

$$x_{\bar{t}} \approx F_t x_{t-1} + V_t n_t + \delta t (f(\mu_{t-1}, u_t, 0) - A_t \mu_{t-1})$$

在状态变化的过程中引入了新的不确定性，根据协方差的乘积公式可得：

$$\bar{\Sigma}_t = F_t \Sigma_{t-1} F_t^T + V_t Q_t V_t^T$$

预测步骤为：

$$\begin{aligned} \bar{\mu}_t &= \mu_{t-1} + \delta t f(\mu_{t-1}, u_t, 0) \\ \bar{\Sigma}_t &= F_t \Sigma_{t-1} F_t^T + V_t Q_t V_t^T \end{aligned}$$

更新模型：

对  $z$  进行线性化处理，

$$\begin{aligned} x &= \bar{\mu}_t, v = 0 \\ g(x, v) &\approx g(\bar{\mu}_t, 0) + \frac{\partial g}{\partial x} \Big|_{\bar{\mu}_t, 0} (x - \bar{\mu}_t) \\ &\quad + \frac{\partial g}{\partial v} \Big|_{\bar{\mu}_t, 0} (v - 0) \end{aligned}$$

令

$$\begin{aligned} C_t &= \frac{\partial g}{\partial x} \Big|_{\bar{\mu}_t, 0} \\ W_t &= \frac{\partial g}{\partial v} \Big|_{\bar{\mu}_t, 0} \end{aligned}$$

$$z_t = g(x_t, v_t) \approx g(\bar{\mu}_t, 0) + C_t(x_t - \bar{\mu}_t) + W_t v_t$$

得到状态空间方程

$$\begin{bmatrix} x_t \\ z_t \end{bmatrix} = \begin{bmatrix} I & 0 \\ C_t & W_t \end{bmatrix} \begin{bmatrix} x_{\bar{t}} \\ v_t \end{bmatrix} + \begin{bmatrix} 0 \\ g(\bar{\mu}_t, 0) - C_t \bar{\mu}_t \end{bmatrix}$$

均值：

$$\begin{aligned} E[X_t] &= \bar{\mu}_t \\ E[Z_t] &= g(\bar{\mu}_t, 0) \end{aligned}$$

协方差：

$$\Sigma = \begin{bmatrix} I & 0 \\ C_t & W_t \end{bmatrix} \begin{bmatrix} \bar{\Sigma}_t & 0 \\ 0 & R_t \end{bmatrix} \begin{bmatrix} I & C_t^T \\ 0 & W_t^T \end{bmatrix}$$

$$\Sigma = \begin{bmatrix} \bar{\Sigma}_t & \bar{\Sigma}_t C_t^T \\ C_t \bar{\Sigma}_t & C_t \bar{\Sigma}_t C_t^T + W_t R W_t^T \end{bmatrix}$$

条件密度函数  $f_{X|Z}(x|Z=z)$  如下

$$\mu_{X|Z} = \mu_X + \Sigma_{XZ} \Sigma_{ZZ}^{-1} (z - \mu_Z)$$

$$\Sigma_{X|Z} = \Sigma_{XX} - \Sigma_{XZ} \Sigma_{ZZ}^{-1} \Sigma_{ZX}$$

更新步骤为：

$$\begin{aligned} \mu_t &= \bar{\mu}_t + K_t (z_t - g(\bar{\mu}_t, 0)) \\ \Sigma_t &= \bar{\Sigma}_t - K_t C_t \bar{\Sigma}_t \\ K_t &= \bar{\Sigma}_t C_t^T (C_t \bar{\Sigma}_t C_t^T + W_t R W_t^T)^{-1} \end{aligned}$$

## 2.3 扩展卡尔曼滤波(EKF)设计实现

### 2.3.1 基础部分

基础部分需要将提供的 IMU 数据和视觉例程数据融合，IMU 的加速度计和陀螺仪分别提供线加速度和角加速度，对状态进行预测。

旋翼无人机状态如下

$$\mathbf{x} = \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \mathbf{x}_3 \\ \mathbf{x}_4 \\ \mathbf{x}_5 \end{bmatrix} = \begin{bmatrix} \mathbf{p} \\ \mathbf{q} \\ \dot{\mathbf{p}} \\ \mathbf{b}_g \\ \mathbf{b}_a \end{bmatrix} = \begin{bmatrix} position \\ orientation \\ linear velocity \\ gyroscope bias \\ accelerometer bias \end{bmatrix} \in R^{15}$$

过程模型为

$$\dot{\mathbf{x}} = \begin{bmatrix} \mathbf{x}_3 \\ \mathbf{G}(\mathbf{x}_2)^{-1}(\boldsymbol{\omega}_m - \mathbf{x}_4 - \mathbf{n}_g) \\ \mathbf{g} + \mathbf{R}(\mathbf{x}_2)(\mathbf{a}_m - \mathbf{x}_5 - \mathbf{n}_a) \\ \mathbf{n}_{bg} \\ \mathbf{n}_{ba} \end{bmatrix}$$

其中  $\mathbf{n}_a$  和  $\mathbf{n}_g$  为加速度计和陀螺仪的噪声， $\mathbf{n}_{ba}$  和  $\mathbf{n}_{bg}$  为零漂的导数，为高斯白噪声。

$\mathbf{G}$  为角速度坐标系转换矩阵：

$$\boldsymbol{\omega} = \begin{bmatrix} \mathbf{p} \\ \mathbf{q} \\ \mathbf{r} \end{bmatrix} = \begin{bmatrix} c\theta & 0 & -c\phi s\theta \\ 0 & 1 & s\phi \\ s\theta & 0 & c\phi c\theta \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \mathbf{G}(\mathbf{q}) \dot{\mathbf{q}}$$

$\mathbf{R}$  为欧拉角和四元数的转换矩阵：

$$\mathbf{q} = \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix} = \begin{bmatrix} roll \\ pitch \\ yaw \end{bmatrix}$$

$\mathbf{R}$

$$\mathbf{R} = \begin{bmatrix} c\psi c\theta - s\phi s\psi s\theta & -c\phi s\psi & c\psi s\theta + c\theta s\phi s\psi \\ c\theta s\psi + c\psi s\phi s\theta & c\phi c\psi & s\psi s\theta - c\psi c\theta s\phi \\ -c\phi s\theta & s\phi & c\phi c\theta \end{bmatrix}$$

AR Tags 提供的视觉里程计信息包含位置和姿态，建立观测方程，对状态进行更新。

观测方程为：

$$\begin{aligned} \mathbf{z} = \begin{bmatrix} \mathbf{p} \\ \mathbf{q} \end{bmatrix} + \mathbf{v} &= \begin{bmatrix} I & 0 & 0 & 0 & 0 \\ 0 & I & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{p} \\ \mathbf{q} \\ \dot{\mathbf{p}} \\ \mathbf{b}_g \\ \mathbf{b}_a \end{bmatrix} + \mathbf{v} \\ \mathbf{z} &= \mathbf{C} \mathbf{x} + \mathbf{v} \end{aligned}$$

EKF 的实现基于 ROS 平台，通过建立节点完成信息的通信。利用 subscriber 节点的回调函数完成预测和更新过程。利用 publisher 发布滤波后的位姿信息 topic，利用 rqt\_plot 实现可视化。

subscriber s1 用于接收 IMU 节点发布的加速度和角速度信息，利用 imu\_callback 的回调函数完成对状态变量和误差协方差矩阵的推算，实现状态的预测，并发布融合的状态信息 ekf\_odom。

subscriber s2 用于接受 tag\_odom 发布的实时里程计位姿检测结果，odom\_callback 计算卡尔曼增益 K、更新状态变量和协方差矩阵，实现位姿更新。

其中对于 tag\_odom 节点发布的实时视觉里程计位姿检测结果，位姿数据是在二维码 tag frame 下的，需要进行坐标转换，将数据转换到 world frame 下，已知 IMU 和 tag 的相关矩阵  $R_{ic}$  和  $T_{ic}$ ，tag 返回的数据为  $T_{cw}$  和  $R_{cw}$ ，通过如下的矩阵变换得到 world frame 下的视觉传感器数据。

$$R_{wi} = R_{wc} * R_{ci}$$

$$T_{wi} = -R_{wc} * (R_{ci} * T_{ic} + T_{cw})$$

### 2.3.2 进阶部分

基础实践在实际使用中，由于视觉传感器依赖于二维码信息，二维码的铺设是有一定范围，飞行器存在飞出二维码范围的情况，这时只剩下 IMU 传感器，无法得到较为精准的位姿估计。因此本文引入新的传感器进行融合，无二维码时，用光流传感器和 TFmini 激光雷达得到位姿信息与 IMU 融合。有二维码时，将所有传感器进行融合。同时考虑到各传感器测量频率不同而需要完成时间同步问题。

光流是视域中物体运动检测中的概念，用来描述对于观察者的运动所造成的观测目标、表面或边缘的运动。光流传感器可以通过测量视觉运动并基于光流法计算线速度测量值  $v_x$ 、 $v_y$ 。光流法是利用图像序列中像素在时间域上的变化以及相邻帧相关性来找到对应关系并得出运动信息的方法。本文使用 opencv 的 LK 光流法函数得到线速度数据。

TFmini 是一个小型激光雷达传感器，它可以无接触的获得高度信息，对于本研究的实验环境，不存在复杂的地形起伏，用 TFmini 极为合适。

vicon mocap 是一个运动捕捉系统。动作捕捉是指运动物体的关键部位设置跟踪器，由 Motion capture 系统捕捉跟踪器位置，再经过计算机处理后得到三维空间坐标的数据。Vicon 系统可提供最高级别精度的位置和角度。此数据作为 EKF 融合 IMU、

视觉、光流、Tfmini 结果的参考值对比数据。

IMU 预测的过程模型为：

$$\dot{\mathbf{x}} = \begin{bmatrix} \mathbf{x}_3 \\ G(\mathbf{x}_2)^{-1}(\omega_m - \mathbf{x}_4 - \mathbf{n}_g) \\ \mathbf{g} + R(\mathbf{x}_2)(\mathbf{a}_m - \mathbf{x}_5 - \mathbf{n}_a) \\ \mathbf{n}_{bg} \\ \mathbf{n}_{ba} \end{bmatrix}$$

由于 TFmini 雷达的高度数据和 tag 的高度数据产生了耦合，且由于引入了光流传感器的速度，观测模型中也需加入速度，速度需要用和欧拉角有关的转换矩阵  $R$  进行变换。对比基础任务时的线性模型，新的观测模型是个非线性的，如下公式所示：

$$\mathbf{z} = \begin{bmatrix} p \\ q \\ R(q)^T \dot{p} \end{bmatrix} + \mathbf{v}$$

$$\mathbf{z} = \begin{bmatrix} I & 0 & 0 & 0 & 0 \\ 0 & I & 0 & 0 & 0 \\ 0 & 0 & R(q)^T & 0 & 0 \end{bmatrix} \begin{bmatrix} p \\ q \\ \dot{p} \\ b_g \\ b_a \end{bmatrix} + \mathbf{v}$$

为了让视觉里程计和光流-TFmini 独立、互不干扰又能做到共同作用，作如下的观测模型分离。

$$\mathbf{z}_1 = \begin{bmatrix} I & 0 & 0 & 0 & 0 \\ 0 & I & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} p \\ q \\ \dot{p} \\ b_g \\ b_a \end{bmatrix} + \mathbf{v}$$

$$\mathbf{z}_2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & R(q)^T & 0 \\ 0 & 0 & 0 & R(q)^T \end{bmatrix} \begin{bmatrix} p \\ q \\ \dot{p} \\ \dot{y} \end{bmatrix} + \mathbf{v}$$

这样一来，有 tag 码时两个观测模型都进行更新，无 tag 时光流和 Tfmini 观测模型单独更新。引入光流 TFmini 后 ROS 节点如图 2.3.1 所示：

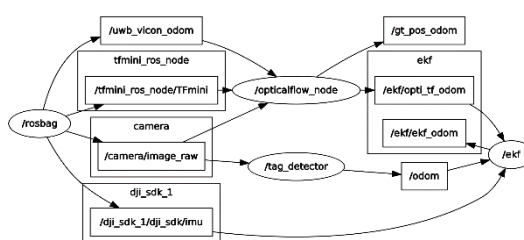


图 2.3.1 ROS 节点图

Rosbag: 提供传感器数据，包括动作捕捉数据、image\_raw 相机数据、Tfmini 高度数据、IMU 数据。

opticalflow\_node: 读取 vicon 数据，并以 Path 消息发布 /gt\_pos\_odom 作为运动位姿的真值参考；处理 image\_raw 图像数据，使用 opencv 的 LK 光流算法获得 camera frame 下的  $v_x$  和  $v_y$  数据；获得 Tfmini 高度数据，并将其和  $v_x$ 、 $v_y$  数据一起以

PointStamped 消息发布 opti\_tf\_odom。

tag\_detector\_node: 处理 image\_raw 图像数据, 使用 opencv 的 PNP 算法将视频的 2D 图像帧转化为 tag frame 下的 3D 位姿数据并以 nav\_msgs::Odometry 消息发布 odom\_ref。

除了基础部分的回调函数, 进阶部分新增了 op\_tf\_callback, 此函数订阅 opticalflow\_node 发布的线速度和高度信息, 使用观测模型 2 进行对状态的更新, 坐标系转换和更新步骤同 tag 回调函数。

## 2.4 多传感器的时间同步原理

时间同步指在时间频率的精确测量或在使用精确时间频率的工作中, 使时间信号在时刻上或频率信号在相位上保持某种严格的特定关系。在本项目中是指将不同传感器得到的对应于同一时间戳的信息进行同步处理。

时间同步的目的在于在对多传感器数据进行融合处理时, 由于不同信息获取的频率不同, 传递时需要的时间不同, 所以在同一个时刻接收到的信息可能不对应与同一个时间点。比如通过相机图像得到的光流信息和 tag\_odom 信息是存在较大延迟的, 但是 IMU 信息存在的延迟较小, 所以接收到的图像信息应该可能对应于几十毫秒前的 IMU 信息, 而不是同时收到的 IMU 量。如果利用之前的图像信息对现在的状态进行更新就会导致较大的误差。

## 2.5 时间同步实现

### 2.5.1 ROS 自带的时间同步工具介绍

消息过滤器 message\_filters 是一个用于 roscpp 和 rospy 的实用程序库。它类似一个消息缓存, 当消息到达消息过滤器的时候, 可能并不会立即输出, 而是在稍后的时间点里满足一定条件下输出。

message\_filters 库中含 synchronizer 时间同步工具, 可以接收来自多信号发送源的不同类型的消息, 且仅当它们在具有相同时间戳的每个源上接收到消息时才输出, 由此起到消息同步输出的效果。

但是在使用中我们发现 synchronizer 只输出相同时间的信息, 其余不同时间的信息都会丢弃, 信息的利用率很低。且当多个主题频率一致的时候也无法保证回调函数的频率等于订阅主题的频率。这样就失去了我们利用 IMU 对视觉信息进行更新的功能。且 IMU 和图像信息的频率差别较大, 出现完全相同时间戳的情况并不常见, 所以用 synchronizer 方法并不适合我们的项目。

### 3.5.2 时间同步实现

最终采用的方法为建立一个数组储存接受到的 IMU 信息, 数组结构及相关参数作用如图 2.5.1:

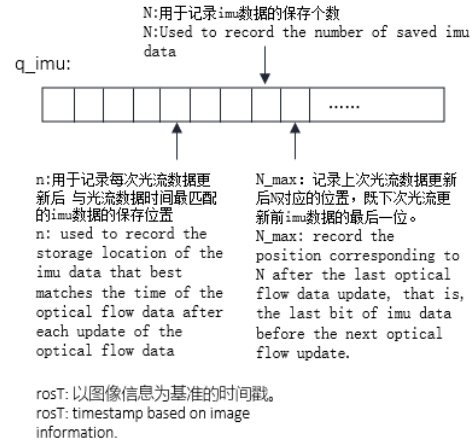


图 2.5.1 储存数组

并且对回调函数也进行了修改, 增加功能的伪代码如图 2.5.2 所示。

```
if imu_callback 被调用
    q_imu[N]=msg->ang_v, lin_a;
    N++;    //在 q_imu 中存储新接收的 imu
           //信息, 记录信息数量的 N++;
    ang_v, lin_a=q_imu[n];
    n++;    //提取 q_imu 中与光流数据时间
           //最匹配的 imu 数据 q_imu[n], 进行预测更新。
           //处理结束后 n++。
    if n>N_max  n=0;
    //被调用的 imu 信息是有效的;
    rosT+= Δ T;
    sys(rosT);    //对 rosT 进行更新, 用 rosT
                 //作为时间戳发布消息。

if odom_callback\opti_callback 被调用
    for(int i=0; i<N ;i++)
        find n'->min(abs(t_image-t_imu));
        n=n';    //找到时间戳最接近的 IMU 信息,
    对 n 进行更新;
    rosT=ros::Time::Now(); //更新 rosT;
    N_max=N;
    N=0;    //更新 N_max 和 N;
```

图 2.5.2 回调函数伪代码

通过每次调用 IMU 回调函数时都对 n 加一, 保证每次被处理的数据不是最新接受的数据, 而是经过同步后的和光流时间最匹配的那一个数据。

### 3 实践结果

#### 3.1 融合结果

##### 3.1.1 基础部分

将 IMU 获得的加速度角速度信息与视觉里程计获得的位姿信息用扩展卡尔曼滤波的方法实现了融合。结果如图 3.1.1、3.1.2 所示。

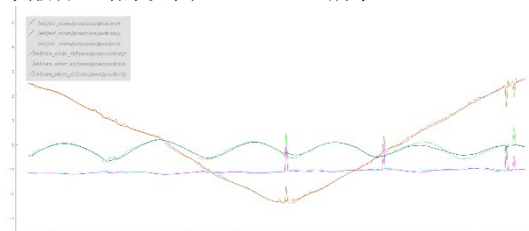


图 3.1.1 位置估计结果

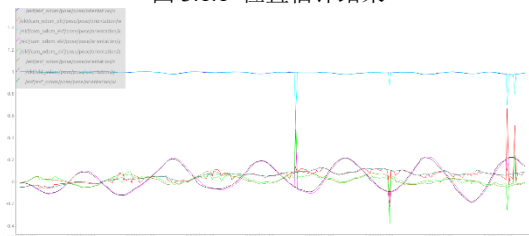


图 3.1.2 姿态估计结果

在 rviz 中对路径进行仿真，可得到图 3.1.3 结果：

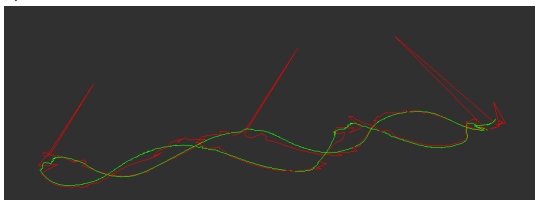


图 3.1.3 路径仿真结果

##### 3.1.2 进阶部分

对存在二维码边界的图像信息的图像包(仅用视觉里程计作观测的结果如图 3.1.4)，除 IMU、视觉里程计外，融合光流得到的飞行器线速度信息、TFmini 获得的高度信息，实现多传感器融合，如图 3.1.5。并完成时间同步，得到结果如图 3.1.6 所示。

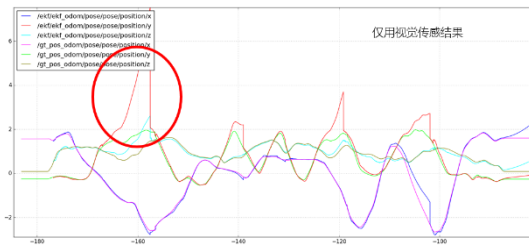


图 3.1.4 仅用视觉传感器

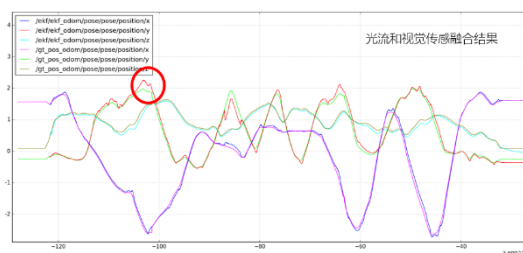


图 3.1.5 光流和视觉传感融合结果

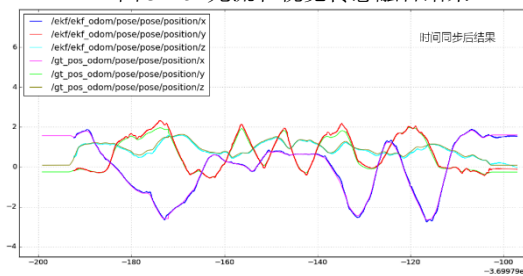


图 3.1.6 时间同步后结果

#### 3.2 时间同步

运用图像信息存在较大延迟的数据包对时间同步代码进行检验，运行结果如图 3.2.1~3.2.3。

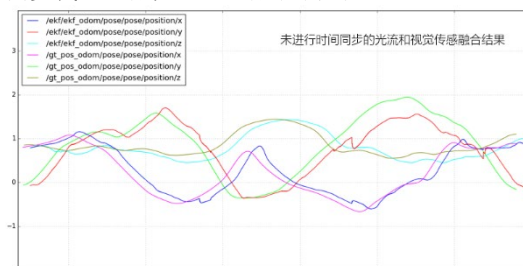


图 3.2.1 未进行时间同步的融合结果

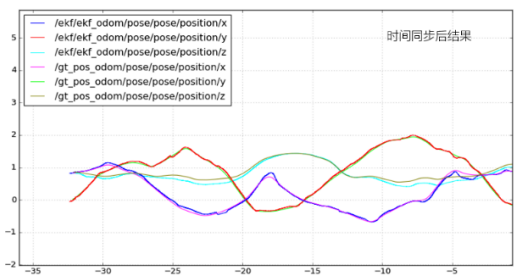


图 3.2.2 进行时间同步的融合结果

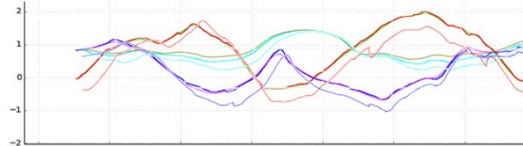


图 3.2.3 将融合结果对齐时间后的图像  
(粗线为时间同步后，细线为时间同步前输出)

## 4 分析讨论

### 4.1 融合结果分析

#### 4.1.1 基础部分

实现了 IMU 与视觉里程计的融合，平滑了位姿估计，对尖刺也有很好的消除作用。

#### 4.1.2 进阶部分

对比仅用二维码进行更新，与同时使用二维码和光流更新的图像结果，在没有二维码的时间段里，使用了光流的图像可以继续和真值保持相近结果，可以实现不间断的位姿估计。

经过时间同步后，可以修正二维码从有到无时使 EKF 结果产生的一些跳变，进一步平滑了位姿。误差图如图 4.1.1 所示，可以发现误差在较小范围。

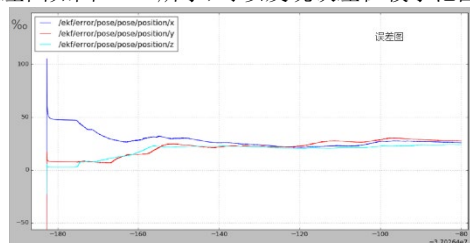


图 4.1.1 误差图

### 4.2 时间同步创新性思考

实际的无人机飞行中可能会出现传感器通信中断或卡住的情况，为模拟此情况：运用图像信息存在较大延迟的数据包对时间同步代码进行检验。

对比时间同步前后的图像，可以发现：与未进行时间同步时相比，EKF 融合结果更好，没有出现时滞现象，更好地和真值保持一致。

对此现象进行分析：

图 3.2.1 中真值 `gt` 和融合后的 `ekf` 曲线存在较大延迟是因为 `gt` 图像用 `rqt_plot` 工具输出时默认使用了 IMU 数据的时间戳，而 `ekf` 图像由于进行了融合，所以在输出图像时用了比较慢的图像信息时间戳。

图 3.2.2 中经过时间同步两组图像延迟被消除。通过 `bag` 相关输出我们得知 `bag` 中 IMU 比图像信息和光流信息提前了 0.6s，在进行时间同步时，收到 `tag` 或光流信息时，会搜寻之前存储的对应的 IMU 信息并对 IMU 的时间戳进行了以 `tag` 或光流为基准的更新，这样就实现了时间戳的对齐，并且 `gt` 输出使用的时间戳也和 `ekf` 输出时间戳进行了同步，消除了显示延迟。

图 3.2.3 在经过了人工对齐后，进行时间同步后的 `ekf` 输出比起进行时间同步前的输出还减少了突变点，并且曲线也更加平滑。

## 5 结论

本文针对旋翼无人飞行器基础传感器相关问题，由浅入深进行了探究。首先，为了解决单传感器或是存在累计误差或是频率低有跳变的缺点，进行传感器融合，以消除误差、获得更加准确的位姿估计并提高位姿估计的频率。由于模型的非线性性，选择了扩展卡尔曼滤波的方法开展研究。实现 IMU 和视觉里程计融合，发现融合对于尖刺有很好的处理效果。随后，针对二维码可能存在边界因而部分位置无法获得里程计信息的问题，引入了光流传感器、TFmini，进一步利用 EKF 完成多传感器融合。最终完成时间同步问题，发现时间同步可以平滑突变的同时，还可以实现对时滞的消除，从而得到光滑、准确的实时位姿信息融合。

## References

1. Alattise M B, Hancke G P. Pose estimation of a mobile robot based on fusion of IMU data and vision data using an extended Kalman filter[J]. *Sensors*, 2017, 17(10): 2164.
2. Sun S L, Deng Z L. Multi-sensor optimal information fusion Kalman filter[J]. *Automatica*, 2004, 40(6): 1017-1023.
3. Nützi G, Weiss S, Scaramuzza D, et al. Fusion of IMU and vision for absolute scale estimation in monocular SLAM[J]. *Journal of intelligent & robotic systems*, 2011, 61(1-4): 287-299.
4. Elson J, Estrin D. Time synchronization for wireless sensor networks[C]//*Proceedings 15th International Parallel and Distributed Processing Symposium. IPDPS 2001. IEEE*, 2001: 1965-1970.
5. Sichitiu M L, Veerarittiphan C. Simple, accurate time synchronization for wireless sensor networks[C]//*2003 IEEE Wireless Communications and Networking, 2003. WCNC 2003. IEEE*, 2003, 2: 1266-1273.