

PDR/GNSS Fusion Algorithm Based on Joint Heading Estimation

Abstract: Pedestrian dead reckoning (PDR) based on smartphone is limited by the low performance of MEMS sensors, and the heading estimation and stride length estimation produce significant accumulated errors over time, which limits the application of PDR. Considering that GNSS of smartphone can provide absolute location information in outdoor applications, and the estimation errors of PDR and GNSS have different characteristics, this paper analyses the characteristics of two different types of errors, and proposes a fusion positioning framework that can real-time fuse PDR and GNSS information. The optimal estimations of heading and stride length are also given. The experimental results show that the fusion algorithm is superior to PDR or GNSS in heading estimation, anti-jamming and noise performance.

Keywords: PDR; heading estimation; MCHE; fusion positioning

基于联合航向估计的PDR/GNSS融合定位算法

【摘要】 基于智能手机的行人航位推算（PDR）由于受限于MEMS器件的低性能，其航向与步长估计会随着时间增长产生显著的积累误差，从而限制了PDR的实际应用。考虑到室外应用时，手机的GNSS能够提供绝对位置信息，而且两种不同的定位方式的航向与步长误差具有不同的特征，因此，本文充分分析了两种不同类型误差的特性，并提出了一种可以实时融合PDR和GNSS信息的融合定位框架，从而给出了最优的航向与步长估计。实验结果表明，融合算法相比于PDR或GNSS在偏航程度、抗干扰以及噪声性能等方面都具有优越性。

【关键词】 PDR；航向估计；平均累积航向误差；融合定位

1 引言

伴随着智能手机的发展，很多传统用于高精度导航领域的专业传感器都被集成到其中，比如GNSS接收机、IMU（陀螺仪和加速度计）等。但是低成本器件的性能相比于专业级设备在性能指标上差别巨大，比如MEMS加速度计的零偏一般大于 0.03m/s^2 ，MEMS陀螺仪的零偏一般大于 $100^\circ/\text{h}$ ，而基于单点定位的手机GNSS定位精度通常在 10m 左右^[1,5]。正因如此，传统的捷联惯导算法（SINS）在MEMS IMU上几乎无法实现，因此研究人员渐渐地转移了关注点。虽然MEMS的低精度会在SINS中导致积分误差，但是在某些应用场景下，比如行人、车辆行驶等，同样会利用MEMS传感器来辅助姿态测量，而且该姿态变化具有一定的规律性，因此通过设计相应的算法可以在适当程度上消除MEMS器件带来的积分误差，由此发展了行人航位推算（PDR）^[3]、车辆航位推算（VDR）等研究方向。

另外，现在的智能手机上还集成了很多其他类型的传感器，比如地磁、蓝牙、WIFI和摄像头等，因此基于智能手机的多源融合定位方案可以多种多样。就以传统的GNSS和INS组合导航为例，这是一种高度地优势互补系统，因此在多个应用领域都取得了成功。然而在智能手机上，一方面是由于前述的MEMS性能不足以支持INS，另一方面受限于各自系统的开放程度，很难实现两个系统更紧的耦合。组合导航根据耦合程度的深浅可以划分成三种类型：在定位结果层面的松耦合，在观测量层面的紧耦合以及在信号跟踪层面的深耦合。伴随着近些年来安卓手机原始GNSS观测量的开放^[6]，目前理论上是支持MEMS与低成本GNSS的紧耦合，但是此类研究仍然存在巨大挑战。

因为基于智能手机的低成本MEMS与GNSS目前仍然无法实现GNSS/INS组合导航，而基于MEMS的PDR与GNSS的融合却被认为是可实现的，因此近些年来大量相关论文被发表。如同上文所述，基于PDR和

低成本GNSS的融合可以是松耦合，也可以是紧耦合。Li-Ta Hsu等人^[2]提出了一种在城市环境下基于3D地图辅助GNSS和PDR的融合算法框架，该算法将PDR的输出步长与航向以及3D-GNSS的位置与精度等信息一并送入卡尔曼滤波器，从而获得融合结果，因此这是一种松耦合方式。其中的卡尔曼滤波器的观测来自3D-GNSS，而PDR的输出则用于控制输入，这是因为行人的轨迹具有很大不确定性，难以用线性状态转移方程描述，因此文中采用PDR的实时观测来更新状态转移方程，但由于PDR仍然存在累积误差，因此这种设计还是不可避免地导致整体系统存在随时间积累的误差，导致最终的偏航。Haiyu Lan等人^[4]的工作采用扩展卡尔曼滤波器（EKF）融合PDR和GNSS，该算法是直接基于MEMS传感器的输出信息，因此可以视为一种紧耦合算法。该算法采用GNSS推导的航向来辅助MEMS陀螺仪的航向估计，从而提高了航向估计准确度，但是GNSS航向存在的严重噪声问题并没有很好地解决，因此导致最终的轨迹输出的初始阶段严重依赖于GNSS的航向。

基于上述分析，本文首先重点讨论了PDR和GNSS航向的误差形式和特点，并据此提出了一套融合框架，该框架充分结合了PDR和GNSS的各自优点，有效地消除了累积误差以及随机噪声。另外，考虑到实践中GNSS和PDR的更新频率不一致问题，讨论了具体的数据同步方法，这在之前的文献中并没有被充分研究。

本文后续章节安排如下：第二节将简要讨论PDR算法的实现细节；第三节将详细讨论融合算法的误差分析、数据同步以及融合框架等问题；第四节将给出实验方法和结果分析；最后一节进行了总结和展望。

2 PDR算法

PDR算法通常包含三个步骤：步态检测、步长估计和航向估计，如图1所示。

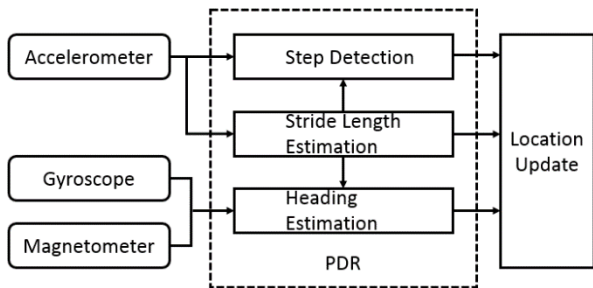


Figure 1. Block diagram of PDR algorithm

图 1.PDR 算法框图

加速度计的3轴输出一般用于检测跨步的发生以及估计步长，而航向则一般由陀螺仪或者磁力计给出。因此PDR算法是在完成一个步态检测之后输出第k步

的航向与步长估计 $\{\theta, SL\}$ ，在获取到初始位置的前提下，由式1迭代更新行人的位置。

$$\begin{cases} x_{k+1} = x_k + SL_k \cdot \sin \theta_k \\ y_{k+1} = y_k + SL_k \cdot \cos \theta_k \end{cases} \quad (1)$$

2.1 步态检测

步态检测主要是对加速度计在行人行走时产生的周期性信号进行检测，以此判断行人是否跨步了。常见的步态检测方法有过零检测法、峰值检测法、自相关法和频谱分析法等。在本文中，首先对加速度数据进行预处理，通过设置低通滤波器的截止频率为3Hz从而有效地降低了原始数据中的噪声，然后采用了文献^[7]中提出的动态门限过零检测法，通过有限状态机来控制行走与非行走状态之间的转换，实现了99%的准确率。

2.2 步长估计

常见的步长估计算法包括经验模型、线性模型、非线性模型和机器学习模型等。本文采用的步长估计模型是以步频 f_s 和加速度方差 σ_a^2 为变量的二元线性回归模型^[8]，其表达式如下：

$$SL = \alpha f_s + \beta \sigma_a^2 + \gamma \quad (2)$$

其中， α 与 β 是权重因子，而 γ 为常量，它们都需要通过离线训练才能确定。一旦确定，便可以用于在实时应用中估计不同行人的步长。

2.3 航向估计

航向估计是PDR算法中最为关键的一环，这是因为在航向上引入的微小估计误差将积累起来，最终导致行人轨迹与真实轨迹差距很大。PDR中的航向估计通常采用陀螺仪、磁力计或者二者联合的方法。由于陀螺仪会在积分时产生累积误差，而磁力计容易受到周围磁场的干扰，因此二者具有很明显的互补特性。本文中采用的航向估计融合了陀螺仪和磁力计，通过利用二者的冗余信息得到了最优的估计航向。

3 PDR/GNSS融合框架

本节首先定义了GNSS航向与步长，将PDR和GNSS的输出统一到相同框架下。然后分析了PDR和GNSS的航向误差与步长误差形式，并提出相应的衡量指标。最后讨论了数据同步问题，并给出了融合算法框架的实现细节。

3.1 GNSS航向与步长

GNSS单点定位给出的是某个时刻用户的位置，通常认为将时间上的单点位置依次连接起来就能得到

GNSS轨迹。通常接收机在输出单点定位结果之前都会进行滤波，但是仍然存在随机噪声的影响，尤其是用智能手机上的低成本GNSS，噪声功率要远大于专业级接收机。因为PDR的观测输出是航向与步长，为了统一GNSS和PDR的输出，需要定义GNSS航向与步长，这两个概念在传统GNSS领域鲜被提及。具体方法如下：

坐标转换：将GNSS的输出经纬高（LLH）首先转换成ECEF XYZ坐标，然后基于先验初始位置将获得的ECEF坐标转为东北天（ENU）坐标，在这种二维平面上，便可以定义类似PDR系统中的航向与步长了；

计算航向与步长：假设 (E_k, N_k, U_k) 和 $(E_{k+1}, N_{k+1}, U_{k+1})$ 是 k 和 $k+1$ 两个时刻的ENU坐标，那么GNSS航向 θ_{k+1}^{GNSS} 与步长 SL_{k+1}^{GNSS} 的计算公式如式3所示。需要说明的是，航向的定义与PDR相同，即前后两个定位结果的矢量与正北方向的夹角，需要注意该式中的航向取值范围是 $(-\pi, +\pi)$ 。

$$\begin{aligned}\theta_{k+1}^{GNSS} &= \arctan\left(\frac{N_{k+1} - N_k}{E_{k+1} - E_k}\right) \\ SL_{k+1}^{GNSS} &= \left\| \begin{bmatrix} E_{k+1} - E_k \\ N_{k+1} - N_k \end{bmatrix} \right\|_2\end{aligned}\quad (3)$$

3.2 PDR与GNSS航向与步长误差

式4给出了式1的另一种写法，其中 $\{x_1, y_1\}$ 是先验的初始位置。在每次迭代过程中，步长和航向都会引入一些误差，这些误差最终将反映在位置 x 和 y 上，利用式4可以推导这种航向与步长的误差对 x 和 y 的影响，但是这个过程是复杂的，而且对消除积累误差也没有太大帮助。

$$\begin{cases} x_{k+1} = x_1 + \sum_{i=1}^k (SL_i \cdot \sin \theta_i) \\ y_{k+1} = y_1 + \sum_{i=1}^k (SL_i \cdot \cos \theta_i) \end{cases}\quad (4)$$

为了融合PDR和GNSS，一种思路是基于坐标点的融合，但是根据上述分析，利用坐标点的融合不便于分析累积误差的影响，因此提出了GNSS航向与步长的概念。下文将给出关于GNSS和PDR的航向与步长估计误差分析，并由此提出一个明确的融合目标。首先，给出如下符号说明：

- $\varepsilon_k^{\theta-PDR}$ ：k步时PDR的航向估计误差；
- ε_k^{SL-PDR} ：k步时PDR的步长估计误差；
- $\varepsilon_k^{\theta-GNSS}$ ：k时刻GNSS航向估计误差；
- $\varepsilon_k^{SL-GNSS}$ ：k时刻GNSS步长估计误差；

为了分析上述误差的特征，既可以从PDR或单点定位算法进行推导，也可以选择通过实验来进行误差特征统计。这类实验需要很多的先验控制点，对于行人航位推算，可以将实验简化，比如让测试人员沿着带有直线标记的路面行进，沿着地面的控制点一步一步地前进，将步长固定为常数，并将实时数据记录下来用后处理软件分析。通过进行了大量上文介绍的控制点实验之后，我们得出了上述四种误差类型的特征：其中PDR的航向和步长误差伴随着 k 的增长而增加，而GNSS的航向与步长误差更加符合零均高斯噪声。为了更好地描述这两种不同特性的误差，本文定义了一个称之为平均累积航向误差（MCHE）的指标来衡量两种系统的航向误差，公式如下：

$$MCHE(k) = \frac{1}{k} \sum_{i=1}^k \varepsilon_i^{\theta-SYS} \quad (5)$$

其中SYS可以是GNSS，也可以是PDR。这个指标可以有效地区分随机噪声误差和累积误差的特征，当 k 增大时，GNSS的零均噪声MCHE会趋于零，而PDR的累积误差MCHE则是发散的。同理，我们定义累积步长误差（CSLE）用以衡量两种系统的步长误差特征，公式如下：

$$CSLE(k) = \sum_{i=1}^k \varepsilon_i^{SL-SYS} \quad (6)$$

根据上文讨论，本文所提出的融合算法的一个目标就是要将两个系统的航向进行融合，使得新的估计航向一方面不随时间积累误差，另一方面也能降低噪声功率。

3.3 数据同步

在融合系统中，PDR的输出是不定时的，当检测到有跨步时才更新输出，而行人的行走又具有随机性，因此PDR的输出是无法预测的。而GNSS则是固定周期输出定位结果的，通常为1Hz的更新率。因此为了能够实时融合二者的数据，需要将两个系统的输出步调变得相一致。当人在正常行走时，通常在1s内能够完成至少一个完整的跨步，而在1s内无法完成一个完整的跨步，更有可能是行人没有考虑好下一步的方向。称第一种情形是快步，第二种情形是慢步，如图2所示。正常情况下，人在连续行走时，都是快步，而慢步是偶尔发生的现象。如果在一段较长的时间内没有跨步，则称之为驻留，也就是行人的双脚合拢，停留在原地。下图给出了GNSS和PDR异步输出的示意图。

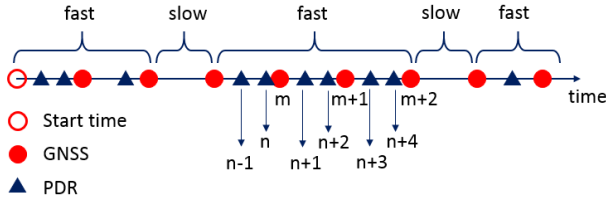


Figure 2. Data synchronization between PDR and GNSS

图 2.PDR 与 GNSS 的数据同步

本文提出的数据同步方法的基本思路是将PDR的数据变成定时输出。在图2中，在GNSS的m和m+1时刻之间，有两个PDR输出，分别是n+1和n+2，对应得到 $\{\theta_{n+1}, SL_{n+1}\}$ 和 $\{\theta_{n+2}, SL_{n+2}\}$ ，定义在GNSS的m时刻得到一个PDR的输出 $\{\theta_m, SL_m\}$ ，公式如下：

$$\begin{cases} \theta_m = \frac{\theta_{n+1} + \theta_{n+2}}{2} \\ SL_m = \sqrt{SL_{n+1}^2 + SL_{n+2}^2 + 2C} \\ C = SL_{n+1}SL_{n+2} \cdot \cos(\theta_{n+1} - \theta_{n+2}) \end{cases} \quad (7)$$

式7给出了在某一时刻计算同步后的PDR航向与步长的方法，这个公式可以扩展到一个GNSS间隔内存在1个、2个或3个等任意个PDR输出的情形，其中，航向的计算是求平均，而步长的计算不是简单的求和，而是求取m时刻后第一个PDR输出坐标与m+1时刻后第一个PDR输出坐标之间的距离。

3.4 PDR/GNSS融合框架

单独的PDR定位具有清晰轨迹，但是因为在航向和步长的估计中，还是会产生累积误差，因此伴随着时间的增长，最后的轨迹会变形。而GNSS的单点轨迹表现为不光滑的曲线，这是随机噪声造成的影响，但是它不会产生累积误差。这是相对定位与绝对定位所具有的很明显的区别。本文融合算法的目标是抑制累积误差，并尽可能降低随机噪声的影响，使得最后的行人轨迹能真实反映行人的所处位置，即轨迹光滑而且不变形。

考虑到PDR/GNSS融合算法可以直接借鉴传统INS和GNSS的松耦合方式，如图3所示。其中同步后的PDR系统将输出位置与航向信息，并且与GNSS的位置与航向做差，之后位置与航向残差将被送入卡尔曼滤波器进行校正，最后将校正量返回到PDR系统，形成组合系统的位置与航向估计。但是这种方法的难点是，在卡尔曼滤波器的设计时仍然需要分析位置与航向的观测噪声，在实际应用时，这些噪声与误差的统计特性是难以获取的，不仅如此，如同3.2小节所讨论的那样，最终的位置观测中的误差与噪声分量难以计算，因此在设计卡尔曼滤波器时，参数的选取直接决定了

融合系统的滤波效果。

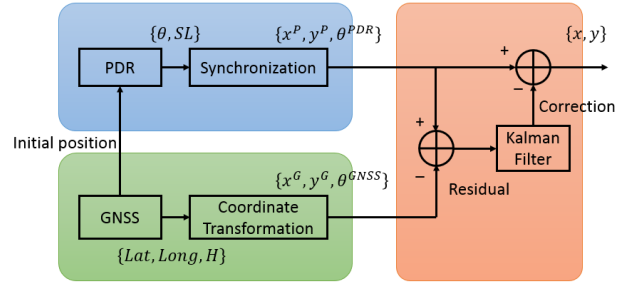


Figure 3. Traditional loose-coupled framework of INS/GNSS

图 3.传统 INS/GNSS 松耦合框架

为了降低对卡尔曼参数选择的依赖，本文给出了另一种滤波思路。该思路基于联合的航向估计，利用自身噪声功率小的PDR航向来平滑噪声大的GNSS航向，不仅如此，因为算法中采用的是差分PDR航向，因此累积误差也得到了抑制。该融合算法的框架示意图如图4所示，在初始阶段由GNSS提供初始位置，这可以用长时间观测的均值。然后PDR输出航向与步长估计，GNSS输出经纬高信息，并且经过坐标转换，从经纬高转换成ECEF XYZ，然后再转换成东北天坐标，最后根据式3转换成GNSS航向和GNSS步长。同时PDR经过同步处理之后，与GNSS在固定频率下输出同步后的航向与步长 $\{\theta^{PDR}, SL^{PDR}\}$ 。

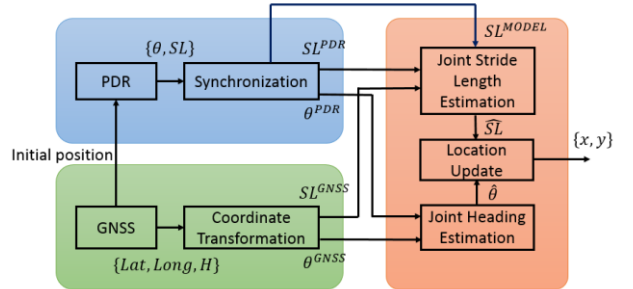


Figure 4. Fusion framework of PDR/GNSS

图 4.PDR/GNSS 融合框架

然后将两系统的航向估计和步长估计分别送入联合航向估计和联合步长估计模块，最终由这两模块输出经过融合滤波之后的航向与步长估计 $\{\hat{\theta}, \hat{SL}\}$ ，并送给位置更新模块，由式1完成行人位置的迭代更新。

联合航向估计模块的目的是将具有累积误差的PDR航向和具有随机噪声的GNSS航向进行融合，实现既减轻了累积误差，同时减小噪声功率的双重作用。它的具体实现由下式展现：

$$\begin{aligned} \hat{\theta}_{k+1} &= \frac{1}{M} \theta_{k+1} + \frac{M-1}{M} (\hat{\theta}_k + \theta_{k+1}^{PDR} - \theta_k^{PDR}) \\ \theta_{k+1} &= p \theta_{k+1}^{GNSS} + (1-p) \theta_{k+1}^{PDR} \end{aligned} \quad (8)$$

上式是一个迭代公式，输入是两个系统的航向估

计 θ^{PDR} 和 θ^{GNSS} ，输出是经融合后的航向估计。式中的 M 和 p 是设计参数，其中 M 是平滑系数，用来对含有噪声的GNSS航向进行滤波平滑，它不宜设置得过大或过小，太大的话使得航向的跟踪变得迟钝，而太小的话则滤波效果不佳，通常设置为30至50。 p 是权重因子，通常取值为1至1.5。

联合步长估计模块的输入包含三方面：同步后PDR步长、GNSS步长以及PDR同步模块的输出 SL^{MODEL} ，其中同步模块会判断行人的模式，也即3.3小节中提到的快步、慢步和驻留。最终的输出是三者的加权输出。

4 实验环境与结果

4.1 实验环境

实验场景如图9中红线所示，选取的路线是一条直线，有地面标记，其长度为127m，该直线与正北方向的偏角为 -19.19° 。为了凸显PDR算法随时间误差积累的特点，测试人员沿着直线从A出发，总共往返三次（A-B-A为一次往返），总耗时为540s。

实验中，测试人员右手平持华为P10手机，沿着直线连续行走，行走的速度较为均匀，偶尔有快有慢。当到达两个端点之一A或B时，双脚靠拢，并转身重新开始行进。实验过程中，两边偶有车辆穿行，手机中低成本GNSS芯片在这种情形下的多径误差会变大，因此可以测试融合算法的抗干扰能力。

4.2 实验结果

4.2.1 误差分析

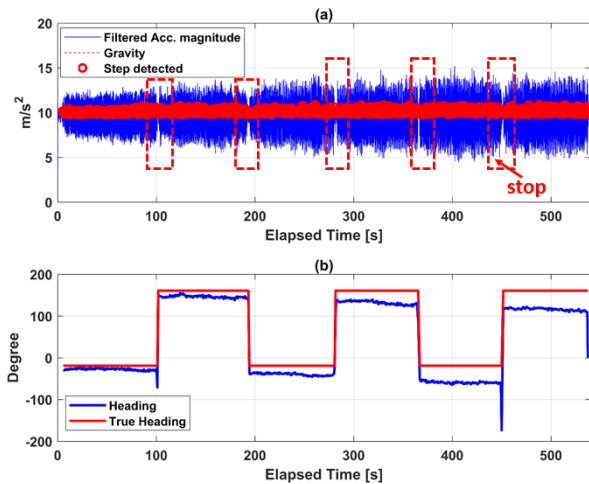


Figure 5. Step detection and heading estimation

图 5.PDR 步态检测与航向估计

图5（a）展示了PDR的步态检测结果，从图中可以明显地看出三次往返的间歇时刻，如同图中红色虚

线框所示。而图（b）展示了PDR计算的航向估计与真值的对比，可以发现在前200s的时间内，航向估计还是比较准确的，但是在后面的300s中，可以明显看出第二次和第三次往返时航向产生了明显的偏差。

下图给出了PDR步长估计误差的特征。

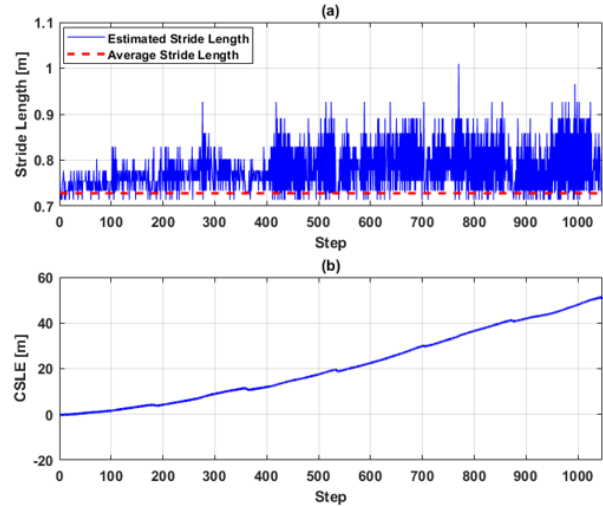


Figure 6. Stride length error of PDR

图 6.PDR 步长误差

图6(a)展示了PDR估计步长与平均步长的对比，可以发现伴随着时间的增长估计步长要明显大于平均步长，其中平均步长大约为0.73米。图6（b）给出了累积步长误差CSLE的曲线图，由图可知，由于每次的PDR步长估计都存在0.1m左右的误差，因此最终导致1000步时将近50m的总误差。因此，图5（b）和图6（b）共同说明在PDR中步长估计和航向估计都会引入较大的累积误差。

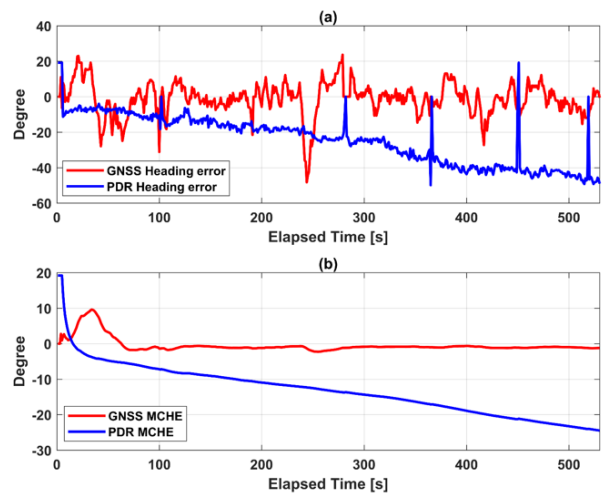


Figure 7. Heading errors of PDR and GNSS

图 7.PDR 与 GNSS 航向误差

图7展示了PDR航向与GNSS航向的误差特点，图（a）中红线给出了GNSS的航向误差，可以看出GNSS航向受到噪声的影响很大，最大有 20° 的偏航，而图

中蓝线表示的PDR航向呈现出时间积累误差，但是可以看得出来，其上的噪声并不是很大，图中蓝线有几个尖峰，那是因为PDR算法中的驻留点检测导致的，说明此刻行人是处于停止状态的。图(b)展示了GNSS航向和PDR航向的平均累积航向误差MCHE，这幅图说明了两种误差所具有的完全不同的特征，即GNSS航向误差是零均值随机噪声，而PDR航向误差呈现出时间函数的特性。

4.2.2 融合效果

GNSS航向、PDR航向以及本文提出的融合算法得到的航向估计的对比如图8所示。从图中可以看出，GNSS航向不具有时间偏移型，但是其噪声影响了航向估计的准确度，这使得GNSS推导出的行人轨迹不光滑，而且GNSS容易受到环境因素的干扰，比如图中红圈1标注的地方，那一段数据存在很大的波动，这是因为在测试过程中，路边偶尔过往的车辆遮挡所致，这种情况会导致GNSS在那时刻的多径误差突然变大。而与此同时，PDR则不会受到外界的干扰，鲁棒性较高。而融合航向则具备了上述两种航向估计的各自优点，比如没有明显的时间偏置、抗干扰能力增强以及随机噪声降低等特点。图中红圈2给出了在某个时刻检测到的一个慢步，由此可见慢步在行人正常的行走过程中是极少出现的，由于PDR航向估计会检测到慢步，因此融合算法的航向估计同样会表现出慢步的影响，但是这并不会影响之后的航向估计，也就是说，这种现象是无害的。

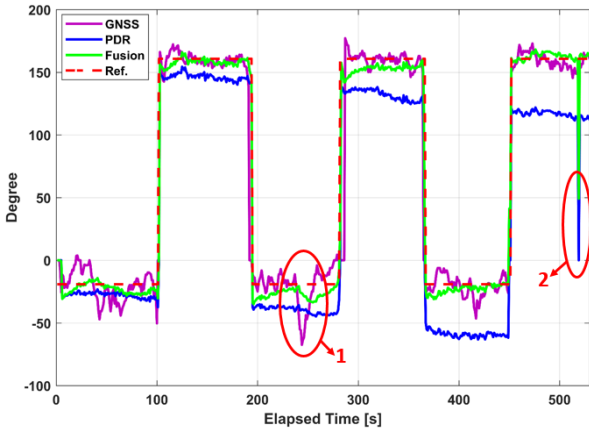


Figure 8. Comparison of heading estimation by three methods

图 8.三种方法的航向估计对比

图9给出了三种方法的最终轨迹图，该图将三次往返过程依次拆解成了(a)(b)(c)三幅图。

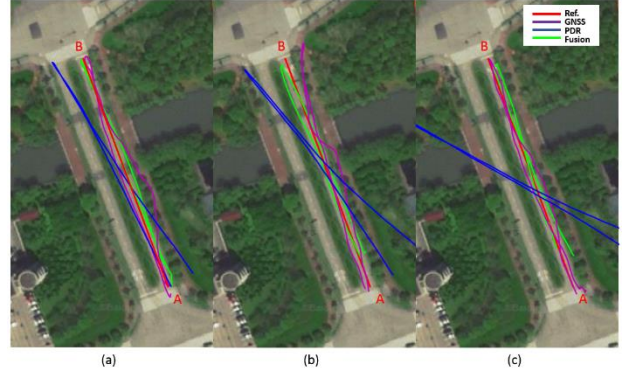


Figure 9. Comparison of trajectory by three methods

图 9.三种方法的行人轨迹对比

图中A是路径起点，B是终点。由图(a)可知，PDR方法在瞬时的精度很高，但是会随时间增长而偏航，而GNSS轨迹则表现出较大的波动性，而融合算法的轨迹表现出了与真实轨迹的高度一致性。图(b)中展示了伴随时间增长，PDR的轨迹产生了严重的偏航，已经无法反映行人的真实轨迹了，而在这次往返过程中，GNSS轨迹的扭曲变形是因为受到测试时车辆的干扰，即便在这种GNSS受到严重多径干扰以及PDR产生严重偏航的情况下，融合算法仍然能给出相对光滑、鲁棒以及无偏的航迹。图(c)中的PDR轨迹已经完全失去了意义，而GNSS在这次往返过程中受干扰影响较小，融合算法的轨迹则依然能保持不偏航，说明此时它受到PDR航向估计的影响较小，但它同时继承了PDR航向噪声小的优点，轨迹更加光滑。从三幅图中可以看出融合算法基本杜绝了时间偏移的特性，而且明显增强了抗干扰的能力，实现PDR和GNSS的优势互补。但是我们也注意到这一现象，相比于真实轨迹，融合轨迹的长度缩短了，在图(b)和(c)中表现明显，这主要是融合算法中的步长估计仍然存在少量的累积误差导致的，因此在未来工作中步长估计的性能仍需提高。

5 总结与展望

本文通过对比分析PDR和GNSS两种不同定位方式所导致的航向与步长误差，提出了一套有效地PDR/GNSS融合框架。该框架给出了PDR和GNSS数据的同步方案以及关键的航向与步长融合算法。实验结果表明：无论是在航偏、抗干扰还是噪声等方面，融合算法都要优于单独的GNSS或PDR算法，较好地体现了绝对定位和相对定位的优势互补。但与此同时，步长估计精度仍需提高，GNSS航向可以考虑基于噪声更小的多普勒进行推导，融合算法在城市多径环境下的表现仍有待验证，这些都是未来工作的重点。

References (参考文献)

- [1] GROVES, Paul D . Principles of GNSS, Inertial, and Multi-sensor Integrated Navigation Systems[J]. Industrial Robot, 2013, 67(3):191-192.
- [2] Hsu L T , Gu Y , Huang Y , et al. Urban Pedestrian Navigation using Smartphone-based Dead Reckoning and 3D Maps Aided GNSS[J]. IEEE Sensors Journal, 2016, 16(5):1281-1293.
- [3] Jimenez A R, Seco F, Prieto C, et al. A comparison of Pedestrian Dead-Reckoning algorithms using a low-cost MEMS IMU[C]// IEEE International Symposium on Intelligent Signal Processing. IEEE, 2009:37-42.
- [4] Lan H, Yu C, El-Sheimy N. An Integrated PDR/GNSS Pedestrian Navigation System[J]. 2015, 342:677-690.
- [5] Liu Q., Ying R., Wang Y., Qian J., Liu P. (2018) Pseudorange Double Difference Algorithm Based on Duty-Cycled Carrier Phase Smoothing on Low-Power Smart Devices. In: Sun J., Yang C., Guo S. (eds) China Satellite Navigation Conference (CSNC) 2018 Proceedings. CSNC 2018. Lecture Notes in Electrical Engineering, vol 497. Springer, Singapore
- [6] Malkos, S. User Location Takes Center Stage in New Android OS: Google to Provide Raw GNSS Measurements. GPS World 2016, 27, 36.
- [7] Qian J, Ma J, Ying R, et al. RPNOS: Reliable Pedestrian Navigation on a Smartphone[M]// Geo-Informatics in Resource Management and Sustainable Ecosystem. Springer Berlin Heidelberg, 2013:188-199.
- [8] Shin S H, Park C G, Kim J W, et al. Adaptive Step Length Estimation Algorithm Using Low-Cost MEMS Inertial Sensors[C]// Sensors Applications Symposium, 2007. SAS '07. IEEE. IEEE, 2007:1-5.