# 2014 Combining Inertial Navigation and ICP for Real-time 3D Surface Reconstruction 翻译

## Abstract摘要

We present a novel method to improve the robustness of real-time 3D surface reconstruction by incorporating inertial sensor data when determining inter-frame alignment. With commodity inertial sensors, we can significantly reduce the number of iterative closest point (ICP) iterations required per frame. Our system is also able to determine when ICP tracking becomes unreliable and use inertial navigation to correctly recover tracking, even after significant time has elapsed. This enables less experienced users to more quickly acquire 3D scans. We apply our framework to several different surface reconstruction tasks and demonstrate that enabling inertial navigation allows us to reconstruct scenes more quickly and recover from situations where reconstructing without IMU data produces very poor results.

我们提出了一个新奇的方法，用来提升实时3D表面重建的鲁棒性，他通过在进行帧间配准时，整合惯性传感器数据（进行工作）。使用日用IMU，我们可以显著减少每帧ICP迭代所需的次数。我们系统还能判定ICP是否不可靠了，并使用惯性导航来正确地恢复跟踪，即使过了较长时间。这能使没经验的用户更快速的获取3D扫描结果。我们将框架应用在一些不同的表面重建任务中，并演示使用惯性导航允许我们更快速进行场景重建，并在没有IMU数据重建结果会很差的情形下恢复。

## Introduction and Related Work。引言与相关工作

With the advent of commodity real-time RGB-D sensors such as the Microsoft Kinect and the Asus Xtion, 3D reconstruction has **gained new momentum** in the computer graphics and **vision community**. In particular, online reconstruction approaches that involve real-time volumetric fusion have received significant attention [**CL96, NIH*∗*11, RV12, WJK*∗*12,CBI13,NZIS13**]. These methods require frame-toframe tracking in order to align input scan data. However, achieving high-quality alignments is challenging. One approach is to use visual SLAM (simultaneous localization and mapping) [Dav03, KM07, KRD08, NLD11]. Computationally cheaper tracking can be realized by leveraging depth data provided by RGB-D cameras, typically using a variant of the iterative closest point algorithm (**ICP**) [BM92,CM92]. While there are depth tracking approaches **beyond ICP (e.g., [HJS08])**, ICP has been established in particular in the context of volumetric fusion. ICP works by projectively aligning adjacent frames to determine correspondences between depth values and solving for the corresponding affine transformation. While ICP variants provide **suitable** tracking results in some scenarios, they fail at scanning scenes lacking sufficient geometric detail and **moderately** large frame-to-frame motion. Thus, 3D scanning applications such as Kinect Fusion (available in the Kinect SDK) are very limited in practice and require significant user training.

随着日用实时RGBD传感器的出现（比如 ms Kinect & asus Xtion），3D重建在CG和视觉领域重新获得了关注。特别是，在线重建，使用实时体素融合的方法，引起了相当大的关注。这些方法需要 f2f 的跟踪，以便配准输入数据。但是获取高质量的配准很难。一种方式是使用视觉SLAM。利用RGBD相机提供的深度数据，廉价计算的跟踪可被实现，通常使用ICP的一个变种。尽管有一些超越ICP的深度跟踪方法。ICP已经在体素融合这一语境中被固定使用。ICP通过投影配准相邻帧，来确立深度值之间的对应点关系，并且求解对应的仿射变换。尽管ICP变种在一些场景下提供了适当的跟踪结果，他们在缺乏充足几何结构细节、以及比较大的 f2f帧间运动情形下会失败。因此，3D扫描应用（如KinectFusion）在实际应用中非常有限，并且需要较多的用户训练。

In this work we focus on making real-time scanning accessible to non-expert users by improving the robustness of ICP tracking. We specifically consider scenarios where depth-based tracking fails, such as large inter-frame motion and planar surfaces (e.g., walls and floors). Most existing online scanning methods cannot detect or recover from poor ICP tracking, resulting in significant visual artifacts like multiple overlapping copies of the scene [BGC13].

Our approach is to achieve high-quality alignments by combining the ICP algorithm with sensor readings from an inertial measurement unit (IMU). An IMU measures inertial forces, typically using gyroscopes to measure angular velocity and accelerometers to measure linear acceleration. More sophisticated IMUs may contain other sensors, such as a magnetometer to correct for orientation drift and a barometer to provide altitude. IMUs are now ubiquitous as they are integrated into many popular smartphones.

本文中，我们致力于实现实时扫描，并能供非专业用户使用，通过提升ICP跟踪的鲁棒性。我们特别考虑基于深度的跟踪失败的场景，例如大的帧间运动以及平面表面（e.g. 墙壁、底面）。大多数现有实时扫描算法不能检测、或从欠佳的ICP跟踪中恢复，导致出现显著视觉缺陷/伪影，就像场景的多个重叠拷贝。

我们的方法是通过结合ICP与IMU读数，来获得高质量的配准结果。IMU测量的是惯性力，通常使用gyro测量角速度，acc测量线性加速度。更复杂的的IMU可能包含别的传感器，比如mag来校正朝向漂移以及气压计来提供海拔信息。IMU现在因为整合进了很多智能手机，所以无处不在。

This paper presents our method for incorporating inertial navigation into a surface reconstruction framework:

*‿*We introduce metrics to evaluate ICP tracking quality and determine when ICP tracking becomes unreliable.

*‿*We show how to use IMU data to improve ICP quality and reduce the number of ICP iterations needed, in some cases requiring only a single ICP iteration per frame.  
*‿*We demonstrate that using an IMU for dead reckoning enables real-time 3D scanning to correctly recover when ICP tracking fails.

本文提出我们的方法，用来整合惯性导航到表面重建框架中：

我们提出度量ICP跟踪质量的标准，判定ICP什么时候会失效。

我们展示如何使用IMU数据提升ICP质量，并降低ICP所需迭代次数，有些情形下，每帧仅需要一次ICP迭代。

我们演示了使用IMU进行定位，能使ICP跟踪失败时，正确地恢复实时3D扫描。

## ICP Failure Analysis。ICP失败分析

In order to perform camera tracking for data fusion, we employ the rigid point-to-plane ICP variant [CM92]. The goal of ICP is to determine the **cumulative** frame-to-frame transform **M** which is composed of a rotation and translation **T**(*tx,ty,tz*) *·* **R**(α*,*β*,*γ). In this work, we further need to determine the quality of an ICP match so that we can use inertial navigation when ICP is unreliable. In the remainder of this section, we detail the **specifics** of our ICP quality metric. We start by obtaining weighted (based on normal variation and distance) **projective correspondences** *{***s***i,***d***i}* between the current and the last observed frame. These define the non-linear least squares minimization problem



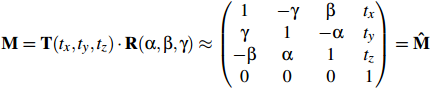
with **T** being a 3*×*4 translation matrix, and **R** a 3*×*4 rotation matrix with **R**(α*,*β*,*γ) = **Rz**(γ) *·* **Ry**(β) *·* **Rx**(α).

为了进行数据融合的相机跟踪，我们使用刚性p2plane ICP变种。ICP的目的是求解累计 f2f 变换 M，他有 T，R组成。本工作中，我们进一步需要确定ICP配准的质量，以便我们可以在ICP不可靠时使用惯性导航。在本节剩余部分，我们详细描述我们ICP质量度量标准。我们通过获取当前帧、上一帧之间的加权（基于法向变化、距离）射影对应关系 {si, di}开始，这定义为非线性最小二乘最小化问题：



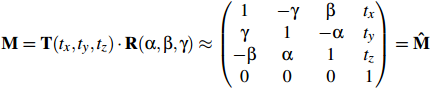
其中T是3x4位移矩阵，R是3x4旋转矩阵，有**R**(α*,*β*,*γ) = **Rz**(γ) *·* **Ry**(β) *·* **Rx**(α).

Following Low [Low04], we linearize the rotations (assuming small rotation angles):



We now rearrange (**M***·***s***i −***d***i*)*·***n***i* into a linear system **Ax** = **b**, where **x** = (α*,*β*,*γ*,tx,ty,tz*)*T* . Next, we solve for **x** in the optimal least squares formulation **ATAx** = **ATb**. We accomplish this using a parallel reduction on the GPU [HSO07] to build both **ATA** (6 *×* 6), and **ATb** (6*×*1) for all valid correspondence pairs *{***s***i,***d***i}*. We then solve the linear 6 *×* 6 system for **x** on the CPU using a singular value decomposition which allows us to compute **M***·***x**. Since **x** approximates the solution of the non-linear least squares system, we iterate this process until convergence.

根据Low04，我们线性化旋转量（假定旋转角很小）



我们重新组织(M·si −di)·ni 到一个线性系统 Ax=b中，其中x = (α,β,γ,tx,ty,tz)T 。下一步，我们求解最小二乘式AtAx=Atb的最优解x。我们使用并行GPU算法计算AtA、Atb，对于所有的{si, di}对应点对。我们然后求解6x6系统的x，在CPU上使用SVD。因为x为所求的非线性最小二乘系统的近似解，我们迭代此过程，直到收敛。

To determine ICP quality, we incorporate the residual **r**= *||***Ax***−***b***||*2 2, the number of non-rejected correspondences *n*, and the summed confidence weight ∑*i wi* (that are used to weight the rows of **A**) into the GPU reduction at each ICP step. We further determine the condition of the system matrix **A** using the previously obtained singular values κ(**A**) = σ σ((**A A**))*max min* . This specifies the descriptiveness of the geometric features; e.g., if the algorithm aligns two planar shapes, σ(**A**)*min* will be close to zero, and in the limit κ(**A**) *→ ∞* which means that the system becomes **underconstrained** and **A** is ill-conditioned. This obtained ICP error allows us to determine ICP convergence, and when to stop iterating. We further employ empirically determined error thresholds to identify lost tracking states as required by our inertial navigation approach described in Section 3.

为了确定ICP质量，我们整合残差 **r**= *||***Ax***−***b***||*2 2 、非拒绝对应关系n，以及置信权重和∑*i wi* （A中每一行的权重）到GPU每步ICP降解过程中。我们进一步确定系统中矩阵A的条件数(condition)，使用之前得到的奇异值。它描述了几何特征的“描述性”；例如，如果算法配准两个平面形体，，结果，意味着系统难以约束，A是病态的。这一解得的ICP误差允许我们确定ICP收敛性，以及何时停止迭代。我们进一步使用经验性误差阈值，来确定跟踪失败的状态，以便使用我们小节3所描述的惯性导航方法。

## ICP Correction using IMU Data使用IMU数据校正ICP

IMUs are comprised of many different sensor types. For this work, we only assume that the IMU is able to estimate the rigid motion of the scanner over a time range. We represent this with a function *InertialEstimate*(*ta, tb*), which returns a matrix representing the rigid transformation of the sensor’s coordinate frame from time *tb* to *ta*. If the IMU has no error, this transform is sufficient to perform surface reconstruction and ICP is unnecessary. In practice, the IMU estimate accumulates error from many different sources such as sensor noise, global drift, and low sampling rates.

IMU包含许多不同类型传感器。本文中，我们仅假设IMU在一定时间范围内可以估计扫描器的刚性运动。我们用函数*InertialEstimate*(*ta, tb*) 来表示，它返回从时间tb->ta，传感器坐标系的刚性运动。如果IMU没有误差，这一运动足以用于表面重建，ICP也就不必要了。事实上，IMU有累计误差，其来源有很多：传感器噪声，全局漂移，以及低采样率。

### Improved ICP Initialization提升ICP初始化

Typically, the ICP algorithm for frame *t* is initialized with the transform computed for frame *t −* 1. This approach can converge to a good result if the motion between frames is small, the previous frame’s transform is accurate, and there are strong features in the frame. Otherwise convergence often becomes slow and a bad local minima is reached.

We use the rigid transform estimated by inertial navigation to provide a **significantly better initial guess** for the ICP algorithm. We compute ∆*IMU*, the inertial estimate from frame *t −*1 to frame *t*. We use the previous frame’s transform followed by ∆*IMU* as our initial seed for ICP. When ∆*IMU* is accurate, this allows ICP to converge to the correct transform in a very small number of iterations. It also improves robustness in cases when ICP produces many possible solutions, such as nearly planar regions. In Section 4, we show that typically only a single ICP iteration is needed when using the inertial estimate.

典型的ICP算法，在t帧，使用t-1帧的姿态做初始化。这种方式在两帧之间运动很小、前一帧姿态很精确、且帧中特征较强的时候能收敛到较好的结果；否则通常收敛缓慢，且得到差的局部极小值。

我们使用惯性导航得到的刚性运动值，为ICP算法提供显著更优的初值。我们计算，即从t-1到t帧的姿态变化量的IMU估计值。我们