Demo Abstract: Visual and Inertial Sensor Fusion for Mobile X-ray Detector Tracking

Yang Zhao
GE Research
Niskayuna, New York

Eric Tkaczyk
GE Research
Niskayuna, New York

Feng Pan
Spark insights
Boston, Massachusetts

ABSTRACT

Robust 3D pose tracking of an object is a critical technique for various mobile sensing applications. Computer vision-based pose tracking method provides a cost-effective solution, but it is sensitive to occlusion and illumination change issues. In this work, we propose a novel visual-inertial sensor fusion framework and demonstrate the real-time implementation of a tightly-coupled sensor fusion algorithm: inertial perspective-n-point (IPNP) algorithm. With measurements from an inertial measurement unit (IMU), the prototype system only needs to detect two keypoints to track all six degrees of freedom of a planar object, e.g., a mobile X-ray detector, a 50% reduction on required number of keypoints, compared with the vision-based perspective-n-point algorithm.

CCS CONCEPTS

• **Computer systems organization** → *Embedded and cyber-physical systems*; • **Human-centered computing** → **Mobile devices**.

KEYWORDS

Pose estimation, sensor fusion, mobile sensing

ACM Reference Format:
Yang Zhao, Eric Tkaczyk, and Feng Pan. 2020. Demo Abstract: Visual and Inertial Sensor Fusion for Mobile X-ray Detector Tracking. In *The 18th ACM Conference on Embedded Networked Sensor Systems (SenSys '20), November 16–19, 2020, Virtual Event, Japan.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3384419.3430435

1 INTRODUCTION

Object position and orientation estimation is important for various mobile sensing applications. In the field of medical imaging, accurate 3D tracking of medical devices can not only improve the quality of medical images, but also speed up the imaging workflow, and even enable autonomous imaging and robotic operations. In mobile X-ray imaging, accurate pose tracking of the X-ray detector is a critical technique to a 2D X-ray imaging system, as well as more innovative 3D tomosynthesis system.

For high-accuracy 3D pose tracking, existing techniques and systems include motion capturing technique, e.g., Vicon and Optitrack systems; electromagnetic tracking technique, e.g., trakSTAR system, etc. However, the cost of these solutions is prohibitive for many applications to adopt them. As a low-cost alternative, various computer vision methods have been developed for augmented reality, robotics and other applications [3]. For example, marker-based method uses fiducial markers with special patterns, such as ARTag, AprilTag to identify and locate objects. Visual odometry (VO) and vision-based simultaneous localization and mapping techniques do not need any fiducial markers. These feature matching-based methods can extract features from the environment to track the ego-motion of the camera [3]. However, both marker-based and feature-based methods are essentially vision solutions, which are sensitive to occlusion, light condition changes and image acquisition errors. To make vision-based pose tracking more robust, visual-inertial odometry (VIO) [2] combines complimentary inertial measurement unit (IMU) sensor with visual sensor. As reviewed in [2], some VIO algorithms use the Kalman filter (KF) or extended Kalman filter (EKF) framework to integrate these two sensing modalities; other algorithms use loosely-coupled or weighted average approach. However, the performance of these methods is affected by motion dynamic model or weight parameters, and installing cameras on objects can be intrusive for many applications.

We propose an alternative visual-inertial solution, and develop a prototype to demonstrate a tightly-coupled sensor fusion algorithm called inertial perspective-n-point (IPNP) algorithm. The same as VIO, IPNP also integrates visual and inertial sensors, but it differs from VIO in the following ways. First, the IPNP algorithm uses an iterative optimization approach, and it does not rely on any motion model or weight parameters as in the KF or EKF framework. Second, a VIO system has both inertial and visual sensors onboard the object, but we only attach an IMU sensor with Bluetooth connectivity on the object, which is less intrusive and also reduces the payload on the object. Finally, IPNP significantly reduces required number of visual keypoints, compared to the classical perspective-n-point (PnP) algorithm [1] and the five-point visual odometry algorithm [3], which we describe in details next.

2 METHOD AND SYSTEM

2.1 Sensor Fusion Algorithm

Let $\mathbf{x}$ denote the pose vector: $\mathbf{x} = [\theta_x, \theta_y, \theta_z, t_x, t_y, tz]^T$ (orientation as θ, position as t), $\mathbf{y}$ denote the XY coordinates of the image markers, i.e., keypoints: $\mathbf{y} = [x_1, y_1, x_2, y_2, \cdots, x_N, y_N]^T$, and let $\mathbf{m}$ represent the model parameters, e.g., the camera intrinsic parameters, then we use a forward model function $\mathbf{y} = f(\mathbf{x}, \mathbf{m})$ to estimate the 2D locations of the image keypoints $\mathbf{y}$, given the pose $\mathbf{x}$ and model parameter vector $\mathbf{m}$.

For the classical perspective-n-point (PnP) algorithm [1], the forward model f can be formulated as: $s\tilde{\mathbf{p}} = \mathbf{A}[\mathbf{R}\ \mathbf{t}]\tilde{\mathbf{P}}$, where s is a

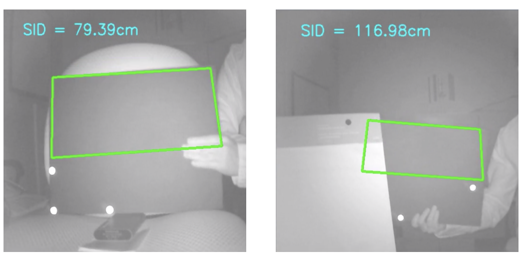
SenSys '20, November 16–19, 2020, Virtual Event, Japan

ACM ISBN 978-1-4503-7590-0/20/11…$15.00
https://doi.org/10.1145/3384419.3430435

scale factor, $\mathbf{A}$ is the camera intrinsic matrix, e.g., model parameters in $\mathbf{m}$, $[\mathbf{R}, \mathbf{t}]$ are rotation matrix and translation vector that contain the camera extrinsic parameters, i.e., the pose parameters in $\mathbf{x}$, and $\tilde{\mathbf{p}} = [x, y, 1]^T$ and $\tilde{\mathbf{P}} = [X, Y, Z, 1]^T$ are augmented vectors by adding 1 as the last element to the 2D point $\mathbf{p} = [x, y]^T$ and 3D point $\mathbf{P} = [X, Y, Z]^T$, respectively [1]. Once the forward model is formulated in this way as a linear matrix equation, a least squares solution can be used for estimating the pose parameters.

A limitation of the PnP algorithm is that its performance significantly depends on the robust detection of keypoints. Occlusion, illumination changes, and errors in correspondences can all lead to failure of keypoint detection. The IPNP algorithm improves the robustness of pose estimation by combining orientation information from an IMU with keypoints from images.

The rotation matrix $\mathbf{R}$ relates the world coordinate system to the camera coordinate system. While the PnP algorithm uses a perspective projection model to obtain $\mathbf{R}$, we can also use the gyroscope measurements from an IMU to compute $\mathbf{R} = \mathbf{R}_z(\gamma)\mathbf{R}_y(\beta)\mathbf{R}_x(\alpha)$, where α, β and γ are Euler angles about axes x, y and z, respectively. Thus, we can use the orientation measurements from the IMU in the PnP forward model, if we can calculate the orientation offset between the IMU and camera systems. Fortunately, we can obtain the 3D orientation offset during a calibration period when at least four keypoints are detected by the computer vision algorithm.



(a) Pose estimate from three keypoints **(b) Pose estimate from two keypoints**

Figure 1: Accurate pose estimation of the top-half of a square obtained using the IPNP algorithm as indicated in green overlay for the case with three (a) and two (b) markers visible (SID indicates the distance from the camera).

Once we use Euler angles to compute the rotation matrix $\mathbf{R}$, we only need to compute the translation vector $\mathbf{t} = [t_x, t_y, t_z]^T$ in the forward model. Thus, only two keypoints (XY coordinates) are required to solve an overdetermined problem. Once we use the forward model with inertial and visual sensor data to predict the XY location of all keypoints, i.e., $\mathbf{y}$, we can compute the Jacobian matrix $\mathbf{J}_{i,j} = d\mathbf{y}_i/d\mathbf{x}_i$ and use gradient descent-based method to solve an optimization problem. For a planar object, a minimum of four keypoints are required to derive a unique PnP solution [1]. Our IPNP algorithm only requires detection of two keypoints, a 50% improvement in terms of minimum number of keypoints.

2.2 Hardware and Sensors

We use commercial-off-the-shelf visual and IMU sensors to demonstrate the IPNP algorithm. In order to simplify the detection of keypoint markers in the camera image, we place an infrared (IR) filter on top of the lens of a regular camera as our IR camera. We cut chromatte tape into one inch circles to create retroreflective markers, and place four markers on the surface of the planar object, so that the IR camera can easily detect them based on brightness thresholding. Note that we follow the standard camera calibration procedure with a checkerboard to calculate the intrinsic camera parameters. Markers are arranged with specific spacing and in a pattern such that unique identification of each marker can be performed from the camera view. To ensure PnP accuracy, markers are far apart enough to create separation by 20 or more pixels in the camera image at the working distance.

For the IMU sensor, we choose a 9-axis LPMS motion sensor with Bluetooth connectivity, so we can collect the orientation measurements wirelessly on a laptop. The LPMS sensor provides both the angular velocity measurements and the orientation measurements, since it has a gyroscope and a magnetometer. While we can integrate angular velocity to obtain orientation, we choose to use the orientation measurements directly, and we calculate the orientation offset between the camera frame and the IMU frame during the calibration, as discussed in Section 2.1.

3 DEMO DESCRIPTION

We configure the hardware sensor as follows to set up the demo. We place our IR camera on a fixed location and connect it to our demo computer. Also connected to the computer is a Bluetooth USB dongle, from which we can collect the IMU sensor data. We choose a planar object such as a paper plate as the tracking target. We put the IMU sensor on one side of the plate, and four retroreflective markers on the other side. We place four markers following a special geometric pattern so that we can identify these markers. Note that keypoint correspondence can be achieved by other ways, such as feature matching in markerless methods. Our iterative optimization-based sensor fusion framework works for both marker-based and markerless methods, and we only demonstrate the fusion of inertial sensor for the marker-based method.

Before the demo, a simple calibration is performed by placing the object in front of the camera with all four markers visible to the camera. After that, users can move the object around in the field of view of the IR camera, and see the real-time location and orientation estimate of the object from a window on the computer. As illustrated in Figure 1, the green bounding box shows the contour of the planar object from the orientation estimate, and source-to-image distance (SID) indicates the estimate of the distance from the camera to the center of the object. To test the accuracy and robustness of the IPNP algorithm, users can block certain markers to see if the demo prototype can still estimate the 3D pose accurately. While the classical PnP algorithm requires at least four keypoints [1], our IPNP algorithm works with three and two markers visible to the IR camera, as shown in Figure 1 (a) and (b), respectively.

REFERENCES

[1] Raul Acuna and Volker Willert. 2018. Robustness of control point configurations for homography and planar pose estimation. *CoRR* abs/1803.03025 (2018).
[2] J. Delmerico and D. Scaramuzza. 2018. A Benchmark Comparison of Monocular Visual-Inertial Odometry Algorithms for Flying Robots. In *2018 IEEE ICRA*.
[3] D. Scaramuzza and F. Fraundorfer. 2011. Visual Odometry [Tutorial]. *IEEE Robotics Automation Magazine* 18, 4 (Dec 2011), 80–92.