PennCOSYVIO: A Challenging Visual Inertial Odometry Benchmark

Bernd Pfrommer[1] Nitin Sanket[1] Kostas Daniilidis[1] Jonas Cleveland[2]

***Abstract*— We present PennCOSYVIO, a new challenging Visual Inertial Odometry (VIO) benchmark with synchronized data from a VI-sensor (stereo camera and IMU), two Project Tango hand-held devices, and three GoPro Hero 4 cameras. Recorded at UPenn's Singh center, the 150m long path of the hand-held rig crosses from outdoors to indoors and includes rapid rotations, thereby testing the abilities of VIO and Simultaneous Localization and Mapping (SLAM) algorithms to handle changes in lighting, different textures, repetitive structures, and large glass surfaces. All sensors are synchronized and intrinsically and extrinsically calibrated. We demonstrate the accuracy with which ground-truth poses can be obtained via optic localization off of fiducial markers. The data set can be found at https://daniilidis-group.github.io/penncosyvio/.**

I. Introduction

In this paper we present a new, challenging data set aimed at benchmarking and supporting the development of new Visual Inertial Odometry (VIO) algorithms.

Originating from the Greek words "odos" (way) and "metron" (measure), odometry is the art and science of estimating traveled distances based on sensor readings. A wide variety of different sensors can be used for this purpose, including global positioning system (GPS) receivers, laser range finders, radio frequency (RF) receivers, sonar, cameras and Inertial Measurement Units (IMUs), with VIO being based on the latter two. The key challenge lies in developing algorithms that efficiently fuse multi-sensory data [1] [2] and estimate the device's motion as quickly and precisely as possible, often also simultaneously building a map of the environment (SLAM) [3]. A great amount of research has been done in this area, lately motivated by the race to build autonomous cars [4] [5] and aerial vehicles [6] [7].

Most recently, augmented reality is coming to cell phones [8], hand-held devices such as Google's Project Tango [9], and head-mounted displays, creating urgent demand for extraordinarily accurate odometry and head tracking in order for augmented reality objects to remain stationary when users move. Pure visual odometry [10] [11] [12] [13] [14] has enjoyed considerable success, but at the moment, VIO based on the fusion of camera (to eliminate drift and establish loop closure) and IMU data (for rapid rotations) [15] [16] [17] appears to be the most promising approach.

In this context we present a VIO benchmark for which we simultaneously record a number of different camera and IMU streams, including data from two hand-held Google Project Tango devices. Our data set comes with an accurate reference (ground truth) position, which is important for benchmarking VIO algorithms. As an immediate application, this allows us to quantitatively assess the reliability of the Tango's proprietary on-device VIO software. Because our method of obtaining ground truth positions via fiducial markers can be used in other contexts as well, we describe it here in detail.

The present work advances the state of the art by containing data from a unique mix of cameras and IMUs, being recorded from a hand-held platform, and by providing accurate ground truth over a relatively large roaming area in the presence of glass surfaces that make localization by other means such as laser range finders difficult.

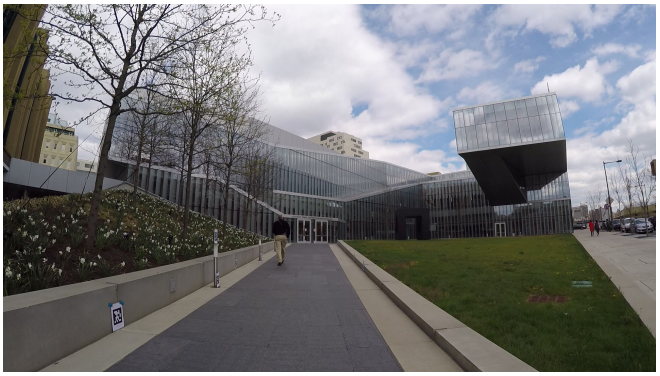

Fig. 1. Outside the Singh center, as seen through GoPro camera C2.

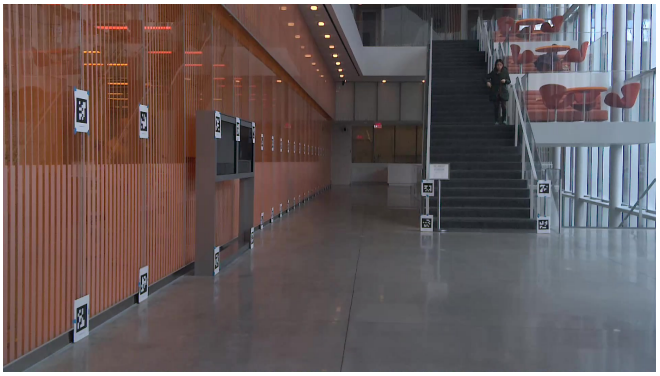

Fig. 2. Inside the Singh center, recorded by the Bottom Tango RGB camera.

II. Related Work

Intense work on SLAM algorithms has produced a large number of related data sets, to which links can be found on sites like OpenSLAM [18] or Radish [19]. The vast majority of them uses ground vehicles as carriers and employs laser

[1]University of Pennsylvania School of Engineering and Applied Science
[2]Cognitive Operational Systems, LLC

Financial support by the I/UCRC Rose-Hub NSF-IIP-1439681 and the ARL RCTA W911NF-10-2-0016 is gratefully acknowledged.

range finders to facilitate localization and mapping. For brevity we discuss only a few popular and representative benchmarks that we consider most closely related to our work. Table I gives an overview of their characteristics.

The very popular KITTI [5] odometry benchmark stands in for several similar data sets [23] [24] [25] [26] geared towards autonomous navigation, where the sensors are carried by a ground vehicle. As all KITTI sequences are recorded outdoors, the authors can use a highly precise GPS/IMU combination to establish a six degree of freedom (6 DOF) ground truth trajectory. The camera frame rate of 10Hz suffices to capture a car's motion, but is fairly low for a hand-held device.

For indoor micro aerial vehicle (MAV) operations, the EuRoC data set [7] has set a high bar by carefully synchronizing the stereo camera and IMU data recorded by a VI-sensor to ground truth pose information provided by an external localization system, either a Vicon Motion Capture (MoCap) system, or, when roaming in the much larger machine hall, a Leica MS 50 laser tracker. While the ground truth is highly precise, the recording is strictly indoors, and for the larger roaming area does not provide full 6 DOF information.

From the TU Munich comes another strictly indoors data set aimed at RGB-D SLAM benchmarking [20], recording RGB and depth information with a Microsoft Kinect sensor. Although the paths traveled are rather long, ground truth is provided only in the relatively small area covered by the MoCap system.

The Rawseeds [21] data set is recorded on a ground robot, comes with ground truth and covers both indoors and outdoors environments. Indoor ground truth however is not provided for the mixed indoor/outdoor situations, the indoors poses only include 2D location and heading direction, and the GPS-based outdoor ground truth does not contain orientation information. The authors also use fiducial markers (AR tags), but attach them to the robot and mount cameras in various places of the room to triangulate the position much like a MoCap system does. In contrast, we attach cameras to the sensor rig, and place static tags as landmarks, which scales better with area size.

The NCLT data set [22] is the only other work we are aware of with accurate ground truth and some indoors/outdoors transitions. Collected at the University of Michigan using a Segway as carrier, its focus is on autonomous systems and seasonal changes over a whole year. The omnidirectional camera (Pointgrey Ladybug3) records images at 5Hz, which is sufficient for VIO on ground vehicles, but is at the lower end for hand-held devices targeted by our data set.

III. Data Set Characteristics

A. Location

We chose the University of Pennsylvania's Singh Center of Nanotechnology as the site for recording the data set for several reasons. It has a long outdoors path, extended glass walls on the outside as well as inside, and features many repetitive patterns (see Figures 1 and 2) that will stress test visual odometry algorithms. The indoor lighting is a mix of artificial and natural daylight, bright enough to capture sufficiently sharp images of the AprilTag fiducial markers [27] employed. Because the Singh center was built very recently (2013), we are able to source accurate architectural CAD drawings and elevation charts for indoors and outdoors that we use to eliminate error accumulation from our own tape-measured distances. We spot checked several of the dimensions in the CAD file with the measuring tape and even for distances of more than 10m the discrepancies were less than 2cm. This is owed due to the highly accurate laser equipment employed in the construction of modern buildings.

B. Sensor Platform

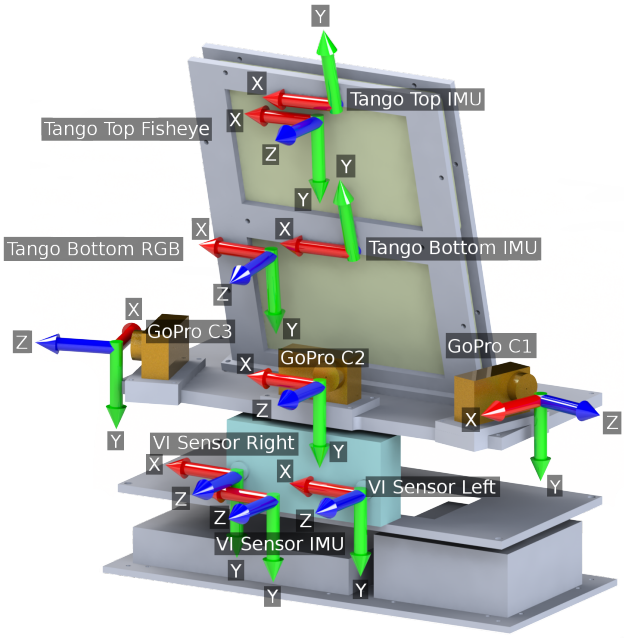

Fig. 3. Sensor rig with orientations of all ten sensors. The Tango Top records IMU data and fisheye camera video, the Tango Bottom records IMU, RGB video, and pose estimates from Google's proprietary VIO algorithm. The three GoPro cameras C1/C2/C3 record HD video of fiducial markers to establish ground truth. The VI-sensor at the bottom captures synchronized monochrome stereo video and IMU data. Extrinsic calibration is provided with the data set.

All devices are attached to a rig (Figure 3) constructed from laser cut MDF board and spacers. A list of sensors and their characteristics can be found in Table II. A Intel NUC5i7RYH (square shaped box at the bottom of the platform) is connected via USB 3.0 to the forward-facing VI-sensors above it, and stores the incoming data to a ROS bag. The flat slab next to the NUC is a 32Ah Li-Ion battery providing power to NUC and VI-sensor.

Precise synchronization was a design goal of the VI-sensor. The two cameras and IMU are hardware synchronized, and the sensor timestamps (not recording timestamps!) contained in the ROS bag must be used to recover full quality. These are also the timestamps we use for synchronization. Since this data is targeted at stereo-based algo-

	PennCOSYVIO	KITTI [5]	EuRoC [7]	RGBD-SLAM [20]	Rawseeds [21]	NCLT [22]
year	2016	2012	2016	2012	2009	2016
focus	hand-held VIO	self-driving car	MAV VIO	RGB-D SLAM	SLAM	long term SLAM
environ.	in/outdoors	outdoors	indoors	indoors	in/outdoors	in/outdoors
carrier	hand held	car	hexacopter	ground robot/hand held	ground robot	Segway
≈ dist.	150m per sequence	39km total	up to 131m	up to 40m	up to 1.9km [23]	up to 7.5km
cameras	- 4 RGB: 1920x1080 @30Hz - 1 stereo gray: 2x752x480 @20Hz - 1 fisheye gray: 640x480 @30Hz	- 1 stereo RGB: 2x1392x512 @10Hz - 1 stereo gray: 2x1392x512 @10Hz	- 1 stereo gray: 2x768x480 @20Hz	- 1 RBG-D: 640x400 @30Hz	- 1 trinocular gray: 3x640x480 @15Hz - 1 RGB: 640x480 @30Hz - 1 fisheye RGB: 640x640 @15Hz	- 1 omnidirectional color: 6x1600x1200 @5Hz
IMUs	- 2 accel. @128Hz - 2 gyros @100Hz - 1 acc/gyro @200Hz	- 1 accel/gyro @10Hz (OXTS RT 3003)	- 1 accel/gyro @200Hz	none	- 1 accel/gyro @128Hz	- 1 accel/gyro @100Hz - 1 fiber gyro @100Hz
GPS	none	- 1 OXTS RT 3003 @10Hz	none	none	- 1 RTK GPS @5Hz	- 1 GPS @5Hz - 1 RTK GPS @1Hz
Laser	none	- 3D: Velodyne HDL-64E @10Hz	- 3D: Leica MS 50 (stationary) @20Hz	none	- 2D: 2 Hokuyo @10Hz - 2D: 2 SICK @75Hz	- 3D: Velodyne HDL-32E @ 10Hz - 2D: Hokuyo @40Hz - 2D: Hokuyo @10Hz
3D point cloud	no	yes	yes	yes	no	yes
ground truth	6DOF (visual tags)	6DOF (GPS/IMU)	- 6DOF (MoCap) - 3D (laser tracker)	6DOF (MoCap)	- 3D (GPS) - 2D+heading (visual tags/laser)	6DOF (GPS/IMU/laser)
≈ accur.	15cm	10cm	1mm	1mm	few cm/m	10cm

TABLE I

OVERVIEW OF RELATED DATA SETS AND BENCHMARKS

rithms, and given storage space constraints while recording the sequence, we provide rectified images only.

Mounted above the GoPro cameras are two Google Project Tango 7in "Yellowstone" tablets. The top one (Tango Top) records video from the fisheye camera and IMU data, whereas Tango Bottom collects RGB camera, IMU, and VIO pose information.

IV. DATA SET CONTENT

The PennCOSYVIO benchmark consists of four sequences that follow a path similar to the one shown in Figure 4. It starts outdoors at the south west end of the walkway, goes up a slight slope (1m elevation difference) for about 30m, and enters through the left main door into the lobby. Once inside it heads east, does a 360° rotation, then continues on until a 180° left turn leads back the same path and out to the starting point. The 360° turn gauges the ability of VIO algorithms to handle rotations in the yaw direction.

Of the four sequences listed in Table III, two (A-S and A-F) are for training and come with ground truth provided, the other two are for testing. Test trajectories should be submitted as directed on the benchmark web site, and will be evaluated against the ground truth using the metrics described in Section VIII. For all sequences the raw data as recorded is provided, except that the time stamps are shifted for synchronization (Section VI-C) and expressed in seconds uniformly. The `mp4` files recorded by the GoPro cameras C1/C2/C3 and on Tango Bottom/Top are cut on key frames without re-encoding, and are accompanied by text files with time stamps for each frame. For convenience we also deliver decoded frames in `png` format for all cameras. The VIO poses from the Tango Bottom are only given for the training sequences.

Aside from the actual VIO sequences, the data set contains text files with the results of our intrinsic and extrinsic calibration, and the corresponding calibration images.

V. COORDINATE SYSTEMS AND NOTATION

In this paper, and for further documentation provided with the data set, vector coordinates are left-superscripted with W (world), B (body, the camera rig to which all sensors are rigidly attached), or the label of the sensor depending on the coordinate basis. The notation for transformations between coordinate systems follows a similar pattern, e.g. ${}^{\mathrm{B}}\mathbf{T}_{\mathrm{C1}}$ transforms coordinates $\mathbf{X}$ from camera C1 to the body system:

$$ {}^{\mathrm{B}}\mathbf{X} = {}^{\mathrm{B}}\mathbf{T}_{\mathrm{C1}}\,{}^{\mathrm{C1}}\mathbf{X}. \tag{1} $$

Symbols without indicators are assumed to be in the world (W) coordinate system.

VI. POST PROCESSING

A. Intrinsic Calibration

To facilitate intrinsic calibration, we record video footage of a large 7x8 checker board pattern with a square size of 108mm, and cut out a sufficient number of frames from different points of view. For the Tango Bottom (RGB) and the VI-sensor cameras we use the MATLAB calibration toolbox with two radial distortion coefficients. The GoPro and Tango Top fisheye lenses are calibrated with the OCamCalib [29]

Sensor	Characteristics
C1,C2,C3	- GoPro Hero 4 Black - RGB 1920x1080@30fps on "W" (wide) setting - rolling shutter - FOV: 69.5° vert., 118.2° horiz.
VI-Sensor [28]	- Skybotix integrated VI-sensor - stereo camera: 2 Aptina MT9V034 - gray 2x752x480 @ 20fps (rectified), global shutter - FOV: 57° vert., 2 x 80° horiz. - IMU: ADIS16488 @200Hz
Tango Bottom	- Google Project Tango "Yellowstone" 7in tablet - RGB 1920x1080@30fps, rolling shutter - FOV: 31° vert., 52° horiz. - proprietary VIO pose estimation - accelerometer @128Hz - gyroscope @100Hz
Tango Top	- Google Project Tango "Yellowstone" 7in tablet - gray 640x480@30fps, global shutter - FOV: 100° vert., 132° horiz. - accelerometer @128Hz - gyroscope @100Hz

TABLE II

LIST OF DEPLOYED SENSORS AND THEIR CHARACTERISTICS

sequence	purpose	pace	distance	time	ground truth
A-S	training	slow	149.2m	155s	yes
A-F	training	fast	148.8m	96s	yes
B-S	test	slow	similar	167s	hidden
B-F	test	fast	similar	101s	hidden

TABLE III

THE FOUR SEQUENCES PROVIDED WITH THE DATA SET

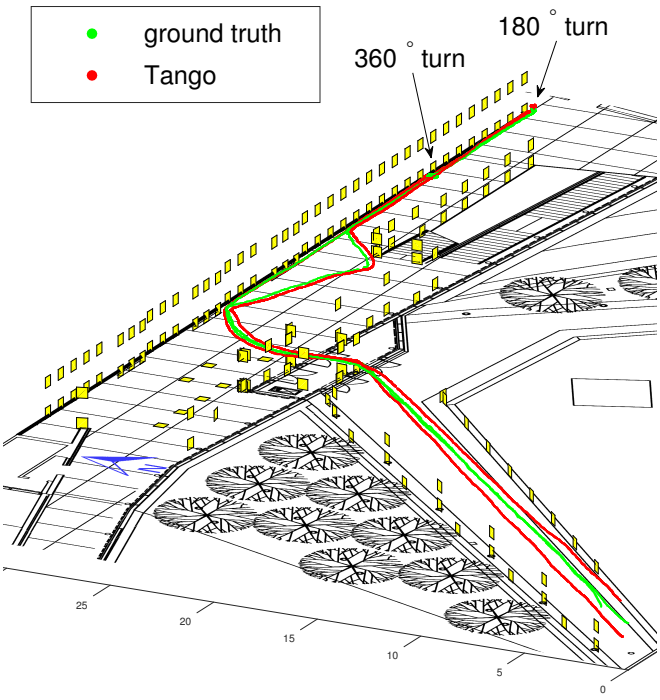

Fig. 4. Visualization of sequence A-S. The yellow squares (enlarged three times for visibility) are the AprilTag markers used for ground truth measurement (green trajectory). Also shown is the pose given by the Tango's proprietary VIO algorithm. Axis units are meters.

toolbox, using a four-parameter imaging function which performs significantly better than the radial distortion model implemented in the MATLAB toolbox. For the GoPros we obtain well less than one pixel average reprojection error on the test images. We use the imaging function to undistort the GoPro frames into equivalent perspective camera images so the fiducial markers can readily be detected by the AprilTag library, and the corner points can subsequently processed by our extrinsic calibration and localization code base. We crop the outer regions of the undistorted image such that all pixels are valid and up-sample the inner region to preserve the original image format. The required MATLAB scripts for this process are included with the data set.

B. Extrinsic Calibration

First the direction of the optical axes of all cameras are determined by recording video of a calibration target consisting of AprilTags placed at well-known locations on a wall. From the video footage 16 synchronized frames are cut and corner points extracted from each tag in the images. With the intrinsic calibration parameters, the pixel coordinates of the corner points, and the 3D positions of the tag corners known, the extrinsic calibration can then be determined by minimizing the reprojection error over all camera and rig poses.

To deal with non-overlapping camera views and with cameras sometimes having no view of the calibration target at all, a pose graph optimizer based on the GTSAM [30] library (v 3.2.1) is used. For illustration, Figure 5 shows a subgraph of the full factor graph used for extrinsic calibration. Circles represent hidden variables to be optimized, solid squares represent factors connecting one or more variables, and free labels next to factors denote the measured data used.

At the bottom of the graph a `Prior` factor [30] with a narrow Gaussian of $\sigma = 25\,\mathrm{mm}$ pins the 3D coordinates $\mathbf{X}_i, i = 1 \ldots 4$ of a single tag's four corners to their measured locations $\mathbf{M}_i$. Thus the optimizer is allowed to move the corner points on the order of a few centimeters, accounting for some inaccuracy in tag placement.

The tag corner points are viewed in the first ($p = 1$, left side of the graph) and second ($p = 2$, right side) rig position extracted from the video. The corresponding unknown rig transformations and 3D points in the body system B(p) are related by $\mathbf{X}_i = \mathbf{T}_{\mathrm{B(p)}}\,{}^{\mathrm{B(p)}}\mathbf{X}_i$, which is modeled [30] by a `ReferenceFrameFactor` with $\sigma = 1\,\mathrm{mm}$.

Finally, the image-level 2D pixel coordinates $\{{}^{\mathrm{C1(p)}}\mathbf{u}_i, {}^{\mathrm{C2(p)}}\mathbf{u}_i\}$ serve as input data to a

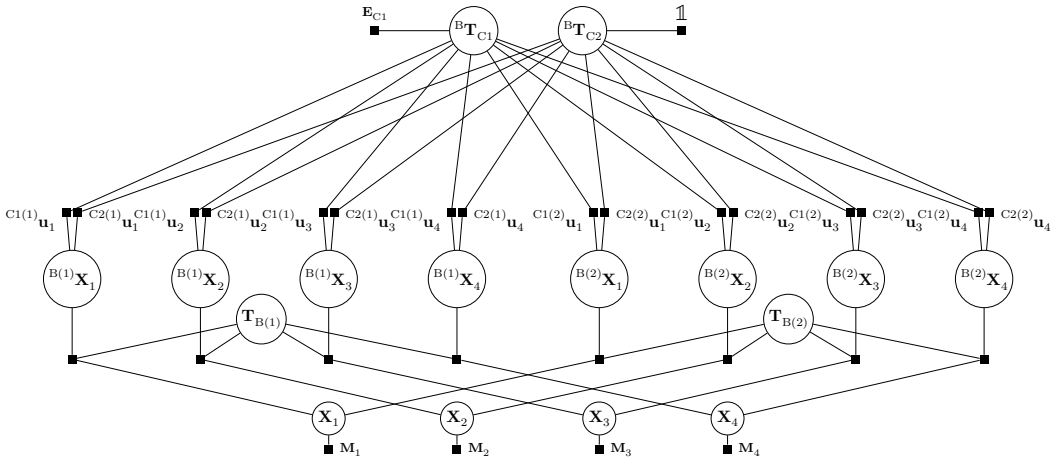

Fig. 5. Example factor graph for a two-camera calibration, viewing the four corners of an AprilTag from two different rig positions. Circles are hidden variables to be optimized, solid squares are factors connecting them and labels next to factors denote the measured data used. A transform $\mathbf{T}_{\mathrm{B(p)}}$ dependent on rig-position p relates the tag corner points $\mathbf{X}_i$ to their body-system coordinates ${}^{\mathrm{B(p)}}\mathbf{X}_i$. Measured data are the 3D tag corner point locations $\mathbf{M}_i$ and the pixel locations ${}^{\mathrm{C(1,2)(p)}}\mathbf{u}_i$ of their projection onto the camera sensor. The measured data, combined with location priors $\mathbf{E}_{\mathrm{C1}}$ and $\mathbb{1}$ ultimately constrains the extrinsic calibration transforms ${}^{\mathrm{B}}\mathbf{T}_{\mathrm{C1}}$ and ${}^{\mathrm{B}}\mathbf{T}_{\mathrm{C2}}$ (top of graph).

`GenericProjectionFactor` ($\sigma = 2$ pixels) relating the body-frame coordinates ${}^{\mathrm{B(p)}}\mathbf{X}_i$ to the sought-for extrinsic calibration transforms $\{{}^{\mathrm{B}}\mathbf{T}_{\mathrm{C1}}, {}^{\mathrm{B}}\mathbf{T}_{\mathrm{C2}}\}$. Since introducing the rig transformation adds an additional degree of freedom, we can now choose one of the camera's extrinsic calibration freely. We pick the forward-facing camera C2 as the rig center by connecting it to the identity transform with a tight ($\sigma = 1\,\mathrm{nm}$) `Prior` factor. On top of that, we constrain the location (not orientation!) of the cameras with a prior ($\sigma = 1\,\mu\mathrm{m}$) to their positions $\mathbf{E}$ as measured manually with a ruler and triangulation. Especially for the fisheye lenses with low resolution this improves accuracy, although fixing the locations makes little difference in practice, and moves the camera positions by at most 30mm (Tango Top).

In addition, since the VI-sensor already provides the extrinsic calibration of its IMU and two stereo cameras, we only fix the optical axis of its left camera to match with the pose graph optimizer's results, then position the IMU and right camera according to the sensor-provided relative poses.

Once the camera positions are established, the IMU positions follow from their known displacement with respect to the camera. In the case of the VI-sensor the full extrinsic calibration between cameras and IMU is directly provided in the ROS bag, and for the Tangos it can be obtained via Android's API.

C. Synchronization

Among the devices on the rig, the VI-sensor stands out due to the built-in hardware synchronization between camera shutters and IMU [7] [28], thereby establishing a solid relationship between optical and inertial data which we leverage to synchronize the other sensors.

To aid with synchronizing the cameras, before the start and after the end of each sequence we perform multiple hand claps in locations where the views of the cameras overlap. Using these visual markers we can see in post-processing analysis that cameras C1, C2, and C3 run at such constant and equal rate that clock drift is negligible during the short time it takes to record a sequence, and further validate the accuracy of the time stamps embedded in the `mp4` files of both Tangos and in the VI-sensor's ROS bag.

Based on the hand claps we align all camera footage to within half a frame (25ms) of the VI-sensor's camera and trim excess footage to leave just the benchmark sequence. Trimming of `mp4` files is done with the `ffmpeg` tool, cutting on key frames to avoid potential quality loss associated with re-coding. Note however that caution is necessary when basing VIO on the C1/C2/C3 video, first because of the rolling shutter and second because of a relatively large uncertainty in the synchronization to the IMU.

Next, the IMUs of Tango Top and Bottom are synchronized to the VI-sensor's IMU by first using the extrinsic calibration to express the yaw-axis angular velocity of all sensors in the rig frame. Then the time series are aligned by finding the peak in the correlation function. Since logging of accelerometer and gyroscope data occurs off of the same system clock, this simultaneously yields synchronization offsets for the acceleration data as well, and in the case of Tango Bottom, the VIO poses.

Finally, the cameras are synchronized to the inertial data. We start by extracting the ground truth poses from the frame-accuracy aligned footage of cameras C1, C2, and C3 as explained in Section VII. Using the same method, we also compute the poses of cameras C1, C2, C3, Tango Top, and Tango Bottom, from which the yaw angular velocities are calculated and synchronized against the VI-sensors IMU data via maximum correlation as before.

VII. Ground Truth

A. Method

Our ground truth poses are obtained by recording images of fiducial markers with cameras C1, C2, and C3. The GoPro cameras' optical axes are all in the horizontal plane, and are offset by an angle of about 75° about the vertical axis (see Figure 3), resulting in a combined horizontal field of view of almost 270° (vertical FOV of 69.5°) when recording in "W" (wide) mode at full HD (1920x1080). The wide FOV (along with a bubble level on the rig to keep it approximately horizontal) ensures that a sufficient number of tags are visible at any one point in time.

As fiducial markers we use AprilTags [27] of size 36h11, printed on letter size paper, and backed by 1/4in MDF board where necessary. A total of 170 tags are placed as shown in Figure 4. *Attention must be paid to place the tags such that they form triangles of sufficient height*. For instance, placing tags just along the bottom of the stone wall at the walkway is *not* sufficient. Additional tags must be placed at different elevations, i.e. on the lamp posts, to yield accurate altitude via triangulation. After placement, we measure the exact position of each tag with a measure tape and record it, along with its orientation. That many tags are placed along precisely built walls expedites this process. Placing all tags, measuring and noting their locations, and recording the video sequence can be accomplished by two persons in a single 10h day.

After the sequence is recorded, the `mp4` video files are downloaded from the cameras and synchronized (see Section VI-C). With the help of the AprilTag library [27] we extract for all three cameras $\mathrm{k} \in \{\mathrm{C1}, \mathrm{C2}, \mathrm{C3}\}$ the pixel coordinates ${}^{\mathrm{k(p)}}\mathbf{u}_i^j$ of the four corner points $i = 1 \ldots 4$ belonging to tag j in each frame p, and via the tag identity relate them to the measured 3D corner coordinates $\mathbf{M}_i^j$ in the world reference frame.

We can now determine the ground truth poses of the rig via maximum likelihood estimation over a pose graph using the GTSAM [30] library in a similar fashion as for the extrinsic calibration in Section VI-B. An example of the graph structure is shown in Figure 6. For simplicity, only a single tag is visible here, allowing us to drop the tag index j. At the bottom of the graph, a narrow `PriorFactor` [30] of $\sigma = 25\,\mathrm{mm}$ constrains the 3D corner coordinates $\mathbf{X}_i$ of a single tag to their physically measured values $\mathbf{M}_i$. The corresponding observed 2D pixel coordinates ${}^{\mathrm{k(p)}}\mathbf{u}_i$ in frame p then impose constraints via a `ProjectionFactor` ($\sigma = 2$ pixels) on the pose $\mathbf{T}_{\mathrm{k(p)}}$ of the observing camera, which in turn establishes the rig pose $\mathbf{T}_{\mathrm{B(p)}}$ through the known extrinsic calibration transforms ${}^{\mathrm{B}}\mathbf{T}_{\mathrm{k}}$ with uncertainties chosen to be $\sigma_{\mathrm{rot}} = 3^\circ$ (rotation) and $\sigma_{\mathrm{trans}} = 10\,\mathrm{mm}$ (translation).

The rig poses $\mathbf{T}_{\mathrm{B(p)}}$ are interconnected with an identity transform ($\sigma_{\mathrm{rot}} = 11^\circ$, $\sigma_{\mathrm{trans}} = 50\,\mathrm{mm}$) that smooths between frames and fills in rig poses in the rare occasion when none of the three cameras observes a tag.

B. Accuracy Tests

Because the camera path includes outdoors scenes, and due to the large area covered, using a MoCap system for ground truth verification as in reference [7] is not possible. Instead, similar to [21] we conducted experiments to estimate the accuracy of our ground truth. For this purpose, the camera rig was placed on a dolly that was guided along a $2\,\mathrm{m}$ long rail at three different places (one outdoors and two indoors), traveling parallel to walls so the orientation of the test path was well established. After rotating and shifting the ground truth to align with the starting point of the test path, the ground truth motion should be strictly in the longitudinal x direction, with no movement in y and z. Figure 7 shows the observed error in the horizontal y direction and the elevation z (upper graph), and an indicator of visible tags (lower graph). The peak-to-peak error observed for outdoors is $\Delta y = 50\,\mathrm{mm}$ and $\Delta z = 17\,\mathrm{mm}$. Notice the fluctuation of the y error caused by the presence or absence of close-by tag #90. For the same experiment conducted at two different indoors locations along the path we find $\Delta y = 31\,\mathrm{mm}$, $\Delta z = 16\,\mathrm{mm}$ (open lobby, near display) and $\Delta y = 16\,\mathrm{mm}$, $\Delta z = 9\,\mathrm{mm}$ (near location of 360° turn).

To put these errors into perspective, Figure 7 also plots the errors in the position estimate of the Google Tango as obtained from its proprietary VIO algorithm. The errors are generally about four times larger than for the ground truth, and until loop closure occurs, are expected to show some accumulation.

Without exhaustive tests along the whole path it is difficult to put a strict bound on the accuracy of the ground truth rig poses. The covariances from the pose graph estimator yield a mean error of $2.5\,\mathrm{cm}$ for the position and 3.5° for the angle. We estimate that our ground truth position is accurate to better than $10\,\mathrm{cm}$ along most of the path, with inaccuracies possibly rising to $15\,\mathrm{cm}$ near the beginning and end of the path, where tags are spread more sparsely.

VIII. Evaluation metric

There are several different ways [5] [20] [31] to evaluate the quality of a trajectory with respect to the ground truth. Closely following reference [20] we adopt two of them, the *Absolute Trajectory Error* (ATE), and the *Relative Pose Error* (RPE). While they are qualitatively often similar [20], they measure accuracy at different length scales. ATE is the more intuitive variant and more relevant for e.g. augmented reality applications because it measures the ability to follow the entire length of the path without drift or rotational errors, making successful loop closure a necessity. RPE on the other hand measures the drift of the trajectory over some length scale, and is used most prominently for the KITTI VO benchmark [5]. We will discuss both metrics below.

In general an odometry algorithm will produce an estimated trajectory $P_{1:N}$ that is a sequence of transformations $P_i := {}^{\mathrm{W}'}\mathbf{T}_{\mathrm{B(i)}}$, describing the transition from body (rig) coordinate system B to an arbitrarily chosen world system W' for a given frame i. This is to be compared to the corresponding ground truth poses $Q_i := {}^{\mathrm{W}}\mathbf{T}^{\mathrm{ref}}_{\mathrm{B(i)}}$.

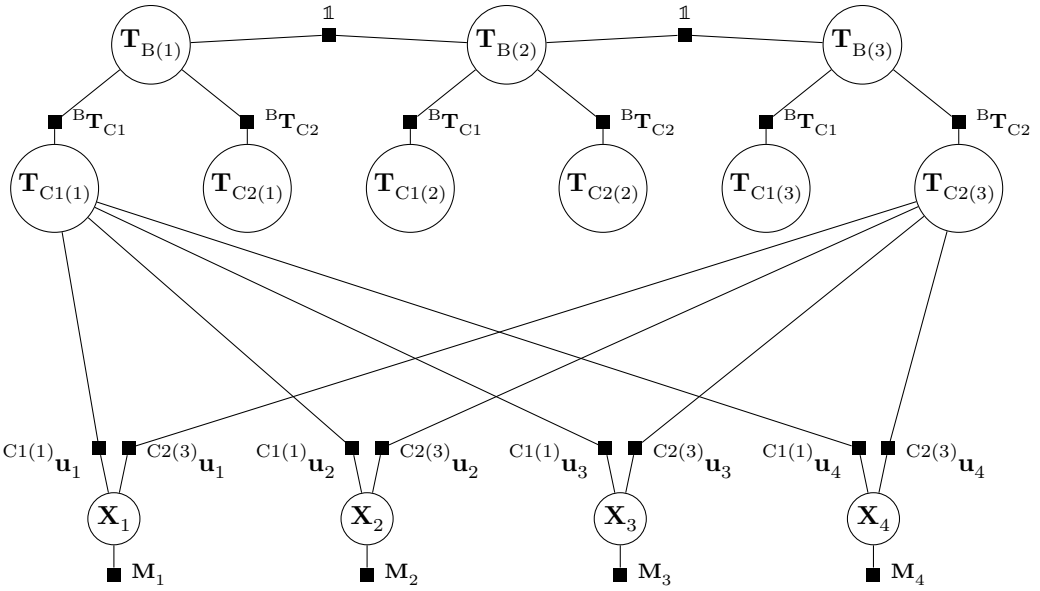

Fig. 6. Example factor graph for a two-camera localization off of a single tag across three frames. The tag's 3D corner points $\mathbf{X}_i$, $i = 1 \ldots 4$ are at physically measured locations $\mathbf{M}_i$. They are observed in frame 1 at image locations ${}^{\mathrm{C1}(1)}\mathbf{u}_i$ by camera C1, and later in frame 3 by camera C2 at locations ${}^{\mathrm{C2}(3)}\mathbf{u}_i$. These sensor measurements constrain the corresponding camera poses $\mathbf{T}_{\mathrm{C1}(1)}$ and $\mathbf{T}_{\mathrm{C2}(3)}$. Together with the known extrinsic calibrations ${}^{\mathrm{B}}\mathbf{T}_{\mathrm{C1}}$ and ${}^{\mathrm{B}}\mathbf{T}_{\mathrm{C2}}$ this determines the rig poses $\mathbf{T}_{\mathrm{B}(1)}$ and $\mathbf{T}_{\mathrm{B}(3)}$, and via the smoothing identity transforms interconnecting frames, fills in the rig pose $\mathbf{T}_{\mathrm{B}(2)}$ where neither camera observes the tag.

The ATE error evaluation starts by using Horn's method [32] to find the global transform $\mathbf{S} := {}^{\mathrm{W}}\mathbf{T}_{\mathrm{W}'}$ that aligns $\mathbf{P}_{1:N}$ and $\mathbf{Q}_{1:N}$ in a least square sense, possibly scaling it as well for trajectories generated by purely visual algorithms that have no scale reference. Then the pose errors $\mathbf{F}_i := \mathbf{Q}_i^{-1}\mathbf{S}\mathbf{P}_i$ are computed, and the root mean square error (RMSE) of their translational component is taken, weighted by the time Δt_i between a pose and its predecessor,

$$\mathrm{ATE}(\mathbf{F}_{1:N}) := \left(\frac{1}{T} \sum_{i=1}^{N} \Delta t_i ||trans(\mathbf{F}_i)||^2 \right)^{1/2}, \qquad (2)$$

normalized by the total time $T = \sum_{i=1}^{N} \Delta t_i$.

In contrast to ATE, RPE focuses on errors between *relative* poses

$$\mathbf{E}_i := \left(\mathbf{Q}_i^{-1}\mathbf{Q}_{i+\Delta}\right)^{-1} \left(\mathbf{P}_i^{-1}\mathbf{P}_{i+\Delta}\right) \qquad (3)$$

between time t_i and a future time $t_{i+\Delta}$, i.e. how well the *change* $\mathbf{Q}_i^{-1}\mathbf{Q}_{i+\Delta}$ in ground truth pose is reproduced. The RPE then follows by taking the RMSE:

$$\mathrm{RPE}(\mathbf{E}_{1:N}) := \left(\frac{1}{T} \sum_{i=1}^{N} \Delta t_i ||trans(\mathbf{E}_i)||^2 \right)^{1/2}. \qquad (4)$$

This leaves the question of how to pick the time horizon $\tau = t_{i+\Delta} - t_i$. One can either average over several τ [20] [5] or match the time scale over which the benchmarked algorithm can reasonably be expected to track. We proceed by first selecting a length scale of $l = 20\,\mathrm{m}$, which is comparable to the dimensions of the outdoors pathways and the lobby hallway, and then compute $\tau = (l/L)T$ from the total trajectory length L and travel time T. We observe that for the Tango trajectories, picking τ this way leads to ATE and RPE of similar magnitude.

Our data set comes with the source code for both evaluation metrics, with and without scaling, and decomposing the errors in x, y and z directions for testing trajectories provided by pure 2D approaches. This should make the benchmark accessible to a wide variety of algorithms. For RPE, we express errors in percentages by dividing all distances with the average path length l, similar to the KITTI benchmark [5]. The identical program will be run to measure the accuracy of trajectories for the test sequences with hidden ground truth.

IX. Conclusion

We present PennCOSYVIO, a new challenging indoor/outdoor VIO data set for hand held devices that covers both indoors and outdoors environments and comes with ground truth trajectories for benchmarking. For more details please visit https://daniilidis-group.github.io/penncosyvio/.

X. Acknowledgments

We thank Bhavya Gupta for developing the Android Apps for data logging on the Tangos. Financial support by the I/UCRC Rose-Hub NSF-IIP-1439681 and the ARL RCTA W911NF-10-2-0016 is gratefully acknowledged.

References

[1] A. I. Mourikis and S. I. Roumeliotis, "A Multi-State Constraint Kalman Filter for Vision-aided Inertial Navigation," in *International Conference on Robotics and Automation (ICRA)*. IEEE, 2007, pp. 10–14.

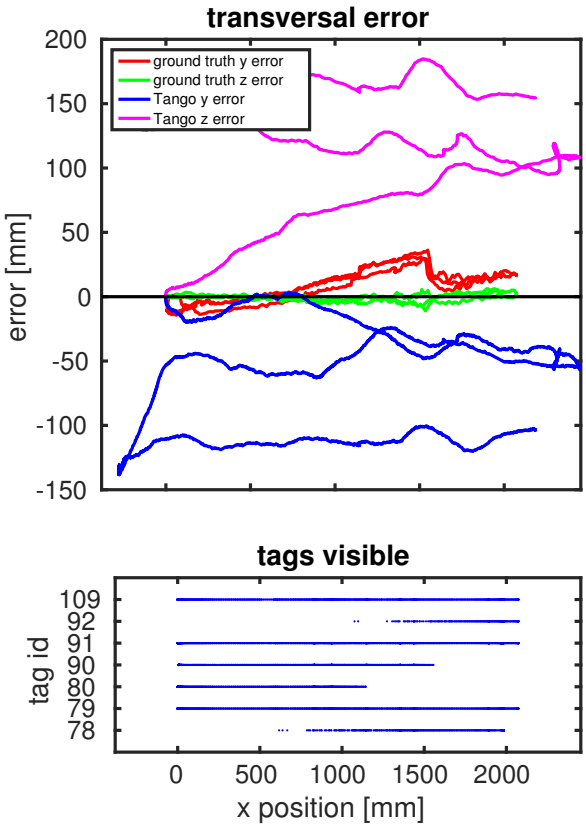

Fig. 7. Outdoors ground truth accuracy tests. The top graph shows the deviation from an ideal straight line, both in z (elevation) and y (left/right) direction, while the cart traveled its path forward, back, and forward again. The bottom graph indicates which tags were visible by any one of the three GoPro cameras. For comparison the error of Project Tango's proprietary VIO algorithm is shown as well.

[2] P. Tiefenbacher, T. Schulze, and G. Rigoll, "Off-the-Shelf Sensor Integration for Mono-SLAM on Smart Devices," in *Conference on Computer Vision and Pattern Recognition Workshops*. IEEE, 2015, pp. 15–20.

[3] A. Concha, G. Loianno, V. Kumar, and J. Civera, "Visual-Inertial Direct SLAM," in *International Conference on Robotics and Automation (ICRA)*. IEEE, 2016, pp. 1331–1338.

[4] H. Lategahn, A. Geiger, and B. Kitt, "Visual SLAM for Autonomous Ground Vehicles," in *International Conference on Robotics and Automation (ICRA)*. IEEE, 2011, pp. 1732–1737.

[5] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite," in *Conference on Computer Vision and Pattern Recognition*. IEEE, 2012, pp. 3354–3361.

[6] C. Forster, S. Lynen, L. Kneip, and D. Scaramuzza, "Collaborative monocular SLAM with multiple Micro Aerial Vehicles." in *International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2013, pp. 3962–3970.

[7] M. Burri, J. Nikolic, P. Gohl, T. Schneider, J. Rehder, S. Omari, M. W. Achtelik, and R. Siegwart, "The EuRoC micro aerial vehicle datasets," *The International Journal of Robotics Research*, vol. 35, no. 10, pp. 1157–1163, 2016.

[8] T. Schöps, J. Engel, and D. Cremers, "Semi-Dense Visual Odometry for AR on a Smartphone," in *International Symposium on Mixed and Augmented Reality (ISMAR)*. IEEE, 2014, pp. 145–150.

[9] T. Schöps, T. Sattler, C. Häne, and M. Pollefeys, "3D Modeling on the Go: Interactive 3D Reconstruction of Large-Scale Scenes on Mobile Devices," in *International Conference on 3D Vision*, 2015, pp. 291–299.

[10] L. M. Paz, P. Pinis, J. D. Tards, and J. Neira, "Large scale 6-dof slam with stereo-in-hand," *IEEE TRANSACTIONS ON ROBOTICS*, vol. 24, no. 5, pp. 946–957, 2008.

[11] G. Sibley, L. H. Matthies, and G. S. Sukhatme, "Sliding window filter with application to planetary landing." *J. Field Robotics*, vol. 27, no. 5, pp. 587–608, 2010.

[12] K. Pirker, M. Rther, and H. Bischof, "CD SLAM - Continuous Localization and Mapping in a Dynamic World," in *International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2011, pp. 3990–3997.

[13] R. A. Newcombe, S. J. Lovegrove, and A. J. Davison, "DTAM: Dense Tracking and Mapping in Real-time," in *Proceedings of the 2011 International Conference on Computer Vision*. IEEE, 2011, pp. 2320–2327.

[14] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardós, "ORB-SLAM: a versatile and accurate monocular SLAM system," *CoRR*, vol. abs/1502.00956, 2015.

[15] M. Li and A. I. Mourikis, "High-precision, Consistent EKF-based Visual-inertial Odometry," *Int. J. Rob. Res.*, vol. 32, no. 6, pp. 690–711, May 2013.

[16] S. Leutenegger, S. Lynen, M. Bosse, R. Siegwart, and P. Furgale, "Keyframe-based visualinertial odometry using nonlinear optimization," *The International Journal of Robotics Research*, vol. 34, pp. 314–334, 2014.

[17] K. Tsotsos, A. Chiuso, and S. Soatto, "Robust Inference for Visual-Inertial Sensor Fusion," in *International Conference on Robotics and Automation (ICRA)*. IEEE, May 2015, pp. 5203–5210.

[18] C. Stachniss, U. Frese, and G. Grisetti, "OpenSLAM: Datasets & Links," 2016. [Online]. Available: http://openslam.org/

[19] A. Howard and N. Roy, "Radish: The Robotics Data Set Repository," 2003. [Online]. Available: http://radish.sourceforge.net/

[20] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A Benchmark for the Evaluation of RGB-D SLAM Systems," in *International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2012, pp. 573–580.

[21] S. Ceriani, G. Fontana, A. Giusti, D. Marzorati, M. Matteucci, D. Migliore, D. Rizzi, D. G. Sorrenti, and P. Taddei, "Rawseeds ground truth collection systems for indoor self-localization and mapping," *Autonomous Robots*, vol. 27, no. 4, pp. 353–371, 2009.

[22] N. Carlevaris-Bianco, A. K. Ushani, and R. M. Eustice, "University of Michigan North Campus long-term vision and lidar dataset," *International Journal of Robotics Research*, vol. 35, no. 9, pp. 1023–1035, 2015.

[23] J.-L. Blanco, F.-A. Moreno, and J. González-Jiménez, "The Málaga Urban Dataset: High-rate Stereo and Lidars in a realistic urban scenario," *International Journal of Robotics Research*, vol. 33, no. 2, pp. 207–214, 2014.

[24] A. Huang, M. Antone, E. Olson, L. Fletcher, D. Moore, S. Teller, and J. Leonard, "A high-rate, heterogeneous data set from the DARPA Urban Challenge," *International Journal of Robotics Research*, vol. 29, no. 13, pp. 1595–1601, Nov 2010.

[25] G. Pandey, J. R. McBride, and R. M. Eustice, "Ford Campus Vision and Lidar Data Set," *International Journal of Robotics Research*, vol. 30, no. 13, pp. 1543–1552, 2011.

[26] M. Smith, I. Baldwin, W. Churchill, R. Paul, and P. Newman, "The New College Vision and Laser Data Set," *The International Journal of Robotics Research*, vol. 28, no. 5, pp. 595–599, May 2009.

[27] E. Olson, "AprilTag: A robust and flexible visual fiducial system," in *International Conference on Robotics and Automation (ICRA)*. IEEE, 2011, pp. 3400–3407.

[28] J. Nikolic, J. Rehder, M. Burri, P. Gohl, S. Leutenegger, P. Furgale, and R. Y. Siegwart, "A Synchronized Visual-Inertial Sensor System with FPGA Pre-Processing for Accurate Real-Time SLAM," in *International Conference on Robotics and Automation (ICRA)*. IEEE, 2014, pp. 431–437.

[29] D. Scaramuzza, A. Martinelli, and R. Siegwart, "A Toolbox for Easily Calibrating Omnidirectional Cameras," in *International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2006, pp. 5695–5701.

[30] F. Dellaert, "Factor graphs and GTSAM: A hands-on introduction," GT RIM, Tech. Rep. GT-RIM-CP&R-2012-002, Sep 2012.

[31] R. Kümmerle, B. Steder, C. Dornhege, M. Ruhnke, G. Grisetti, C. Stachniss, and A. Kleiner, "On Measuring the Accuracy of SLAM Algorithms," *Autonomous Robots*, vol. 27, no. 4, pp. 387–407, 2009.

[32] B. K. Horn, "Closed-form solution of absolute orientation using unit quaternions," *Journal of the Optical Society of America B*, vol. 4, no. 4, Apr 1987.