OrcVIO: Object residual constrained Visual-Inertial Odometry

Mo Shan Qiaojun Feng Nikolay Atanasov

***Abstract*— Introducing object-level semantic information into simultaneous localization and mapping (SLAM) system is critical. It not only improves the performance but also enables tasks specified in terms of meaningful objects. This work presents OrcVIO, for visual-inertial odometry tightly coupled with tracking and optimization over structured object models. OrcVIO differentiates through semantic feature and bounding-box reprojection errors to perform batch optimization over the pose and shape of objects. The estimated object states aid in real-time incremental optimization over the IMU-camera states. The ability of OrcVIO for accurate trajectory estimation and large-scale object-level mapping is evaluated using real data.**

I. Introduction

The foundations of visual understanding in robotics, machine learning, and computer vision lie in the twin technologies of inferring geometric structure and semantic content. Researchers have made a significant progress to infer the structure of the scene using techniques like visual-inertial odometry (VIO) [1] and SLAM [2]. State of the art VIO approaches work with monocular or stereo cameras [3], often complemented by inertial information [4], [5]. However, most real-time incremental SLAM results provide only geometric representations that lack a semantic understanding of the environment.

At the other end of the spectrum, impressive results have been achieved in object recognition and semantic understanding using deep neural networks [6]. Methods related to VIO and SLAM focus on learning to regress camera poses and image depth directly from images [7], [8]. For instance, monocular depth, optical flow, and ego-motion are jointly optimized from video in [9] by relying on a view-synthesis loss. Deep learning techniques have shown impressive performance in localization, object recognition, and semantic segmentation but do not yet provide global positioning of the semantic content.

This paper focuses on the joint visual-inertial odometry and object-level mapping (for rigid, static objects). Generating geometrically consistent and semantically meaningful maps allows compressed representation, improved loop closure (recognizing already visited locations), and robot mission specifications in terms of human-interpretable objects. There are mainly two groups of object-based SLAM techniques. Category-specific approaches optimize the pose and shape of object instances, using semantic keypoints [11], [12] or 3D shape models [13], [14]. Category-agnostic approaches use geometric shapes, such as spheres [15], cuboids [16], or ellipsoids [17], to represent objects. CubeSLAM [16] generates and refines 3D cuboid proposals using multi-view bundle adjustment without relying on prior models. QuadricSLAM [17] uses an ellipsoid representation, suitable for defining a bounding-box detection model. Structural constraints based on supporting and tangent planes, commonly observed under a Manhattan assumption, may also be introduced [18]. Using generic symmetric shapes, however, makes the orientation of object instances potentially irrecoverable.

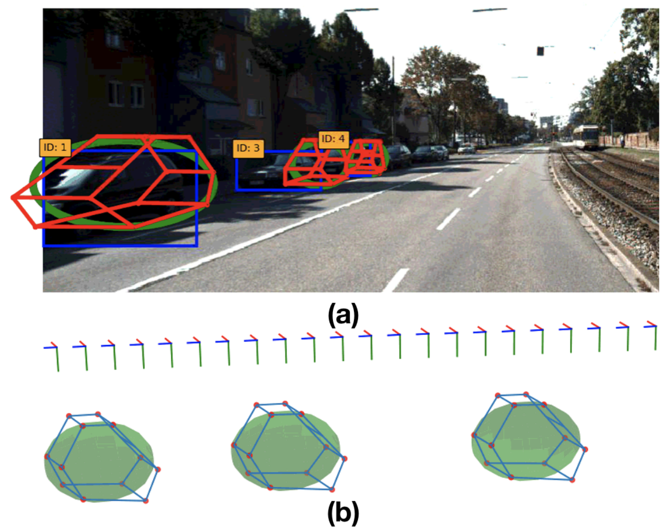

Fig. 1. We propose a tightly coupled visual-inertial odometry and object state optimization algorithm. (a) Back-projection of the estimated object shapes (red) and ellipsoids (green) on an image from the KITTI dataset [10]. Blue rectangles are the bounding-boxes from the object detector, with the object ID label at the top left corner. (b) Camera poses (colored axes) and object states (red: semantic keypoints, blue: car shape, green: ellipsoid) are shown in the global frame. A demo can be found at `https://youtu.be/fPfBzfGnEcY`.

Our work takes advantage of both specific and generic representations and proposes a coarse-to-fine object model. We use an ellipsoid at the coarse level to restrict an object's pose variation, and semantic keypoints at the fine level to obtain a precise shape deformation. Our **contribution** is a lightweight incremental semantic visual-inertial odometry algorithm, tightly coupled with iterative multi-view optimization of object poses and shapes. The approach relies on residuals and Jacobians obtained from inertial measurements, geometric features, object bounding-box detections and mid-level object part features (e.g., car wheels, windshield, doors). Inspired by the multi-state constraint Kalman filter (MSCKF) [4], we combine fast filter-based propagation of

We gratefully acknowledge support from ARL DCIST CRA W911NF-17-2-0181 and ONR N00014-18-1-2828.

The authors are with the Department of Electrical and Computer Engineering, University of California, San Diego, La Jolla, CA 92093, USA {moshan,qjfeng,natanasov}@ucsd.edu

IMU-camera states with corrections based on object states, optimized over multiple views. We dub our method *Object residual constrained Visual-Inertial Odometry* (OrcVIO) to emphasize the role of the semantic error terms in the optimization process. OrcVIO is capable of producing meaningful object maps and estimating accurate sensor trajectories, as shown in Fig. 1. We will make our source-code publicly available for the benefit of the research community.

II. Background

We denote the IMU, camera, object, and global reference frames as $\{I\}$, $\{C\}$, $\{O\}$, $\{G\}$, respectively. The transformation from frame $\{A\}$ to $\{B\}$ is specified by a 4×4 matrix:

$$ {}^B_A\mathbf{T} \triangleq \begin{bmatrix} {}^B_A\mathbf{R} & {}^B_A\mathbf{p} \\ \mathbf{0}^\top & 1 \end{bmatrix} \in SE(3) \tag{1} $$

where ${}^B_A\mathbf{R} \in SO(3)$ is a rotation matrix and ${}^B_A\mathbf{p} \in \mathbb{R}^3$ is a translation vector. To simplify the notation, we will not explicitly indicate the global frame when specifying transformations. For example, the pose of the IMU frame $\{I\}$ in $\{G\}$ at time t is specified by ${}_I\mathbf{T}_t$. We overload $\boldsymbol{\theta}_\times$ to denote the mapping from an axis-angle vector $\boldsymbol{\theta} \in \mathbb{R}^3$ to a 3×3 skew-symmetric matrix $\boldsymbol{\theta}_\times \in \mathfrak{so}(3)$ and the mapping from a position-rotation vector $\boldsymbol{\xi} \in \mathbb{R}^6$ to a 4×4 twist matrix $\boldsymbol{\xi}_\times \in \mathfrak{se}(3)$. We define an infinitesimal change of pose $\mathbf{T} \in SE(3)$ using a left perturbation $\exp\left(\boldsymbol{\xi}_\times\right)\mathbf{T} \in SE(3)$ (see [19, Ch.7]).

Let $\underline{\mathbf{x}}$ be the homogeneous coordinates [20, Ch.1] of a vector $\mathbf{x}$. We will use the operators [19, Ch.7]:

$$ \underline{\mathbf{x}}^\odot \triangleq \begin{bmatrix} \mathbf{I}_3 & -\mathbf{x}_\times \\ \mathbf{0}^\top & \mathbf{0}^\top \end{bmatrix} \in \mathbb{R}^{4\times 6} \quad \underline{\mathbf{x}}^\circledcirc \triangleq \begin{bmatrix} \mathbf{0} & \mathbf{x} \\ -\mathbf{x}_\times & \mathbf{0} \end{bmatrix} \in \mathbb{R}^{6\times 4}. \tag{2} $$

An axis-aligned ellipsoid centered at $\mathbf{0}$ can be described as:

$$ \mathcal{E}_\mathbf{u} \triangleq \left\{\mathbf{x} \mid \mathbf{x}^\top \mathbf{U}^{-\top}\mathbf{U}^{-1}\mathbf{x} \le 1\right\}, \tag{3} $$

where $\mathbf{U} \triangleq \mathbf{diag}(\mathbf{u})$ and the elements of the vector $\mathbf{u}$ are the lengths of the semi-axes of $\mathcal{E}_\mathbf{u}$. In homogeneous coordinates, $\mathcal{E}_\mathbf{u}$ can be represented as a quadric surface [20, Ch.3], $\left\{\mathbf{x} \mid \underline{\mathbf{x}}^\top \mathbf{Q}_\mathbf{u}\underline{\mathbf{x}} \le 0\right\}$, where $\mathbf{Q}_\mathbf{u} = \mathbf{diag}(\mathbf{U}^{-\top}\mathbf{U}^{-1}, -1)$. This describes the ellipsoid as a collection of points lying on its surface. Alternatively, a quadric can be defined by the set of planes $\underline{\boldsymbol{\pi}} = \mathbf{Q}_\mathbf{u}\underline{\mathbf{x}}$ tangent to its surface at $\underline{\mathbf{x}}$. This dual quadric surface is defined as $\left\{\boldsymbol{\pi} \mid \underline{\boldsymbol{\pi}}^\top \mathbf{Q}^*_\mathbf{u}\underline{\boldsymbol{\pi}} = 0\right\}$, where $\mathbf{Q}^*_\mathbf{u} = \mathbf{adj}(\mathbf{Q}_\mathbf{u})$[1]. A dual quadric defined by $\mathbf{Q}^*_\mathbf{u}$ can be transformed by $\mathbf{T} \in SE(3)$ to another reference frame as $\mathbf{T}\mathbf{Q}^*_\mathbf{u}\mathbf{T}^\top$. Similarly, it can be projected to a lower-dimensional space by $\mathbf{P} = \begin{bmatrix}\mathbf{I} & \mathbf{0}\end{bmatrix}$ as $\mathbf{P}\mathbf{Q}^*_\mathbf{u}\mathbf{P}^\top$.

III. Problem Formulation

Let $\mathbf{x}_t \triangleq ({}_I\mathbf{x}_t,\ {}_C\mathbf{x}_t)$ be the state of an IMU-camera sensor at time t. The IMU state ${}_I\mathbf{x}_t \triangleq ({}_I\mathbf{R}_t,\ {}_I\mathbf{p}_t,\ {}_I\mathbf{v}_t,\ \mathbf{b}_g,\ \mathbf{b}_a)$ consists of orientation ${}_I\mathbf{R}_t \in SO(3)$, position ${}_I\mathbf{p}_t \in \mathbb{R}^3$, velocity ${}_I\mathbf{v}_t \in \mathbb{R}^3$, gyroscope bias $\mathbf{b}_g \in \mathbb{R}^3$, and accelerometer bias $\mathbf{b}_a \in \mathbb{R}^3$. The camera state ${}_C\mathbf{x}_t \triangleq ({}_C\mathbf{T}_{t-W+1},\ \ldots,\ {}_C\mathbf{T}_t)$ consists of a history of W camera poses ${}_C\mathbf{T}_i \in SE(3)$. Ideally, the camera state would contain the camera pose trajectory for all time but to maintain bounded computational complexity, only a subset of the camera poses are kept. The system trajectory over time is a collection $\mathcal{X} \triangleq \{\mathbf{x}_t\}_{t=1}^T$.

[1]If $\mathbf{Q}$ is invertible, $\mathbf{Q}^* = \mathbf{adj}(\mathbf{Q}) = \det(\mathbf{Q})\mathbf{Q}^{-1}$ can be simplified to $\mathbf{Q}^* = \mathbf{Q}^{-1}$ due to the scale-invariance of the dual quadric definition.

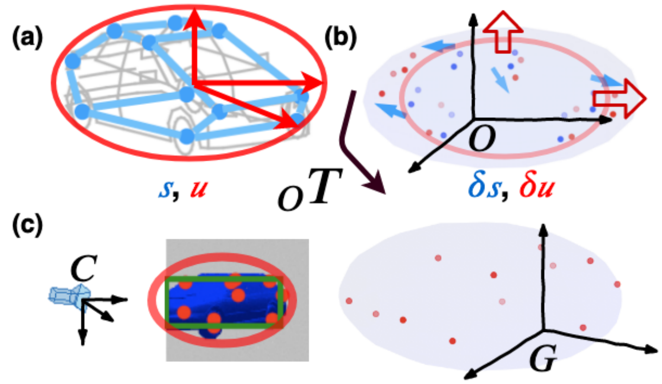

Fig. 2. (a) An object class is defined by a semantic class σ and a mean shape specified by semantic keypoints $\mathbf{s}$ (blue) and an ellipsoid with shape $\mathbf{u}$ (red). (b) A specific instance has keypoints and shape deformations, parameterized by $\delta\mathbf{s}$ (blue arrows) and $\delta\mathbf{u}$ (red arrows). (c) The keypoints are transformed from the object frame $\{O\}$ to the global frame $\{G\}$ via the instance pose ${}_O\mathbf{T}$.

The system evolves in an environment that contains *geometric landmarks* $\mathcal{L} \triangleq \{\boldsymbol{\ell}_m\}_{m=1}^{N_m}$ and *objects* $\mathcal{O} \triangleq \{\mathbf{o}_i\}_{i=1}^{N_i}$, represented in a global frame $\{G\}$. A geometric landmark $\boldsymbol{\ell}_m$ is a static point in $\mathbb{R}^3$, detectable via image corner feature algorithms such as FAST [21]. Each object $\mathbf{o}_i = (\mathbf{c}_i, \mathbf{i}_i)$ is an instance $\mathbf{i}_i$ of a semantic class $\mathbf{c}_i$ detectable via object recognition algorithms such as YOLO [22]. The precise definitions of an object class and instance are as follows.

Definition. An *object class* is a tuple $\mathbf{c} \triangleq (\sigma, \mathbf{s}, \mathbf{u})$, where $\sigma \in \mathbb{N}$ specifies a semantic class (e.g., chair, table, monitor) and $\mathbf{s} \in \mathbb{R}^{3\times N_s}$, $\mathbf{u} \in \mathbb{R}^3$ specify a mean shape. The shape is determined by *semantic landmarks* $\{\mathbf{s}_j\}_{j=1}^{N_s} \in \mathbb{R}^3$ in an object canonical frame $\{O\}$, corresponding to mid-level parts (e.g., front wheel of a car), and an axis-aligned ellipsoid $\mathcal{E}_\mathbf{u}$.

Definition. An *object instance* of class $\mathbf{c}$ is a tuple $\mathbf{o}_i \triangleq ({}_O\mathbf{T}, \delta\mathbf{s}, \delta\mathbf{u})$, where ${}_O\mathbf{T} \in SE(3)$ is the instance pose, and $\delta\mathbf{s} \in \mathbb{R}^{3\times N_s}$ are the deformations of the average semantic landmarks $\mathbf{s}$, $\delta\mathbf{u} \in \mathbb{R}^3$ represents the ellipsoid semi-axes lengths $\mathbf{u}$.

The shape of an object $(\mathbf{c}, \mathbf{i})$ in the global frame $\{G\}$ is specified by semantic landmarks ${}_O\mathbf{T}\left(\underline{\mathbf{s}}_j + \underline{\delta\mathbf{s}}_j\right)$ and dual ellipsoid ${}_O\mathbf{T}\mathbf{Q}^*_{(\mathbf{u}+\delta\mathbf{u})}{}_O\mathbf{T}^\top$. These definitions are illustrated for a car model with 12 semantic landmarks in Fig. 2.

The IMU-camera sensor provides three sources of information: inertial, geometric, and semantic, illustrated in Fig. 3. The inertial observations ${}^i\mathbf{z}_t \triangleq \left({}^i\mathbf{a}_t, {}^i\boldsymbol{\omega}_t\right) \in \mathbb{R}^6$ are the IMU's body frame linear acceleration and angular velocity at time t. The geometric observations are noisy detections ${}^g\mathbf{z}_{t,n} \in \mathbb{R}^2$ of the image projections of the geometric landmarks $\mathcal{L}$ visible to the camera at time t. To obtain semantic observations, an object detection algorithm [22]

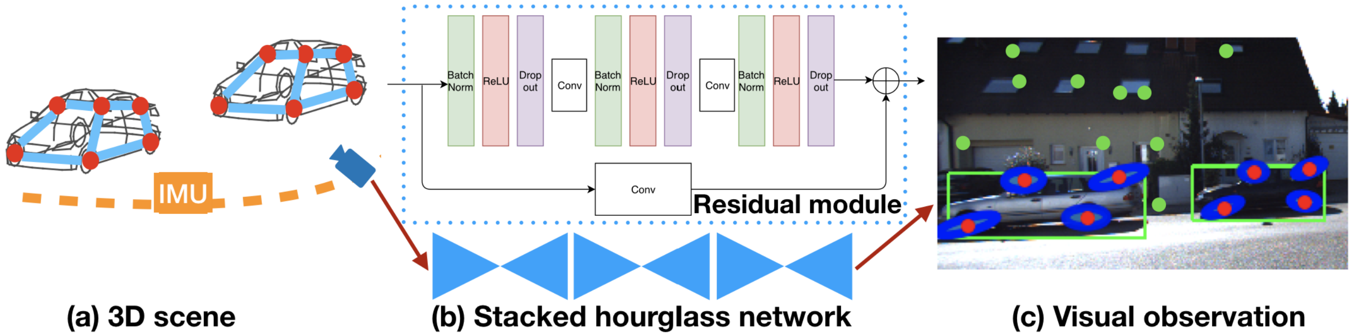

Fig. 3. OrcVIO utilizes visual-inertial information to optimize the sensor trajectory and the shapes and poses of objects. The observations include geometric keypoints (FAST keypoints indicated by green dots in (c)), semantic keypoints (car parts indicated by red dots in (c)), semantic keypoint covariances (blue ellipses in (c)), bounding-boxes (green boxes in (c)), and inertial data (orange dotted lines in (a)). The semantic keypoints and their covariances are obtained from a Bayesian stacked hourglass CNN (b), composed of residual modules (blue dotted rectangle in (b)) including convolution, ReLU, batch normalization, and dropout layers. The dropout layers are used to sample different weight realizations to enable a test-time estimate of the semantic keypoint covariances.

is applied to the image at time t, followed by semantic keypoint extraction [23] within each detected bounding-box. The k-th object detection includes its class ${}^{c}\mathbf{z}_{t,k} \in \mathbb{N}$, bounding-box ${}^{b}\mathbf{z}_{t,j,k} \in \mathbb{R}$, described by $j = 1, \ldots, 4$ lines in normalized pixel coordinates[2] and semantic keypoints ${}^{s}\mathbf{z}_{t,j,k} \in \mathbb{R}^2$ in normalized pixel coordinates associated with the $j = 1, \ldots, N_s$ semantic landmarks[3].

Let $\mathbb{1}_{t,m,n} \in \{0,1\}$ indicate whether the n-th geometric keypoint observed at time t is associated with the m-th geometric landmark. Similarly, let $\mathbb{1}_{t,i,k} \in \{0,1\}$ indicate whether the k-th object detection at time t is associated with the i-th object instance. These data association functions are unknown and need to be estimated. We describe an approach for geometric keypoint and object tracking to determine the data associations in Sec. IV. Given the associations, we introduce error functions:

$$
\begin{aligned}
{}^{i}\mathbf{e}_{t,t+1} &\triangleq {}^{i}\mathbf{e}\left(\mathbf{x}_t, \mathbf{x}_{t+1}, {}^{i}\mathbf{z}_t\right) & {}^{g}\mathbf{e}_{t,m,n} &\triangleq {}^{g}\mathbf{e}\left(\mathbf{x}_t, \boldsymbol{\ell}_m, {}^{g}\mathbf{z}_{t,n}\right) \\
{}^{s}\mathbf{e}_{t,i,j,k} &\triangleq {}^{s}\mathbf{e}\left(\mathbf{x}_t, \mathbf{o}_i, {}^{s}\mathbf{z}_{t,j,k}\right) & {}^{b}\mathbf{e}_{t,i,j,k} &\triangleq {}^{b}\mathbf{e}\left(\mathbf{x}_t, \mathbf{o}_i, {}^{b}\underline{\mathbf{z}}_{t,j,k}\right)
\end{aligned}
$$

for the inertial, geometric, semantic keypoint and bounding-box measurements, respectively, defined precisely in Sec. V. We also introduce a regularization error term ${}^{r}\mathbf{e}\left(\mathbf{o}_i\right)$ to ensure that the instance deformations $(\delta\mathbf{s}, \delta\mathbf{u})$ remain small. We consider the following problem.

Problem. Determine the sensor trajectory $\mathcal{X}^*$, geometric landmarks $\mathcal{L}^*$, and object states $\mathcal{O}^*$ that minimize the weighted sum of squared errors:

$$
\begin{aligned}
\min_{\mathcal{X},\mathcal{L},\mathcal{O}} \; & {}^{i}w \sum_{t} \|{}^{i}\mathbf{e}_{t,t+1}\|^2_{{}^{i}\mathbf{V}} + {}^{g}w \sum_{t,m,n} \mathbb{1}_{t,m,n} \|{}^{g}\mathbf{e}_{t,m,n}\|^2_{{}^{g}\mathbf{V}} \\
& + {}^{s}w \sum_{t,i,j,k} \mathbb{1}_{t,i,k} \|{}^{s}\mathbf{e}_{t,i,j,k}\|^2_{{}^{s}\mathbf{V}} + {}^{b}w \sum_{t,i,j,k} \mathbb{1}_{t,i,k} \|{}^{b}\mathbf{e}_{t,i,j,k}\|^2_{{}^{b}\mathbf{V}} \\
& + {}^{r}w \sum_{i} \|{}^{r}\mathbf{e}\left(\mathbf{o}_i\right)\|^2 \qquad (4)
\end{aligned}
$$

[2]Given pixel coordinates $\mathbf{z} \in \mathbb{R}^2$ and a camera intrinsic calibration matrix $\mathbf{K} \in \mathbb{R}^{3\times 3}$, the *normalized pixel coordinates* of $\mathbf{z}$ are $\mathbf{K}^{-1}\underline{\mathbf{z}}$.

[3]The semantic landmark-keypoint correspondence is provided by the semantic keypoint detector. Some landmarks may not be detected due to occlusion but we do not make this explicit for simplicity.

where ${}^{*}w$ are positive constants determining the relative importance of the error terms and ${}^{*}\mathbf{V}$ are matrices specifying the covariances associated with the inertial, geometric, semantic, and bounding-box measurements and define a quadratic (Mahalanobis) norm $\|\mathbf{e}\|^2_{\mathbf{V}} \triangleq \mathbf{e}^\top \mathbf{V}^{-1} \mathbf{e}$.

Inspired by the MSCKF [4], we decouple the optimization over $\mathcal{L}$ and $\mathcal{O}$ from the optimization over $\mathcal{X}$. When a geometric keypoint or object track is lost, we perform multi-view iterative optimization over its state based on the latest IMU-camera state estimate. The IMU-camera state is propagated using the inertial observations ${}^{i}\mathbf{z}_t$ and updated using the optimized landmark and object states and the geometric and semantic observations. This decoupling leads to higher efficiency compared to window or batch keyframe optimization [24]. Our approach is among the first to offer tight coupling between semantic information and geometric structure in visual-inertial odometry. The error functions and Jacobians, defined in Sec. V, can also be used for batch keyframe optimization.

IV. Keypoint and Object Tracking

Geometric keypoints ${}^{g}\mathbf{z}_{t,n}$ are detected using the FAST detector [21] and are tracked temporally using the Lucas-Kanade (LK) algorithm [25]. Keypoint-based tracking has lower accuracy but higher efficiency than descriptor-based methods, allowing our method to use a high frame-rate camera. Outliers are eliminated by estimating the essential matrix between consecutive views and removing those keypoints that do not fit the estimated model. Assuming that the time between subsequent images is short, the relative orientation is obtained by integrating the gyroscope measurements ${}^{i}\boldsymbol{\omega}_t$ and only the unit translation vector is estimated using two-point RANSAC [26].

The YOLO detector [22] is used to detect object classes ${}^{c}\mathbf{z}_{t,k}$ and bounding-box lines ${}^{b}\mathbf{z}_{t,j,k}$. Semantic keypoints ${}^{s}\mathbf{z}_{t,j,k}$ are extracted within each bounding box using the StarMap stacked hourglass convolutional neural network [23]. We augment the original StarMap network with dropout layers as shown in Fig. 3b). Several stochastic forward passes are preformed using Monte Carlo dropout [27] to obtain semantic keypoint covariances ${}^{s}\mathbf{V}$ (see Fig. 3 c)). The bounding boxes ${}^{b}\mathbf{z}_{t,j,k}$ are tracked temporally using the SORT algorithm [28], which performs intersection over union (IoU) matching via the Hungarian algorithm. The semantic keypoints ${}^{s}\mathbf{z}_{t,j,k}$ within each bounding box are tracked via a Kalman filter, which uses the LK algorithm for prediction and the StarMap keypoint detections for update.

V. Landmark and Object Reconstruction

This section first introduces the linearization of the errors in (4) around perturbations for iterative optimization, and then defines the error functions and their Jacobians.

We linearize the error functions in (4) around estimates of the IMU-camera state $\hat{\mathbf{x}}_t$, geometric landmarks $\hat{\boldsymbol{\ell}}_m$, and object instances $\hat{\mathbf{o}}_i$ using perturbations $\tilde{\mathbf{x}}_t$, $\tilde{\boldsymbol{\ell}}_m$, and $\tilde{\mathbf{o}}_i$:

$$\mathbf{x}_t = \tilde{\mathbf{x}}_t \oplus \hat{\mathbf{x}}_t, \qquad \boldsymbol{\ell}_m = \tilde{\boldsymbol{\ell}}_m + \hat{\boldsymbol{\ell}}_m, \qquad \mathbf{o}_i = \tilde{\mathbf{o}}_i \oplus \hat{\mathbf{o}}_i, \quad (5)$$

where $\oplus$ emphasizes that some additions are over the $SE(3)$ manifold, defined as follows:

$$\begin{aligned}
&{}_{I}\mathbf{R} = \exp\left({}_{I}\boldsymbol{\theta}_{\times}\right){}_{I}\hat{\mathbf{R}} \quad && {}_{I}\mathbf{p} = {}_{I}\tilde{\mathbf{p}} + {}_{I}\hat{\mathbf{p}} \quad && {}_{I}\mathbf{v} = {}_{I}\tilde{\mathbf{v}} + {}_{I}\hat{\mathbf{v}} \\
&{}_{C}\mathbf{T} = \exp\left({}_{C}\boldsymbol{\xi}_{\times}\right){}_{C}\hat{\mathbf{T}} && \mathbf{b}_g = \tilde{\mathbf{b}}_g + \hat{\mathbf{b}}_g && \mathbf{b}_a = \tilde{\mathbf{b}}_a + \hat{\mathbf{b}}_a \quad (6)\\
&{}_{O}\mathbf{T} = \exp\left({}_{O}\boldsymbol{\xi}_{\times}\right){}_{O}\hat{\mathbf{T}} && \delta\mathbf{s} = \delta\tilde{\mathbf{s}} + \delta\hat{\mathbf{s}} && \delta\mathbf{u} = \delta\tilde{\mathbf{u}} + \delta\hat{\mathbf{u}}.
\end{aligned}$$

Next, we define the geometric-keypoint, semantic-keypoint, bounding-box, and regularization error terms and describe how to perform the optimization in (4) over the object states $\mathcal{O}$. We emphasize that the error function arguments include the object, camera, and IMU poses, defined on the $SE(3)$ manifold, and, hence, particular care should be taken when obtaining the error Jacobians.

A. Landmark and Object Error Functions

Define the geometric keypoint error as the difference between the image projection of a geometric landmark $\boldsymbol{\ell}$ using camera pose ${}_{C}\mathbf{T}$ and its associated keypoint observation ${}^{g}\mathbf{z}$:

$${}^{g}\mathbf{e}\left(\mathbf{x}, \boldsymbol{\ell}, {}^{g}\mathbf{z}\right) \triangleq \mathbf{P}\pi\left({}_{C}\mathbf{T}^{-1}\underline{\boldsymbol{\ell}}\right) - {}^{g}\mathbf{z}, \quad (7)$$

where $\mathbf{P} = \begin{bmatrix}\mathbf{I}_2 & \mathbf{0}\end{bmatrix} \in \mathbb{R}^{2\times 4}$ and $\pi(\underline{\mathbf{s}}) \triangleq \frac{1}{\mathbf{s}_3}\underline{\mathbf{s}} \in \mathbb{R}^4$ is the perspective projection function.

Proposition 1. The Jacobians of ${}^{g}\mathbf{e}$ with respect to perturbations ${}_{C}\boldsymbol{\xi}$, $\tilde{\boldsymbol{\ell}}$, evaluated at estimates $\hat{\mathbf{x}}_t$, $\hat{\boldsymbol{\ell}}$, are:

$$\begin{aligned}
\frac{\partial^{g}\mathbf{e}}{\partial_{C}\boldsymbol{\xi}_t} &= -\mathbf{P}\frac{d\pi}{d\underline{\mathbf{s}}}\left({}_{C}\hat{\mathbf{T}}_t^{-1}\underline{\hat{\boldsymbol{\ell}}}\right){}_{C}\hat{\mathbf{T}}_t^{-1}\left[\underline{\hat{\boldsymbol{\ell}}}\right]^{\odot} \in \mathbb{R}^{2\times 6}\\
\frac{\partial^{g}\mathbf{e}}{\partial\tilde{\boldsymbol{\ell}}} &= \mathbf{P}\frac{d\pi}{d\underline{\mathbf{s}}}\left({}_{C}\hat{\mathbf{T}}_t^{-1}\underline{\hat{\boldsymbol{\ell}}}\right){}_{C}\hat{\mathbf{T}}_t^{-1}\begin{bmatrix}\mathbf{I}_3\\ \mathbf{0}^{\top}\end{bmatrix} \in \mathbb{R}^{2\times 3}
\end{aligned} \quad (8)$$

where $\frac{d\pi}{d\underline{\mathbf{s}}}(\underline{\mathbf{s}})$ is the Jacobian of $\pi(\underline{\mathbf{s}})$ evaluated at $\underline{\mathbf{s}}$.

The semantic-keypoint error is defined as the difference between a semantic landmark $\mathbf{s}_j + \delta\mathbf{s}_j$, projected to the image plane using instance pose ${}_{O}\mathbf{T}$ and camera pose ${}_{C}\mathbf{T}_t$, and its corresponding semantic keypoint observation ${}^{s}\mathbf{z}_{t,j,k}$:

$${}^{s}\mathbf{e}(\mathbf{x}, \mathbf{o}, {}^{s}\mathbf{z}) \triangleq \mathbf{P}\pi\left({}_{C}\mathbf{T}^{-1}{}_{O}\mathbf{T}\left(\underline{\mathbf{s}} + \delta\underline{\mathbf{s}}\right)\right) - {}^{s}\mathbf{z}. \quad (9)$$

Proposition 2. The Jacobians of ${}^{s}\mathbf{e}$ with respect to perturbations ${}_{C}\boldsymbol{\xi}_t, {}_{O}\boldsymbol{\xi}, \delta\tilde{\mathbf{s}}$, evaluated at estimates $\hat{\mathbf{x}}_t$, $\hat{\mathbf{o}}$, are:

$$\begin{aligned}
\frac{\partial^{s}\mathbf{e}}{\partial_{C}\boldsymbol{\xi}_t} &= -\frac{\partial^{s}\mathbf{e}}{\partial_{O}\boldsymbol{\xi}} \in \mathbb{R}^{2\times 6} \quad (10)\\
\frac{\partial^{s}\mathbf{e}}{\partial_{O}\boldsymbol{\xi}} &= \mathbf{P}\frac{d\pi}{d\underline{\mathbf{s}}}\left({}_{C}\hat{\mathbf{T}}_t^{-1}{}_{O}\hat{\mathbf{T}}\left(\underline{\mathbf{s}}_j + \delta\underline{\hat{\mathbf{s}}}\right)_j\right){}_{C}\hat{\mathbf{T}}_t^{-1}\left[{}_{O}\hat{\mathbf{T}}\left(\underline{\mathbf{s}}_j + \delta\underline{\hat{\mathbf{s}}}\right)_j\right]^{\odot}\\
\frac{\partial^{s}\mathbf{e}}{\partial\delta\tilde{\mathbf{s}}_j} &= \mathbf{P}\frac{d\pi}{d\underline{\mathbf{s}}}\left({}_{C}\hat{\mathbf{T}}_t^{-1}{}_{O}\hat{\mathbf{T}}\left(\underline{\mathbf{s}}_j + \delta\underline{\hat{\mathbf{s}}}\right)_j\right){}_{C}\hat{\mathbf{T}}_t^{-1}{}_{O}\hat{\mathbf{T}}\begin{bmatrix}\mathbf{I}_3\\ \mathbf{0}^{\top}\end{bmatrix} \in \mathbb{R}^{2\times 3}.
\end{aligned}$$

The Jacobians with resp. to other perturbations in (6) are $\mathbf{0}$.

To define a bounding-box error, we observe that if the dual ellipsoid $\mathbf{Q}^{*}_{(\mathbf{u}+\delta\mathbf{u})}$ of instance $\mathbf{o}_i$ is estimated accurately, then the lines ${}^{b}\underline{\mathbf{z}}_{t,j,k}$ of the k-th bounding-box at time t should be tangent to the image plane conic projection of $\mathbf{Q}^{*}_{(\mathbf{u}+\delta\mathbf{u})}$:

$${}^{b}\mathbf{e}(\mathbf{x}, \mathbf{o}, {}^{b}\underline{\mathbf{z}}) \triangleq {}^{b}\underline{\mathbf{z}}^{\top}\mathbf{P}{}_{C}\mathbf{T}^{-1}{}_{O}\mathbf{T}\mathbf{Q}^{*}_{(\mathbf{u}+\delta\mathbf{u})}{}_{O}\mathbf{T}^{\top}{}_{C}\mathbf{T}^{-\top}\mathbf{P}^{\top}{}^{b}\underline{\mathbf{z}}. \quad (11)$$

Proposition 3. The Jacobians of ${}^{b}\mathbf{e}$ with respect to perturbations ${}_{C}\boldsymbol{\xi}_t, {}_{O}\boldsymbol{\xi}, \delta\tilde{\mathbf{u}}$, evaluated at estimates $\hat{\mathbf{x}}_t$, $\hat{\mathbf{o}}$, are:

$$\begin{aligned}
\frac{\partial^{b}\mathbf{e}}{\partial_{C}\boldsymbol{\xi}_t} &= -\frac{\partial^{b}\mathbf{e}}{\partial_{O}\boldsymbol{\xi}} \in \mathbb{R}^{1\times 6}\\
\frac{\partial^{b}\mathbf{e}}{\partial_{O}\boldsymbol{\xi}} &= 2{}^{b}\underline{\mathbf{z}}^{\top}\mathbf{P}{}_{C}\hat{\mathbf{T}}_t^{-1}{}_{O}\hat{\mathbf{T}}\hat{\mathbf{Q}}^{*}_{(\mathbf{u}+\delta\hat{\mathbf{u}})}{}_{O}\hat{\mathbf{T}}^{\top}\left[{}_{C}\hat{\mathbf{T}}_t^{-\top}\mathbf{P}^{\top}{}^{b}\underline{\mathbf{z}}\right]^{\circledcirc\top}\\
\frac{\partial^{b}\mathbf{e}}{\partial\delta\tilde{\mathbf{u}}} &= \left(2(\mathbf{u}+\delta\hat{\mathbf{u}})\odot\mathbf{y}\odot\mathbf{y}\right)^{\top} \in \mathbb{R}^{1\times 3} \quad (12)\\
\mathbf{y} &\triangleq \begin{bmatrix}\mathbf{I}_3 & \mathbf{0}\end{bmatrix}{}_{O}\hat{\mathbf{T}}^{\top}{}_{C}\hat{\mathbf{T}}_t^{-\top}\mathbf{P}^{\top}{}^{b}\underline{\mathbf{z}}.
\end{aligned}$$

where $\odot$ denotes element wise multiplication. The Jacobians with resp. to other perturbations in (6) are $\mathbf{0}$.

Finally, the object shape regularization error is defined as:

$${}^{r}\mathbf{e}\left(\mathbf{o}\right) \triangleq \begin{bmatrix}\delta\mathbf{u} & \frac{1}{N_s^2}\delta\mathbf{s}\end{bmatrix} \in \mathbb{R}^{3\times(1+N_s)}, \quad (13)$$

whose Jacobians with respect to the perturbations $\delta\tilde{\mathbf{u}}$, $\delta\tilde{\mathbf{s}}$ are:

$$\frac{\partial\delta\mathbf{u}}{\partial\delta\tilde{\mathbf{u}}} = \mathbf{I}_3 \qquad \frac{\partial\delta\mathbf{s}}{\partial\delta\tilde{\mathbf{s}}} = \frac{1}{N_s^2}\mathbf{I} \in \mathbb{R}^{3\times N_s\times 3\times N_s}. \quad (14)$$

B. Landmark and Object State Optimization

We temporarily assume that the sensor trajectory $\mathcal{X}$ is known. Given $\mathcal{X}$, the optimization over $\mathcal{L}$ and $\mathcal{O}$ is decoupled into individual geometric landmark and object instance optimization problems. The error terms in these decoupled problems can be linearized around initial estimates $\hat{\boldsymbol{\ell}}_m$ and $\hat{\mathbf{o}}_i$, using the Jacobians in Prop. 1, 2, and 3, leading to:

$$\begin{aligned}
&\min_{\tilde{\boldsymbol{\ell}}_m} {}^{g}w\sum_{t,n}\mathbb{1}_{t,m,n}\left\|{}^{g}\hat{\mathbf{e}}_{t,m,n} + \frac{\partial^{g}\hat{\mathbf{e}}_{t,m,n}}{\partial\tilde{\boldsymbol{\ell}}_m}\tilde{\boldsymbol{\ell}}_m\right\|^2_{{}^{g}\mathbf{V}}\\
&\min_{\tilde{\mathbf{o}}_i} {}^{s}w\sum_{t,j,k}\mathbb{1}_{t,i,k}\left\|{}^{s}\hat{\mathbf{e}}_{t,i,j,k} + \frac{\partial^{s}\hat{\mathbf{e}}_{t,i,j,k}}{\partial\tilde{\mathbf{o}}_i}\tilde{\mathbf{o}}_i\right\|^2_{{}^{s}\mathbf{V}} + \quad (15)\\
&{}^{b}w\sum_{t,j,k}\mathbb{1}_{t,i,k}\left\|{}^{b}\hat{\mathbf{e}}_{t,i,j,k} + \frac{\partial^{b}\hat{\mathbf{e}}_{t,i,j,k}}{\partial\tilde{\mathbf{o}}_i}\tilde{\mathbf{o}}_i\right\|^2_{{}^{b}\mathbf{V}} + {}^{r}w\left\|{}^{r}\mathbf{e}(\hat{\mathbf{o}}_i) + \frac{\partial^{r}\mathbf{e}(\hat{\mathbf{o}}_i)}{\partial\tilde{\mathbf{o}}_i}\tilde{\mathbf{o}}_i\right\|^2
\end{aligned}$$

These unconstrained quadratic programs in $\tilde{\boldsymbol{\ell}}_m$ and $\tilde{\mathbf{o}}_i$ can be solved iteratively via the Levenberg-Marquardt algorithm [19, Ch.4], updating $\hat{\boldsymbol{\ell}}_m \leftarrow \tilde{\boldsymbol{\ell}}_m + \hat{\boldsymbol{\ell}}_m$ and $\hat{\mathbf{o}}_i \leftarrow \tilde{\mathbf{o}}_i \oplus \hat{\mathbf{o}}_i$ until convergence to a local minimum.

The geometric landmarks are initialized by solving the linear system of equations:

$$\mathbf{0} = {}^{g}\hat{\mathbf{e}}_{t,m,n} = \mathbf{P}_C \hat{\mathbf{T}}_t^{-1} \underline{\hat{\boldsymbol{\ell}}}_m - \lambda_{t,n}{}^{g}\mathbf{z}_{t,n} \tag{16}$$

for all t, m, n such that $\mathbb{1}_{t,m,n} = 1$, where the unknowns are $\hat{\boldsymbol{\ell}}_m$ and the keypoint depths $\lambda_{t,n}$. The deformations of an object instance $\hat{\mathbf{o}}_i$ are initialized as $\delta\hat{\mathbf{s}} = \mathbf{0}$ and $\delta\hat{\mathbf{u}} = \mathbf{0}$. The instance pose is determined from the system of equations:

$$\begin{aligned}
\mathbf{0} &= {}^{s}\hat{\mathbf{e}}_{t,i,j,k} = \mathbf{P}_C \hat{\mathbf{T}}_t^{-1} {}_{O}\hat{\mathbf{T}} \underline{\mathbf{s}}_j - \lambda_{t,j,k}{}^{s}\mathbf{z}_{t,j,k} \\
0 &= {}^{b}\hat{\mathbf{e}}_{t,i,j,k} = {}^{b}\underline{\mathbf{z}}_{t,j,k}^{\top} \mathbf{P}_C \hat{\mathbf{T}}_t^{-1} {}_{O}\hat{\mathbf{T}} \mathbf{Q}_{\mathbf{u}}^{*} {}_{O}\hat{\mathbf{T}}^{\top} {}_{C}\hat{\mathbf{T}}_t^{-\top} \mathbf{P}^{\top} {}^{b}\underline{\mathbf{z}}_{t,j,k}
\end{aligned} \tag{17}$$

for all j and all t, k such that $\mathbb{1}_{t,i,k} = 1$, where the unknowns are ${}_{O}\hat{\mathbf{T}}$ and the semantic keypoint depths $\lambda_{t,j,k}$. This is a generalization of the pose from n point correspondences (PnP) problem [29]. While this system may be solved using polynomial equations [30], we perform a more efficient initialization by defining $\boldsymbol{\zeta}_j \triangleq {}_{O}\hat{\mathbf{R}}\mathbf{s}_j + {}_{O}\hat{\mathbf{p}}$ and solving the first set of (now linear) equation in (17) for $\boldsymbol{\zeta}_j$ and $\lambda_{t,j,k}$. We recover ${}_{O}\hat{\mathbf{T}}$ via the Kabsch algorithm [31] between $\{\boldsymbol{\zeta}_j\}$ and $\{\mathbf{s}_j\}$. This approach works well as long as there is a sufficient number of semantic keypoints ${}^{s}\mathbf{z}_{t,j,k}$ (at least two per landmark across time for at least three semantic landmarks $\mathbf{s}_j$) associated with the object. If fewer semantic keypoints are available, we use an initial guess ${}_{O}\hat{\mathbf{R}}$ provided by the keypoint detection algorithm StarMap [23] and solve (17) for ${}_{O}\hat{\mathbf{p}}$ and $\lambda_{t,j,k}$. Given ${}_{O}\hat{\mathbf{R}}$, we eliminate ${}_{O}\hat{\mathbf{p}}$ in (17), reducing the problem to a set of quadratic equations in the positive scalars $\lambda_{t,j,k}$ and then recover ${}_{O}\hat{\mathbf{p}}$ from $\lambda_{t,j,k}$ and ${}_{O}\hat{\mathbf{R}}$.

VI. The OrcVIO Algorithm

We return to the problem of joint IMU-camera, geometric landmark, and object instance optimization. The IMU-camera state is tracked using an Extended Kalman filter. Predictions are performed using the inertial observations ${}^{i}\mathbf{z}_t$. When a geometric keypoint or object track is lost, iterative optimization is performed over $\hat{\boldsymbol{\ell}}_m$ or $\hat{\mathbf{o}}_i$ as discussed in Sec. V and the final landmark and instance estimates are used to update the IMU-camera state mean $\hat{\mathbf{x}}_t$ and covariance $\boldsymbol{\Sigma}_t$. This object-based version of the multi-state constraint Kalman filter [4] is described next.

A. Prediction Step

The continuous-time IMU dynamics are [32]:

$$\begin{aligned}
&{}_{I}\dot{\mathbf{R}} = {}_{I}\mathbf{R}\left({}^{i}\boldsymbol{\omega} - \mathbf{b}_g - \mathbf{n}_\omega\right)_\times \qquad \dot{\mathbf{b}}_g = \mathbf{n}_g \qquad \dot{\mathbf{b}}_a = \mathbf{n}_a \\
&{}_{I}\dot{\mathbf{p}} = {}_{I}\mathbf{v} \qquad {}_{I}\dot{\mathbf{v}} = {}_{I}\mathbf{R}\left({}^{i}\mathbf{a} - \mathbf{b}_a - \mathbf{n}_a\right) + \mathbf{g}
\end{aligned} \tag{18}$$

where $\mathbf{n}_\omega, \mathbf{n}_a, \mathbf{n}_g, \mathbf{n}_a \in \mathbb{R}^3$ are Brownian motion noise terms associated with the angular velocity measurements, linear acceleration measurements, gyroscope bias, and accelerometer bias. Using the perturbations in (6), we can split (18) into deterministic nominal and stochastic error dynamics:

$$\begin{aligned}
&{}_{I}\dot{\hat{\mathbf{R}}} = {}_{I}\hat{\mathbf{R}}\left({}^{i}\boldsymbol{\omega} - \hat{\mathbf{b}}_g\right)_\times \qquad \dot{\hat{\mathbf{b}}}_g = \mathbf{0} \qquad \dot{\hat{\mathbf{b}}}_a = \mathbf{0} \\
&{}_{I}\dot{\hat{\mathbf{p}}} = {}_{I}\hat{\mathbf{v}} \qquad {}_{I}\dot{\hat{\mathbf{v}}} = {}_{I}\hat{\mathbf{R}}\left({}^{i}\mathbf{a} - \hat{\mathbf{b}}_a\right) + \mathbf{g} \\
&{}_{I}\dot{\boldsymbol{\theta}} = -{}_{I}\hat{\mathbf{R}}\left(\tilde{\mathbf{b}}_g + \mathbf{n}_\omega\right) \qquad \dot{\tilde{\mathbf{b}}}_g = \mathbf{n}_g \qquad \dot{\tilde{\mathbf{b}}}_a = \mathbf{n}_a \\
&{}_{I}\dot{\tilde{\mathbf{p}}} = {}_{I}\tilde{\mathbf{v}} \qquad {}_{I}\dot{\tilde{\mathbf{v}}} = -\left[{}_{I}\hat{\mathbf{R}}\left({}^{i}\mathbf{a} - \hat{\mathbf{b}}_a\right)\right]_\times {}_{I}\boldsymbol{\theta} - {}_{I}\hat{\mathbf{R}}\left(\tilde{\mathbf{b}}_a + \mathbf{n}_a\right)
\end{aligned} \tag{19}$$

Given time discretization τ and assuming ${}^{i}\boldsymbol{\omega}$ and ${}^{i}\mathbf{a}$ remain constant over intervals of length τ, we can integrate the nominal dynamics in *closed-form* to obtain the predicted IMU mean ${}_{I}\hat{\mathbf{x}}_{t+1}^{p}$:

$$\begin{aligned}
&{}_{I}\hat{\mathbf{R}}_{t+1}^{p} = {}_{I}\hat{\mathbf{R}}_t \exp\left(\tau\left({}^{i}\boldsymbol{\omega}_t - \hat{\mathbf{b}}_{g,t}\right)_\times\right) \\
&\hat{\mathbf{b}}_{g,t+1}^{p} = \hat{\mathbf{b}}_{g,t} \qquad \hat{\mathbf{b}}_{a,t+1}^{p} = \hat{\mathbf{b}}_{a,t} \\
&{}_{I}\hat{\mathbf{p}}_{t+1}^{p} = {}_{I}\hat{\mathbf{p}}_t + {}_{I}\hat{\mathbf{v}}_t\tau + \mathbf{g}\frac{\tau^2}{2} + {}_{I}\hat{\mathbf{R}}_t\mathbf{H}_L\left(\tau\left({}^{i}\boldsymbol{\omega}_t - \hat{\mathbf{b}}_{g,t}\right)\right)\left({}^{i}\mathbf{a}_t - \hat{\mathbf{b}}_{a,t}\right)\tau^2 \\
&{}_{I}\hat{\mathbf{v}}_{t+1}^{p} = {}_{I}\hat{\mathbf{v}}_t + \mathbf{g}\tau + {}_{I}\hat{\mathbf{R}}_t\mathbf{J}_L\left(\tau\left({}^{i}\boldsymbol{\omega}_t - \hat{\mathbf{b}}_{g,t}\right)\right)\left({}^{i}\mathbf{a}_t - \hat{\mathbf{b}}_{a,t}\right)\tau
\end{aligned} \tag{20}$$

where $\mathbf{J}_L(\boldsymbol{\omega}) \triangleq \left(\mathbf{I} + \frac{\boldsymbol{\omega}_\times}{2!} + \frac{\boldsymbol{\omega}_\times^2}{3!} + \ldots\right)$ is the left Jacobian of $SO(3)$ and $\mathbf{H}_L(\boldsymbol{\omega}) \triangleq \left(\frac{\mathbf{I}}{2!} + \frac{\boldsymbol{\omega}_\times}{3!} + \frac{\boldsymbol{\omega}_\times^2}{4!} + \ldots\right)$. Both $\mathbf{J}_L(\boldsymbol{\omega})$ and $\mathbf{H}_L(\boldsymbol{\omega})$ have closed-form (Rodrigues-like) expressions:

$$\begin{aligned}
\mathbf{J}_L(\boldsymbol{\omega}) &= \mathbf{I}_3 + \frac{1 - \cos\|\boldsymbol{\omega}\|}{\|\boldsymbol{\omega}\|^2}\boldsymbol{\omega}_\times + \frac{\|\boldsymbol{\omega}\| - \sin\|\boldsymbol{\omega}\|}{\|\boldsymbol{\omega}\|^3}\boldsymbol{\omega}_\times^2 \\
\mathbf{H}_L(\boldsymbol{\omega}) &= \frac{1}{2}\mathbf{I}_3 + \frac{\|\boldsymbol{\omega}\| - \sin\|\boldsymbol{\omega}\|}{\|\boldsymbol{\omega}\|^3}\boldsymbol{\omega}_\times + \frac{2(\cos\|\boldsymbol{\omega}\| - 1) + \|\boldsymbol{\omega}\|^2}{2\|\boldsymbol{\omega}\|^4}\boldsymbol{\omega}_\times^2.
\end{aligned} \tag{21}$$

To obtain ${}_{C}\hat{\mathbf{x}}_{t+1}^{p}$, the camera poses are augmented with a new predicted pose ${}_{C}\hat{\mathbf{T}}_{t+1}^{p}$ based on ${}_{I}\hat{\mathbf{x}}_{t+1}^{p}$:

$${}_{C}\hat{\mathbf{T}}_{t+1}^{p} = {}_{I}\hat{\mathbf{T}}_{t+1}^{p}{}_{C}^{I}\mathbf{T} \tag{22}$$

where ${}_{C}^{I}\mathbf{T}$ is assumed known from an offline IMU-camera calibration [33]. The oldest pose ${}_{C}\hat{\mathbf{T}}_{t+1-W}$ is removed from ${}_{C}\hat{\mathbf{x}}_{t+1}^{p}$ to ensure that W is not exceeded.

Next, consider the propagation of the state covariance $\boldsymbol{\Sigma}_t \in \mathbb{R}^{(15+6W)\times(15+6W)}$. We use Euler discretization of the IMU error dynamics in (19) with time step τ to obtain:

$$\tilde{\mathbf{x}}_{t+1}^{p} = \begin{bmatrix} \mathbf{F}_t & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_{6W} \end{bmatrix} \tilde{\mathbf{x}}_t + \mathbf{n}_t \qquad \mathbf{n}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}) \tag{23}$$

where the matrices $\mathbf{F}_t$ and $\mathbf{Q}$ are:

$$\mathbf{F}_t = \begin{bmatrix}
\mathbf{I}_3 & \mathbf{0} & \mathbf{0} & -\tau{}_{I}\hat{\mathbf{R}}_t & \mathbf{0} \\
\mathbf{0} & \mathbf{I}_3 & \tau\mathbf{I}_3 & \mathbf{0} & \mathbf{0} \\
-\tau\left[{}_{I}\hat{\mathbf{R}}_t\left({}^{i}\mathbf{a}_t - \hat{\mathbf{b}}_{a,t}\right)\right]_\times & \mathbf{0} & \mathbf{I}_3 & \mathbf{0} & -\tau{}_{I}\hat{\mathbf{R}}_t \\
\mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I}_3 & \mathbf{0} \\
\mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I}_3
\end{bmatrix}$$

$$\mathbf{Q} = \mathbf{diag}\left(\sigma_{\boldsymbol{\omega}}^2\tau^2\mathbf{I}_3, \mathbf{0}_3, \sigma_{\mathbf{a}}^2\tau^2\mathbf{I}_3, \sigma_g^2\tau\mathbf{I}_3, \sigma_a^2\tau\mathbf{I}_3, \mathbf{0}_{6W}\right) \tag{24}$$

and $\sigma_{\boldsymbol{\omega}}$ $[rad/s]$, $\sigma_{\mathbf{a}}$ $[m/s^2]$, σ_g $[rad/s^{3/2}]$, σ_a $[m/s^{5/2}]$ can be obtained from the IMU datasheet or experimental data [34, Appendix E]. The propagated state covariance is:

$$\boldsymbol{\Sigma}_{t+1}^{p} = \begin{bmatrix} \mathbf{F}_t & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_{6W} \end{bmatrix} \boldsymbol{\Sigma}_t \begin{bmatrix} \mathbf{F}_t & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_{6W} \end{bmatrix}^{\top} + \mathbf{Q}. \tag{25}$$

Finally, after adding a new camera pose and dropping the oldest one, the covariance matrix becomes:

$$\boldsymbol{\Sigma}^p_{t+1} = \mathbf{A}_{t+1}\boldsymbol{\Sigma}^p_{t+1}\mathbf{A}^\top_{t+1} \tag{26}$$

$$\mathbf{A}_{t+1} \triangleq \begin{bmatrix} \mathbf{I}_{15} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I}_{6(W-1)} \\ \mathbf{B}_{t+1} & \mathbf{0} & \mathbf{0} \end{bmatrix} \quad \mathbf{B}_{t+1} \triangleq \begin{bmatrix} ({}_I\hat{\mathbf{p}}_{t+1})_\times & \mathbf{I}_3 & \mathbf{0} \\ \mathbf{I}_3 & \mathbf{0} & \mathbf{0} \end{bmatrix}$$

where $\mathbf{B}_{t+1}$ is the Jacobian of the camera pose perturbation ${}_C\boldsymbol{\xi}^p_{t+1}$ with respect to the IMU perturbation ${}_I\tilde{\mathbf{x}}^p_{t+1}$.

B. Update Step

We perform an update to $\hat{\mathbf{x}}^p_{t+1}$ and $\boldsymbol{\Sigma}^p_{t+1}$ without storing the geometric landmarks $\hat{\boldsymbol{\ell}}_m$ or object instances $\hat{\mathbf{i}}_i$ in the filter state using the null-space projection idea of [4]. Let $\hat{\mathbf{y}}_i$ denote an estimate (from Sec. V) of a geometric landmark or object instance whose track gets lost at time $t+1$. The geometric and semantic error functions, linearized using perturbations ${}_C\boldsymbol{\xi}^p_t$, $\tilde{\mathbf{y}}_i$ around estimates ${}_C\hat{\mathbf{T}}^p_t$, $\hat{\mathbf{y}}_i$ are of the form:

$$\mathbf{e}_{t,i} \approx \hat{\mathbf{e}}_{t,i} + \frac{\partial \hat{\mathbf{e}}_{t,i}}{\partial_C \boldsymbol{\xi}^p_t} {}_C\boldsymbol{\xi}^p_t + \frac{\partial \hat{\mathbf{e}}_{t,i}}{\partial \tilde{\mathbf{y}}_i}\tilde{\mathbf{y}}_i + \mathbf{n}_{t,i} \tag{27}$$

where $\mathbf{n}_{t,i}$ is the associated noise term with covariance $\mathbf{V}_{t,i}$. Stacking the errors for all camera poses in $\hat{\mathbf{x}}^p_{t+1}$ associated with i, leads to:

$$\mathbf{e}_i \approx \hat{\mathbf{e}}_i + \frac{\partial \hat{\mathbf{e}}_i}{\partial \tilde{\mathbf{x}}^p_{t+1}}\tilde{\mathbf{x}}^p_{t+1} + \frac{\partial \hat{\mathbf{e}}_i}{\partial \tilde{\mathbf{y}}_i}\tilde{\mathbf{y}}_i + \mathbf{n}_i. \tag{28}$$

The perturbations $\tilde{\mathbf{y}}_i$ can be eliminated by left-multiplication of the errors in (28) with unitary matrices $\mathbf{N}_i$ whose columns form the basis of the left nullspaces of $\frac{\partial \hat{\mathbf{e}}_i}{\partial \tilde{\mathbf{y}}_i}$:

$$\mathbf{N}_i^\top \mathbf{e}_i \approx \mathbf{N}_i^\top \hat{\mathbf{e}}_i + \mathbf{N}_i^\top \frac{\partial \hat{\mathbf{e}}_i}{\partial \tilde{\mathbf{x}}^p_{t+1}}\tilde{\mathbf{x}}^p_{t+1} + \mathbf{N}_i^\top \mathbf{n}_i \tag{29}$$

Finally, let $\hat{\mathbf{e}}$, $\mathbf{J}$, $\mathbf{V}$ be the stacked errors, Jacobians, and noise covariances (after null-space projection) across all geometric landmarks and object instances, whose tracks are lost at $t+1$. The updated IMU-camera mean and covariance are:

$$\begin{aligned} \mathbf{K} &= \boldsymbol{\Sigma}^p_{t+1}\mathbf{J}^\top \left(\mathbf{J}\boldsymbol{\Sigma}^p_{t+1}\mathbf{J}^\top + \mathbf{V}\right)^{-1} \\ \hat{\mathbf{x}}_{t+1} &= (-\mathbf{K}\hat{\mathbf{e}}) \oplus \hat{\mathbf{x}}^p_{t+1} \\ \boldsymbol{\Sigma}_{t+1} &= (\mathbf{I} - \mathbf{K}\mathbf{J})\boldsymbol{\Sigma}^p_{t+1}(\mathbf{I} - \mathbf{K}\mathbf{J})^\top + \mathbf{K}\mathbf{V}\mathbf{K}^\top. \end{aligned} \tag{30}$$

Note that the dimension of $\mathbf{J}$ can be reduced in the computation above via QR factorization as described in [4].

VII. Evaluation

We evaluate OrcVIO on the KITTI dataset [10] both qualitatively and quantitatively. We use the raw data sequences with object annotations to evaluate the object state estimation, and the odometry sequences without object annotations for trajectory accuracy evaluation. Since KITTI provides synchronized velocity measurements, we use a simpler velocity-based prediction step, described in Sec. IX.

Fig. 4 shows the IMU-camera trajectory and object states estimated on KITTI odometry sequence 07. The result demonstrates that OrcVIO is capable of producing meaningful object-level maps and accurate sensor trajectories (without loop closure). Moreover, the estimated car instance shapes vary in accordance with the semantic observations.

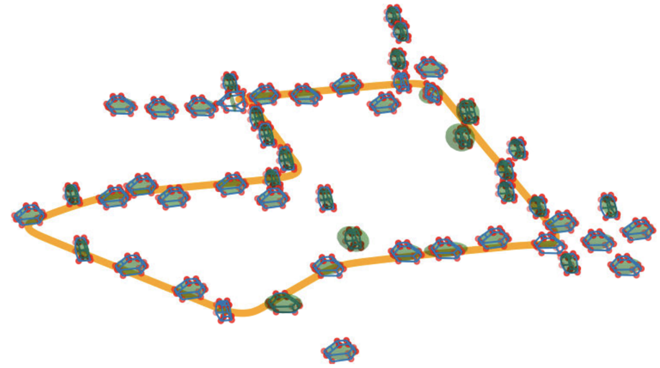

Fig. 4. Estimated IMU-camera trajectory (yellow) and object states (red landmarks and green ellipsoids) from KITTI Odom. Seq. 07.

The quality of the estimated object poses and shapes is evaluated using 3D Intersection over Union (IoU). We obtain a 3D bounding box $\hat{b}_i$ from the semantic landmarks of each estimated object instance. The 3D IoU is defined as the ratio of the intersection volume over the union volume,

$$\text{3D IoU} := \sum_i \frac{\text{Volume of Intersection}(\hat{b}_i, b_i)}{\text{Volume of Union}(\hat{b}_i, b_i)}, \tag{31}$$

with respect to the 3D bounding box b_i of the closest ground truth object. The ground truth 3D annotations are obtained from the KITTI tracklets and the KITTI detection benchmark labels. Table I reports 3D IoU results comparing OrcVIO against state-of-the-art methods including a deep learning approach for single-view bounding box estimation (Single-View [35]), and the multi-view bundle-adjustment algorithm that uses cuboids to represent objects (CubeSLAM [16]). The performances of OrcVIO and CubeSLAM are similar since both rely on point and bounding box measurements to optimize the object states.

There are two approaches for evaluating VIO quantitatively: Absolute Trajectory Error (ATE) and the Relative Pose Error (RPE). We show the RPE [10] in Table I, defined as the average norm of the position component of the error poses $\left({}_I\hat{\mathbf{T}}_{t+1}{}_I\hat{\mathbf{T}}_t^{-1}\right)\left({}_I\mathbf{T}_{t+1}{}_I\mathbf{T}_t^{-1}\right)^{-1}$ over distance travelled. Table I shows that OrcVIO is also very close to CubeSLAM for RPE.

Since the object mapping evaluation in Table I does not contain the same number of detected objects, and to better understand the distribution for the rotation/translation errors of the estimated objects, we compare the precision and recall of OrcVIO on the KITTI raw sequences (2011_09_26_00XX, XX = [01, 19, 22, 23, 35, 36, 39, 61, 64, 93]) against a single-view, end-to-end object estimation approach (SubCNN [36]), and a visual-inertial object detector (VIS-FNL [14]). An object estimate is *true positive* if a ground truth object exists within a specific rotation/translation error threshold. We define *precision* as the fraction of true positives over all estimated objects, whereas *recall* as the fraction of true positives over all ground truth objects. In Table II, the

TABLE I

OBJECT DETECTION AND POSE ESTIMATION ON KITTI OBJECT SEQUENCES

Metric	KITTI Sequence →	22	23	36	39	61	64	95	96	117	Mean
3D IoU	SingleView [35]	0.52	0.32	0.50	0.54	**0.54**	0.43	0.40	0.26	0.25	0.42
	CubeSLAM [16]	**0.58**	0.35	**0.54**	**0.59**	0.50	**0.48**	**0.52**	0.29	**0.35**	**0.47**
	OrcVIO	0.44	**0.56**	0.52	0.54	0.48	0.44	0.38	0.34	0.29	0.44
Trans. error (%)	CubeSLAM [16]	1.68	**1.72**	2.93	1.61	**1.24**	**0.93**	1.49	1.81	2.21	**1.74**
	OrcVIO	**1.64**	2.51	**2.11**	1.03	3.11	2.48	**1.05**	1.40	**1.36**	1.85

TABLE II

PRECISION-RECALL EVALUATION ON KITTI OBJECT SEQUENCES

	Translation error →	≤ 0.5 m		≤ 1.0 m		≤ 1.5 m	
Rotation error	Method	Precision	Recall	Precision	Recall	Precision	Recall
$\leq 30°$	SubCNN [36]	0.10	0.07	0.26	0.17	0.38	0.26
	VIS-FNL [14]	**0.14**	0.10	**0.34**	**0.24**	**0.49**	**0.35**
	OrcVIO	0.10	**0.12**	0.18	0.21	0.22	0.25
$\leq 45°$	SubCNN [36]	0.10	0.07	0.26	0.17	0.38	0.26
	VIS-FNL [14]	**0.15**	0.11	**0.35**	0.25	**0.50**	**0.36**
	OrcVIO	**0.15**	**0.17**	0.25	**0.28**	0.31	0.35
–	SubCNN [36]	0.10	0.07	0.27	0.18	0.41	0.28
	VIS-FNL [14]	0.16	0.11	0.40	0.29	0.58	0.42
	OrcVIO	**0.29**	**0.33**	**0.50**	**0.56**	**0.62**	**0.69**

TABLE III

TRAJECTORY RMSE (M) ON KITTI ODOMETRY SEQUENCES

KITTI Sequence →	00	02	04	05	06	07	08	09	10	Mean
Object BA [15]	73.4	55.5	10.7	50.8	73.1	47.1	72.2	31.2	53.5	51.9
CubeSLAM [16]	**13.9**	**26.2**	**1.1**	4.8	7.0	2.7	**10.7**	10.7	**8.4**	**9.5**
OrcVIO	25.7	27.1	1.2	**4.6**	**4.5**	**1.8**	14.3	**10.4**	8.8	10.9

first six rows are the precision and recall associated with different translation error (row) and rotation error (column) thresholds, whereas the last 3 rows ignore the rotation error. The results demonstrate that OrcVIO is able to retrieve a reasonable amount of the groundtruth objects and provide a high-quality object map. When both rotation and translation errors are considered (in the first six rows), OrcVIO is better than SubCNN, since the latter does not rely on temporal association of objects. In contrast, OrcVIO is slight worse than VIS-FNL possibly explained by the fact that VIS-FNL uses multiple object hypotheses, while OrcVIO only keeps one object state. Moreover, OrcVIO outperforms SubCNN and VIS-FNL when only translation error is taken into account, which suggests that the object position estimates are accurate but the orientation estimates could be improved.

Although the main contribution of OrcVIO is object mapping, we also evaluate the ATE of the IMU-camera trajectory estimation in Table III for completeness. Suppose that at time t the groundtruth pose is ${}_I\mathbf{T}_t$, while the estimate is ${}_I\hat{\mathbf{T}}_t$ and then the error is $\Delta_I\mathbf{T}_t = \{\Delta_I\mathbf{R}_t, \Delta_I\mathbf{p}_t\}$ for rotation, position, where $\Delta_I\mathbf{R}_t = {}_I\mathbf{R}_t\left({}_I\hat{\mathbf{R}}_t\right)^\top, \Delta_I\mathbf{p}_t = {}_I\mathbf{p}_t - \Delta_I\mathbf{R}_{tI}\hat{\mathbf{p}}_t$. The root mean square error (RMSE) of translational ATE [37] is: $ATE_{pos} = \frac{1}{N}\sum_{t=0}^{t=N-1}(||\Delta_I\mathbf{p}_t||^2)^{\frac{1}{2}}$. OrcVIO is compared with two visual object SLAM methods: CubeSLAM [16], and a monocular visual SLAM that integrates spherical object models to estimate the scale via bundle-adjustment (Object BA [15]). Table III shows that OrcVIO outperforms Object BA, because spheres are very coarse object representations compared to our coarse-to-fine object representations, and thus Object BA cannot maintain the object scales as accurately as OrcVIO. Moreover, OrcVIO uses inertial data while Object BA is a visual odometry. CubeSLAM has better performance on some of the sequences, and one possible reason is that OrcVIO uses low frequency velocity measurements but we assume the velocity stays constant during the prediction step in Sec. VI-A, which could increase the drift.

VIII. CONCLUSION

This paper presents a joint ego-motion, object pose and shape estimation algorithm, which may enable robots to perform tasks involving object perception and manipulation. We have also shown that OrcVIO is capable of estimating the trajectory and producing object-level maps on real world KITTI dataset. Future work will focus on more general models of object shape, multiple object categories and object-level data association for loop closure. We will also include more object categories for mapping.

IX. Appendix: Velocity-based State Propagation

For a robot, such as a ground wheeled vehicle [38], equipped with a velocity sensor, we may use a simpler kinematic motion model based on the linear and angular velocity measurements. The IMU state becomes ${}_I\mathbf{x} \triangleq ({}_I\mathbf{R},\ {}_I\mathbf{p},\ \mathbf{b}_g,\ \mathbf{b}_v)$, and the continuous-time dynamics in Eq. (18) simplify to:

$$\begin{aligned} {}_I\dot{\mathbf{R}} &= {}_I\mathbf{R}\left({}^i\boldsymbol{\omega} - \mathbf{b}_g - \mathbf{n}_{\boldsymbol{\omega}}\right)_\times & \dot{\mathbf{b}}_g &= \mathbf{n}_g \\ {}_I\dot{\mathbf{p}} &= {}_I\mathbf{R}\left({}^i\mathbf{v} - \mathbf{b}_v - \mathbf{n}_{\mathbf{v}}\right) & \dot{\mathbf{b}}_v &= \mathbf{n}_v \end{aligned} \tag{32}$$

where ${}^i\mathbf{v}$ is the velocity measurement in the body frame. In this case, the matrices $\mathbf{F}_t$ and $\mathbf{Q}$ in Eq. (23) become:

$$\mathbf{F}_t = \begin{bmatrix} \mathbf{I}_3 & \mathbf{0} & -\tau\,{}_I\hat{\mathbf{R}}_t & \mathbf{0} \\ -\tau\left[{}_I\hat{\mathbf{R}}_t\left({}^i\mathbf{v}_t - \hat{\mathbf{b}}_{v,t}\right)\right]_\times & \mathbf{I}_3 & \mathbf{0} & -\tau\,{}_I\hat{\mathbf{R}}_t \\ \mathbf{0} & \mathbf{0} & \mathbf{I}_3 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I}_3 \end{bmatrix}$$

$$\mathbf{Q} = \mathbf{diag}\left(\sigma_{\boldsymbol{\omega}}^2\tau^2\mathbf{I}_3, \sigma_{\mathbf{v}}^2\tau^2\mathbf{I}_3, \sigma_g^2\tau\mathbf{I}_3, \sigma_v^2\tau\mathbf{I}_3, \mathbf{0}_{6W}\right). \tag{33}$$

References

[1] J. Delmerico and D. Scaramuzza, "A Benchmark Comparison of Monocular Visual-Inertial Odometry Algorithms for Flying Robots," in *IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2018.

[2] C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. Reid, and J. J. Leonard, "Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age," *IEEE Transactions on Robotics*, vol. 32, no. 6, pp. 1309–1332, Dec 2016.

[3] R. Mur-Artal and J. D. Tardós, "Orb-SLAM2: An open-source SLAM system for monocular, stereo, and rgb-d cameras," *IEEE Transactions on Robotics*, vol. 33, no. 5, pp. 1255–1262, Oct 2017.

[4] A. I. Mourikis and S. I. Roumeliotis, "A Multi-State Constraint Kalman Filter for Vision-aided Inertial Navigation," in *IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2007, pp. 3565–3572.

[5] T. Qin, P. Li, and S. Shen, "VINS-Mono: A Robust and Versatile Monocular Visual-Inertial State Estimator," *IEEE Transactions on Robotics*, vol. 34, no. 4, pp. 1004–1020, Aug 2018.

[6] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *IEEE International Conference on Computer Vision (ICCV)*, Oct 2017, pp. 2980–2988.

[7] A. Kendall, M. Grimes, and R. Cipolla, "Posenet: A convolutional network for real-time 6-dof camera relocalization," in *IEEE International Conference on Computer Vision (ICCV)*, Dec 2015, pp. 2938–2946.

[8] R. Li, S. Wang, Z. Long, and D. Gu, "Undeepvo: Monocular visual odometry through unsupervised deep learning," in *IEEE International Conference on Robotics and Automation (ICRA)*, May 2018, pp. 7286–7291.

[9] Z. Yin and J. Shi, "Geonet: Unsupervised learning of dense depth, optical flow and camera pose," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018, pp. 1983–1992.

[10] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the KITTI vision benchmark suite," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012, pp. 3354–3361.

[11] J. K. Murthy, G. V. S. Krishna, F. Chhaya, and K. M. Krishna, "Reconstructing vehicles from a single image: Shape priors for road scene understanding," in *IEEE International Conference on Robotics and Automation (ICRA)*, May 2017, pp. 724–731.

[12] P. Parkhiya, R. Khawad, J. K. Murthy, B. Bhowmick, and K. M. Krishna, "Constructing category-specific models for monocular object-SLAM," in *IEEE International Conference on Robotics and Automation (ICRA)*, May 2018, pp. 1–9.

[13] N. Atanasov, S. Bowman, K. Daniilidis, and G. Pappas, "A unifying view of geometry, semantics, and data association in slam," in *International Joint Conference on Artificial Intelligence (IJCAI)*, 2018.

[14] J. Dong, X. Fei, and S. Soatto, "Visual-inertial-semantic scene representation for 3D object detection," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017, pp. 3567–3577.

[15] D. Frost, V. Prisacariu, and D. Murray, "Recovering stable scale in monocular SLAM using object-supplemented bundle adjustment," *IEEE Transactions on Robotics*, vol. 34, no. 3, pp. 736–747, June 2018.

[16] S. Yang and S. Scherer, "CubeSLAM: Monocular 3-D Object SLAM," *IEEE Transactions on Robotics*, vol. 35, no. 4, pp. 925–938, Aug 2019.

[17] L. Nicholson, M. Milford, and N. Sünderhauf, "QuadricSLAM: Dual Quadrics From Object Detections as Landmarks in Object-Oriented SLAM," *IEEE Robotics and Automation Letters*, vol. 4, no. 1, pp. 1–8, 2019.

[18] M. Hosseinzadeh, K. Li, Y. Latif, and I. Reid, "Real-time monocular object-model aware sparse SLAM," in *International Conference on Robotics and Automation (ICRA)*, May 2019, pp. 7123–7129.

[19] T. D. Barfoot, *State Estimation for Robotics*. Cambridge University Press, 2017.

[20] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*. Cambridge university press, 2003.

[21] E. Rosten, R. Porter, and T. Drummond, "Faster and better: A machine learning approach to corner detection," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 32, no. 1, pp. 105–119, 2010.

[22] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," *arXiv preprint arXiv:1804.02767*, 2018.

[23] X. Zhou, A. Karpur, L. Luo, and Q. Huang, "Starmap for category-agnostic keypoint and viewpoint estimation," in *Computer Vision – ECCV*, 2018, pp. 328–345.

[24] S. L. Bowman, N. Atanasov, K. Daniilidis, and G. J. Pappas, "Probabilistic data association for semantic SLAM," in *IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2017, pp. 1722–1729.

[25] B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *Int. Joint Conf. on Artificial Intelligence (IJCAI)*, 1981, pp. 674–679.

[26] D. G. Kottas, K. J. Wu, and S. I. Roumeliotis, "Detecting and dealing with hovering maneuvers in vision-aided inertial navigation systems," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Nov 2013, pp. 3172–3179.

[27] Y. Gal and Z. Ghahramani, "Dropout as a bayesian approximation: Representing model uncertainty in deep learning," in *Proceedings of The 33rd International Conference on Machine Learning*, vol. 48, Jun 2016, pp. 1050–1059.

[28] A. Bewley, Z. Ge, L. Ott, F. Ramos, and B. Upcroft, "Simple online and realtime tracking," in *IEEE International Conference on Image Processing (ICIP)*, Sep 2016, pp. 3464–3468.

[29] J. A. Hesch and S. I. Roumeliotis, "A direct least-squares (DLS) method for PnP," in *International Conference on Computer Vision (ICCV)*, Nov 2011, pp. 383–390.

[30] H. Yang, J. Shi, and L. Carlone, "Teaser: Fast and certifiable point cloud registration," 2020. [Online]. Available: https://github.com/MIT-SPARK/TEASER-plusplus

[31] W. Kabsch, "A discussion of the solution for the best rotation to relate two sets of vectors," *Acta Crystallographica Section A*, vol. 34, no. 5, pp. 827–828, 1978.

[32] N. Trawny and S. Roumeliotis, "Indirect Kalman Filter for 3D Attitude Estimation," University of Minnesota, Dept. of Comp. Sci. & Eng., Tech. Rep, Tech. Rep., 2005.

[33] P. Furgale, J. Rehder, and R. Siegwart, "Unified temporal and spatial calibration for multi-sensor systems," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2013, pp. 1280–1286.

[34] J. Sola, "Quaternion kinematics for the error-state KF," *LAAS-CNRS, Toulouse, France, Tech. Rep*, 2012.

[35] A. Mousavian, D. Anguelov, J. Flynn, and J. Košecká, "3D bounding box estimation using deep learning and geometry," in *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

[36] Y. Xiang, W. Choi, Y. Lin, and S. Savarese, "Subcategory-aware convolutional neural networks for object proposals and detection," in *IEEE Winter Conference on Applications of Computer Vision (WACV)*, March 2017, pp. 924–933.

[37] Z. Zhang and D. Scaramuzza, "A tutorial on quantitative trajectory evaluation for visual(-inertial) odometry," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct 2018, pp. 7244–7251.

[38] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The KITTI dataset," *The International Journal of Robotics Research*, vol. 32, no. 11, pp. 1231–1237, 2013.