OrcVIO: Object residual constrained Visual Inertial Odometry
Mo Shan, Student Member, IEEE, Vikas Dhiman, Member, IEEE, Qiaojun Feng, Student Member, IEEE, Jinzhao Li, and Nikolay Atanasov, Member, IEEE

Abstract—Introducing object models in the variables optimized by simultaneous localization and mapping (SLAM) algorithms not only improves performance but also supports specification and execution of semantically meaningful robotic tasks. This work presents an Object residual constrained Visual Inertial Odometry (OrcVIO) algorithm for online sensor localization, tightly coupled with 3D object pose and shape estimation. OrcVIO initializes and optimizes the pose and shape of detected and tracked objects by differentiating through two new optimization terms capturing object-feature and bounding-box measurements. The estimated object poses and shapes aid in real-time incremental multi-state constraint Kalman filtering (MSCKF) over the visual-inertial sensor states. The ability of OrcVIO for accurate sensor trajectory estimation and large-scale object mapping is evaluated using simulated and real data.

Supplementary Material
Software and videos supplementing this paper are available at: http://erl.ucsd.edu/pages/orcvio.html

I. INTRODUCTION

The foundations of visual environment understanding in robotics, machine learning, and computer vision lie in the twin technologies of inferring geometric structure and semantic content. Researchers have made significant progress in geometric structure reconstruction using Structure from Motion (SfM) [1], [2] and SLAM [3] techniques. State of the art SLAM approaches work with monocular or stereo cameras [4], often complemented by inertial information [5], [6]. However, most real-time incremental SLAM techniques provide purely geometric representations, e.g., of points, lines, or planes, that lack semantic interpretation of the environment.

Recently, tremendous progress has been achieved in semantic scene understanding using deep neural networks for object detection [7], instance segmentation [8], and object tracking [9]. Nevertheless, the literature in deep learning is sparse in techniques that provide global positioning of the detected and tracked objects to obtain an object-level map online.

This paper focuses on joint visual-inertial odometry and object mapping, bridging the gap between geometric and semantic inference in SLAM. 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, e.g., for safe navigation, manipulation, multi-stage interaction [10]–[13]. Our work introduces object states, modeling the position, orientation, and shape of object instances in the environment. We utilize a coarse category-agnostic and a fine category-specific model of object shape. The coarse model uses an ellipsoid to restrict an object’s pose variation and relate its coarse shape to bounding-box detections. The fine model uses a set of mid-level object parts (e.g., car wheels, windshield, doors), called semantic keypoints, to obtain a precise part-based shape description.

Given streaming inertial measurement unit (IMU) and monocular camera measurements, we develop an algorithm to simultaneously estimate the IMU-camera trajectory and the states of the objects, detected and tracked in the camera images. The contributions of this paper are summarized as follows.

• We introduce object states in the formulation of a SLAM problem, modeling position, orientation, coarse ellipsoid shape, and fine semantic-keypoint shape.
• We define residuals relating object states and IMU-
camera states to inertial measurements, geometric features, object semantic features, and object bounding-box detections, and explicitly derive their Jacobians.

- We develop an extension of the multi-state constraint Kalman filter (MSCKF) [5] to enable online tightly coupled estimation of object and IMU-camera states. Our innovations include closed-form mean and covariance propagation over the SE(3) pose and velocity manifold of the IMU-camera states and measurement updates based on our new semantic residuals with object states optimized over multiple views. Our algorithm is suitable for real-time incremental odometry and object mapping and is more efficient than nonlinear batch optimization.

We name our method Object residual constrained Visual Inertial Odometry (OrcVIO) to emphasize the role of the semantic residuals in the optimization process. OrcVIO is capable of producing meaningful object maps and estimating accurate sensor trajectories, as shown in Fig. 1.

II. RELATED WORK

Many visual SLAM approaches work with monocular [5], [6], [14]–[16] or stereo cameras [4], [17], [18]. Featureless approaches [19]–[21] that minimize image intensity directly have been proposed, and inertial information is often used to complement the visual information [5], [18], [22]–[24]. The MSCKF algorithm [5] leverages both inertial data and visual features in an extended Kalman filter formulation. Each visual feature whose track is lost provides multi-frame constraints for a corresponding 3D landmark. The constraint residuals are linearized to perform the filter update step. A key idea is to marginalize the landmark states via projection to the null space of the visual feature Jacobians, allowing structureless estimation of the IMU-camera states only. A key extension of OrcVIO over the MSCKF and visual-inertial SLAM algorithms in general is to introduce object measurements (bounding boxes and semantic features) and object states whose residuals constrain the IMU-camera states.

Recent methods have considered learning to regress camera poses directly from raw images [25]–[31]. For instance, monocular depth, optical flow, and ego-motion are jointly optimized from video in [30] by relying on a view-synthesis loss. These unsupervised learning techniques have shown impressive performance in localization but do not generate global maps. In this paper, we focus on obtaining a joint geometric-semantic representations from measurements in real time, i.e., spatial perception [32]. Prior works that utilize both spatial and semantic information include [33]–[40], but the spatial and semantic states are estimated independently and merged later. On the other hand, [41]–[46] consider joint metric and semantic mapping. Recent works focus on the tightly coupled spatial and semantic estimation, and there are mainly two groups of object-based SLAM techniques: category-specific and category-agnostic.

Category-specific approaches optimize the pose and shape of object instances, using semantic keypoints [47], [48] or 3D shape models [49]–[57]. For example, [49] introduces a real-time joint 3D object pose and camera pose estimation via pose graph optimization. The objects stored in a database that are also present in the current frame are detected and optimized, using the vertex and normal map from a RGBD sensor. Object pose and shape are optimized in [48] using semantic keypoints to provide additional error terms related to object pose in the SLAM factor graph. Visual-inertial information for object mapping is used in [53], relying on a database to retrieve object shapes. Hu et al. [54] embed object priors into least-squares minimization to incrementally track and map chairs. The object shape represented by a binary voxel grid is compactly described by a latent code obtained from an auto-encoder, which is used for shape initialization and iterative residual minimization. These methods in general are computationally demanding due to iterative batch optimization and reliance on instance-specific CAD models.

Category-agnostic approaches use geometric shapes, such as spheres [58], [59], cuboids [60], [61], or ellipsoids [62]–[69], to represent objects. SSFM [60] uses the tightest bounding cube enclosing an object to parameterize the object location and pose. The object measurements include the location, size of the object bounding box and the object pose obtained from a 3D object detector [70]. Assuming the semantic measurements are consistent across frames, an object state is optimized via maximum likelihood estimation. CubeSLAM [61] generates and refines 3D cuboid proposals using multi-view bundle adjustment without relying on prior models. QuadricSLAM [65] 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, have also been introduced [71]. Using generic symmetric shapes, however, makes the orientation of object instances potentially irrecoverable. For instance, the front and back of an object become indistinguishable.

This paper extends our prior conference publication [72] with several theoretical contributions and a large-scale experimental evaluation of OrcVIO. While [72] used linear velocity measurements, this paper uses a complete description of all IMU states for odometry and derives a novel closed-form expression for covariance propagation. A new bounding-box residual is developed to ensure that it scales equivalently as the keypoint residual. In [72], the bounding-box residual was quadratic instead of linear as a function of the bounding-box lines. This version also implements a zero-velocity residual to reduce VIO drift when the sensor is completely static. Extensive evaluation is conducted on the KITTI dataset, a photo-realistic Unity dataset, and in indoor and outdoor real-time experiments. We also extend OrcVIO to handle multiple object classes, including cars, doors, barriers, monitors, chairs.

III. BACKGROUND AND NOTATION

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:

\[
\mathbf{R}_A^B \triangleq \begin{bmatrix} \mathbf{R}^B_A & \mathbf{p}_A^B \\ 0 & 1 \end{bmatrix} \in SE(3)
\]

(1)

where \(\mathbf{R}_A^B \in SO(3)\) is a rotation matrix and \(\mathbf{p}_A^B \in \mathbb{R}^3\) is a translation vector. To simplify the notation, we will not explic-
Fig. 2: (a) An object class is defined by a semantic class $\sigma$ and average shape specified by semantic landmarks $s$ (blue) and an ellipsoid with semi-axes lengths $u$ (red). (b) A specific instance has landmark and ellipsoid deformations parameterized by $\delta s$ (blue arrows) and $\delta u$ (red arrows). (c) The landmarks and ellipsoid are transformed from the object frame $\{O\}$ to the global frame $\{G\}$ via the instance pose $\phi \cdot T$.

We overload $(\cdot)_x$ to denote the mapping from an axis-angle vector $\theta$ to a $3 \times 3$ skew-symmetric matrix $\theta \times \in so(3)$ as well as from a vector $\xi \in \mathbb{R}^6$ to a $4 \times 4$ twist matrix:

$$\xi = \begin{bmatrix} \rho \\ \theta \end{bmatrix} \in \mathbb{R}^6, \quad \xi_x := \begin{bmatrix} \theta^T \\ 0 \end{bmatrix} \in se(3).$$

We define an infinitesimal change of pose $T \in SE(3)$ using a right perturbation $T \exp(\xi_T) \in SE(3)$ (see [73, Ch.7]).

Let $x = [x^T 1]^T \in \mathbb{R}^2$ be the homogeneous coordinates of a vector $x$. For $x \in \mathbb{R}^3$, we define the operators $x^0$ and $x^\circ$:

$$x^0 \triangleq \begin{bmatrix} I_3 \\ -x^T \end{bmatrix} \in \mathbb{R}^{4 \times 6}, \quad x^\circ \triangleq \begin{bmatrix} 0 \\ x \end{bmatrix} \in \mathbb{R}^{6 \times 4},$$

where $I_3$ is the $3 \times 3$ identity matrix. A quadric shape [74, Ch.3] is a set $\{ x \mid x^T Q x \leq 0 \}$, where $Q$ is a symmetric matrix. Consider an axis-aligned ellipsoid centered at 0:

$$E_u \triangleq \{ x \mid x^T U^{-1} U^{-1} x \leq 1 \},$$

where $U \triangleq \text{diag}(u)$ and the elements of the vector $u$ are the lengths of the semi-axes of $E_u$. In homogeneous coordinates, $E_u$ is a special case of a quadric shape $\{ x \mid x^T Q u x \leq 0 \}$ with $Q_u \triangleq \text{diag}(U^{-2}, -1)$. A quadric shape can also be defined in dual form, as the set of planes $\pi = Q x$ that are tangent to the shape surface at each $x$. A dual quadric surface is defined as $\{ \pi \mid \pi^T Q^* \pi = 0 \}$, where $Q^* = \text{adj}(Q)^T$. A dual quadric surface defined by $Q^* \in \mathbb{R}^{4 \times 4}$ can be transformed by $T \in SE(3)$ to another reference frame as $T^{*} Q^{*} T^T$. Similarly, it can be projected to a lower-dimensional space by a projection matrix $P \triangleq [I \ 0]$ as $P Q^* P^T$.

IV. PROBLEM FORMULATION

Consider a system equipped with an IMU-camera sensor. Let $X_k \triangleq (x_k, v_k, p_k, b_{k,h}, b_{k,s})$ be the IMU state at time $t_k$, consisting of orientation $R_k \in SO(3)$, velocity $v_k \in \mathbb{R}^3$, position $p_k \in \mathbb{R}^3$, gyroscope bias $b_{k,h} \in \mathbb{R}^3$, and accelerometer bias $b_{k,s} \in \mathbb{R}^3$. Assume that the camera is rigidly attached to the IMU with relative transformation $I \cdot T \in SE(3)$, known from extrinsic calibration. Given $X_k$, the camera pose can be obtained as $C_k = I \cdot T_{k}^{*} T$, where $I \cdot T_k \in SE(3)$ is the IMU pose. To facilitate the use of multi-frame camera information, define an augmented state $x_k \triangleq (x_k, I \cdot T_{k-1}, \ldots, I \cdot T_{k-W})$, containing a sliding window of $W$ past IMU poses in addition to the current IMU state $x_k$.

The system state over time is $X \triangleq \{ x_k \}_k$.

Suppose that the system evolves in an unknown environment, containing geometric landmarks $L \triangleq \{ l_m \}_m$ and objects $O \triangleq \{ o_i \}_i$, represented in a global frame $\{ G \}$. A geometric landmark is a static point $l_m \in \mathbb{R}^3$, detectable via image keypoint algorithms, such as FAST [75]. An object $o_i = (c_i, l_i)$ is an instance $i$ of a semantic class $c_i$, detectable via object recognition algorithms, such as YOLO [76]. The precise definitions of object class and instance follow.

Definition. An object class is a tuple $c \triangleq (\sigma, s, u)$, where $\sigma \in \mathbb{N}$ specifies a semantic type (e.g., chair, table, monitor) and $s \in \mathbb{R}^{3 \times N_c}$ and $u \in \mathbb{R}^3$ specify the average class shape. The class shape is modeled by an axis-aligned ellipsoid $E_u$ and a set of semantic landmarks $s_i \in \mathbb{R}^3$ corresponding to the columns of $s$. The semantic landmarks $s_i$ define the 3D positions of mid-level object parts $O_i$ in the object class canonical frame $\{ O \}$.

Definition. An object instance of class $c$ is a tuple $i \triangleq (\sigma, T, \delta s, \delta u)$, where $\sigma \in \mathbb{N}$ specifies a semantic type (e.g., chair, table, monitor) and $s \in \mathbb{R}^{3 \times N_c}$ and $u \in \mathbb{R}^3$ specify the average class shape. The shape of an object $i$ in the global frame $\{ G \}$ is obtained by deforming and transforming the semantic landmark positions, $\sigma \cdot T \cdot (\delta s + \delta s_i)$, and the dual ellipsoid, $\sigma \cdot T Q^* \cdot (\delta u + \delta u_i) \cdot T^T$, using the instance pose $\sigma \cdot T$ and deformations $\delta s, \delta u$. Fig. 2 shows an illustration for a car model with 12 semantic landmarks.

The IMU-camera sensor provides inertial measurements $\dot{x}_k, \dot{y}_k, \dot{z}_k$, geometric keypoint measurements $\dot{g} x_{k,n}$, and semantic measurements, containing object class $\dot{c} z_{k,j}$, bounding-box $\dot{b} z_{k,i,j}$, and semantic keypoint $\dot{a} z_{k,l,j}$ detections, illustrated in Fig. 3. The inertial measurements $\dot{z}_k \triangleq (\dot{\omega}_k, \dot{a}_k) \in \mathbb{R}^6$ are the IMU’s body-frame angular velocity $\dot{\omega}_k$ and linear acceleration $\dot{a}_k$ at time $t_k$. The geometric keypoint measurements are noisy detections $\dot{g} x_{k,n} \in \mathbb{R}^2$ in normalized pixel coordinates of the image projections of the subset of geometric landmarks $L$ visible to the camera at time $t$. To obtain semantic observations, an object detection algorithm is applied to the image at time $t_k$, followed by semantic keypoint extraction within each detected bounding-box (see Sec. V-A for details). The $j$-th

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

2Given pixel coordinates $x \in \mathbb{R}^2$ and a camera intrinsic calibration matrix $K \in \mathbb{R}^{3 \times 3}$, the normalized pixel coordinates of $x$ are $P K^{-1} x \in \mathbb{R}^2$. 


object detection provides the object class \( c_{k,j} \in \mathbb{N} \), bounding box \( b_{k,l,j} \in \mathbb{R}^2 \), described by \( l = 1, \ldots, 4 \) lines in normalized pixel coordinates, and semantic keypoints \( s_{k,l,j} \in \mathbb{R}^2 \) in normalized pixel coordinates associated with the \( l = 1, \ldots, N_s \) semantic landmarks\(^9\).

Let \( I_{k,m,n} \in \{0, 1\} \) indicate whether the \( n \)-th geometric keypoint observed at time \( t_k \) is associated with the \( m \)-th geometric landmark. Similarly, let \( I_{k,i,j} \in \{0, 1\} \) indicate whether the \( j \)-th object detection at time \( t_k \) is associated with the \( i \)-th object instance. This data association information can be obtained by keypoint and object tracking as described in Sec. V-A. Given the associations, we introduce error functions:

\[
\begin{align*}
&\xi_{k,k+1} \triangleq \xi(x_{k}, x_{k+1}, i_{z_k}) \\
&\gamma_{k,m,n} \triangleq \gamma(x_k, \ell_m, i_{z_{k,n}}) \\
&s_{k,l,j} \triangleq s(x_k, o_i, i_{z_{k,l,j}}) \\
&b_{k,l,j} \triangleq b(x_k, o_i, i_{z_{k,l,j}})
\end{align*}
\]

for the inertial, geometric, semantic keypoint and bounding-box measurements, respectively, defined precisely in Sec. V. We also introduce an object shape regularization error term \( r^*e(o_i) \) to ensure that the instance deformations \((\delta s, \delta u)\) remain small, and consider the following problem.

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

\[
\begin{align*}
&\min_{X, L, O} \sum_k \xi_{k,k+1}^2 \|e_{k,k+1}\|^2_V + g\sum_{k,m,n} \gamma_{k,m,n}^2 \|e_{k,m,n}\|^2_V \\
&+ s \sum_{k,l,j} I_{k,i,j} \|s_{k,l,j}\|^2 + b \sum_{k,l,j} I_{k,i,j} \|b_{k,l,j}\|^2 \\
&+ r \sum_i \|r^*e(o_i)\|^2 \text{ for } i = 1, \ldots, N_s
\end{align*}
\]

where \( *w \) are positive constants determining the relative importance of the error terms and \( *V \) are positive-definite matrices specifying the covariances associated with the inertial, geometric, semantic, and bounding-box measurements. A measurement covariance \( V \) defines a quadratic (Mahalanobis) norm \( \|e\|^2_V \triangleq e^T V^{-1} e \).

\(^9\)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.

Inspired by the MSCKF algorithm [5], we decouple the optimization over \( L \) and \( O \) from that over \( X \) to design an efficient real-time algorithm. When a geometric-keypoint or object track is lost, we perform multi-view iterative optimization over the corresponding geometric landmark \( \ell_m \) or object \( o_i \) based on the estimate of the latest IMU-camera state \( x_t \). The IMU-camera state is propagated using the inertial observations and updated using the optimized geometric landmark and object states and the geometric and semantic observations. This decoupling leads to potentially lower accuracy but higher efficiency compared to window or batch keyframe optimization techniques [50]. Our approach is among the first to offer tight coupling between semantic information and geometric structure in visual-inertial odometry. Error functions and Jacobians derived in Sec. V can be used for batch keyframe optimization in factor-graph formulation of object SLAM [65].

**V. LANDMARK AND OBJECT RECONSTRUCTION**

Our approach consists for a front-end measurement generation stage and a back-end landmark state and sensor pose optimization stage. This section discusses the detection and tracking of geometric keypoint measurements \( g z_{k,n} \) and object class \( c z_{k,j} \), bounding-box \( b z_{k,l,j} \), and semantic keypoint \( s z_{k,l,j} \) measurements in the front-end. It also defines the error functions in (5) and their Jacobians needed for the back-end optimization. Finally, it presents the back-end optimization over the geometric landmarks \( L \) and the object states \( O \) for a given sensor trajectory \( X \).

**A. Keypoint and Object Tracking**

Geometric keypoints \( g z_{k,n} \) are detected in the camera images using the FAST detector [75] and are tracked temporarily using the Lucas-Kanade (LK) algorithm [78]. Keypoint-based tracking has lower accuracy but higher efficiency than descriptor-based methods, allowing our method to use a high frame-rate camera and process more keypoints. 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
consecutive images is short, the relative orientation is obtained by integrating the gyro measurements $\omega_k$ as described in Sec. VI-A and only the unit translation vector is estimated using two-point RANSAC [79].

The YOLO detector [7] is used to detect object classes $s z_{k,i,j}$ and bounding-box lines $b z_{k,i,j}$. Semantic keypoints $s z_{k,i,j}$ are extracted within each bounding box using the StarMap stacked hourglass convolutional neural network [77]. We augment the original StarMap network with dropout layers as shown in Fig. 3(b). Several stochastic forward passes may be preformed using Monte Carlo dropout [80] to obtain semantic keypoint covariances $s V$, illustrated in Fig. 3(c).

The bounding boxes $b z_{k,i,j}$ are tracked temporally using the SORT algorithm [81], which performs intersection over union (IoU) matching via the Hungarian algorithm. The semantic keypoints $s z_{k,i,j}$ within each bounding box are tracked via a Kalman filter, which uses the DK algorithm for prediction and the StarMap keypoint detections for update.

### B. Landmark and Object Error Functions

Next, we define the geometric-keypoint $s e_{k,m,n}$, semantic-keypoint $s e_{k,i,j}$, bounding-box $b e_{k,i,j}$, and regularization $s e(\partial x)$ error terms in (5) and derive their Jacobians. The error function arguments include the IMU, camera, and object poses, defined on the $S E(3)$ manifold, and, hence, particular care should be taken when obtaining the Jacobians. The error functions are linearized around estimates of the IMU-camera state $x_k$, geometric landmarks $l_m$, and objects $o_i$ using perturbations $x_k$, $l_m$, and $o_i$:

$$x_k = \hat{x}_k \oplus \tilde{x}_k, \quad l_m = \hat{l}_m + \tilde{l}_m, \quad o_i = \hat{o}_i \oplus \tilde{o}_i, \quad (6)$$

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

$$\begin{align*}
\hat{R} &= \hat{R} \exp (\hat{\theta}_\times) \\
\hat{P} &= \hat{P} + \hat{\tilde{P}} \\
\hat{v} &= \hat{\tilde{v}} + \hat{\tilde{v}} \\
\hat{c} &= \hat{c} \exp (\hat{c}_\times) \\
\hat{b}_g &= \hat{b}_g + \hat{\tilde{b}}_g \\
\bar{b}_a &= \hat{b}_a + \hat{\tilde{b}}_a \\
\hat{o}_T &= \hat{o} \exp (\hat{o}_\times) \\
\hat{\delta}_s &= \hat{\delta}_s + \hat{\tilde{\delta}}_s \\
\bar{\delta}_u &= \hat{\bar{\delta}}_u + \hat{\tilde{\delta}}_u,
\end{align*}$$

where we use right perturbations $\hat{\theta}_\times \in so(3)$, $\hat{c}_\times \in se(3)$, and $\hat{o}_\times \in se(3)$ for the IMU orientation $\hat{R}$, camera pose $\hat{c} T$, and object pose $\hat{o}_T$, respectively.

We define the geometric-keypoint error as the difference between the image projection of a geometric landmark $l$ in camera frame $c T = \hat{c} T l c T l$ and its associated keypoint observation $s e$:

$$s e(x, l, s e) \triangleq P (c T^{-1} l c T) - s e,$$  \hspace{1cm} (8)

where $P = [I_2 \ 0] \in R^{2 \times 4}$ is a projection matrix and $\pi(s) \triangleq \frac{1}{s T s} \in R^2$ is the perspective projection function.

#### Proposition 1

The Jacobians of $s e$ in (8) with respect to the IMU pose perturbation $\partial x$, $\partial c T$, and the landmark position perturbation $\hat{l}$, evaluated at estimates $\hat{x}$, $\hat{c}_T$, and $\hat{l}$, are:

$$\begin{align*}
\frac{\partial s e}{\partial \hat{x}} &= -P \frac{\partial \pi}{\partial \delta s} \left( c T^{-1} l c T \right) \otimes \partial c T \in R^{2 \times 6}, \\
\frac{\partial s e}{\partial \hat{c}_T} &= P \frac{\partial \pi}{\partial \delta s} \left( c T^{-1} l c T \right) c T^{-1} I_2 \in R^{2 \times 3}, \quad (9)
\end{align*}$$

where $[\cdot] \otimes$ is defined in (3), $\frac{\partial \pi}{\partial \delta s}(s)$ is the Jacobian of $\pi(s)$ and:

$$\frac{\partial c T}{\partial \hat{l}} = \left[ -I R T, \frac{1}{l} c R T, 0 \right] \in R^{6 \times 6} \quad (10)$$

The Jacobians with respect to other perturbations in (7) are 0.

#### Proof

See Sec. IX-A.

The semantic-keypoint error is defined as the difference between the projection of a semantic landmark $s_i + \delta s_i$ from the object frame to the image plane, using instance pose $o_i T$ and camera pose $c T$, and its corresponding semantic-keypoint observation $\in z$:

$$s e(x, o, s e) \triangleq P \pi (c T^{-1} o_i T (s_i + \delta s_i)) - s e.$$  \hspace{1cm} (11)

#### Proposition 2

The Jacobians of $s e$ in (11) with respect to perturbations $\partial x$, $\partial o_i$, $\partial s_i$, evaluated at estimates $\hat{x}$, $\hat{o}_i$, and $\hat{s}_i$, are:

$$\frac{\partial s e}{\partial \hat{x}} = -P \frac{\partial \pi}{\partial \delta s} \left( c T^{-1} o_i T (s_i + \delta s_i) \right) \left[ c T^{-1} o_i T (s_i + \delta s_i) \right]^\circ \in R^{2 \times 6},$$

$$\frac{\partial s e}{\partial \hat{o}_i} = P \frac{\partial \pi}{\partial \delta s} \left( c T^{-1} o_i T (s_i + \delta s_i) \right) c T^{-1} o_i T \left[ (I_2 \ 0) \right] \in R^{2 \times 3}.$$  \hspace{1cm} (12)

The Jacobians with respect to other perturbations in (7) are 0.

#### Proof

See Sec. IX-B.

We define the bounding-box error as the distance between the hyperplane $b = [b^T - b_h]^T \triangleq o_i T c T^{-1} P b z$ induced by projecting the bounding-box line $b z$ to the object frame and the closest hyperplane that is tangent to the quadric surface $Q_{u + \delta u}(u + \delta u) (u + \delta u)$.

$$b e(x, o, b e) \triangleq \frac{1}{\|b\|} \left( \text{sgn}(b_h) \sqrt{b^T \text{diag}(u + \delta u)^2 b - b_h} \right) \quad (13)$$

where $\text{sgn}(x) = \frac{\partial \pi}{\partial x}$ is the sign function.

#### Proposition 3

The Jacobians of $b e$ in (13) with respect to perturbations $\partial x$, $\partial o_i$, and $\partial s_i$, evaluated at estimates $\hat{x}$, $\hat{o}_i$, and $\hat{s}_i$, are:

$$\begin{align*}
\frac{\partial b e}{\partial \hat{x}} &= \text{sgn}(b_h) \sqrt{b^T \text{diag}(u + \delta u)^2 b - b_h} \quad (14)
\end{align*}$$

where $\bar{b} = o_i T c T^{-1} P b z$ and with $\bar{U} \triangleq \text{diag}(u + \delta u)$:

$$\begin{align*}
\frac{\partial b e}{\partial \bar{b}} &= \text{sgn}(b_h) \sqrt{b^T \text{diag}(u + \delta u)^2 b - b_h} \quad (15)
\end{align*}$$

The Jacobians with respect to other perturbations in (7) are 0.
Proof. See Sec. IX-C.

Finally, the object shape regularization error is defined as:
\[
\mathbf{r}(\mathbf{o}) = \left[ \frac{\partial \mathbf{e}}{\partial \mathbf{u}} \right]^T \sum_{k=N}^1 \mathbf{g}_{k,m,n} + \frac{\partial^2 \mathbf{e}}{\partial \mathbf{u}^2} \mathbf{e}_{k,m,n} + \mathbf{f}_{k,m,n} \mathbf{e}_{k,m,n} \mathbf{e}_{k,m,n}^T \mathbf{f}_{k,m,n}^T.
\]
whose Jacobians with respect to the perturbations \(\mathbf{u}, \mathbf{u}\) are:
\[
\frac{\partial \mathbf{e}}{\partial \mathbf{u}} = [I_{3} \quad 0_{3N}]^T, \quad \frac{\partial^2 \mathbf{e}}{\partial \mathbf{u}^2} = [0_{3N} \quad I_{3} \quad 0_{3(N-I)}]^T.
\]

C. Landmark and Object State Optimization

We temporarily assume that the sensor trajectory \(\chi\) is known. Given \(\chi\), the optimization over \(\mathcal{L}\) and \(O\) decouples into individual geometric landmark and object optimization problems. The error terms in the decoupled problems can be linearized around initial estimates \(\mathbf{e}_{k,m,n}\) and \(\mathbf{o}_{i}\), using the Jacobians in Propositions 1, 2, and 3, leading to:
\[
\min_{\mathbf{e}_{k,m,n}} \mathbf{g}_{k,m,n} \mathbf{e}_{k,m,n} + \frac{\partial \mathbf{f}_{k,m,n}}{\partial \mathbf{e}_{k,m,n}} \mathbf{e}_{k,m,n} \mathbf{e}_{k,m,n}^T \mathbf{f}_{k,m,n}^T.
\]
\[
\min_{\mathbf{o}_{i}} \mathbf{g}_{k,m,n} \mathbf{e}_{k,m,n} + \frac{\partial \mathbf{f}_{k,m,n}}{\partial \mathbf{e}_{k,m,n}} \mathbf{e}_{k,m,n} \mathbf{e}_{k,m,n}^T \mathbf{f}_{k,m,n}^T.
\]
for all \(k, m, n\) such that \(\mathbf{f}_{k,m,n} = 1\), where the unknowns are \(\mathbf{e}_{k,m,n}\) and the keypoint depths \(\lambda_{k,m,n}\). The deformations of an object instance \(\mathbf{o}_{i}\) are initialized as \(\mathbf{d}_{k,m,n} = 0\) and \(\mathbf{u}_{k,m,n} = 0\). The instance pose is determined from the system of equations consisting of semantic keypoint and bounding-box line residuals:
\[
\mathbf{P}_{k} \mathbf{C}_{k} \mathbf{T}_{k}^{-1} \mathbf{e}_{k,m,n} = \lambda_{k,m,n} \mathbf{g}_{k,m,n} \mathbf{e}_{k,m,n} = 0
\]
for all \(k, m, n\) such that \(\mathbf{f}_{k,m,n} = 1\), where the unknowns are \(\mathbf{e}_{k,m,n}\) and the semantic keypoint depths \(\lambda_{k,m,n}\).

The least squares keypoint for semantic keypoint is a generalization of the pose from \(n\) point correspondences (PnP) problem [82]. While this system may be solved using polynomial equations [83], we perform a more efficient initialization by defining \(\mathbf{\zeta}_{k,m,n} = \mathbf{R}_{k,m,n} \mathbf{s}_{k,m,n} + \mathbf{\hat{o}}_{k,m,n}\) and solving the first set of (now linear) equations in (19) for \(\mathbf{\zeta}_{k,m,n}\) and \(\mathbf{\hat{o}}_{k,m,n}\). We recover \(\mathbf{T}_{k}\) via the Kabsch algorithm [84] between \(\{\mathbf{\zeta}_{i}\}\) and \(\{\mathbf{s}_{j}\}\). This approach works well as long as there is a sufficient number of semantic keypoints \(\mathbf{s}_{k,m,n}\) (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, \(\mathbf{\hat{o}}_{k,m,n}\) can be recovered from the second equation in (19). Let
\[
\mathbf{Q}^* \triangleq \mathbf{O} \mathbf{T}_{k} \mathbf{O} \mathbf{T}_{k}^{-1}, \quad \mathbf{p}_{k,m,n} \triangleq \mathbf{b}_{k,m,n} \mathbf{P}_{k} \mathbf{C}_{k}^{-1}, \quad \text{then a linear system } \mathbf{M} \mathbf{w} = 0 \text{ can be formed:}
\]
\[
\mathbf{M} \triangleq \begin{bmatrix}
\mathbf{Q}^*_{k,m,n} \otimes \mathbf{T}_{k,m,n}^{-1} \\
\vdots \\
\mathbf{Q}^*_{k,m,n} \otimes \mathbf{T}_{k,m,n}^{-1}
\end{bmatrix} \mathbf{w} \triangleq \text{vec}(\mathbf{Q}^*).
\]
for all \(l\) and all \(k, m, n\), the optimization over \(\mathbf{\hat{o}}_{k,m,n}\) and \(\mathbf{\hat{o}}_{k,m,n}\) can be solved iteratively via the Levenberg-Marquardt algorithm [73, Ch.4], updating \(\mathbf{e}_{k,m,n} = \mathbf{e}_{k,m,n} + \mathbf{e}_{k,m,n} + \mathbf{e}_{k,m,n} \mathbf{e}_{k,m,n}^T \mathbf{e}_{k,m,n}^T\).

The power spectral densities \(\sigma_{\mathbf{x}_{k,m,n}}\) are white Gaussian noise terms associated with the angular velocity, linear acceleration, gyroscope bias, and accelerometer bias, respectively. The power spectral densities \(\sigma_{\mathbf{x}_{k,m,n}} [\text{rad}^2/s^2], \sigma_{\mathbf{x}_{k,m,n}} [m^2/s^3], \sigma_{\mathbf{x}_{k,m,n}} [m^2/s^3]\) can be obtained from the IMU datasheet or experimental data [86, Appendix E]. Using

VI. The ORCVIO ALGORITHM

We return to the problem of joint IMU-camera, geometric-landmark, and object optimization and describe the ORCVIO algorithm. The IMU-camera state \(\mathbf{x}_{k} \) is tracked using an extended Kalman filter with mean \(\mathbf{x}_{k}\) and covariance \(\mathbf{\Sigma}_{k}\). Prediction of the mean \(\mathbf{\hat{x}}_{k+1}\) and covariance \(\mathbf{\Sigma}_{k}^{p}\) is performed using the inertial measurements \(\mathbf{\hat{x}}_{k}\). When a geometric-landmark or object track is lost, iterative optimization is performed over \(\mathbf{e}_{k,m,n}\) and \(\mathbf{\hat{o}}_{k,m,n}\) as discussed in Sec. V. The optimized geometric landmark and object estimates are used to update the IMU-camera mean \(\mathbf{\hat{x}}_{k+1}\) and covariance \(\mathbf{\Sigma}_{k+1}^{p}\). ORCVIO is an extension of the MSCKF [5], which performs closed-form prediction and integrates object states in the update.

A. Prediction Step

The continuous-time IMU dynamics are [85]:
\[
\dot{\mathbf{\hat{r}}} = \mathbf{\hat{r}} \left( \mathbf{w} - \mathbf{b}_{g} - \mathbf{g} \right), \quad \dot{\mathbf{b}}_{g} = \mathbf{n}_{g}, \quad \dot{\mathbf{b}}_{a} = \mathbf{n}_{a},
\]
\[
\dot{\mathbf{v}} = \mathbf{\hat{v}} \left( \mathbf{a} - \mathbf{b}_{a} - \mathbf{g} \right), \quad \dot{\mathbf{p}} = \mathbf{\hat{p}}.
\]
for all \(\mathbf{\hat{x}}_{k,m,n}, \mathbf{\hat{y}}_{k,m,n}, \mathbf{\hat{z}}_{k,m,n} \in \mathbb{R}^3\) are white Gaussian noise terms associated with the angular velocity, linear acceleration, gyroscope bias, and accelerometer bias, respectively. The power spectral densities \(\sigma_{\mathbf{\hat{x}}_{k,m,n}} [\text{rad}^2/s^2], \sigma_{\mathbf{\hat{x}}_{k,m,n}} [m^2/s^3], \sigma_{\mathbf{\hat{x}}_{k,m,n}} [m^2/s^3]\) can be obtained from the IMU datasheet or experimental data [86, Appendix E]. Using
the perturbations in (7), we can split (22) into deterministic nominal dynamics:
\[ i\dot{\mathbf{R}} = i\dot{\mathbf{R}}(i\omega - \mathbf{b}_g) \times, \quad \dot{\mathbf{b}}_g = 0, \quad \dot{\mathbf{a}}_a = 0, \]
\[ i\dot{\mathbf{v}} = i\dot{\mathbf{R}}(i\mathbf{a} - \mathbf{b}_a) + \mathbf{g}, \quad i\dot{\mathbf{p}} = i\dot{\mathbf{v}}, \tag{23} \]
and stochastic error dynamics:
\[ i\dot{\mathbf{\theta}} = -(i\omega - \mathbf{b}_g) \times \mathbf{\theta} - (\dot{\mathbf{b}}_g + \mathbf{n}_\omega), \]
\[ i\dot{\mathbf{v}} = -i\dot{\mathbf{R}}(i\mathbf{a} - \mathbf{b}_a) i\mathbf{\theta} - i\dot{\mathbf{R}}(\mathbf{b}_a + \mathbf{n}_a), \tag{24} \]
Given time discretization \( \tau_k \) and assuming \( i\omega(t) \) and \( i\mathbf{a}(t) \) remain constant over the interval \( t \in [t_k, t_k + \tau_k) \) with values \( i\mathbf{z}_k = (i\omega_k, i\mathbf{a}_k) \), we can compute the predicted IMU-camera state mean and covariance from (23) and (24), respectively. Let \( \mathbf{x}_k \) and \( \Sigma_k \) be the prior mean and covariance.

**Proposition 4.** The nominal dynamics (23) can be integrated in closed-form to obtain the predicted mean \( \mathbf{x}_{k+1}^p \):
\[ i\dot{\mathbf{R}}_{k+1}^p = i\dot{\mathbf{R}}_k \exp \left( \tau_k (i\omega_k - \mathbf{b}_g) \times \right), \]
\[ i\dot{\mathbf{v}}_{k+1}^p = i\dot{\mathbf{v}}_k + \mathbf{g}\tau_k + i\dot{\mathbf{R}}_k \mathbf{J}_L (\tau_k (i\omega_k - \mathbf{b}_g), (i\mathbf{a}_k - \mathbf{b}_a, \mathbf{n}_a), \tau_k, \mathbf{m}_L), \]
\[ i\dot{\mathbf{p}}_{k+1}^p = i\dot{\mathbf{p}}_k + i\dot{\mathbf{v}}_k \tau_k + \mathbf{g}\tau_k + i\dot{\mathbf{R}}_k \mathbf{H}_L (\tau_k (i\omega_k - \mathbf{b}_g), (i\mathbf{a}_k - \mathbf{b}_a, \mathbf{n}_a), \tau_k, \mathbf{m}_L), \]
where \( \mathbf{J}_L (\omega) \triangleq \mathbf{I}_3 + \frac{i\omega}{||\omega||^2} + \frac{\omega}{||\omega||} + \ldots \) is the left Jacobian of \( SO(3) \) and \( \mathbf{H}_L (\omega) \triangleq \mathbf{I}_3 + \frac{i\omega}{||\omega||^2} + \frac{\omega}{||\omega||} + \ldots \). The full covariation matrix, \( \Sigma_k \), is:
\[ \Sigma_k(t, s) = \mathbf{F}(\tau_k, t) \Sigma_k(\tau_k, s) \mathbf{F}^\top(\tau_k, t) ds \]
where \( \mathbf{F}(t) = \mathbf{F}(t) \exp \left( \left\{ t(i\omega_k - \mathbf{b}_g) \right\} \times \right) \). Since \( \mathbb{E} \left[ i\mathbf{x}(t) \right] \) remains zero over \([0, \tau_k)\), the covariance of \( i\mathbf{x}(\tau_k) \) is:
\[ \Sigma_{k+1}^p = \mathbb{E} \left[ i\mathbf{x}(\tau_k) i\mathbf{x}(\tau_k) \right]^T \]
\[ = \Phi(\tau_k, 0) \Sigma_k \Phi(\tau_k, 0)^T + \int_0^{\tau_k} \Phi(\tau, s)\Sigma_k \Phi(\tau, s)^T ds \]
where \( \Phi(t, s) \) is the transition matrix\(^5\) of (27).

**Proposition 5.** The LTV SDE in (27) has a closed-form transition matrix:
\[ \Phi(t, 0) \]
\[ = \begin{bmatrix} \exp(\mathbf{t}\mathbf{I}_3) & 0 & 0 & 0 \\ 0 & \mathbf{I}_3 & 0 & 0 \\ 0 & 0 & \mathbf{I}_3 & 0 \\ 0 & 0 & 0 & \mathbf{I}_3 \end{bmatrix} \]
where \( \mathbf{w} = i\omega_k - \mathbf{b}_g, \mathbf{a} = i\mathbf{a}_k - \mathbf{b}_a, \) and the blocks are:
\[ \mathbf{F}_\mathbf{v}(t) = -t \mathbf{F}_\mathbf{p}(t) \mathbf{J}_L (t \omega) \mathbf{a}_x \]
\[ \mathbf{F}_\mathbf{p}(t) = \begin{bmatrix} \exp(-t\mathbf{I}_3) & \mathbf{I}_3 & -t \mathbf{I}_3 & \mathbf{I}_3 \\ -t \mathbf{I}_3 & \mathbf{I}_3 & -t \mathbf{I}_3 & \mathbf{I}_3 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \]
\[ \Phi_{\mathbf{v}}(t) = -t \mathbf{F}_\mathbf{p}(t) \mathbf{J}_L (t \omega) \mathbf{a}_x \]
\[ \Phi_{\mathbf{p}}(t) = -t^2 \mathbf{F}_\mathbf{p}(t) \mathbf{J}_L (t \omega) \mathbf{a}_x \]
\[ \Phi_{\mathbf{a}}(t) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \mathbf{I}_3 & 0 & 0 \\ 0 & 0 & \mathbf{I}_3 & 0 \\ 0 & 0 & 0 & \mathbf{I}_3 \end{bmatrix} \]
where \( \Delta(t) = \exp \left( t(i\omega_x) \right) (\mathbf{I}_3 - t(i\omega_x)) - \mathbf{I}_3 \).

**Proof.** See Sec. IX-D.

To compute the predicted covariance \( \Sigma_{k+1}^p \), we need to integrate the error dynamics in (24). The IMU error state \( i\mathbf{x} \) satisfies a linear time-variant (LTV) stochastic differential equation (SDE) for \( t \in [0, \tau_k) \):
\[ i\dot{\mathbf{x}} = \mathbf{F}(t) i\mathbf{x} + i\mathbf{n}, \quad i\mathbf{x}(0) \sim \mathcal{N}(0, \mathbf{I} \Sigma_k) \tag{27} \]
where \( \mathbf{I} \Sigma_k \) is the top-left 15 \( \times \) 15 block of \( \Sigma_k \) corresponding to the IMU state. \( i\mathbf{n} \) is white Gaussian noise with power spectral density \( Q = \text{diag} (\sigma_1^2, \sigma_1^2, \sigma_0^2, \sigma_0^2, \sigma_2^2, \sigma_2^2, \sigma_3, \sigma_3) \) and \( \mathbf{F}(t) \) is:
\[ \mathbf{F}(t) = \begin{bmatrix} - (i\omega_k - \mathbf{b}_g, k) \times & 0 & 0 & - \mathbf{I}_3 & 0 \\ - i\dot{\mathbf{R}}(t) (i\mathbf{a}_k - \mathbf{b}_a, k) \times & 0 & 0 & - i\dot{\mathbf{R}}(t) & 0 \\ 0 & \mathbf{I}_3 & 0 & 0 & 0 \\ 0 & 0 & \mathbf{I}_3 & 0 & 0 \\ 0 & 0 & 0 & \mathbf{I}_3 & 0 \end{bmatrix} \]
\[ \mathbf{F}(t) = \mathbf{F}(t) \exp \left( \left\{ t(i\omega_k - \mathbf{b}_g) \right\} \times \right) \].

\(^5\)Note that \( \dot{\mathbf{R}}(t) \) in \( \mathbf{F}(t) \) is time varying. Time-invariant approximations of the transition matrix are presented in [56, App. B, E] using right-perturbation error dynamics and in [72] using left-perturbation error dynamics.

**B. Update Step**

We perform an update to \( \mathbf{x}_{k+1}^p \) and \( \Sigma_{k+1}^p \) without storing the geometric landmarks \( \tilde{\ell}_m \) or object instances \( \tilde{i}_o \) in the filter state using the null-space projection idea of [5]. Let \( \tilde{y}_i \) denote an estimate (from Sec. V) of a geometric landmark or object instance whose track gets lost at time \( t_{k+1} \). The geometric and
semantic error functions, linearized using perturbations $C \xi_k^p$,  
\[ e_{k,i} \approx \hat{e}_{k,i} + \frac{\partial \hat{e}_{k,i}}{\partial \xi_k} \xi_k^p \]  
where $n_{k,i}$ is the associated noise term with covariance $\Sigma_{n_{k,i}}$.  
Stacking the errors for all camera poses in $x_{k+1}$ associated with $i$, leads to:  
\[ e_i \approx \hat{e}_i + \frac{\partial \hat{e}_i}{\partial \hat{y}_i} x_{k+1} + \hat{e}_i \hat{y}_i + n_i. \]  
The perturbations $\hat{y}_i$ can be eliminated by left-multiplication of the errors in (32) with unitary matrices $N_i$ whose columns form the basis of the left nullspaces of $\frac{\partial \hat{y}_i}{\partial \hat{y}_i}$:  
\[ N_i^T e_i \approx N_i^T \hat{e}_i + N_i^T \frac{\partial \hat{e}_i}{\partial \hat{y}_i} x_{k+1} + N_i^T n_i. \]  
In Sec. IX-F, we also introduce a zero-velocity residual to account for complete stops of the system, which is common for autonomous ground and some aerial robots.

Finally, let $\hat{e}$, $\hat{J}$, $\Sigma$ 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:  
\[ K = \Sigma_{k+1}^p J^T (J \Sigma_{k+1}^p J^T + \Sigma)^{-1} \]  
\[ \hat{x}_{k+1} = \hat{x}_{k+1}^p \oplus (-K \hat{e}) \]  
\[ \Sigma_{k+1} = (I - KJ) \Sigma_{k+1}^p (I - KJ)^T + KV^K. \]

Note that the dimension of $J$ can be reduced in the computation above via QR factorization as described in [5].

VII. EVALUATION

This section presents results from large-scale evaluation of OrcVIO using indoor and outdoor simulated and real data. We present experiments in photo-realistic Unity simulation (Fig. 4), on raw and odometry data sequences from the KITTI dataset [87], and on real data collected on UCSD’s campus.

A. Metrics

We use two standard metrics for quantitative VIO evaluation: position Root Mean Square Error (RMSE) [88] (also referred to as position ATE [89]) and KITTI’s translation error (TE) metric [87]. Let $\text{Trans}(\mathbf{T})$ return the position component of a pose $\mathbf{T} \in SE(3)$. Let $\mathbf{fT}_k$ be the ground-truth pose trajectory and $\mathbf{fT}'_k$ be the estimated pose trajectory. To measure error, the estimated trajectory is first aligned to the initial frame of the ground-truth trajectory via $\mathbf{fT}_k = \mathbf{fT}_0 \mathbf{T}_0^{-1} \mathbf{fT}'_k$. After alignment, RMSE (m) and TE (%) are measured as:

\[ \text{RMSE} \triangleq \left( \frac{1}{K} \sum_{k=0}^{K-1} \| \text{Trans} \left( \mathbf{fT}_k \mathbf{T}_k^{-1} \mathbf{T}_k \right) \|_2 \right)^{1/2}, \]  
\[ \text{TE} \triangleq \frac{1}{|F|} \sum_{(i,j) \in F} \frac{\| \text{Trans} \left( \mathbf{fT}_j^{-1} \mathbf{T}_j \right)^{-1} \left( \mathbf{fT}_j^{-1} \mathbf{T}_j \right) \|_2}{\text{length}(i,j)}, \]

where $F$ is a set of frames with fixed distances $\text{length}(i,j)$ over the values $\{100, ..., 800\} m$ [87].

The object estimates are evaluated using 3D Intersection over Union (IoU). A 3D bounding box $\mathbf{b}_i$ is obtained from each estimated object $\hat{b}_i$, and IoU is defined as the ratio of the intersection volume over the union volume with respect to the bounding box $b_i$ of the closest ground truth object:

\[ \text{IoU}(\hat{b}_i, b_i) \triangleq \frac{\text{Volume of Intersection}(\hat{b}_i, b_i)}{\text{Volume of Union}(\hat{b}_i, b_i)}. \]  

To understand the distribution of the object orientation and translation errors, we define an estimate as true positive if the
closest ground-truth object pose is within a specific rotation or translation error threshold. Specifically, a rotation error of \(\alpha^o\) means \(\|\log(\hat{\Omega}^T \Omega)^\top\|_2 \leq \alpha^o\), and translation error of \(\beta m\) means \(\|\hat{\Omega} \hat{p} - \Omega p\|_2 \leq \beta m\). We define precision as the fraction of true positives over all estimated objects and recall is the fraction of true positives over all ground-truth objects.

B. Unity Dataset Results

OrcVIO is evaluated in a Unity simulation containing 50 car, road barrier, and door object instances, shown in Fig. 4. A ROS bridge between Unity and Gazebo is used to simulate a quadrotor robot, navigating in the environment and providing IMU and camera measurements. The object map reconstructed by OrcVIO is shown in Fig. 5. Additional qualitative results and a video demo are available in the Supplementary Material. The estimated objects are generally quite close to the ground-truth ones. The object poses near the starting position are less accurate due to insufficient motion parallax, since the quadrotor performs a pure rotation in the beginning. The trajectory RMSE and TE are 0.97 m and 0.40%, respectively. The odometry drift is mainly due to pure rotation maneuvers at planned path corners executed by the quadrotor controller.

The 3D IoU of the object estimates is 0.49. The Precision and Recall of the object reconstruction is shown in Table I. Despite that doors and barriers have a thin structure, causing even small pose estimation drift to reduce the overlap with the ground-truth object instances, OrcVIO is able to produce an accurate object map with good 3D IoU.

C. KITTI Dataset Results

We also evaluate OrcVIO on the KITTI dataset [87], 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 the original inertial data from KITTI is not reliable, we use a VIO simulator [92] to generate realistic noisy high-frequency IMU measurements at 250 Hz from the ground-truth poses.

Fig. 6 shows the IMU-camera trajectory and object states estimated on KITTI odometry sequence 07, and a video demo is provided in the Supplementary Material. The results demonstrate that OrcVIO produces meaningful object maps. We obtained ground-truth 3D annotations from the KITTI tracklets and the KITTI detection benchmark labels for quantiative evaluation of the object reconstruction. Table II reports 3D IoU results comparing OrcVIO against state-of-the-art methods, including a deep learning approach for single-view 3-D bounding-box estimation (SingleView [90]), and a multi-view bundle-adjustment algorithm that uses cuboids to represent objects (CubeSLAM [61]). OrcVIO has better 3D IoU than SingleView for the majority of the sequences (23, 36, 39, 64, 96) because, unlike SingleView, OrcVIO use multi-view optimization over the object states. The performances of OrcVIO and CubeSLAM are similar since both rely on point and bounding-box measurements to optimize the object states. CubeSLAM is better than OrcVIO in terms of 3D IoU, possibly because OrcVIO does not model dynamic objects. Table II also shows that OrcVIO is slightly better than CubeSLAM according to the TE odometry metric. In contrast with CubeSLAM, OrcVIO uses inertial residuals in addition to the geometric and object residuals.

In Table III, 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 [91]), and a visual-inertial object detector (VIS-FNL [52]). The 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 retrieves a reasonable amount of the ground-truth objects and provides a high-quality object map. When both rotation and translation errors are considered (first six rows), OrcVIO is better than SubCNN, since the latter does not rely on temporal object tracking. OrcVIO is comparable with VIS-FNL, even though VIS-FNL uses multiple object hypotheses while OrcVIO only keeps one object state. OrcVIO outperforms SubCNN and VIS-FNL.
when only translation error is taken into account (last three rows), which suggests that the object position estimates are accurate but the orientation estimates could be improved.

We evaluate the RMSE of the IMU-camera trajectory estimation in Table IV. OrcVIO is compared with two visual object SLAM methods: a monocular visual SLAM that integrates spherical object models to estimate the scale via bundle-adjustment (Object BA [58]), and CubeSLAM [61]. Table IV shows that OrcVIO outperforms Object BA because spheres are a very coarse shape representation, compared to our ellipsoid and semantic keypoint representation, and thus Object BA cannot maintain the object scales as accurately as OrcVIO. The use of inertial data in OrcVIO leads to superior results as well. OrcVIO also outperforms CubeSLAM, possibly due to the incorporation of a zero-velocity update (Sec. IX-F), which is critical in driving datasets with frequent stops.

### D. UCSD Campus Dataset Results

We also evaluated OrcVIO using real data collected with two commercial VIO sensors on UCSD’s campus. The results are qualitative due to the lack of ground truth information.

First, we used Intel RealSense D435i with image frequency of 30 Hz, image resolution of $640 \times 480$, IMU frequency of 200 Hz in an indoor lab scene with chairs and monitors as shown in Fig. 7. The estimated sensor trajectory and reconstructed object map by OrcVIO are shown in the figure. A video demo can be found in the Supplementary Material. The results demonstrate that OrcVIO can map object instances from different categories and operates at real-time speed in a cluttered indoor scene. Since OrcVIO does not currently have a loop-closing mechanism for object re-identification, objects reappearing after getting lost will be mapped twice. Thus, there are more reconstructed chairs in Fig. 7 than in the reality.

Semantic keypoint detection is challenging due to occlusion, viewpoint change, and the lack of distinctive features on the monitors as shown in Fig. 7 (b). To handle the monitor class successfully, we decreased the weight of the semantic keypoint residual in the object Levenberg-Marquardt optimization (17). Although removing the reliance on semantic keypoints leads to worse object orientation estimation, it allows OrcVIO to work with bounding-box detections only. This simplifies the front-end to an object detector and tracker and makes the algorithm more efficient and easier to deploy on resource-constrained...
robots. We release code for both the full OrcVIO algorithm and the OrcVIO-Lite version in the Supplementary Material.

Finally, we used an INDEMIND Binocular Visual-Inertial Camera to run OrcVIO outdoors with images at 25 Hz with resolution of $640 \times 400$ and IMU measurements at 200 Hz. The sensor initially observes bikes and chairs, and then makes a transition into a parking structure, as shown in Fig. 8 (b), (c). A video demonstrating the performance of OrcVIO on this dataset can be found in the Supplementary Material. The resulting object map is shown in Fig. 8 (a), demonstrating that OrcVIO is able to estimate object states from different categories in both indoor and outdoor scenes. This experiment also shows that OrcVIO can handle large illumination changes, transitioning from direct outdoor sunlight to dim lighting inside the parking structure.

VIII. CONCLUSION

This paper presented a unified formulation of ego-motion and object pose and shape estimation and developed a real-time simultaneous localization and object mapping algorithm. OrcVIO provides computationally constrained robot platforms with the ability to understand their surroundings at geometric and semantic levels, which may enable further capabilities such as object-level loop closure in challenging environments, collaborative semantic SLAM, and object interaction. Estimating object motion and performing collaborative mapping and loop closure are promising directions for future research.

IX. APPENDIX

A. Proof of Proposition 1

We first derive the Jacobian of the camera pose perturbation $c\xi = [cR^\top, c\theta^\top]^\top$ with respect to the IMU pose perturbation $i\xi = [i\theta^\top, i\tilde{p}^\top]^\top$ in (10). The camera and IMU poses are related via $cT = I^T_C T$. Approximating $cT$ using $c\hat{T}$ and $c\xi$ leads to:

$$cT \approx cT (I_4 + c\xi_x) = \begin{bmatrix} 0 & c\tilde{p} & 0 & c\hat{R}\theta \times c\hat{R}\rho \end{bmatrix}.$$

Similarly, approximating $iT$ using $i\hat{T}$ and $i\xi$ leads to:

$$i\hat{T}^T T \approx \begin{bmatrix} 0 & i\tilde{p} & 0 & i\hat{R}\theta \times i\hat{R}\rho \end{bmatrix}.$$

Equating the two expression above, we get:

$$c\hat{R}\theta \times = i\hat{R}\theta \times I_c, c\hat{R}\rho = i\hat{R}\rho,$$

and (10) can be concluded from:

$$c\theta = I_c^T c\hat{R}^T i\theta, c\rho = -I_c^T c\hat{R}^T i\rho.$$

Next, we derive the expressions in (9). Note that:

$$cT^{-1} \xi \approx (I_4 + c\xi_x)^{-1} cT^{-1} \xi  = (I_4 - c\xi_x)cT^{-1} \xi$$

$$= c\hat{T}^{-1} \xi - [c\hat{T}^{-1} \xi]^\top c\xi,$$

where $(I_4 + \xi_x)(I_4 - \xi_x) = I_4 - \xi_x^2 \approx I_4$ by discarding high-order terms in $\xi$, and we used (7.159) in [73, Ch.7] in the last equality. Applying the chain rule to (8):

$$\frac{\partial \xi}{\partial p} = F \frac{d\pi}{ds} (cT^{-1} \xi) \frac{\partial cT^{-1} \xi}{\partial c\xi},$$

and evaluating at $c\hat{T}$ and $\hat{\xi}$, using $\frac{\partial cT^{-1} \xi}{\partial c\xi} = -\frac{1}{cT^{-1} \xi}$ from (39), leads to the first equation in (9). The second equation in (9) follows directly from the chain rule:

$$\frac{\partial \xi}{\partial \ell} = F \frac{d\pi}{ds} (cT^{-1} \xi) \frac{cT^{-1} \xi}{cT^{-1} \ell},$$

$$\frac{\partial cT^{-1} \xi}{\partial \xi} = \frac{\partial \xi}{\partial \ell} = \begin{bmatrix} I_3 \\ 0 \end{bmatrix}.$$
B. Proof of Proposition 2

The derivation of \( \frac{\partial \text{e}}{\partial \text{e}} \) is identical to the derivation of \( \frac{\partial \text{e}}{\partial \text{e}} \) in (40) with \( \ell \) replaced by \( \text{O}(\text{s}_\ell + \delta \text{s}_\ell) \), which does not depend on \( \text{e} \). To derive the Jacobians of \( \text{e} \) with respect to \( \partial \text{e} \) and \( \delta \text{s}_\ell \), apply the chain rule to (11):

\[
\frac{\partial \text{e}}{\partial \text{e}} = \mathbf{P} \mathbf{d}_\text{e} \left( \text{O}^{-1}(\text{s}_\ell + \delta \text{s}_\ell) \right) \text{O}^{-1} \frac{\partial \text{O}(\text{s}_\ell + \delta \text{s}_\ell)}{\partial \text{e}} \frac{\partial \text{e}}{\partial \text{s}_\ell}.
\]

As in (41), \( \frac{\partial \text{e}}{\partial \text{e}} \approx \left[ I_\text{e} \right] \left( \text{O}^{-1}(\text{s}_\ell + \delta \text{s}_\ell) \right) \left( \text{O}^{-1}(\text{s}_\ell + \delta \text{s}_\ell) \right) \frac{\partial \text{O}(\text{s}_\ell + \delta \text{s}_\ell)}{\partial \text{e}} \frac{\partial \text{e}}{\partial \text{s}_\ell} \)\( \text{O} \).

C. Proof of Proposition 3

**Lemma 1.** Given a hyperplane \( \text{b} = [\text{b}^T - \text{b}_0]^T \) and an origin-centered ellipsoid as a dual quadric \( \text{Q}^* = \text{diag}(\text{U}^2, -1) \), there are two hyperplanes parallel to \( \text{b} \) that are tangent to the ellipsoid: \( \hat{\text{t}} = [\text{b}^T \pm \sqrt{\text{b}^T \text{U}^2 \text{b}}]^T \) and the signed distance between \( \text{b} \) and the closest tangent hyperplane is:

\[
d(\text{b}, \text{U}) = \frac{1}{2\|\text{b}\|} \left( \text{sgn}(\text{b}) \sqrt{\text{b}^T \text{U}^2 \text{b}} - \text{b}_0 \right) .
\]

**Proof.** Let the tangent parallel to \( \text{b} \) be \( \text{t} = [\text{b}^T - \alpha]^T \).

Recall that the normal to an isosurface at any point \( x \) is the gradient of the isosurface \( \nabla_x \text{x}^T \text{U}^{-2} \text{x} = 2 \text{U}^{-2} \text{x} \). For a tangent plane \( t \), the plane normal and ellipsoid normal must be in the same direction, \( \text{b} \propto 2 \text{U}^{-2} \text{x} \). Assuming \( \beta \in \mathbb{R} \) as unknown constant, we can solve for \( x = \beta \text{U}^2 \text{x} \). Because \( x \) lies on the ellipsoid \( \beta^2 \text{U}^2 \text{b} = 1 \) or \( \beta = \pm \frac{1}{\sqrt{\text{b}^T \text{U}^2 \text{b}}} \).

If \( \text{x} \) also lies on the hyperplane \( \text{t} \), then \( 0 = \text{b}^T \text{x} - \alpha = \beta \text{b}^T \text{U}^2 \text{x} - \alpha = \pm \sqrt{\text{b}^T \text{U}^2 \text{b}} \), and we get \( \alpha = \pm \frac{1}{\sqrt{\text{b}^T \text{U}^2 \text{b}}} \).

The perpendicular distance between two parallel hyperplanes \( [\text{b}^T - \alpha_1]^T \) and \( [\text{b}^T - \alpha_2]^T \) is \( \frac{\|\alpha_1 - \alpha_2\|}{\sqrt{\text{b}^T \text{U}^2 \text{b}}} \). The distance from the nearest tangent \( \text{t} \) to the hyperplane \( \text{b} \) is \( d(\text{b}, \text{U}) = \frac{1}{\|\text{b}\|} \text{min}_{\alpha} (\alpha - \text{b}_0) = \frac{1}{\|\text{b}\|} \left( \text{sgn}(\text{b}) \sqrt{\text{b}^T \text{U}^2 \text{b}} - \text{b}_0 \right) \). The signed distance is chosen such that the residual from \( \text{t} \) takes it closer to \( \text{b} \).

**Lemma 2.** Given a hyperplane \( \text{b} = [\text{b}^T - \text{b}_0]^T \) and an origin-centered ellipsoid as a dual quadric \( \text{Q}^* = \text{diag}(\text{U}^2, -1) \), the Jacobian of the distance \( d(\text{b}, \text{U}) \) in (44) with respect to \( \text{b} \) is:

\[
\frac{\partial d(\text{b}, \text{U})}{\partial \text{b}} = \frac{\text{sgn}(\text{b}) \text{b}^T \text{U}^2}{\|\text{b}\| \sqrt{\text{b}^T \text{U}^2 \text{b}}} \left( \text{I}_3 - \frac{\text{b} \text{b}^T}{\|\text{b}\|^2} \right) + \frac{[\text{b} \text{b}^T \|\text{b}\|^2]}{\|\text{b}\|^3}.
\]

**Proof.** We rewrite \( d(\text{b}, \text{U}) \) explicitly in terms of \( \text{b} \) by replacing \( \text{b}_0 = -[0^T 1]^T \) to get:

\[
d(\text{b}, \text{U}) = \frac{1}{\|\text{b}\|} \left( -\text{sgn}(0^T \text{b}) \sqrt{\text{b}^T \text{U}^2(\text{b} + 0^T \text{b})} \right).
\]

Finally, \( \frac{\partial d}{\partial \text{U}} \) is obtained by the chain rule:

\[
\frac{\partial \text{e}}{\partial \text{U}} = \frac{\text{sgn}(\text{b})}{2\|\text{b}\| \sqrt{\text{b}^T \text{U}^2 \text{b} + \text{b}^T \text{U}^2 \text{b}}} \frac{\partial}{\partial \text{U}} \text{b}^T \text{U}^2 \text{b} + \text{b}^T \text{U}^2 \text{b} \cdot \text{U}^2 \text{b}.
\]

D. Proof of Proposition 4

**Lemma 3.** Let \( \omega \in \mathbb{R}^3 \), \( J_L(\omega) \triangleq \sum_{n=0}^{\infty} \omega^n_{(n+1)} \), and \( H_L(\omega) \triangleq \sum_{n=0}^{\infty} \omega^n_{(n+2)} \). For \( \omega \neq 0 \), \( J_L(\omega) \) and \( H_L(\omega) \) admit closed-form expressions, shown in (26).
Proof. Using that \( \omega_{x}^{2n+1} = (-1)^n \| \omega \|^{2n} \omega_x \) for \( n \geq 0 \):

\[
J_L(\omega) = \mathbf{I}_3 + \sum_{n=1}^{\infty} \frac{\omega^n_{x}}{(n+1)!} \int_0^t \exp(-t \| \omega \|^{2n} \omega_x) dt.
\]

\[
= \mathbf{I}_3 + \sum_{n=0}^{\infty} \frac{(-1)^n \| \omega \|^{2n}}{(2n+2)!} \omega_x \int_0^t \exp(-t \| \omega \|^{2n} \omega_x) dt + \mathbf{I}_3 + \frac{1 - \cos \| \omega \|}{\| \omega \|^2} \omega_x + \frac{\| \omega \| - \sin \| \omega \|}{\| \omega \|^2} \omega_x^2.
\]

The derivation for \( H_L(\omega) \) is equivalent.

\[\square\]

Lemma 4. For \( \omega \in \mathbb{R}^3 \), \( t \in \mathbb{R} \), the matrix \( \exp(\tau \omega \times) \) satisfies:

\[
\int_0^t \int_0^s \exp(r \omega \times) dr ds = \int_0^t s \mathbf{J}_L(s \omega \times) ds = t^2 \mathbf{H}_L(t \omega).
\]

Proof. The result follows by integrating the terms of the Taylor series of \( \exp(t \omega \times) = \sum_{n=0}^{\infty} \frac{(t \omega \times)^n}{n!} \) and since \( J_L(\omega) \) and \( H_L(\omega) \) are well defined by Lemma 3.

To obtain (25), we compute the solutions to the linear time-invariant (LTI) ordinary differential equations (ODEs) in (23). With \( \dot{\omega} = \dot{\omega}_k - \mathbf{b}_g, k \) and \( t \in [0, \tau_k] \), the solution to:

\[
\dot{\mathbf{R}} = \mathbf{J}_L(\dot{\omega}_k), \quad \dot{\mathbf{R}}(0) = \mathbf{I}_3,
\]

is \( \dot{\mathbf{R}}(t) = \mathbf{J}_L(\mathbf{R}) \exp(\tau \omega \times) \) and, hence:

\[
\mathbf{R}_{k+1} = \mathbf{J}_L(\tau_k) \mathbf{R}_k \exp(\tau_{k} (\omega_k - \mathbf{b}_g, k \times)).
\]

Similarly, with \( \mathbf{a} = \mathbf{a}_k - \mathbf{b}_n, k \in [0, \tau_k] \), and initial condition \( \dot{\mathbf{v}}(0) = \mathbf{I}_3 \), the solution of the LTI ODE for \( \dot{\mathbf{v}} \) in (23) is:

\[
\dot{\mathbf{v}}(t) = \mathbf{I}_3 + \int_0^t \mathbf{J}_L(s \omega \times) \mathbf{R} \mathbf{a} + t \mathbf{g} \times 
\]

where the second equality uses Lemma 4. Hence, \( \dot{\mathbf{v}}_{k+1} = \dot{\mathbf{v}}(\tau_k) \) satisfies the second expression in (25). Also, by Lemma 4, for \( t \in [0, \tau_k] \) with initial condition \( \dot{\mathbf{p}}(0) = \mathbf{I}_3 \), the solution of the LTI ODE for \( \dot{\mathbf{p}} \) in (23) is:

\[
\dot{\mathbf{p}}(t) = \mathbf{I}_3 + \mathbf{J}_L(\mathbf{R}) \mathbf{a} + t \mathbf{g} \times
\]

where the second equality uses Lemma 4. Hence, \( \dot{\mathbf{p}}_{k+1} = \dot{\mathbf{p}}(\tau_k) \) satisfies the third expression in (25). Finally, \( \mathbf{b}_g(t) = \mathbf{b}_g(0) = \mathbf{b}_g, k \) and \( \mathbf{b}_n(t) = \mathbf{b}_n(0) = \mathbf{b}_n, k \) for all \( t \in [0, \tau_k] \). The IMU pose history in (25) is updated by adding the pose \( (\mathbf{R}_k \mathbf{p}_k) \) to the sliding window and dropping the oldest pose \( T_{k-W} \).

E. Proof of Proposition 5

The transition matrix \( \mathbf{F}(t, 0) \) of (27) can be determined by computing the solution \( \dot{x}(t) = \mathbf{F}(t, 0) \dot{x}(0) \) to the homogeneous system \( \dot{x} = \mathbf{F}(t) \dot{x} \) for an arbitrary initial condition \( \dot{x}(0) = (\mathbf{I}(\dot{\theta}(0)), \mathbf{I}(\dot{\mathbf{v}}(0)), \mathbf{I}(\dot{\mathbf{p}}(0)), \mathbf{b}_g(0), \mathbf{b}_n(0)) \). Since the last two rows of \( \mathbf{F}(t) \) are zero, the bias terms remain constant in the homogeneous system:

\[
\mathbf{b}_g(t) = \mathbf{b}_g(0) = [0 \quad 0 \quad 0 \quad 0 \quad \mathbf{I}_3 \quad 0 \times \mathbf{x}(0), \mathbf{b}_n(t) = \mathbf{b}_n(0) = [0 \quad 0 \quad 0 \quad 0 \quad \mathbf{I}_3 \times \mathbf{x}(0).
\]

Next, consider \( \dot{\theta}(t) = -\omega_x \mathbf{J}_L(\dot{\omega}) - \mathbf{b}_g(t) \) with \( \omega = \dot{\omega}_k - \mathbf{b}_g, k \), which is a linear time-invariant (LTI) system in \( \dot{\theta}(t) \).

Using \( \mathbf{b}_g(t) = \mathbf{b}_g(0) \) and Lemma 4, the system has solution:

\[
\dot{\theta}(t) = \exp(-t \omega_x \mathbf{J}_L) \mathbf{I}_3 - \int_0^t \exp(-s \omega_x \mathbf{J}_L) \mathbf{b}_g(s) ds = \exp(-t \omega_x \mathbf{J}_L) \mathbf{I}_3 - \int_0^t \exp(-s \omega_x \mathbf{J}_L) \mathbf{b}_g(s) ds.
\]

Next, consider \( \dot{\mathbf{v}}(t) = \mathbf{I}_3 \mathbf{R} \mathbf{a}_k - \mathbf{b}_n \mathbf{J}_L \mathbf{a}_k \mathbf{J}_L - \mathbf{b}_n(0) \) with \( \mathbf{a} = \mathbf{a}_k - \mathbf{b}_n, k \), which is an LTI system in \( \dot{\mathbf{v}}(t) \).

Using \( \mathbf{b}_n(t) = \mathbf{b}_n(0) \), the LTI system has solution:

\[
\dot{\mathbf{v}}(t) = \mathbf{I}_3 \dot{\mathbf{R}}(s) \mathbf{a}_k - \mathbf{b}_n(0) \mathbf{J}_L \mathbf{a}_k \mathbf{J}_L - \mathbf{b}_n(0) = \mathbf{I}_3 \dot{\mathbf{R}}(s) \mathbf{a}_k - \mathbf{b}_n(0) \mathbf{J}_L \mathbf{a}_k \mathbf{J}_L - \mathbf{b}_n(0) = \mathbf{I}_3 \dot{\mathbf{R}}(s) \mathbf{a}_k - \mathbf{b}_n(0) \mathbf{J}_L \mathbf{a}_k \mathbf{J}_L - \mathbf{b}_n(0).
\]
Finally, consider $\dot{\tilde{\Phi}}(t) = \dot{\tilde{\Phi}}(0) + \int_0^t \dot{\tilde{\Phi}}(s) \, ds$

$= [\Phi_{p\theta}(t) \quad tI_3 \quad I_3 \quad \Phi_{p\omega}(t) \quad \Phi_{pa}(t)] \dot{x}(t), \quad (62)$

where:

\[
\Phi_{p\theta}(t) = \int_0^t \Phi_{\theta\theta}(s) \, ds = -t^2 \bar{\mathbf{R}}_k \mathbf{H}_L(t) \omega \mathbf{a} \| \omega \|^2, \quad (63)
\]

\[
\Phi_{p\omega}(t) = \int_0^t \Phi_{\omega\omega}(s) \, ds, \quad (64)
\]

\[
\Phi_{pa}(t) = \int_0^t \Phi_{va}(s) \, ds = -t^2 \bar{\mathbf{R}}_k \mathbf{H}_L(t) \omega, \quad (65)
\]

where (63) and (65) follow from Lemma 4. To integrate (64), we use that:

\[
\int_0^t \Delta(s) \, ds = \int_0^t \exp(s \omega \times) \, ds \omega \times - tI_3 = t \mathbf{J}_L(\omega) - \frac{\omega \times \Delta(t)}{||\omega||^2} - tI_3, \quad (66)
\]

where in the second integral we used (60) and that $\omega \times$ and $\Delta(t)$ commute. We integrate the second and third term in (61) using Lemma 4 to obtain the final result for $\Phi_{p\omega}(t)$ in (29). Since $\dot{x}(0)$ was arbitrary, the rows of $\Phi(t, 0)$ are provided by (52), (53), (62), and (51).

### F. Zero-Velocity Update

Zero-velocity conditions are frequently encountered in autonomous driving and autonomous flight applications, and thus the ability to determine whether the robot is static is important for reducing drift in VIO estimation [92], [93]. In EncVIO zero-velocity is detected similarly as in [92] by using pseudo inertial measurements and then checking the velocity magnitude. For the zero-velocity update, we need to compute the residuals and the Jacobians of the inertial measurements with respect to the state. Based on (22) the residuals are:

\[
e(z | x_i, z) \triangleq \begin{bmatrix} (\omega - \mathbf{b}_g) - 0 \\ (\mathbf{r} \times (\mathbf{a} - \mathbf{b}_a) + \mathbf{g}) - 0 \end{bmatrix}. \quad (67)
\]

The corresponding Jacobians are presented as follows:

\[
\begin{aligned}
\frac{\partial^2 e}{\partial \theta} &= [0 - \mathbf{r} \times (\mathbf{a} - \mathbf{b}_a)]^T, \\
\frac{\partial^2 e}{\partial b_g} &= [-I_3 \quad 0]^T, \\
\frac{\partial^2 e}{\partial b_a} &= [0 - \mathbf{r} \times]^T. 
\end{aligned} \quad (68)
\]

### REFERENCES

[1] S. Agarwal, N. Snavely, I. Simon, S. Seitz, and R. Szeliski, “Building Rome in a Day,” in *IEEE International Conference on Computer Vision (ICCV)*, 2009.

[2] J. L. Schonberger and J.-M. Frahm, “Structure-from-motion revisited,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 4104–4113.

[3] 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, 2016.

[4] 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.

[5] A. I. Mourikis and S. I. Roumeliotis, “A multi-state constraint kalman filter for vision-aided inertial navigation,” in *IEEE International Conference on Robotics and Automation (ICRA)*, Apr 2007, pp. 3565–3572.

[6] 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.

[7] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, “Yolov4: Optimal speed and accuracy of object detection,” arXiv preprint: 2004.10934, 2020.

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

[9] N. Wojke and A. Bewley, “Deep cosine metric learning for person re-identification,” in *IEEE Winter Workshop on Applications of Computer Vision (WACV)*, 2018, pp. 748–756.

[10] M. Lorbach, S. Höfer, and O. Brock, “Prior-assisted propagation of spatial information for object search,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2014, pp. 2904–2909.

[11] C. Sahin, G. Garcia-Hernando, J. Sock, and T.-K. Kim, “A review on object pose recovery: from 3d bounding box detectors to full 6d pose estimators,” *Image and Vision Computing*, p. 103898, 2020.

[12] E. Vitinopoulos, G. Pavlakos, S. Bowman, J. D. Caporale, K. Daniilidis, G. J. Pappas, and D. Kolditschek, “Reactive semantic planning in unexplored_semantic environments using deep perceptual feedback,” *IEEE Robotics and Automation Letters*, 2020.

[13] S. Garg, N. Sünderrauf, F. Dayoub, D. Morrison, A. Cosgun, G. Carneiro, Q. Wu, T.-J. Chin, I. Reid, S. Gould et al., “Semantics for Robotic Mapping, Perception and Interaction: A Survey,” *Foundations and Trends in Robotics*, vol. 8, no. 1–2, pp. 1–224, 2020.

[14] J. Civera, A. J. Davison, and J. M. M. Montiel, “Inverse depth parametrization for monocular slam,” *IEEE Transactions on Robotics*, vol. 24, no. 5, pp. 932–945, Oct 2008.

[15] J. Civera, O. G. Grasa, A. J. Davison, and J. M. M. Montiel, “1-Point RANSAC for extended Kalman filtering: Application to real-time structure from motion and visual odometry,” *Journal of Field Robotics*, vol. 27, no. 5, pp. 609–631, 2010.

[16] C. Forster, Z. Zhang, M. Gassner, M. Werlberger, and D. Scaramuzza, “SVO: Semidirect Visual Odometry for Monocular and Multicamera Systems,” *IEEE Transactions on Robotics*, vol. 33, no. 2, pp. 249–265, April 2017.

[17] A. Howard, “Real-time stereo visual odometry for autonomous ground vehicles,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Sep 2008, pp. 3946–3952.

[18] K. Sun, K. Mohta, B. Pfrommer, M. Watters, S. Liu, Y. Mulgaonkar, C. J. Taylor, and V. Kumar, “Robust stereo visual inertial odometry for fast autonomous flight,” *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 965–972, April 2018.

[19] R. A. Newcombe, S. J. Lovegrove, and A. J. Davison, “Otam: Dense tracking and mapping in real-time,” in *International Conference on Computer Vision (ICCV)*, Nov 2011, pp. 2320–2327.

[20] J. Engel, T. Schöps, and D. Cremers, “LSD-SLAM: Large-Scale Direct Monocular SLAM,” in *European Conference on Computer Vision (ECCV)*, 2014, pp. 834–849.

[21] X. Gao, R. Wang, N. Demmel, and D. Cremers, “Lsdso: Direct sparse odometry with loop closure,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct 2018, pp. 2198–2204.

[22] S. Weiss and R. Siegwart, “Real-time metric state estimation for modular vision-inertial systems,” in *IEEE International Conference on Robotics and Automation (ICRA)*, 2011, pp. 4531–4537.

[23] K. Wu, A. Ahmed, G. A. Georgiou, and S. I. Roumeliotis, “A square root inverse filter for efficient vision-aided inertial navigation on mobile devices,” in *Robotics: Science and Systems XI*, 2015.

[24] S. Leutenegger, S. Lynen, M. Bose, R. Siegwart, and P. Furgale, “Keyframe-based visual-inertial odometry using nonlinear optimization,” *The International Journal of Robotics Research*, vol. 34, no. 3, pp. 314–334, 2015.

[25] 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.

[26] N. Tian, M. Brown, and D. G. Lowe, “Unsupervised learning of depth and ego-motion from video,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017, pp. 6612–6619.
Y. Xiang, W. Choi, Y. Lin, and S. Savarese, “Subcategory-aware models that unify geometry and semantics and on optimal control and reinforcement learning approaches for minimizing uncertainty in these models. Dr. Atanasov’s work has been recognized by the Joseph and Rosaline Wolf award for the best Ph.D. dissertation in Electrical and Systems Engineering at the University of Pennsylvania in 2015 and the best conference paper award at the IEEE International Conference on Robotics and Automation in 2017.

Mo Shan (S’20) is a PhD student of Electrical and Computer Engineering at the University of California San Diego. He obtained a B.S. degree in Electrical Engineering from National University of Singapore, Singapore, in 2014 and M.S. degree in Electrical Engineering from the University of California San Diego. His research focuses on semantic SLAM and visual inertial odometry.

Vikas Dhiman (M’15) is a Postdoctoral Researcher at the Contextual Robotics Institute and the Existentiaal Robotics Lab at the University of California, San Diego. His work focuses on localization, mapping, and control algorithms for robotics applications. He graduated in Electrical Engineering (2008) from Indian Institute of Technology, Roorkee, earned an M.S. (2014) from the University at Buffalo, and received a Ph.D. (2019) from the University of Michigan, Ann Arbor.

Qiaojun Feng (S’19) obtained a B.Eng. in Automation from Tsinghua University, Beijing, China, in 2017 and an M.S. in Electrical Engineering from the University of California San Diego, La Jolla, CA, in 2019. He is currently a Ph.D. student of Electrical and Computer Engineering at the University of California San Diego. His research focuses on robotics and autonomy, especially SLAM. He works on environment perception and map reconstruction, combining geometric and semantic information.

Jinzhuo Li obtained a B.Eng. in Automation from Tsinghua University, Beijing, China, in 2018 and an M.S. in Electrical and Computer Engineering from the University of California San Diego, La Jolla, CA, in 2020. His research focuses on SLAM, visual inertial odometry, and object-level reconstruction.

Nikolay Atanasov (S’07-M’16) is an Assistant Professor of Electrical and Computer Engineering at the University of California San Diego. He obtained a B.S. degree in Electrical Engineering from Trinity College, Hartford, CT, in 2008 and M.S. and Ph.D. degrees in Electrical and Systems Engineering from the University of Pennsylvania, Philadelphia, PA, in 2012 and 2015, respectively. His research focuses on robotics, control theory, and machine learning, applied to active sensing using ground and aerial robots. He works on probabilistic environment models that unify geometry and semantics and on optimal control and reinforcement learning approaches for minimizing uncertainty in these models. Dr. Atanasov’s work has been recognized by the Joseph and Rosaline Wolf award for the best Ph.D. dissertation in Electrical and Systems Engineering at the University of Pennsylvania in 2015 and the best conference paper award at the IEEE International Conference on Robotics and Automation in 2017.