Visual-Inertial Odometry and Object Mapping with Structural Constraints

Mo Shan and Nikolay Atanasov

Department of Electrical and Computer Engineering

SLAM

- Simultaneous Localization And Mapping (SLAM): a model of the environment (the map), and the estimation of the state of the robot moving within it (C. Cadena et al., 2016).

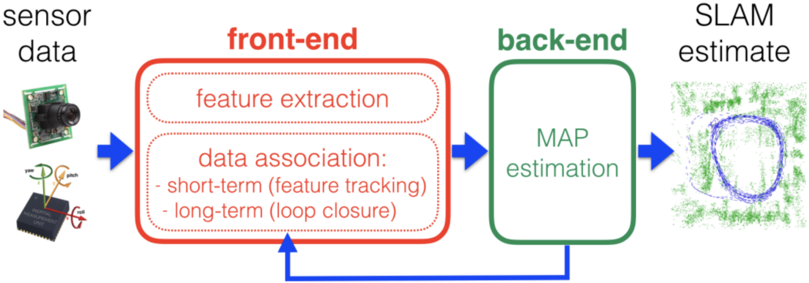


Figure: SLAM framework.

Factor graph

- SLAM as a factor graph

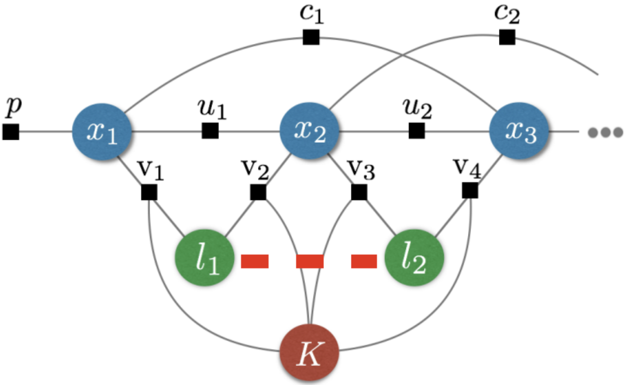


Figure: Factor graph. Blue circles: robot poses, green circles: landmark positions, red circle: variable of intrinsic parameters (K). u: odometry constraints, v: camera observations, c: loop closures, p: prior factors.

Motivation

Object-level semantics are important for

- improving performance of feature tracking
- reducing drift via loop closure
- obtaining compressed maps of objects for subsequent tasks

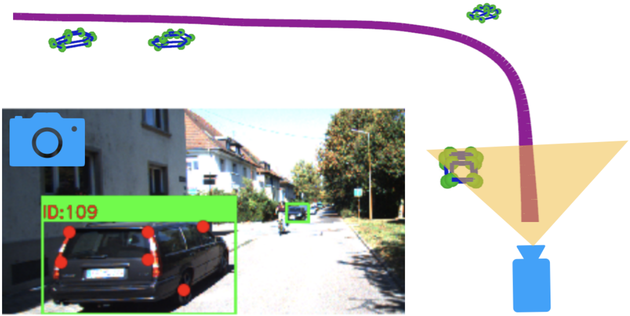
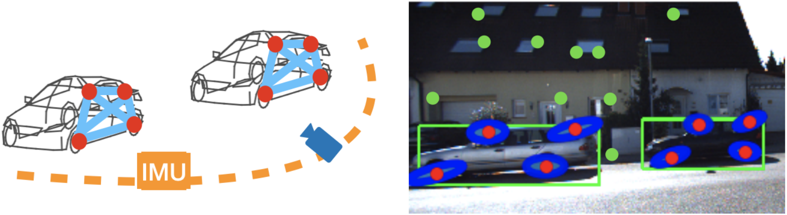


Figure: An object map.

Objective

A robot equipped with an IMU and RGB camera, localize the robot using visual-inertial odometry (VIO), and map the objects composed of semantic landmarks in the scene using:

- inertial observations: linear acceleration and angular velocity
- geometric measurements from geometric landmarks
- semantic measurements from keypoints on objects

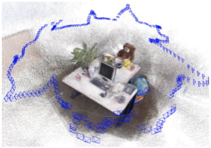


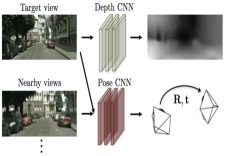
(a) Inertial data

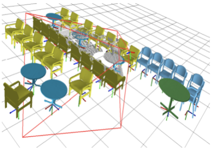
(b) Visual observation

State of the Art

- Traditional VIO, SLAM approaches such as ORB SLAM (Mur-Artal et al., 2017), DSO (J. Engel et al., 2016) rely on geometric features, eg ORB, SIFT, but overlook objects
- Learning-based approaches that use convolutional neural networks (CNNs) only regress camera pose but do not produce meaningful maps
- Initial attempts on object-level SLAM often use iterative optimization as well as complicated object CAD models







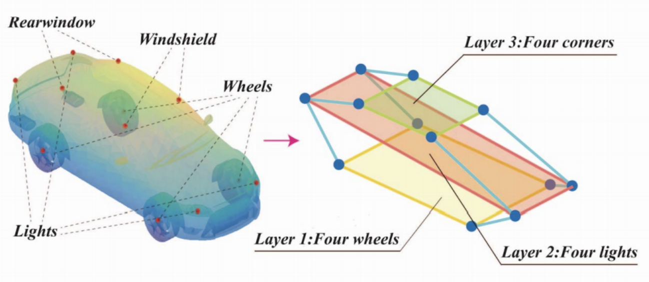
Contribution

We exploits the object semantics to

- obtain uncertainty estimates for the semantic feature locations
- achieve probabilistic tracking of composite semantic features, i.e., at the object level
- exploit object structure constraints (e.g., the wheels of a car should not be very close or far away to each other) to execute an accurate estimate

Objects

- Objects in the environment $\mathcal{O} \triangleq \{(o_i, c_i)\}_{i=1}^{N_o}$
- Object of class $c_i \in \mathcal{C}_o$ defined by $N_s(c_i)$ semantic keypoints.
- There also exits the pairwise *category-specific constraint* arising from the shape prior



Problem formulation

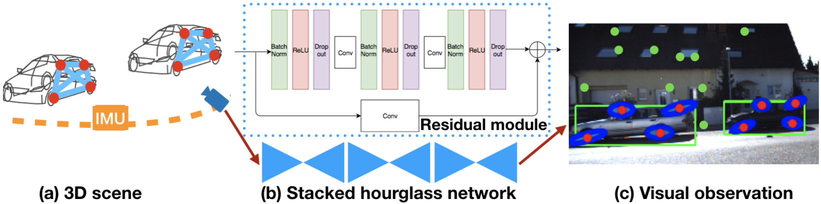
Given measurements $\{{}^{i}\mathbf{z}_t, {}^{g}\mathbf{z}_t, {}^{c}\mathbf{z}_t, {}^{s}\mathbf{z}_t, {}^{b}\mathbf{z}_t\}_{t=1}^{T}$, determine the sensor trajectory $\mathcal{X}$ and the object states $\mathcal{O}$ that maximize the measurement likelihood:

$$\max_{\mathcal{O},\mathcal{X}} \sum_{t=1}^{T} \log(p({}^{i}\mathbf{z}_t|\mathcal{X})p({}^{g}\mathbf{z}_t|\mathcal{X})p({}^{c}\mathbf{z}_t, {}^{b}\mathbf{z}_t, {}^{s}\mathbf{z}_t|\mathcal{O}, \mathcal{X})) \tag{1}$$

The likelihood terms above can be defined as Gaussian density functions. Variances are determined by the measurement noise. Means are determined by the dynamic equations of motion over the $SE(3)$ Lie group and the camera perspective model.

Front-end

- We use a stacked hourglass convolutional network to extract mid-level semantic features and their uncertainties, used for the probabilistic tracking of composite semantic features



Keypoint detection

- StarMap produces heatmap for all keypoints.
- Corresponding features as 3D locations in the canonical object view (CanViewFeature)
- Augmented with an additional depth channel (DepthMap) to lift the 2D keypoints to 3D

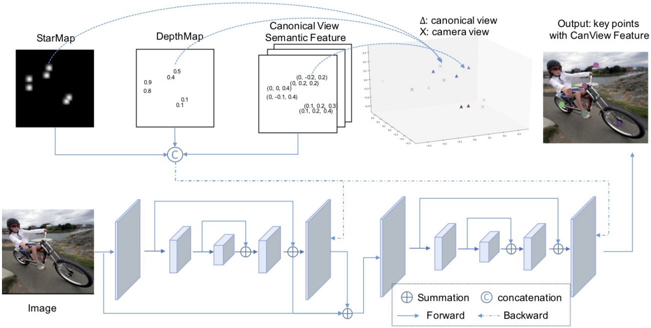


Figure: Starmap.

MC dropout

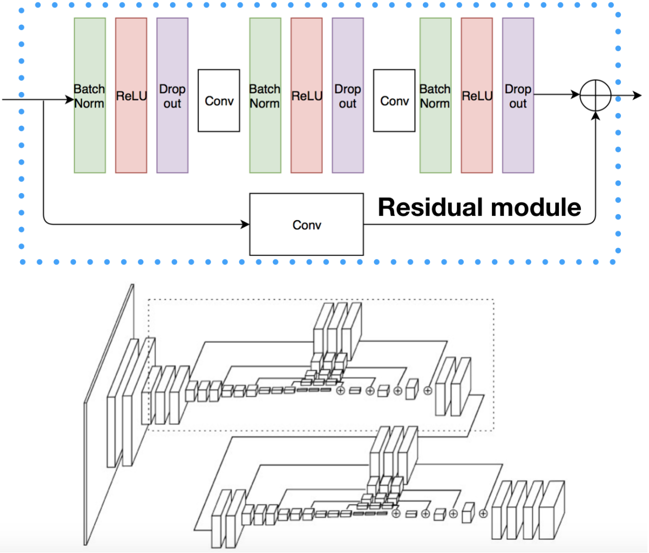


Figure: Starmap.

MC dropout

The Monte Carlo estimate is named MC dropout, and defined as in Eq. 2

$$\begin{aligned}\hat{y}_{mc} &= \frac{1}{B}\sum_{i=1}^{B}\hat{y}_i \\ \hat{\eta}_{mc} &= \frac{1}{B}\sum_{i=1}^{B}(\hat{y}_i - \hat{y})^2\end{aligned} \tag{2}$$

MC dropout approximately integrates over the models weights and can be interpreted as a Bayesian approximation of a Gaussian process (Y. Gal, 2016).

Object-level tracking

- Use Kalman Filter to fuse the detection and tracking: KanadeLucasTomasi (KLT) feature tracker for prediction and keypoint detection as update.
- The state for object i at time t is

$$\mathbf{a}_t^i = \begin{pmatrix} \mathbf{x}_t^b & \mathbf{y}_t^1 & \dots & \mathbf{y}_t^{N_{kp}} \end{pmatrix} \tag{3}$$

  where $\mathbf{x}_t^b \triangleq ({}_b x_t^1, \ {}_b \dot{x}_t^1, \ {}_b y_t^1, \ {}_b \dot{y}_t^1, \ {}_b x_t^2, \ {}_b \dot{x}_t^2, \ {}_b y_t^2, \ {}_b \dot{y}_t^2)$ contains the coordinates of the object bounding box and their velocities, and $\mathbf{y}_t^j \triangleq ({}_k x_t, \ {}_k \dot{x}_t, \ {}_k y_t, \ {}_k \dot{y}_t), \ j \in 1 \dots N_{kp}$ represents the coordinates and velocities of semantic keypoints.
- The tracker jointly tracks the bounding box and all the N_{kp} semantic keypoints on each car.

Notation

- We denote the global frame by $\{G\}$, the IMU frame by $\{I\}$, and the camera frame by $\{C\}$.
- The transformation from $\{I\}$ to $\{C\}$ is specified by a translation ${}^C_I\mathbf{p} \in \mathbb{R}^3$ and unit quaternion ${}^C_I\mathbf{q}$ using a left-handed JPL convention
- Alternatively via a transformation matrix:

$$ {}^C_I\mathcal{T} \triangleq \begin{pmatrix} {}^C_I\mathbf{R} & {}^C_I\mathbf{p} \\ \mathbf{0} & 1 \end{pmatrix} \in SE(3), \tag{4} $$

Back-end

- EKF prediction:

$$\begin{aligned}\hat{\boldsymbol{x}}_{k|k-1} &= f\left(\hat{\boldsymbol{x}}_{k-1|k-1}, \boldsymbol{u}_k\right)\\ \boldsymbol{P}_{k|k-1} &= \boldsymbol{F}_k \boldsymbol{P}_{k-1|k-1} \boldsymbol{F}_k^\top + \boldsymbol{Q}_k\end{aligned}$$

- EKF update:

$$\begin{aligned}\tilde{\boldsymbol{y}}_k &= \boldsymbol{z}_k - h\left(\hat{\boldsymbol{x}}_{k|k-1}\right)\\ \boldsymbol{S}_k &= \boldsymbol{H}_k \boldsymbol{P}_{k|k-1} \boldsymbol{H}_k^\top + \boldsymbol{R}_k\\ \boldsymbol{K}_k &= \boldsymbol{P}_{k|k-1} \boldsymbol{H}_k^\top \boldsymbol{S}_k^{-1}\\ \hat{\boldsymbol{x}}_{k|k} &= \hat{\boldsymbol{x}}_{k|k-1} + \boldsymbol{K}_k \tilde{\boldsymbol{y}}_k\\ \boldsymbol{P}_{k|k} &= \left(\boldsymbol{I} - \boldsymbol{K}_k \boldsymbol{H}_k\right) \boldsymbol{P}_{k|k-1}\end{aligned}$$

- where

$$\begin{aligned}\boldsymbol{F}_k &= \left.\frac{\partial f}{\partial \boldsymbol{x}}\right|_{\hat{\boldsymbol{x}}_{k-1|k-1}, \boldsymbol{u}_k}\\ \boldsymbol{H}_k &= \left.\frac{\partial h}{\partial \boldsymbol{x}}\right|_{\hat{\boldsymbol{x}}_{k|k-1}}\end{aligned}$$

VIO background

- The state of the IMU is defined as

$$ {}_I\mathbf{x} \triangleq \begin{pmatrix} {}_I\bar{\mathbf{q}} & \mathbf{b}_g & {}_I\mathbf{v} & \mathbf{b}_a & {}_I\mathbf{p} \end{pmatrix} \in \mathbb{R}^{16}, \tag{5} $$

- Our objective: estimate the true state ${}_I\mathbf{x}$ with an estimate ${}_I\hat{\mathbf{x}}$:

$$ {}_I\hat{\mathbf{x}} \triangleq \begin{pmatrix} {}_I\hat{\bar{\mathbf{q}}} & \hat{\mathbf{b}}_g & {}_I\hat{\mathbf{v}} & \hat{\mathbf{b}}_a & {}_I\hat{\mathbf{p}} \end{pmatrix} \in \mathbb{R}^{16}. \tag{6} $$

- The IMU error state is:

$$ {}_I\tilde{\mathbf{x}} \triangleq \begin{pmatrix} {}_I\tilde{\bar{\boldsymbol{\theta}}} & \tilde{\mathbf{b}}_g & {}_I\tilde{\mathbf{v}} & \tilde{\mathbf{b}}_a & {}_I\tilde{\mathbf{p}} \end{pmatrix} \in \mathbb{R}^{15}. \tag{7} $$

- ${}_I\tilde{\bar{\boldsymbol{\theta}}}$ is the angle axis representation of ${}_I\tilde{\bar{\mathbf{q}}}$, and $\tilde{\bar{\mathbf{q}}} \simeq \begin{bmatrix} \frac{1}{2}\tilde{\bar{\boldsymbol{\theta}}}^\top & 1 \end{bmatrix}^\top$

State augmentation

- Keep a history of the camera poses of length $W+1$. The camera state and error state are:

$$ {}_C\mathbf{x} \triangleq ({}_C\bar{\mathbf{q}},\ {}_C\mathbf{p}), \quad {}_C\tilde{\mathbf{x}} \triangleq ({}_C\tilde{\bar{\theta}},\ {}_C\tilde{\mathbf{p}}) \in \mathbb{R}^{6(W+1)}. \tag{8} $$

- The complete state and error state at time t are:

$$ \mathbf{x}_t \triangleq \left({}_I\mathbf{x}_t \quad {}_C\mathbf{x}_{t-W:t} \right), \quad \tilde{\mathbf{x}}_t \triangleq \left({}_I\tilde{\mathbf{x}}_t \quad {}_C\tilde{\mathbf{x}}_{t-W:t} \right). \tag{9} $$

Prediction

- We can discretize the state estimate dynamics to obtain the prediction step for the IMU state mean
- Linearized continuous-time IMU error state dynamics satisfy:

$$ {}_I\dot{\tilde{\mathbf{x}}} = \mathbf{F}(t)\,{}_I\tilde{\mathbf{x}} + \mathbf{G}(t)\mathbf{n}_I \tag{10} $$

- The propagated covariance of the IMU state is

$$ \mathbf{P}_{II_{t+1|t}} = \mathbf{\Phi}_t \mathbf{P}_{II_{t|t}} \mathbf{\Phi}_t + \mathbf{Q}_t \tag{11} $$

- where $\mathbf{Q} = \mathbb{E}\left[\mathbf{n}_I \mathbf{n}_I^\top\right]$ is continuous noise covariance

$$ \begin{aligned} \mathbf{\Phi}_t =& \mathbf{\Phi}(t, t+1) = \exp\left(\int_{t+1}^{t} \mathbf{F}(\tau)\right) d\tau \\ \mathbf{Q}_t =& \int_t^{t+1} \mathbf{\Phi}\left(t+1, \tau\right) \mathbf{GQG} \mathbf{\Phi}\left(t+1, \tau\right)^\top d\tau \end{aligned} $$

Prediction

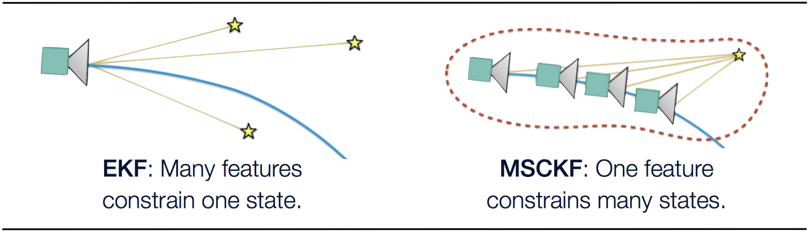
- The covariance matrix after augmentation with a new camera state is

$$\mathbf{P}_{t+1|t} = \begin{pmatrix} \mathbf{I}_{15+6(W+1)} \\ \mathbf{J}_t \end{pmatrix} \mathbf{P}_{t+1|t} \begin{pmatrix} \mathbf{I}_{15+6(W+1)} \\ \mathbf{J}_t \end{pmatrix}^\top \tag{12}$$

- We obtain the Gaussian pdf $p({}^{i}\mathbf{z}_t \mid \mathcal{X})$ in (1)

EKF vs MSCKF

- EKF: Many features constrain one state.
- MSCKF: One feature constrains many states.



EKF: Many features constrain one state.

MSCKF: One feature constrains many states.

Figure: Comparison of EKF, MSCKF.

Update

- The measurement model relating a landmark $\boldsymbol{\ell} \in \mathcal{L}$ to its observation $\mathbf{z}_t$ in camera frame $\{C_t\}$ is:

$$\mathbf{z}_t = \pi\left({}_{C_t}\mathbf{R}^\top(\boldsymbol{\ell} - {}_{C_t}\mathbf{p})\right) + \mathbf{n}_t \tag{13}$$

- The estimate ${}^g\hat{\boldsymbol{\ell}}_j$ is used to define a residual $\mathbf{r}^j$ via first-order Taylor series linearization of ${}^g z^j_{t-W:t}$ based on (13):

$$\mathbf{r}^j = {}^g\mathbf{z}^j_{t-W:t} - {}^g\hat{\mathbf{z}}^j_{t-W:t} \approx \mathbf{H}^j_x\tilde{\mathbf{x}} + \mathbf{H}^j_\ell {}^g\tilde{\boldsymbol{\ell}}_j + \mathbf{n}^j \tag{14}$$

- MSCKF update, $p({}^g\mathbf{z}_t \mid \mathcal{X})$ in (1):

$$\mathbf{r}^j_o = \mathbf{A}^\top\mathbf{r}^j \approx \mathbf{A}^\top\mathbf{H}^j_x\tilde{\mathbf{x}} + \mathbf{A}^\top\mathbf{n}^j = \mathbf{H}^j_o\tilde{\mathbf{x}} + \mathbf{n}^j_o. \tag{15}$$

Constrained filtering

- MSCKF with Persistent Object States

$$\mathbf{x}_t = \begin{pmatrix} {}_I\tilde{\mathbf{x}}_t & {}_C\tilde{\mathbf{x}}_{t-W:t} & {}_{C_1}\boldsymbol{\ell}_1^{\vee} & \cdots & {}_{C_k}\boldsymbol{\ell}_k^{\vee} \end{pmatrix} \tag{16}$$

- The original measurement model in EKF SLAM as in Eq. 13 is

$$\mathbf{z} = \mathbf{H}\mathbf{x}_t + \mathbf{n}$$

where $\mathbf{x}_t$ is the state vector defined in eq. 16. The measurement model could be augmented to

$$\begin{bmatrix} \mathbf{z} \\ \mathbf{d} \end{bmatrix} = \begin{bmatrix} \mathbf{H} \\ \mathbf{D} \end{bmatrix} \mathbf{x}_t + \begin{bmatrix} \mathbf{n} \\ \mathbf{n}_c \end{bmatrix} \tag{17}$$

where the constraint is enforced as $\mathbf{D}\mathbf{x}_t + \mathbf{n}_c = \mathbf{d}$, and $\mathbf{n}_c$ is noise with covariance $\boldsymbol{\Sigma}_c$.

Constrained filtering

- Landmarks annotations $\ell_p \sim \mathcal{N}(\boldsymbol{\mu}_p, \boldsymbol{\Sigma}_p)$, $\ell_q \sim \mathcal{N}(\boldsymbol{\mu}_q, \boldsymbol{\Sigma}_q)$
- The Euclidean distance $\mathbf{d} = ||\ell_p - \ell_q||_2$, where $\Delta\ell = \ell_p - \ell_q \sim \mathcal{N}(\boldsymbol{\mu_p} - \boldsymbol{\mu_q}, \boldsymbol{\Sigma}_p + \boldsymbol{\Sigma}_q)$.
- Covariance of $\mathbf{d}$ is $\boldsymbol{A}(\boldsymbol{\Sigma}_p + \boldsymbol{\Sigma}_q)\boldsymbol{A}^\top$, where $\boldsymbol{A}$ is the Jacobian of the L_2 norm.

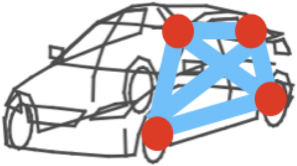


Figure: Pairwise constraints.

Constrained filtering

- Constrained filtering could fuse all available sources of information (S. Tully et al., 2012)

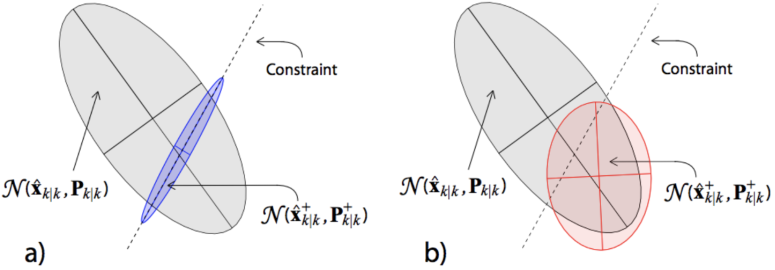

Constraint
$\mathcal{N}(\hat{\mathbf{x}}_{k|k}, \mathbf{P}_{k|k})$
$\mathcal{N}(\hat{\mathbf{x}}^+_{k|k}, \mathbf{P}^+_{k|k})$
a)
Constraint
$\mathcal{N}(\hat{\mathbf{x}}_{k|k}, \mathbf{P}_{k|k})$
$\mathcal{N}(\hat{\mathbf{x}}^+_{k|k}, \mathbf{P}^+_{k|k})$
b)

Figure: Posterior with equalities and inequalities constraints.

Quantitative Comparison

Enforcing constraints could keep the points close to groundtruth with large measurement noise

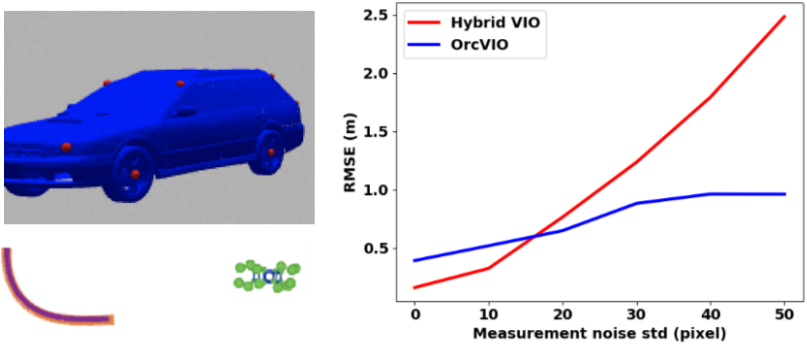
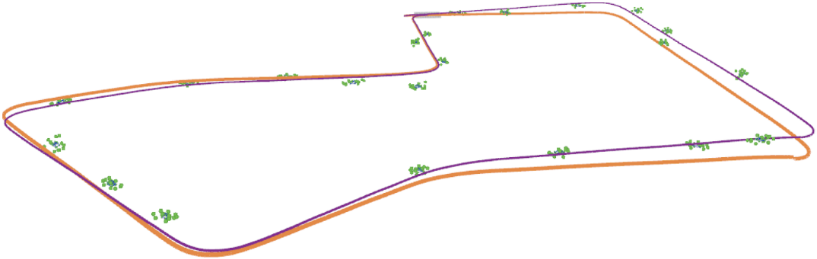


Figure: Left: 640×480 image, birdeye view. Right: RMSE comparison between Hybrid VIO and OrcVIO in Gazebo Simulation.

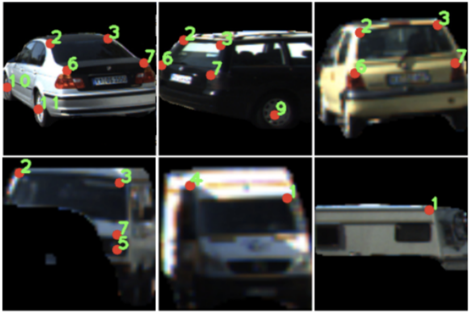
Qualitative evaluation

- Gazebo simulation using real-world IMU data
- Reconstruction for 22 cars
- Drift in Z is large due to insufficient movement



Qualitative evaluation

- Semantic keypoint detection using StarMap. Upper row: successes. Lower row: failures.

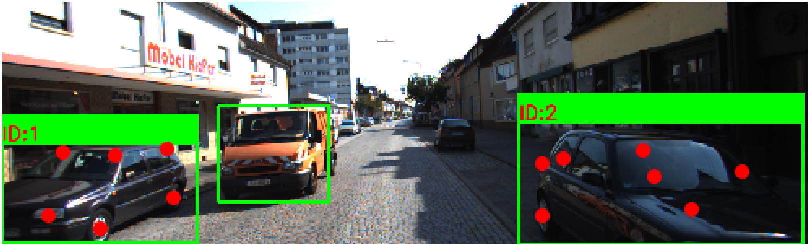


Listing 1: Keypoint labels

```
0  upper_left_windshield
1  upper_right_windshield
2  upper_right_rearwindow
3  upper_left_rearwindow
4  left_front_light
5  right_front_light
6  right_back_trunk
7  left_back_trunk
8  left_front_wheel
9  left_back_wheel
10  right_front_wheel
11  right_back_wheel
```

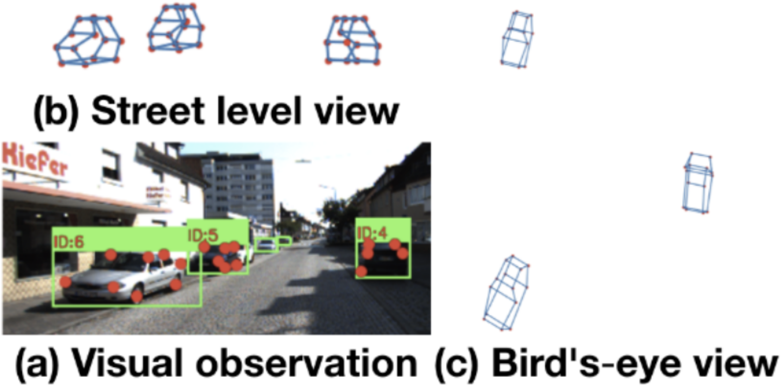
Qualitative evaluation

- Semantic feature detection on real-world dataset



Qualitative evaluation

Reconstruction snapshot on real-world dataset



(b) Street level view

(a) Visual observation (c) Bird's-eye view

Figure: Visulization of reconstruction.

Qualitative evaluation

- Birds eye view of reconstruction
- Both the precision and recall for the reconstruction have to be improved for real-world data
- Orange path is groundtruth trajectory, purple path is ours
- Red bounding boxes are groundtruth car positions Green wireframes are results from OrcVIO

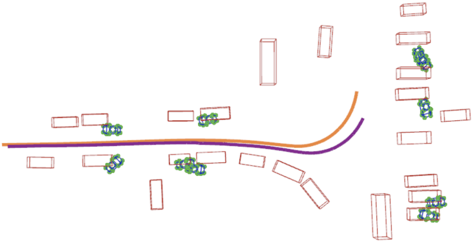


Figure: Visulization of reconstruction.

Weaknesses

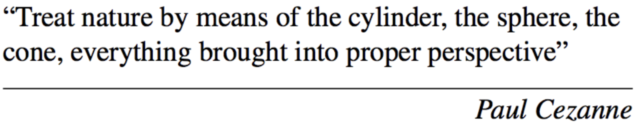
- We use triangulation and LevenbergMarquardt to obtain initial positions
- However, triangulation requires a sufficient baseline
- When baseline is small, depth estimation is inaccurate and landmarks will be pruned as outliers
- For some inliers depth is not accurate either which lead to incorrect object pose

Conclusion

- We present OrcVIO, which incorporates object structures for constrained state estimation
- The key insight is that there are objects in the scene and their keypoints are not independent
- The advantages include a more accurate estimation structure and an object map
- However, there is a lack of an object-level prior to restrict the depth estimation in triangulation and LM

Future work

- *Shape-Aware Adjustment*: given an initialization, use planarity and symmetry, etc to improve reconstruction
- QuadricSLAM (L. Nicholson et al., 2018) uses ellipsoids, CubeSLAM (S. Yang, 2019) uses cuboids. We will also explore how to use geometric shapes to help improve depth estimation

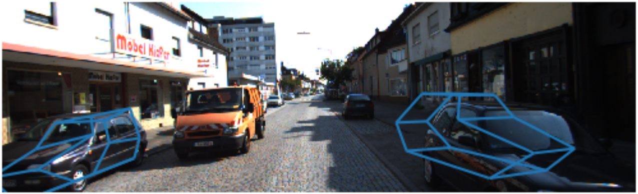


"Treat nature by means of the cylinder, the sphere, the cone, everything brought into proper perspective"

Paul Cezanne

Figure: A quote from a painter.

Initial results



References

- Cadena, Cesar, et al. "Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age." IEEE Transactions on robotics 32.6 (2016): 1309-1332.
- Mur-Artal, Raul, and Juan D. Tards. "Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras." IEEE Transactions on Robotics 33.5 (2017): 1255-1262.
- Engel, Jakob, Vladlen Koltun, and Daniel Cremers. "Direct sparse odometry." IEEE transactions on pattern analysis and machine intelligence 40.3 (2018): 611-625.
- Gal, Yarin, and Zoubin Ghahramani. "Dropout as a bayesian approximation: Representing model uncertainty in deep learning." international conference on machine learning. 2016.
- Tully, Stephen, et al. "Constrained filtering with contact detection data for the localization and registration of continuum robots in flexible environments." 2012 IEEE