# ClusterVO: Clustering Moving Instances and Estimating Visual Odometry for Self and Surroundings

Jiahui Huang[1] Sheng Yang[2] Tai-Jiang Mu[1] Shi-Min Hu[1]*
[1]BNRist, Department of Computer Science and Technology, Tsinghua University, Beijing
[2]Alibaba Inc., China
huang-jh18@mails.tsinghua.edu.cn, shengyang93fs@gmail.com
taijiang@tsinghua.edu.cn, shimin@tsinghua.edu.cn

## Abstract

*We present ClusterVO, a stereo Visual Odometry which simultaneously clusters and estimates the motion of both ego and surrounding rigid clusters/objects. Unlike previous solutions relying on batch input or imposing priors on scene structure or dynamic object models, ClusterVO is online, general and thus can be used in various scenarios including indoor scene understanding and autonomous driving. At the core of our system lies a multi-level probabilistic association mechanism and a heterogeneous Conditional Random Field (CRF) clustering approach combining semantic, spatial and motion information to jointly infer cluster segmentations online for every frame. The poses of camera and dynamic objects are instantly solved through a sliding-window optimization. Our system is evaluated on Oxford Multimotion and KITTI dataset both quantitatively and qualitatively, reaching comparable results to state-of-the-art solutions on both odometry and dynamic trajectory recovery.*

## 1. Introduction

Understanding surrounding dynamic objects is an important step beyond ego-motion estimation in the current visual Simultaneous Localization and Mapping (SLAM) community for the frontier requirements of advanced Augmented Reality (AR) or autonomous things navigation: In typical use cases of Dynamic AR, these dynamics need to be explicitly tracked to enable interactions of virtual object with moving instances in the real world. In outdoor autonomous driving scenes, a car should not only accurately localize itself but also reliably sense other moving cars to avoid possible collisions.

Despite the above need from emerging applications to perceive scene motions, most classical SLAM systems [4, 19, 20, 28] merely regard dynamics as outliers during pose estimation. Recently, advances in vision and robotics have demonstrated us with new possibilities of developing motion-aware Dynamic SLAM systems by coupling various different vision techniques like detection and tracking [5, 32, 36]. Nevertheless, currently these systems are often tailored for special use cases: For indoor scenes where dense RGB-D data are available, geometric features including convexity or structure regularities are used to assist segmentation [6, 34, 35, 38, 46]. For outdoor scenes, object priors like car sizes or road planar structure are exploited to constrain the solution spaces [2, 24, 26, 48]. These different assumptions render existing algorithms hardly applicable to general dynamic scenarios. Contrarily, ClusterSLAM [15] incorporates no scene prior, but it acts as a backend instead of a full system whose performance relies heavily on the landmark tracking and association quality.

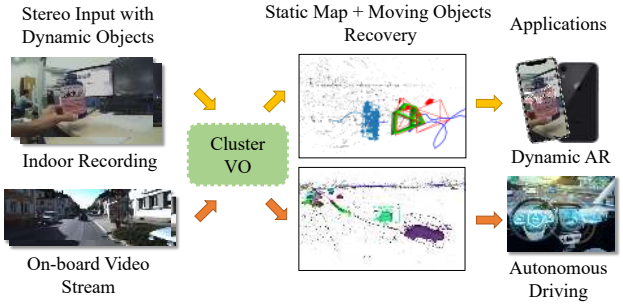



Figure 1. Our proposed system *ClusterVO* can simultaneously recover the camera ego-motion as well as cluster trajectories.

To bridge the above gap in current Dynamic SLAM solutions in the literature, we propose ClusterVO, a stereo visual odometry system for dynamic scenes, which simultaneously optimizes the poses of camera and multiple moving objects, regarded as clusters of point landmarks, in a unified manner, achieving a competitive frame-rate with promising tracking and segmentation ability as listed in Table 1. Because no geometric or shape priors on the scene or dy-

*corresponding author.

Table 1. Comparison with other dynamic SLAM solutions. 📷: Sensor(s) used. 🏠: Applicable in indoor scene? 🚗: Applicable in outdoor driving scenarios? 🚙: Recover poses of moving rigid bodies? ✈: Is online? 'NR' represents single Non-Rigid body.

| | 📷 | 🏠 | 🚗 | 🚙 | ✈ | FPS |
|---|---|---|---|---|---|---|
| ORB-SLAM2 [28] | Multiple | ✓ | ✓ | | ✓ | 10 |
| DynamicFusion [30] | RGB-D | ✓ | | NR | ✓ | - |
| MaskFusion [35] | RGB-D | ✓ | | ✓ | ✓ | 30 |
| Li *et al*. [24] | Stereo | | ✓ | ✓ | | 5.8 |
| DynSLAM [2] | Stereo | | ✓ | ✓ | ✓ | 2 |
| ClusterSLAM [15] | Stereo | ✓ | ✓ | ✓ | | 7 |
| **ClusterVO** | Stereo | ✓ | ✓ | ✓ | ✓ | **8** |

namic objects are imposed, our proposed system is general and adapts to many various applications ranging from autonomous driving, indoor scene perception to augmented reality development. Our novel strategy is solely based on sparse landmarks and 2D detections [32]; to make use of such a lightweight representation, we propose a robust multi-level probabilistic association technique to efficiently track both low-level features and high-level detections over time in the 3D space. Then a highly-efficient heterogeneous CRF jointly considering semantic bounding boxes, spatial affinity and motion consistency is applied to discover new clusters, cluster novel landmarks and refine existing clusterings. Finally, Both static and dynamic parts of the scene are solved in a sliding-window optimization fashion.

## 2. Related Works

**Dynamic SLAM / Visual Odometry.** Traditional SLAM or VO systems are based on static scene assumption and dynamic contents need to be carefully handled which would otherwise lead to severe pose drift. To this end, some systems explicitly detect motions and filter them either with motion consistency [8, 19, 20] or object detection modules [4, 49, 50]. The idea of simultaneously estimating ego motion and multiple moving rigid objects, same as our formulation, originated from the seminal SLAMMOT [44] project. Follow-ups like [6, 34, 35, 38, 46] use RGB-D as input and reconstruct dense models for the indoor scene along with moving objects. For better segmentation of object identities, [35, 46] combine heavy instance segmentation module and geometric features. [22, 40, 43] can track and reconstruct rigid object parts on a predefined articulation template (*e.g*. human hands or kinematic structures). [9, 31] couple existing visual-inertial system with moving objects tracked using markers. Many other methods are specially designed for road scenes by exploiting modern vision modules [3, 24, 26, 27, 29, 45]. Among them, [24] proposes a batch optimization to accurately track the motions of moving vehicles but a real-time solution is not presented.

Different from ClusterSLAM [15], which is based on motion affinity matrices for hierarchical clustering and SLAM, this work focuses on developing a relatively lightweight visual odometry, and faces challenges from real-time clustering and state estimation.

**Object Detection and Pose Estimation.** With the recent advances in deep learning technologies, the performance of 2D object detection and tracking have been boosted [5, 12, 14, 25, 32, 36]. Detection and tracking in 3D space from video sequences is a relatively unexplored area due to the difficulty in the 6-DoF (six degrees of freedom) pose estimation. In order to accurately estimate 3D positions and poses, many methods [13, 23] leverages a predefined object template or priors to jointly infer object depth and rotations. In ClusterVO, the combination of low-level geometric feature descriptors and semantic detections inferred simultaneously in the localization and mapping process can provide additional cues for efficient tracking and accurate object pose estimation.

## 3. ClusterVO

ClusterVO takes synchronized and calibrated stereo images as input, and outputs camera and object pose for each frame. For each incoming frame, semantic bounding boxes are detected using YOLO object detection network [32], and ORB features [33] are extracted and matched across stereo images. We first associate detected bounding boxes and extracted features to previously found clusters and landmarks, respectively, through a multi-level probabilistic association formulation (Sec. 3.1). Then, we perform heterogeneous conditional random field (CRF) over all features with associated map landmarks to determine the cluster segmentation for current frame (Sec. 3.2). Finally, the state estimation step optimizes all the states over a sliding window with marginalization and a smooth motion prior (Sec. 3.3). The pipeline is illustrated in Figure 2.

**Notations.** At frame $t$, ClusterVO outputs: the pose of the camera $\mathbf{P}_t^{\mathbf{c}}$ in the global reference frame, the state of all clusters (rigid bodies) $\{\boldsymbol{x}_t^{\mathbf{q}}\}_{\mathbf{q}}$, and the state of all landmarks $\boldsymbol{x}_t^{\mathbf{L}}$. The $\mathbf{q}$-th cluster state $\boldsymbol{x}_t^{\mathbf{q}} = (\mathbf{P}_t^{\mathbf{q}}, \boldsymbol{v}_t^{\mathbf{q}})$ contains current 6-DoF pose $\mathbf{P}_t^{\mathbf{q}} \in \mathrm{SE}(3)$ and current linear speed in 3D space $\boldsymbol{v}_t^{\mathbf{q}} \in \mathbb{R}^3$. Specially we use $\mathbf{q} = 0$ to denote the static scene for convenience. Hence $\forall t, \mathbf{P}_t^0 \equiv \mathbf{I}, \boldsymbol{v}_t^0 \equiv \mathbf{0}$. As a short hand, we denote the transformation from coordinate frame $\mathbf{a}$ to frame $\mathbf{b}$ as $\mathbf{T}_t^{\mathbf{ab}} := (\mathbf{P}_t^{\mathbf{a}})^{-1}\mathbf{P}_t^{\mathbf{b}}$. For the landmark state $\boldsymbol{x}_t^{\mathbf{L}} = \{(\boldsymbol{p}_t^i, \mathbf{q}^i, w^i)\}_i$, each landmark $i$ has the property of its global position $\boldsymbol{p}_t^i \in \mathbb{R}^3$, the cluster assignment $\mathbf{q}^i$ and its confidence $w^i \in \mathbb{N}^+$ defining the cluster assignment confidence. For observations, we denote the location of the $k$-th low-level ORB stereo feature extracted at frame $t$ as $\boldsymbol{z}_t^k = (u_L, v_L, u_R) \in \mathbb{R}^3$, and the $m$-th high-level semantic bounding box detected at frame $t$ as $\boldsymbol{B}_t^m$. Assuming the feature observation $\boldsymbol{z}_t^k$ is subject to a Gaussian noise with covariance ${}_{\boldsymbol{z}}\Sigma$, the noise of the triangulated points $\boldsymbol{Z}_t^i$ in camera space can be calculated as ${}_{\boldsymbol{Z}}\Sigma_t^i = \mathbf{J}_{\pi^{-1}}({}_{\boldsymbol{z}}\Sigma)\mathbf{J}_{\pi^{-1}}^\top$,

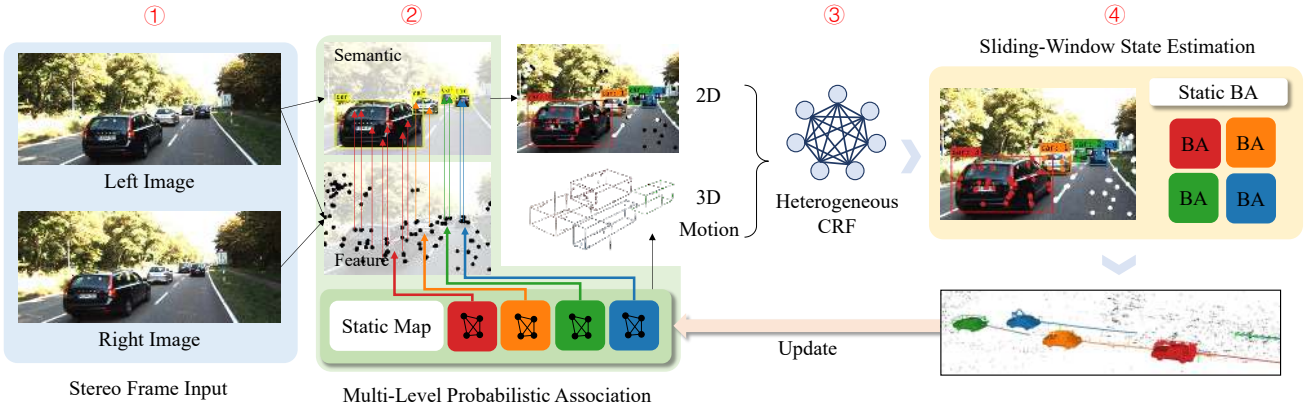



Figure 2. **Pipeline of ClusterVO.** ① For each incoming stereo frame ORB features and semantic bounding boxes are extracted. ② We apply multi-level probabilistic association to associate features with landmarks and bounding boxes with existing clusters. ③ Then we cluster the landmarks observed in the current frame into different rigid bodies using the Heterogeneous CRF module. ④ The state-estimation is performed in a sliding window manner with specially designed keyframe mechanism. Optimized states are used to update the static maps and clusters.

where $\pi$ is the stereo projection function, $\pi^{-1}$ is the corresponding back-projection function and $\mathbf{J}_f$ is the Jacobian matrix of function $f$.

For generality, we do not introduce a category-specific canonical frame for each cluster. Instead, we initialize the cluster pose $\mathbf{P}_t^{\mathbf{q}}$ with the center and the three principal orthogonal directions of the landmark point clouds belonging to the cluster as the translational and rotational part respectively and track the relative pose ever since.

### 3.1. Multi-level Probabilistic Association

For the landmarks on static map (*i.e.* $\mathbf{q}^i = 0$), the features can be robustly associated by nearest neighbour search and descriptor matching [28]. However, tracking dynamic landmarks which move fast on the image space is not a trivial task. Moreover, we need to associate each detected bounding box $\boldsymbol{B}_t^m$ to an existing map cluster if possible, which is required in the succeeding Heterogeneous CRF module.

To this end, we propose a multi-level probabilistic association scheme for dynamic landmarks (*i.e.* $\mathbf{q}^i \neq 0$), assigning low-level feature observation $\boldsymbol{z}_t^k$ to its source landmark id $k \to i$ and high-level bounding box $\boldsymbol{B}_t^m$ to a cluster $m \to \mathbf{q}$. The essence of the probabilistic approach is to model the position of a landmark by a Gaussian distribution with mean $\boldsymbol{p}_t^i$ and covariance $\Sigma_t^i$ and consider the uncertainty throughout the matching.

Ideally, $\Sigma_t^i$ should be extracted from the system information matrix from the last state estimation step, but the computation burden is heavy. We hence approximate $\Sigma_t^i$ as transformed ${}_{\boldsymbol{Z}}\Sigma_{t'<t}^i$ with the smallest determinant, *i.e.*:

$$\Sigma_t^i := \mathbf{R}_{t'}^{\mathbf{c}}\, {}_{\boldsymbol{Z}}\Sigma_{t'}^i \mathbf{R}_{t'}^{\mathbf{c}\top}, \quad t' := \underset{t'<t}{\operatorname{argmin}} |\, {}_{\boldsymbol{Z}}\Sigma_{t'}^i|, \tag{1}$$

which can be incrementally updated. $\mathbf{R}_t^{\mathbf{c}}$ is the rotational part of $\mathbf{P}_t^{\mathbf{c}}$.

For each new frame, we perform motion prediction for each cluster using $\boldsymbol{v}_t^{\mathbf{q}}$. The predicted 3D landmark positions as well as its noise covariance matrix are re-projected back into the current frame using $\boldsymbol{\zeta}_t^i = \pi(\boldsymbol{p}_t^i + \boldsymbol{v}_t^{\mathbf{q}}), \Gamma_t^i = \mathbf{J}_\pi \Sigma_t^i \mathbf{J}_\pi^\top$. The probability score of assigning the $k$-th observation to landmark $i$ becomes:

$$p_i(k) \propto \left[\|\boldsymbol{\zeta}_t^i - \boldsymbol{z}_t^k\|_{\Gamma_t^i}^2 < \gamma\right] \cdot s_{ik}, \tag{2}$$

where $[\cdot]$ is an indicator function, $s_{ik}$ is the descriptor similarity between landmark $i$ and observation $\boldsymbol{z}_t^k$ and $\gamma = 4.0$ in our experiments. For each observation $k$, we choose its corresponding landmark $i$ with the highest assignment probability score: $k \to \operatorname{argmax}_i p_i(k)$ if possible. In practice, Eq. 2 is only evaluated on a small neighbourhood of $\boldsymbol{z}_t^k$.

We further measure the uncertainty of the association $m \to \mathbf{q} := \operatorname{argmax}_{\mathbf{q}'} p_{\mathbf{q}'}(m)$ by calculating the Shannon cross-entropy $\mathcal{E}_t^{\mathbf{q}}$ as:

$$\begin{aligned} \mathcal{E}_t^{\mathbf{q}} &:= -\sum_m p_{\mathbf{q}}(m) \log p_{\mathbf{q}}(m), \\ p_{\mathbf{q}}(m) &\propto \sum_{\boldsymbol{\zeta}_t^k \in \boldsymbol{B}_t^m} (1/|\Gamma_t^i|), \end{aligned} \tag{3}$$

where $p_{\mathbf{q}}(m)$ is the probability of assigning the $m$-th bounding box to cluster $\mathbf{q}$. If $\mathcal{E}_t^{\mathbf{q}}$ is smaller than 1.0, we consider this as a successful high-level association, in which case we perform additional brute force low-level feature descriptor matching within the bounding box to find more feature correspondences.

### 3.2. Heterogeneous CRF for Cluster Assignment

In this step, we determine the cluster assignment $\mathbf{q}^i$ of each landmark $i$ observed in the current frame. A conditional random field model combining semantic, spatial and motion information, which we call 'heterogeneous CRF', is applied, minimizing the following energy:

$$E(\{\mathbf{q}^i\}_i) \coloneqq \sum_i \psi_u(\mathbf{q}^i) + \alpha \sum_{i<j} \psi_p(\mathbf{q}^i, \mathbf{q}^j), \quad (4)$$

which is a weighted sum ($\alpha > 0$ being the balance factor) of unary energy $\psi_u$ and pairwise energy $\psi_p$ on a complete graph of all the observed landmarks. The total number of classes for CRF is set to $M = N_1 + N_2 + 1$, where $N_1$ is the number of *live* clusters, $N_2$ is the number of unassociated bounding boxes in this frame and the trailing 1 allows for an outlier class. A cluster is considered *live* if at least one of its landmarks is observed during the past $L$ frames.

**Unary Energy.** The unary energy decides the probability of the observed landmark $i$ belonging to a specific cluster $\mathbf{q}^i$ and contains three sources of information:

$$\psi_u(\mathbf{q}^i) \propto p_{2\mathrm{D}}(\mathbf{q}^i) \cdot p_{3\mathrm{D}}(\mathbf{q}^i) \cdot p_{\mathrm{mot}}(\mathbf{q}^i). \quad (5)$$

The first multiplier $p_{2\mathrm{D}}$ incorporates information from the detected semantic bounding boxes. The probability should be large if the landmark lies within a bounding box. Let $\mathbf{C}_t^i$ be the set of cluster indices corresponding to the bounding boxes where the observation of landmark $\boldsymbol{z}_t^i$ resides and $\eta$ be a constant for the detection confidence, then:

$$p_{2\mathrm{D}}(\mathbf{q}^i) \propto \begin{cases} \eta/|\mathbf{C}_t^i| & \mathbf{q}^i \in \mathbf{C}_t^i \\ (1-\eta)/(M - |\mathbf{C}_t^i|) & \mathbf{q}^i \notin \mathbf{C}_t^i \end{cases}. \quad (6)$$

The second multiplier $p_{3\mathrm{D}}$ emphasizes the spatial affinity by assigning a high probability to the landmarks near the center of a cluster:

$$p_{3\mathrm{D}}(\mathbf{q}^i) \propto \exp\left(-\|\boldsymbol{Z}_t^i - \boldsymbol{c}_{\mathbf{q}^i}\|^2_{\boldsymbol{z}\Sigma_t^i} / l^2_{\mathbf{q}^i}\right), \quad (7)$$

where $\boldsymbol{c}_{\mathbf{q}^i}$ and $l_{\mathbf{q}^i}$ are the cluster center and dimension, respectively, determined by the center and the $30^{\mathrm{th}}$/$70^{\mathrm{th}}$ percentiles (found empirically) of the cluster landmark point cloud.

The third multiplier defines how the trajectories of cluster $\mathbf{q}^i$ over a set of timesteps $\mathcal{T}$ can explain the observation:

$$p_{\mathrm{mot}}(\mathbf{q}^i) \propto \prod_{t' \in \mathcal{T}} \frac{\exp(-\|\boldsymbol{z}_{t'}^i - \pi(\mathbf{T}_{t'}^{\mathbf{cq}^i}(\mathbf{P}_t^{\mathbf{q}^i})^{-1}\boldsymbol{p}_t^i)\|^2_{\boldsymbol{z}\Sigma})}{\sqrt{|\boldsymbol{z}\Sigma|}}, \quad (8)$$

which is a simple reprojection error w.r.t. the observations. In our implementation we set $\mathcal{T} = \{t-5, t\}$. For the first 5 frames this term is not included in Eq. 5.

The single 2D term only considers the 2D semantic detection, which possibly contains many outliers around the edge of the bounding box. By adding the 3D term, landmarks belonging to faraway background get pruned. However, features close to the 3D boundary, *e.g.*, on the ground nearby a moving vehicle, still have a high probability belonging to the cluster, whose confidence is further refined by the motion term. Please refer to Sec. 4.4 for evaluations and visual comparisons on these three terms.

**Pairwise Energy.** The pairwise energy is defined as:

$$\psi_p(\mathbf{q}^i, \mathbf{q}^j) \coloneqq [\mathbf{q}^i \neq \mathbf{q}^j] \cdot \exp(-\|\boldsymbol{p}_t^i - \boldsymbol{p}_t^j\|^2), \quad (9)$$

where the term inside the exponential operator is the distance between two landmarks $\boldsymbol{p}_t^i, \boldsymbol{p}_t^j$ in 3D space. The pairwise energy can be viewed as a noise-aware Gaussian smoothing kernel to encourage spatial labeling continuity.

We use an efficient dense CRF inference method [21] to solve for the energy minimization problem. After successful inference, we perform Kuhn-Munkres algorithm to match current CRF clustering results with previous cluster assignments. New clusters are created if no proper cluster assignment is found for an inferred label. We then update the weight $w^i$ for each landmark according to a strategy introduced in [41] and change its cluster assignment if necessary: When the newly assigned cluster is the same as the landmark's previous cluster, we increase the weight $w^i$ by 1, otherwise the weight is decreased by 1. When $w^i$ is decreased to 0, a change in cluster assignment is triggered to accept the currently assigned cluster.

### 3.3. Sliding-Window State Estimation

**Double-Track Frame Management.** Keyframe-based SLAM systems like ORB-SLAM2 [28] select keyframes by the spatial distance between frames and the number of commonly visible features among frames. For ClusterVO where the trajectory of each cluster is incorporated into the state estimation process, the aforementioned strategy for keyframe selection is not enough to capture the relatively fast-moving clusters.

Instead of the chunk strategy proposed in ClusterSLAM [15], we employ a sliding window optimization scheme in accordance with a novel *double-track* frame management design (Figure 3). The frames maintained and optimized by the system are divided into two sequential tracks: a temporal track $\mathcal{T}_t$ and a spatial track $\mathcal{T}_s$. $\mathcal{T}_t$ contains the most recent input frames. Whenever a new frame comes, the oldest frame in $\mathcal{T}_t$ will be moved out. If this frame is spatially far away enough from the first frame in $\mathcal{T}_s$ or the number of commonly visible landmarks is sufficiently small, this frame will be appended to the tail of $\mathcal{T}_s$, otherwise it will be discarded. This design has several advantages. First, frames in the temporal track record all recent observations and hence allow for enough observations to

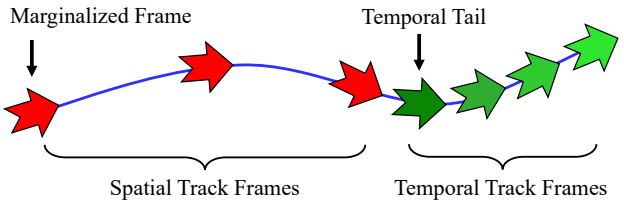



Figure 3. **Frame Management in ClusterVO.** Frames maintained by the system consist of spatial track (red) and temporal track (green). When a new frame comes, the oldest frame in the temporal track (Temporal Tail) will either be discarded or promoted into the spatial track. The last spatial frame is to be marginalized if the total number of spatial frames exceeds a given threshold.

track a fast-moving cluster. Second, previous wrongly clustered landmarks can be later corrected and re-optimization based on new assignments is made possible. Third, features detected in the spatial track help create enough parallax for accurate landmark triangulation and state estimation.

**For static scene $\mathbf{q} = 0$ and camera pose**, the energy function for optimization is a standard Bundle Adjustment [42] augmented with an additional marginalization term:

$$\mathbf{E}(\{\boldsymbol{x}_t^{\mathbf{c}}, \boldsymbol{x}_t^{\mathbf{L}}\}_{t\in\mathcal{T}_a}) \coloneqq \sum_{i\in\mathcal{I}_0, t\in\mathcal{T}_a} \rho(\|\boldsymbol{z}_t^i - \pi((\mathbf{P}_t^{\mathrm{c}})^{-1}\boldsymbol{p}_t^i)\|^2_{\boldsymbol{z}\Sigma}) + \sum_{t\in\mathcal{T}_a} \|\delta\boldsymbol{x}_t^{\mathbf{c}} - \mathbf{H}^{-1}\boldsymbol{\beta}\|^2_{\mathbf{H}}, \tag{10}$$

where $\mathcal{T}_a \coloneqq \mathcal{T}_s \cup \mathcal{T}_t$, $\mathcal{I}_{\mathbf{q}} = \{i|\mathbf{q}^i = \mathbf{q}\}$ indicates all landmarks belonging to cluster $\mathbf{q}$ and $\rho(\cdot)$ is robust Huber M-estimator. As the static scene involves a large number of variables and simply dropping these variables out of the sliding window will cause information loss, leading to possible drifts, we *marginalize* some variables which would otherwise be removed and summarize the influence to the system with the marginalization term in Eq. 10. Marginalization is only performed when a frame is discarded from the spatial track $\mathcal{T}_s$. To restrict dense fill-in of landmark blocks in the information matrix, the observations from the frame to be removed will be either deleted if the corresponding landmark is observed by the newest frame or marginalized otherwise. This marginalization strategy only adds dense Hessian block onto the frames instead of landmarks, making the system still solvable in real-time.

More specifically, in the marginalization term, $\delta\boldsymbol{x}$ is the state change relative to the critical state $\boldsymbol{x}^*$ captured when marginalization happens. For the computation of $\mathbf{H}$ and $\boldsymbol{\beta}$, we employ the standard Schur Complement: $\mathbf{H} = \boldsymbol{\Lambda}_{aa} - \boldsymbol{\Lambda}_{ab}\boldsymbol{\Lambda}_{bb}^{-1}\boldsymbol{\Lambda}_{ba}, \boldsymbol{\beta} = \boldsymbol{b}_a - \boldsymbol{\Lambda}_{ab}\boldsymbol{\Lambda}_{bb}^{-1}\boldsymbol{b}_b$, where $\boldsymbol{\Lambda}_{(\cdot)}$ and $\boldsymbol{b}_{(\cdot)}$ are components of the system information matrix $\boldsymbol{\Lambda}$ and information vector $\boldsymbol{b}$ extracted by linearizing around $\boldsymbol{x}^*$:

$$\boldsymbol{\Lambda} = \begin{bmatrix} \boldsymbol{\Lambda}_{aa} & \boldsymbol{\Lambda}_{ab} \\ \boldsymbol{\Lambda}_{ba} & \boldsymbol{\Lambda}_{bb} \end{bmatrix}, \quad \boldsymbol{b} = \begin{bmatrix} \boldsymbol{b}_a \\ \boldsymbol{b}_b \end{bmatrix}. \tag{11}$$

**For dynamic clusters $\mathbf{q} \neq 0$**, the motions are modeled using a white-noise-on-acceleration prior [1], which can be written in the following form in continuous time $t, t' \in \mathbb{R}$:

$$\ddot{\boldsymbol{t}}^{\mathbf{q}}(t) \sim \mathcal{GP}(\mathbf{0}, \boldsymbol{Q}\delta(t - t')), \tag{12}$$

where $\boldsymbol{t}^{\mathbf{q}}$ is the translational part of the continuous cluster pose $\mathbf{P}^{\mathbf{q}}$ (hence $\ddot{\boldsymbol{t}}^{\mathbf{q}}$ is the cluster acceleration), $\mathcal{GP}$ stands for the Gaussian Process, and $\boldsymbol{Q}$ denotes its power spectral matrix. We define the energy function for optimizing the $\mathbf{q}$-th cluster trajectories and its corresponding landmark positions as follows:

$$\mathbf{E}(\{\boldsymbol{x}_t^{\mathbf{q}}, \boldsymbol{x}_t^{\mathbf{L}}\}_{t\in\mathcal{T}_t}) \coloneqq \sum_{t,t^+\in\mathcal{T}_t} \left\| \begin{bmatrix} \boldsymbol{t}_{t^+}^i \\ \boldsymbol{v}_{t^+}^i \end{bmatrix} - \mathbf{A} \begin{bmatrix} \boldsymbol{t}_t^i \\ \boldsymbol{v}_t^i \end{bmatrix} \right\|^2_{\hat{\boldsymbol{Q}}} + \sum_{i\in\mathcal{I}_{\mathbf{q}}, t\in\mathcal{T}_t} \rho(\|\boldsymbol{z}_t^i - \pi(\mathbf{T}_t^{\mathbf{cq}^i}(\mathbf{P}_t^{\mathbf{c}})^{-1}\boldsymbol{p}_t^i)\|^2_{\boldsymbol{z}\Sigma}), \tag{13}$$

in which

$$\mathbf{A} \coloneqq \begin{bmatrix} \mathbf{I} & \Delta t\mathbf{I} \\ \mathbf{0} & \mathbf{I} \end{bmatrix}, \hat{\boldsymbol{Q}}^{-1} \coloneqq \begin{bmatrix} 12/\Delta t^3 & -6/\Delta t^2 \\ -6/\Delta t^2 & 4/\Delta t \end{bmatrix} \otimes \boldsymbol{Q}^{-1}, \tag{14}$$

where $\otimes$ is the Kronecker product and $\Delta t = t^+ - t$, $t^+$ being the next adjacent timestamp of frame $t$. Eq. 13 is the sum of motion prior term and reprojection term. The motion prior term is obtained by querying the random process model of Eq. 12, which intuitively penalizes the change in velocity over time and smooths cluster motion trajectory which would otherwise be noisy due to fewer features on clusters than static scenes. Note that different from the energy term for the static scene which optimizes over both $\mathcal{T}_s$ and $\mathcal{T}_t$, for dynamic clusters only $\mathcal{T}_t$ is considered.

During the optimization of cluster state, the camera state $\boldsymbol{x}_t^{\mathbf{c}}$ stays unchanged. The optimization process for each cluster can be easily paralleled because their states are mutually independent (in practice the system speed is 8.5Hz & 7.8Hz for 2 & 6 clusters, resp.).

## 4. Experiments

### 4.1. Datasets and Parameter Setup

The effectiveness and general applicability of ClusterVO system is mainly demonstrated in two scenarios: indoor scenes with moving objects and autonomous driving with moving vehicles.

For indoor scenes, we employ the stereo Oxford Multimotion dataset (OMD) [17] for evaluation. This dataset is specially designed for indoor simultaneous camera localization and rigid body motion estimation, with the ground-truth trajectories recovered using a motion capture system. Evaluations and comparisons are performed on two sequences: `swinging_4_unconstrained` (S4,

500 frames, with four moving bodies: S4-C1, S4-C2, S4-C3, S4-C4) and `occlusion_2_unconstrained` (O2, 300 frames, with two moving bodies: O2-Tower and O2-Block), because these are the only sequences with baseline results reported in sequential works from Judd *et al*. [18, 16] named 'MVO'.

For autonomous driving cases, we employ the challenging KITTI dataset [10] for demonstration. As most of the sequences in the odometry benchmark have low dynamics and comparisons on these data can hardly lead to sensible improvements over other SLAM solutions (*e.g*. ORB-SLAM), similar to Li *et al*. [24], we demonstrate the strength of our method in selected sequences from the raw dataset as well as the full 21 tracking training sequences with many moving cars. The ground-truth camera ego-motion is obtained from the OxTS packets (combining GNSS and inertial navigation) provided by the dataset.

The CRF weight is set to $\alpha = 5.0$ and the 2D unary energy constant $\eta = 0.95$. The power spectral matrix $\boldsymbol{Q} = 0.01\mathbf{I}$ for the motion prior. The maximum sizes of the double-track are set to $|\mathcal{T}_s| = 5$ and $|\mathcal{T}_t| = 15$. The threshold for determining whether the cluster is still live is set to $L = |\mathcal{T}_t|$. All of the experiments are conducted on an Intel Core i7-8700K, 32GB RAM desktop computer with an Nvidia GTX 1080 GPU.

### 4.2. Indoor Scene Evaluations

We follow the same evaluation protocol as in [18], by computing the maximum drift (deviation from ground-truth pose) across the whole sequence in translation and rotation (represented in three Euler angles, namely roll, yaw and pitch) for camera ego-motion as well as for all moving cluster trajectories. As our method does not define a canonical frame for detected clusters, we need to register the pose recovered by our method with the ground-truth trajectory. To this end, we multiply our recovered pose with a rigid transformation $\mathbf{T}_\mathrm{r}$ which minimizes the sum of the difference between $\mathbf{P}_t^\mathbf{q}\mathbf{T}_\mathrm{r}$ and the ground-truth pose for all $t$. This is based on the assumption that the local coordinates of the recovered landmarks can be registered with the positions of ground-truth landmarks using this rigid transformation.

For the semantic bounding box extraction, the YOLOv3 network [32] is re-trained to detect an additional class named 'block' representing the swinging or rotating blocks in the dataset. The detections used for training are labeled using a combined approach with human annotations and a median flow tracker on the rest frames from S4 and O2.

Figure 4 shows the ratio of decrease in the drift compared with the baseline MVO [18, 16]. More than half of the trajectory estimation results improve by over 25%, leading to accurate camera ego-motion and cluster motion recoveries. Two main advantages of ClusterVO over MVO have made the improvement possible: First, the pipeline in

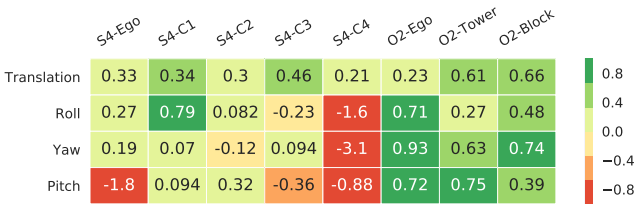
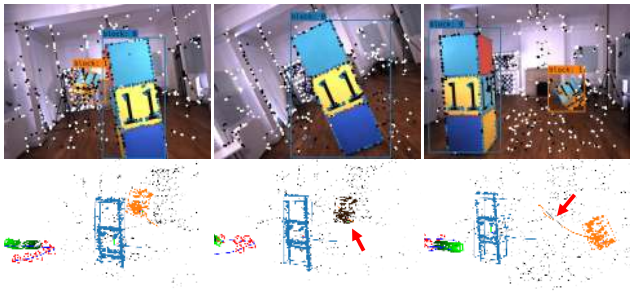



Figure 4. Performance comparison with MVO on S4 and O2 sequence in Oxford Multimotion [17] dataset. The numbers in the heatmap show the ratio of decrease in error using ClusterVO for different trajectories and measurements.



(a) Before Occlusion (b) During Occlusion (c) Completed Trajectory

Figure 5. Qualitative results in OMD Sequence O2. The three subfigures demonstrate an occlusion handling process by ClusterVO.

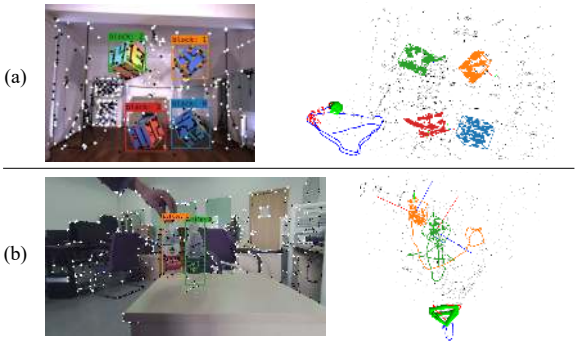



Figure 6. Other indoor qualitative results. (a) OMD Sequence S4; (b) A laboratory scene where two bottles are reordered.

MVO requires a stable tracking of features in each input batch of ∼50 frames and this keeps only a small subset of landmarks where the influence of noise becomes more dominating, while ClusterVO maintains consistent landmarks for each individual cluster and associates both low-level and high-level information to maximize the utility of historical information. Second, if the motion in a local window is small, the geometric-based method will tend to misclassify dynamic landmarks and degrade the recovered pose results; ClusterVO, however, leverages additional semantic and spatial information to achieve more accurate and meaningful classification and estimation.

Meanwhile, the robust association strategy and double-track frame management design allow ClusterVO to contin-

Table 2. Camera ego-motion comparison with state-of-the-art systems on KITTI raw dataset. The unit of ATE and T.RPE is meters and the unit for R.RPE is radians.

| Sequence | ORB-SLAM2 [28] | | | DynSLAM [2] | | | Li *et al*. [24] | ClusterSLAM [15] | | | ClusterVO | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | ATE | R.RPE | T.RPE | ATE | R.RPE | T.RPE | ATE | ATE | R.RPE | T.RPE | ATE | R.RPE | T.RPE |
| 0926-0009 | 0.91 | **0.01** | **1.89** | 7.51 | 0.06 | 2.17 | 1.14 | 0.92 | 0.03 | 2.34 | **0.79** | 0.03 | 2.98 |
| 0926-0013 | 0.30 | **0.01** | **0.94** | 1.97 | 0.04 | 1.41 | 0.35 | 2.12 | 0.07 | 5.50 | **0.26** | **0.01** | 1.16 |
| 0926-0014 | 0.56 | **0.01** | 1.15 | 5.98 | 0.09 | 2.73 | 0.51 | 0.81 | 0.03 | 2.24 | **0.48** | **0.01** | **1.04** |
| 0926-0051 | **0.37** | **0.00** | **1.10** | 10.95 | 0.10 | 1.65 | 0.76 | 1.19 | 0.03 | 1.44 | 0.81 | 0.02 | 2.74 |
| 0926-0101 | 3.42 | 0.03 | 14.27 | 10.24 | 0.13 | **12.29** | 5.30 | 4.02 | **0.02** | 12.43 | **3.18** | **0.02** | 12.78 |
| 0929-0004 | 0.44 | **0.01** | **1.22** | 2.59 | 0.02 | 2.03 | **0.40** | 1.12 | 0.02 | 2.78 | **0.40** | 0.02 | 1.77 |
| 1003-0047 | 18.87 | 0.05 | 28.32 | 9.31 | 0.05 | 6.58 | **1.03** | 10.21 | 0.06 | 8.94 | 4.79 | 0.05 | **6.54** |

uously track cluster motion even it is temporarily occluded. This feature is demonstrated in figure 5 on the O2 sequence where the block is occluded by the tower for $\sim$10 frames. The cluster's motion is predicted during the occlusion and finally the prediction is probabilistically associated with the re-detected semantic bounding box of the block. The state estimation module is then relaunched to recover the motion using the information both before and after the occlusion.

Figure 6(a) shows qualitative results on the S4 sequence and in Figure 6(b) another result from a practical indoor laboratorial scene with two moving bottles recorded using a Mynteye stereo camera is shown.

### 4.3. KITTI Driving Evaluations

Similar to Li *et al*. [24], we divide the quantitative evaluation into ego-motion comparisons and 3D object detection comparisons. Our results are compared to state-of-the-art systems including ORB-SLAM2 [28], DynSLAM [2], Li *et al*. [24] and ClusterSLAM [15] using the TUM metrics [39]. These metrics evaluate ATE, R.RPE and T.RPE, which are short for the Root Mean Square Error (RMSE) of the Absolute Trajectory Error, the Rotational and Translational Relative Pose Error, respectively.

As shown in Table 2, for most of the sequences we achieve the best results in terms of ATE, meaning that our method can maintain globally correct camera trajectories in challenging scenes (*e.g*. 1003-0047) where even ORB-SLAM2 fails due to its static scene assumption. Although DynSLAM maintains a dense mapping of both the static scenes and dynamic objects, the underlying sparse scene flow estimation is based on a frame-to-frame visual odometry *libviso* [11], which will inherently lead to remarkable drift over long travel distances. The batch Multibody SfM formulation of Li *et al*. results in a highly nonlinear factor graph optimization problem whose solution is not trivial. ClusterSLAM [15] requires associated landmarks and the inaccurate feature tracking frontend affects the localization performance even if the states are solved via full optimization. In contrast, our ClusterVO achieves comparable or even better results than all previous methods due to the fusing of multiple sources of information and the robust sliding-window optimization.

The cluster trajectories are evaluated in 3D object detection benchmark in KITTI tracking dataset. We compute the Average Precision (AP) of the 'car' class in both bird view ($\mathrm{AP}_{\mathrm{bv}}$) and 3D view ($\mathrm{AP}_{\mathrm{3D}}$). Our detected 3D box center is $\boldsymbol{c}_{\mathrm{q}}$ (in Eq. 7) and the dimension is taken as the average car size. The box orientation is initialized to be vertical to the camera and tracked over time later on. The detection is counted as a true positive if the Intersection over Union (IoU) score with an associated ground-truth detection is larger than 0.25. All ground-truth 3D detections are divided into three categories (Easy, Moderate and Hard) based on the height of 2D reprojected bounding box and the occlusion/truncation level.

Table 3. 3D object detection comparison on KITTI dataset.

| | $\mathrm{AP}_{\mathrm{bv}}$ | | | $\mathrm{AP}_{\mathrm{3D}}$ | | | Time |
|---|---|---|---|---|---|---|---|
| | Easy | Moderate | Hard | Easy | Moderate | Hard | (ms) |
| Chen *et al*. [7] | **81.34** | **70.70** | **66.32** | **80.62** | **70.01** | **65.76** | 1200 |
| DynSLAM [2] | 71.83 | 47.16 | 40.30 | 64.51 | 43.70 | 37.66 | 500 |
| ClusterVO | 74.65 | 49.65 | 42.65 | 55.85 | 38.93 | 33.55 | **125** |

We compare the performance of our method with the state-of-the-art 3D object detection solution from Chen *et al*. [7] and DynSLAM [2]. The evaluation is performed in camera coordinate system so the inaccuracies in ego-motion estimations are eliminated.

The methods of Chen *et al*. and DynSLAM are similar in that they both perform a dense stereo matching (*e.g*. [47]) to precompute the 3D structure. While DynSLAM crops the depth map using 2D detections to generate spatial detections, Chen *et al*. generates and scores object proposals directly in 3D space incorporating many scene priors in autonomous driving scenarios including the ground plane and car dimension prior. These priors are justified to be critical comparing the results in Table 3: DynSLAM witnesses a sharp decrease in both Moderate and Hard categories which contain faraway cars and small 2D detection bounding boxes.

In the case of ClusterVO, which is designed to be general-purpose, the natural uncertainty of stereo triangulation becomes larger when the landmark becomes distant from the camera without object size priors. Also, we do not detect the canonical direction (*i.e*., the front of the car) of the cluster if its motion is small, so the orientation can be imprecise as well. This explains the gap in detecting hard

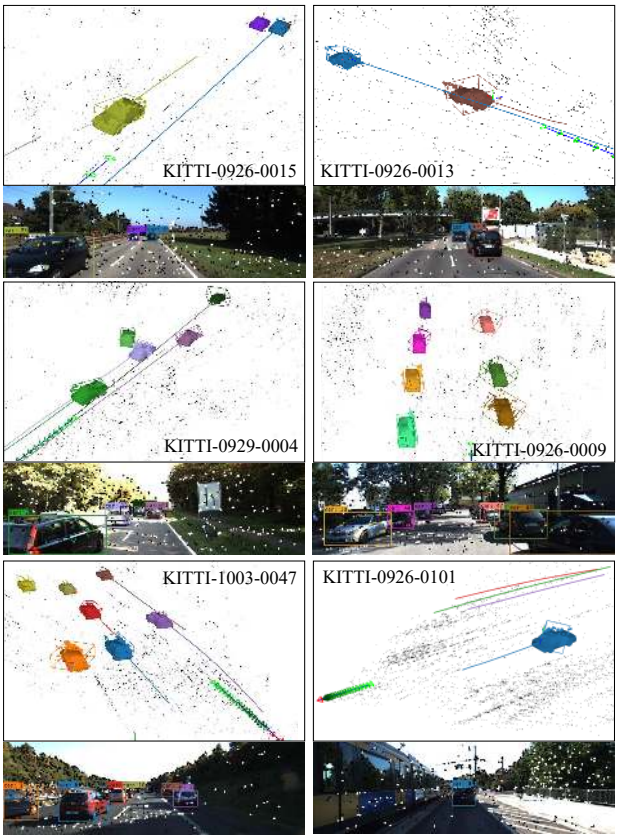



Figure 7. Qualitative results on KITTI raw dataset. The image below each sequence shows the input image and detections of the most recent frame.

examples between ours and a specialized system like [7]. Compared to DynSLAM, the average precision improves because ClusterVO is able to track the moving object over time consistently and predicts their motions even if the 2D detection network misses some targets. Additionally, we emphasize the high efficiency of ClusterVO system by comparing the time cost in Table 1 while the work of Chen *et al*. requires 1.2 seconds for each stereo input pair.

Some qualitative results of KITTI raw dataset are shown in Figure 7. We refer our readers to the supplementary video for animated results.

### 4.4. Ablation study

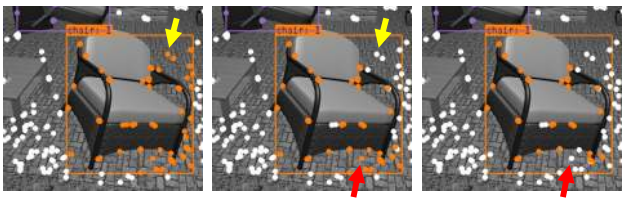
We test the importance of each probabilistic term in our Heterogeneous CRF formulation (Eq. 5) using synthesized motion dataset rendered from SUNCG [37]. Following the same stereo camera parameter as in [15], we generate 4 indoor sequences with moving chairs and balls, and compare the accuracies of ego motion and cluster motions in Table 4.

By gradually adding different terms of Eq. 5 into the system, our performance on estimating cluster motions improves especially in terms of absolute trajectory error (decreases by 45.8% compared to 2D only CRF) while the accuracy of ego motion is not affected. This is due to the more accurate moving object clustering combining both geometric and semantic cues. It should be noted that our results are even comparable to the most recent ClusterSLAM [15], a backend method with full batched Bundle Adjustment optimization: This shows that incorporating semantic information into the motion detection problem helps effectively regularize the solution and achieves more consistent trajectory estimation. Figure 8 visualizes this effect further by computing the classification result based only on the unary term $\psi_u$. Some mis-classified landmarks are successfully filtered out by incorporating more information.

Table 4. Ablation comparisons on SUNCG dataset in terms of ego-motion and cluster trajectories.

| | Ego Motion* | | Cluster Motion | |
|---|---|---|---|---|
| | ATE | R./T.RPE | ATE | R./T.RPE |
| ORB-SLAM2 [28] | **0.35** | 0.14/**0.59** | - | - |
| DynSLAM [2] | 54.07 | 11.07/49.24 | 0.26 | 1.23/0.59 |
| ClusterSLAM [15] | 1.34 | 0.41/1.89 | 0.17 | 0.34/**0.30** |
| ClusterVO 2D | 0.62 | 0.19/0.95 | 0.24 | **0.31**/0.53 |
| ClusterVO 2D+3D | 0.52 | **0.11**/0.87 | 0.15 | 0.50/0.53 |
| ClusterVO Full | 0.61 | 0.19/0.91 | **0.13** | 0.37/0.36 |

* Values are multiplied by 100.



(a) (b) (c)

Figure 8. Unary term visualizations on one indoor sequence from SUNCG dataset. (a) ClusterVO 2D; (b) ClusterVO 2D+3D; (c) ClusterVO Full.

## 5. Conclusion

In this paper we present ClusterVO, a general-purpose fast stereo visual odometry for simultaneous moving rigid body clustering and motion estimation. Comparable results to state-of-the-art solutions on both camera ego-motion and dynamic objects pose estimation demonstrate the effectiveness of our system. In the future, one direction would be to incorporate specific scene priors as pluggable components to improve ClusterVO performance on specialized applications (*e.g*. autonomous driving); another direction is to fuse information from multiple sensors to further improve localization accuracy.

**Acknowledgements.** We thank anonymous reviewers for the valuable discussions. This work was supported by the Natural Science Foundation of China (Project Number 61521002, 61902210), the Joint NSFC-DFG Research Program (Project Number 61761136018) and Research Grant of Tsinghua-Tencent Joint Laboratory for Internet Innovation Technology.

## References


[1] Timothy D Barfoot. *State Estimation for Robotics*. Cambridge University Press, 2017. 5

[2] Ioan Andrei Bârsan, Peidong Liu, Marc Pollefeys, and Andreas Geiger. Robust dense mapping for large-scale dynamic environments. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 7510–7517. IEEE, 2018. 1, 2, 7, 8

[3] Aseem Behl, Omid Hosseini Jafari, Siva Karthik Mustikovela, Hassan Abu Alhaija, Carsten Rother, and Andreas Geiger. Bounding boxes, segmentations and object coordinates: How important is recognition for 3d scene flow estimation in autonomous driving scenarios? In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 2574–2583, 2017. 2

[4] Berta Bescos, José M Fácil, Javier Civera, and José Neira. Dynaslam: Tracking, mapping, and inpainting in dynamic scenes. *IEEE Robotics and Automation Letters*, 3(4):4076–4083, 2018. 1, 2

[5] Goutam Bhat, Joakim Johnander, Martin Danelljan, Fahad Shahbaz Khan, and Michael Felsberg. Unveiling the power of deep tracking. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 483–498. Springer, 2018. 1, 2

[6] Sergio Caccamo, Esra Ataer-Cansizoglu, and Yuichi Taguchi. Joint 3d reconstruction of a static scene and moving objects. In *Proceedings of the International Conference on 3D Vision*, pages 677–685. IEEE, 2017. 1, 2

[7] Xiaozhi Chen, Kaustav Kundu, Yukun Zhu, Huimin Ma, Sanja Fidler, and Raquel Urtasun. 3d object proposals using stereo imagery for accurate object class detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 40(5):1259–1272, 2017. 7, 8

[8] Weichen Dai, Yu Zhang, Ping Li, and Zheng Fang. Rgb-d slam in dynamic environments using points correlations. *arXiv preprint arXiv:1811.03217*, 2018. 2

[9] Kevin Eckenhoff, Yulin Yang, Patrick Geneva, and Guoquan Huang. Tightly-coupled visual-inertial localization and 3-d rigid-body target tracking. *IEEE Robotics and Automation Letters*, 4(2):1541–1548, 2019. 2

[10] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3354–3361. IEEE, 2012. 6

[11] Andreas Geiger, Julius Ziegler, and Christoph Stiller. Stereoscan: Dense 3d reconstruction in real-time. In *Proceedings of the IEEE Intelligent Vehicles Symposium (IV)*, pages 963–968. Ieee, 2011. 7

[12] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 1440–1448, 2015. 2

[13] Alexander Grabner, Peter M Roth, and Vincent Lepetit. Gp2c: Geometric projection parameter consensus for joint 3d pose and focal length estimation in the wild. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 2222–2231, 2019. 2

[14] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 2961–2969, 2017. 2

[15] Jiahui Huang, Sheng Yang, Zishuo Zhao, Yu-Kun Lai, and Shi-Min Hu. ClusterSLAM: A slam backend for simultaneous rigid body clustering and motion estimation. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 5875–5884, 2019. 1, 2, 4, 7, 8

[16] Kevin M Judd and Jonathan D Gammell. Occlusion-robust mvo: Multimotion estimation through occlusion via motion closure. *arXiv preprint arXiv:1905.05121*, 2019. 6

[17] Kevin Michael Judd and Jonathan D Gammell. The oxford multimotion dataset: Multiple se3 motions with ground truth. *IEEE Robotics and Automation Letters*, 4(2):800–807, 2019. 5, 6

[18] Kevin M Judd, Jonathan D Gammell, and Paul Newman. Multimotion visual odometry (mvo): Simultaneous estimation of camera and third-party motions. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3949–3956. IEEE, 2018. 6

[19] M. Keller, D. Lefloch, M. Lambers, S. Izadi, T. Weyrich, and A. Kolb. Real-time 3d reconstruction in dynamic scenes using point-based fusion. In *Proceedings of the International Conference on 3D Vision*, pages 1–8, 2013. 1, 2

[20] Deok-Hwa Kim and Jong-Hwan Kim. Effective background model-based rgb-d dense visual odometry in a dynamic environment. *IEEE Transactions on Robotics*, 32(6):1565–1573, 2016. 1, 2

[21] Philipp Krähenbühl and Vladlen Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. In *Advances in Neural Information Processing Systems*, pages 109–117, 2011. 4

[22] Suren Kumar, Vikas Dhiman, Madan Ravi Ganesh, and Jason J Corso. Spatiotemporal articulated models for dynamic slam. *arXiv preprint arXiv:1604.03526*, 2016. 2

[23] Peiliang Li, Xiaozhi Chen, and Shaojie Shen. Stereo r-cnn based 3d object detection for autonomous driving. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7644–7652, 2019. 2

[24] Peiliang Li, Tong Qin, et al. Stereo vision-based semantic 3d object and ego-motion tracking for autonomous driving. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 646–661. Springer, 2018. 1, 2, 6, 7

[25] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 21–37. Springer, 2016. 2

[26] Jonathon Luiten, Tobias Fischer, and Bastian Leibe. Track to reconstruct and reconstruct to track. *IEEE Robotics and Automation Letters*, 5(2):1803–1810, 2020. 1, 2

[27] Moritz Menze and Andreas Geiger. Object scene flow for autonomous vehicles. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3061–3070, 2015. 2

[28] Raul Mur-Artal and Juan D Tardós. ORB-SLAM2: An open-source slam system for monocular, stereo, and rgb-d

cameras. *IEEE Transactions on Robotics*, 33(5):1255–1262, 2017. 1, 2, 3, 4, 7, 8

[29] Gokul B Nair, Swapnil Daga, Rahul Sajnani, Anirudha Ramesh, Junaid Ahmed Ansari, and K Madhava Krishna. Multi-object monocular slam for dynamic environments. *arXiv preprint arXiv:2002.03528*, 2020. 2

[30] Richard A Newcombe, Dieter Fox, and Steven M Seitz. Dynamicfusion: Reconstruction and tracking of non-rigid scenes in real-time. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 343–352, 2015. 2

[31] Kejie Qiu, Tong Qin, Wenliang Gao, and Shaojie Shen. Tracking 3-d motion of dynamic objects using monocular visual-inertial sensing. *IEEE Transactions on Robotics*, 35(4):799–816, 2019. 2

[32] Joseph Redmon and Ali Farhadi. Yolo9000: better, faster, stronger. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7263–7271, 2017. 1, 2, 6

[33] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski. ORB: An efficient alternative to SIFT or SURF. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 2564–2571, 2011. 2

[34] Martin Rünz and Lourdes Agapito. Co-fusion: Real-time segmentation, tracking and fusion of multiple objects. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 4471–4478, May 2017. 1, 2

[35] Martin Runz, Maud Buffier, and Lourdes Agapito. Maskfusion: Real-time recognition, tracking and reconstruction of multiple moving objects. In *Proceedings of the IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 10–20. IEEE, 2018. 1, 2

[36] Sarthak Sharma, Junaid Ahmed Ansari, J Krishna Murthy, and K Madhava Krishna. Beyond pixels: Leveraging geometry and shape cues for online multi-object tracking. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 3508–3515. IEEE, 2018. 1, 2

[37] Shuran Song, Fisher Yu, Andy Zeng, Angel X Chang, Manolis Savva, and Thomas Funkhouser. Semantic scene completion from a single depth image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1746–1754, 2017. 8

[38] Michael Strecke and Jorg Stuckler. Em-fusion: Dynamic object-level slam with probabilistic data association. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 5865–5874, 2019. 1, 2

[39] Jürgen Sturm, Nikolas Engelhard, Felix Endres, Wolfram Burgard, and Daniel Cremers. A benchmark for the evaluation of rgb-d slam systems. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 573–580. IEEE, 2012. 7

[40] Andrea Tagliasacchi, Matthias Schröder, Anastasia Tkach, Sofien Bouaziz, Mario Botsch, and Mark Pauly. Robust articulated-icp for real-time hand tracking. In *Computer Graphics Forum*, volume 34, pages 101–114. Wiley Online Library, 2015. 2

[41] Keisuke Tateno, Federico Tombari, and Nassir Navab. Real-time and scalable incremental segmentation on dense slam. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4465–4472. IEEE, 2015. 4

[42] Bill Triggs, Philip F McLauchlan, Richard I Hartley, and Andrew W Fitzgibbon. Bundle adjustment—a modern synthesis. In *International workshop on vision algorithms*, pages 298–372. Springer, 1999. 5

[43] Dimitrios Tzionas and Juergen Gall. Reconstructing articulated rigged models from rgb-d videos. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 620–633. Springer, 2016. 2

[44] Chieh-Chih Wang, Charles Thorpe, Sebastian Thrun, Martial Hebert, and Hugh Durrant-Whyte. Simultaneous localization, mapping and moving object tracking. *The International Journal of Robotics Research*, 26(9):889–916, 2007. 2

[45] Lingzhu Xiang, Zhile Ren, Mengrui Ni, and Odest Chadwicke Jenkins. Robust graph slam in dynamic environments with moving landmarks. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2543–2549. IEEE, 2015. 2

[46] Binbin Xu, Wenbin Li, Dimos Tzoumanikas, Michael Bloesch, Andrew Davison, and Stefan Leutenegger. Midfusion: Octree-based object-level multi-instance dynamic slam. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 5231–5237. IEEE, 2019. 1, 2

[47] Koichiro Yamaguchi, David McAllester, and Raquel Urtasun. Efficient joint segmentation, occlusion labeling, stereo and flow estimation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 756–771. Springer, 2014. 7

[48] Shichao Yang and Sebastian Scherer. Cubeslam: Monocular 3-d object slam. *IEEE Transactions on Robotics*, 35(4):925–938, 2019. 1

[49] Chao Yu, Zuxin Liu, Xin-Jun Liu, Fugui Xie, Yi Yang, Qi Wei, and Qiao Fei. Ds-slam: A semantic visual slam towards dynamic environments. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1168–1174. IEEE, 2018. 2

[50] Fangwei Zhong, Sheng Wang, Ziqi Zhang, and Yizhou Wang. Detect-slam: Making object detection and slam mutually beneficial. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1001–1010. IEEE, 2018. 2