DVLO: Deep Visual-LiDAR Odometry with Local-to-Global Feature Fusion and Bi-Directional Structure Alignment

Jiuming Liu[1], Dong Zhuo[1], Zhiheng Feng[1], Siting Zhu[1]
Chensheng Peng[2], Zhe Liu[3], and Hesheng Wang[1]*

[1] Department of Automation, Shanghai Jiao Tong University
[2] University of California, Berkeley
[3] MoE Key Lab of Artificial Intelligence, Shanghai Jiao Tong University
{liujiuming, paryi555, wanghesheng}@sjtu.edu.cn

Abstract. Information inside visual and LiDAR data is well complementary derived from the fine-grained texture of images and massive geometric information in point clouds. However, it remains challenging to explore effective visual-LiDAR fusion, mainly due to the intrinsic data structure inconsistency between two modalities: Image pixels are regular and dense, but LiDAR points are unordered and sparse. To address the problem, we propose a local-to-global fusion network (DVLO) with bi-directional structure alignment. To obtain locally fused features, we project points onto the image plane as cluster centers and cluster image pixels around each center. Image pixels are pre-organized as pseudo points for image-to-point structure alignment. Then, we convert points to pseudo images by cylindrical projection (point-to-image structure alignment) and perform adaptive global feature fusion between point features and local fused features. Our method achieves state-of-the-art performance on KITTI odometry and FlyingThings3D scene flow datasets compared to both single-modal and multi-modal methods. Codes are released at https://github.com/IRMVLab/DVLO.

Keywords: Visual-LiDAR Odometry · Multi-Modal Fusion · Local-to-Global Fusion · Bi-Directional Structure Alignment

1 Introduction

Visual/LiDAR odometry is a fundamental task in the field of computer vision and robotics, which estimates the relative pose transformation between two consecutive images or point clouds. It is widely applied in autonomous driving [19,55,62,71], SLAM [8,9,61,72], navigation [33–35] etc. Recently, the multi-modal odometry [15, 57, 73] has gained increasing attention because it can take advantage of complementary information from different modalities and possess the strong robustness for asymmetric sensor degradation [10].

* Corresponding author. The first two authors contribute equally.

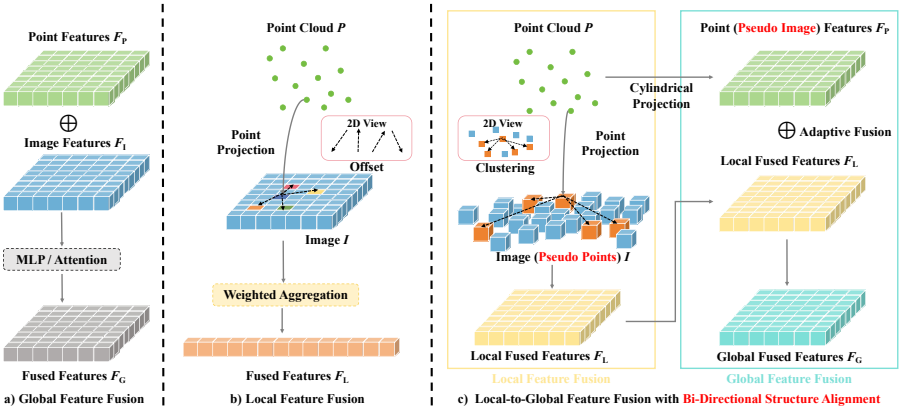


Fig. 1: Different fusion strategies for images and points. Most previous works only perform the fusion globally [53] or locally [73]. Our DVLO designs a local-to-global fusion strategy that facilitates the interaction of global information while preserving local fine-grained information. Furthermore, a bi-directional structure alignment is designed to maximize the inter-modality complementarity.

Previous visual-LiDAR odometry works can be classified into two categories: traditional methods [15, 16, 57, 64] and learning-based methods [51, 53, 73]. Traditional methods accomplish the odometry task through a pipeline consisting of feature extraction, frame-to-frame feature matching, motion estimation, and optimization [64]. However, these methods suffer from inaccurate pose estimation because of the poor quality and low resolution of extracted features [57]. With the development of deep learning, some methods [17, 53] attempt to utilize CNN-based methods for visual-LiDAR fusion and pose estimation. However, the receptive field of feature fusion is limited by the stride and kernel size of CNN. To enlarge the receptive field for multi-modal fusion, attention-based methods are recently proposed, which leverage the cross-attention mechanism for the multi-modal fusion [51, 73]. Attention-based methods can fuse multi-modal features globally and establish the cross-frame association with larger receptive fields because of their long-range dependencies. However, due to the quadratic computational complexity, attention-based methods commonly require larger computational consumption and longer inference time [37], which challenges the real-time applications [51]. Moreover, previous learning-based methods mostly adopt the only feature-level fusion strategy as illustrated in Fig. 1 a), which fails to capture fine-grained pixel-to-point correspondences [29]. Recently, some networks [27, 73] design point-to-image projection and local feature aggregation as in Fig. 1 b). However, their performances are still limited by the intrinsic data structure misalignment between sparse LiDAR points and dense camera pixels [29].

To address these problems, we propose a novel local-to-global fusion network (DVLO) with bi-directional structure alignment in Fig. 1 c). Our fusion module consists of two parts: 1) An image is first viewed as a set of pseudo points

inspired by [38] for fine-grained local fusion with LiDAR points (*image-to-point structure alignment*). 2) Point clouds are also converted into pseudo images by cylindrical projection for global adaptive fusion (*point-to-image structure alignment*). Specifically, a novel clustering-based local fusion module (Local Fuser) is designed to perform local fine-grained feature fusion. We first project LiDAR points onto the image plane based on their coordinate calibration matrices to find the corresponding pixels as cluster centers. Meanwhile, for the image-to-point structure alignment, image pixels are reshaped as a set of pseudo points [38]. Then, within a certain scope of each cluster center, we aggregate pseudo point features to generate the local fused features based on the similarities to the cluster centers dynamically. In the global fusion module, we project point clouds onto the cylindrical surface to obtain pseudo images. Then, an adaptive fusion mechanism is leveraged to merge the above local fused image features and point (pseudo image) features for global fusion. It is worth noting that our fusion module is hierarchically utilized at multi-scale feature maps between images and points. The local fusion module can provide more fine-grained point-to-pixel correspondence information, while the global fusion has a larger receptive field and achieves more global information interaction.

Overall, our contributions are as follows:

1. We propose a local-to-global fusion odometry network with bi-directional structure alignment. We cluster image pixels viewed as a set of pseudo points for local fusion with LiDAR points. Point clouds are also converted into pseudo images through cylindrical projection for global adaptive fusion.
2. A pure clustering-based fusion module is designed to obtain the fine-grained local fused features. To the best of our knowledge, our method is the first deep clustering-based multi-modal fusion attempt, serving as an effective and efficient fusion strategy alternative apart from CNN and transformer.
3. Extensive experiments on the KITTI odometry dataset [13, 14] demonstrate that our method outperforms all recent deep LiDAR, visual, and visual-LiDAR fusion odometry works on most sequences. Furthermore, our fusion strategy can generalize well to other multi-modal tasks, like scene flow estimation, even surpassing recent SOTA method CamLiRAFT [29].

2 Related Work

Deep Visual Odometry. Recently, learning-based methods have shown impressive performance in the visual odometry field [21, 22, 40, 45, 56]. The pioneering work [22] uses deep neural networks for odometry estimation with the prediction of both speed and direction for individual images. PoseNet [21] initially employs Convolution Neural Networks (CNNs) to extract features from the input image and then estimate the pose. DeepVO [56] applies deep recurrent neural networks to capture the temporal dynamics and interdependency information of sequences, thereby facilitating the estimation of ego-motion. Li *et al.* [25] utilize the knowledge distillation technique based on pre-trained visual-LiDAR odometry as a teacher for guiding the training of the visual odometry. Deng

et al. [10] propose a long-term visual SLAM system with map prediction and dynamics removal. NeRF-VO [40] improves the geometric accuracy of the scene representation by optimizing a set of keyframe poses and the underlying dense geometry through training the radiance field with volume rendering [59, 60].

Deep LiDAR Odometry. In contrast to visual odometry, deep LiDAR odometry is still a challenging task because of the large number, irregularity, and sparsity of raw LiDAR points [5, 54, 69]. Nicolai *et al*. [41] first introduce the deep learning technique into LiDAR odometry. They project 3D LiDAR points onto the 2D plane to obtain 2D depth images, and then employ 2D learning methods for pose estimation. DeepPCO [58] projects point clouds to panoramic depth images and applies two sub-networks to estimate the translation and rotation respectively. LO-Net [26] also converts points to 2D format by projection and uses the normal of each 3D point and dynamical masks to further improve the performance. PWCLO [55] introduces the PWC [50] structure for the LiDAR odometry task, which hierarchically refines estimated poses by an iterative warp-refinement module. EfficientLO [54] proposes a projection-aware operator for improving the efficiency of LiDAR odometry. TransLO [30] designs a window-based masked point transformer to enhance global feature embeddings and remove outliers. DELO [1] introduces partial optimal transportation of LiDAR descriptors and predictive uncertainty for robust pose estimation. NeRF-LOAM [7] applies a neural radiation field to the LiDAR odometry system, showing excellent generalization capabilities across various environments.

Viusal-LiDAR Odometry. Recently, there has been an increasing focus on visual-LiDAR odometry, which takes advantage of both 2D texture and 3D geometric features. Existing visual-LiDAR odometry can be classified into two categories: traditional methods and learning-based methods. For traditional methods, V-LOAM [67] leverages the high-frequency estimated poses from visual odometry as a motion prior for the low-frequency LiDAR odometry, resulting in refined motion estimation. LIMO [15] utilizes depth information derived from LiDAR points to alleviate the scale uncertainty that is intrinsic to monocular visual odometry. PL-LOAM [16] provides a pure visual motion tracking method and a novel scale correction algorithm. DV-LOAM [57] is a SLAM framework including a two-stage direct visual odometry module, a LiDAR mapping module with considerations of dynamic objects, and a parallel global and local search loop closure detection module. SDV-LOAM [64] combines the semi-direct visual odometry with an adaptive sweep-to-map LiDAR odometry to tackle the challenges of 3D-2D depth correlation. For learning-based methods, MVL-SLAM [3] employs the RCNN network architecture, fusing RGB images and multi-channel depth images from 3D LiDAR points. LIP-Loc [48] proposes a pre-training strategy for cross-modal localization, which utilizes contrastive learning to jointly train image and point encoders. However, the problem of natural data structure inconsistency between points and images has not been fully considered before. To the best of our knowledge, our work is the first visual-LiDAR odometry network with bi-directional structure alignment.

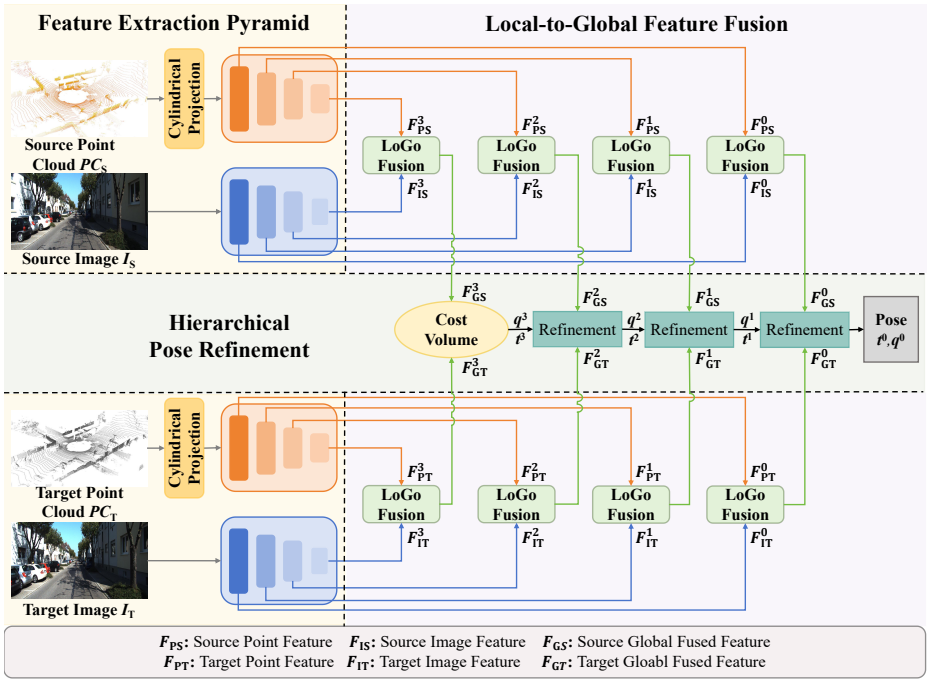


Fig. 2: The pipeline of our proposed DVLO. We propose a novel Local-to-Global (LoGo) fusion module, which consists of a clustering-based Local Fuser and an adaptive Global Fuser. The pose is initially regressed from the cost volume of the coarsest fused features and then refined iteratively from fused features in shallower layers.

3 Methodology

3.1 Overall Architecture

The overall architecture of our proposed DVLO is illustrated in Fig. 2. Given two point clouds PC_S, $PC_T \in \mathbb{R}^{N\times 3}$ and their corresponding monocular camera images I_S, $I_T \in \mathbb{R}^{H\times W\times 3}$ from a pair of consecutive frames, the goal of our odometry is to estimate the relative pose including both rotation quaternion $q \in \mathbb{R}^4$ and translation vector $t \in \mathbb{R}^3$ between two frames.

In detail, we first feed the camera images and LiDAR points into the hierarchical feature extraction module in Sec. 3.2 to obtain multi-level image features and point features. Then, in Sec. 3.3 and Sec. 3.4, we delve into the details of our designed local-to-global fusion module. Finally, poses are estimated and iteratively refined from fused features as in Sec. 3.5.

3.2 Hierarchical Feature Extraction

Point Feature Extraction. Due to the irregularity and sparsity of the raw point clouds, we first project them onto a cylindrical surface [30, 54] to orderly

organize points. Their corresponding 2D positions are:

$$u = \arctan 2\left(y/x\right)/\Delta\theta, \tag{1}$$

$$v = \arcsin\left(z/\sqrt{x^2+y^2+z^2}\right)/\Delta\phi, \tag{2}$$

where x, y, z are the raw 3D coordinates of point cloud. u, v are the corresponding 2D pixel positions on the projected pseudo image. $\Delta\theta$ and $\Delta\phi$ are horizontal and vertical resolutions of the LiDAR sensor, respectively. To make the best use of geometric information of raw 3D points, we fill each projected 2D position with its corresponding original 3D coordinates. In this case, LiDAR points can not only be converted into the pseudo image structure [26] for better alignment and global feature fusion with images in Sec. 3.4, but also retain the original 3D geometric information for effective feature extraction. Then, pseudo images of size $H_P \times W_P \times 3$ in Fig. 2 will be fed into the hierarchical feature extraction module [54] to extract multi-level point features $F_P \in \mathbb{R}^{H_P \times W_P \times D}$, where D is the number of channels of the pseudo image features.

Image Feature Extraction. Given the camera images $I \in \mathbb{R}^{H \times W \times 3}$, we utilize the convolution-based feature pyramid in [17] to extract image features $F_I \in \mathbb{R}^{H_I \times W_I \times C}$, where H_I, W_I are the height and width of the feature map. C is the number of channels of the image features.

3.3 Local Fuser Module

Inspired by Context Clusters [38] which proposes a generic clustering-based visual backbone viewing images as a set of points, we extend it and propose a novel clustering-based feature fusion module (Local Fuser) without any CNN or transformer. The module can locally merge more fine-grained 2D texture from images and geometric features from points within each cluster as shown in Fig. 3. Our clustering-based method also maintains high efficiency, where the total inference time is only half of the attention-based methods as in Tab. 7.

From Image to Pseudo Points. Given image features $F_I \in \mathbb{R}^{H_I \times W_I \times C}$, we first reshape them as a collection of pseudo points $F_{pp} \in \mathbb{R}^{M \times C}$, where $M = H_I \times W_I$ is the number of pseudo points. In this case, images have the same data structure as LiDAR points, which facilitates the local pixel-to-point correspondence establishment and further clustering-based feature aggregation.

Pseudo Point Clustering. We first project LiDAR points onto the image plane to obtain their corresponding 2D coordinates x' and y' in the image coordinate system as cluster centers. The center feature $F_c \in \mathbb{R}^{N \times C}$ is computed by the bilinear interpolation on F_I based on x', y'. Then, we divide all pseudo points into several clusters according to the pair-wise cosine similarities between center features F_c and pseudo point features F_{pp}. Here, we allocate each pseudo point to the most similar center, resulting in N clusters. For efficiency, following Swin Transformer [37], we use the region partition while computing similarity.

Local Feature Aggregation. Following [38], we aggregate all pseudo point features within the same cluster based on the similarities to the cluster center

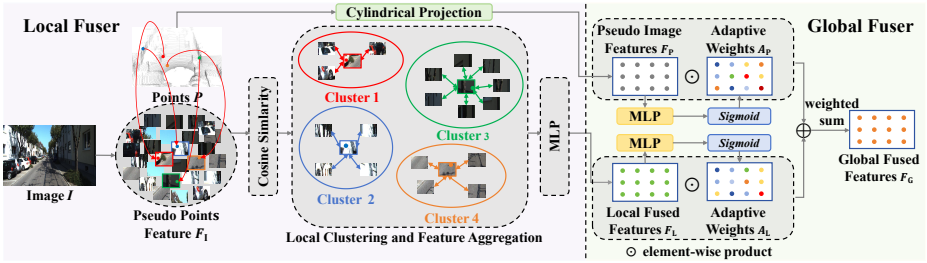


Fig. 3: Our designed Local-to-Global (LoGo) Fusion module. We project points onto the image plane based on the coordinate system transformation matrix as cluster centers and convert the image into a set of pseudo points. Then, we locally aggregate pseudo point features based on the similarities to each cluster center.

dynamically. Given the cluster which contains k pseudo points around the i-th cluster center, the local fused feature $F_L^i \in \mathbb{R}^{1\times C}$ is calculated by:

$$F_L^i = \frac{1}{X}\left(F_c^i + \sum_{j=1}^{k} sigmoid\left(\alpha s_{ij} + \beta\right)\cdot F_{pp}^j\right), \tag{3}$$

$$X = 1 + \sum_{j=1}^{k} sigmoid\left(\alpha s_{ij} + \beta\right), \tag{4}$$

where F_{pp}^j is the j-th pseudo point's feature. s_{ij} is the similarity score between the j-th pseudo point and i-th cluster center. α and β are learnable scalars to scale and shift the similarity. $sigmoid\left(\cdot\right)$ is a sigmoid function to re-scale the similarity to $(0, 1)$. X is the normalization factor. Since we project LiDAR points onto the image plane as cluster centers and aggregate features for each center, local fused feature $F_L \in \mathbb{R}^{N\times C}$ has the same dimension with original LiDAR points. Therefore, we can also reshape the local fused feature F_L like a pseudo image with the size of $H_P \times W_P \times C$ as the input of the Global Fuser module.

3.4 Global Fuser Module

Since local feature fusion is only conducted within a partitioned region, the above Local Fuser module has a limited receptive field. To expand the receptive field for sufficient feature fusion, we introduce the global adaptive fusion mechanism [43] between local fused feature F_L and point (pseudo image) feature F_P as shown in Fig. 3. Our Global Fuser can enable the global information interaction between two modalities, which facilitates to recognize dynamics and occlusion because they introduce inconsistent global motions [31].

From Points to Pseudo Image. We convert the sparse LiDAR points to structured pseudo images by the cylindrical projection in Sec. 3.2. In this case, point feature F_P has the size of $H_P \times W_P \times D$. This process reorganizes the origi-

nally sparse unstructured points as dense structured pseudo images, enabling the following dense feature map fusion with image features.

Adaptive Fusion. Given the local fused features $F_L \in \mathbb{R}^{H_P \times W_P \times C}$ and point features $F_P \in \mathbb{R}^{H_P \times W_P \times D}$, we perform the adaptive global fusion as:

$$A_P = sigmoid(MLP(F_P)), \tag{5}$$

$$A_L = sigmoid(MLP(F_L)), \tag{6}$$

$$F_G = \frac{A_P \odot F_P + A_L \odot F_L}{A_p + A_L}, \tag{7}$$

where A_P and A_L are the adaptive weights for point (pseudo image) features and local fused features, which are obtained by the sigmoid function and MLP layers. $\odot$ represents the element-wise product. We then reshape the global fused feature F_G back to the size of $N \times D$ as the input of the iterative pose estimation.

3.5 Iterative Pose Estimation

Following [54, 55], we use the attentive cost volume to generate the coarse embedding features $E \in \mathbb{R}^{N \times D}$ by associating global fused features F^3_{GS} and F^3_{GT} of two frames in the coarsest layer. The embedding features contain the correlation information between two consecutive frames. Then, we utilize the weighting embedding mask M on the embedding features E to regress the pose transformation. The weighting embedding mask M is calculated by:

$$M = softmax(MLP(E \oplus F^3_{GS})), \tag{8}$$

where $M \in \mathbb{R}^{N \times D}$ is the learnable masks. $F^3_{GS} \in \mathbb{R}^{N \times D}$ is the global fused features in the source frame. Then, the quaternion $q \in \mathbb{R}^4$ and translation vector $t \in \mathbb{R}^3$ are generated by weighting embedding features and FC layers:

$$q = \frac{FC(E \odot M)}{|FC(E \odot M)|}, \tag{9}$$

$$t = FC(E \odot M). \tag{10}$$

After the initial estimation q and t, we refine them by the iterative refinement module in [55] to get the final pose. The refined quaternion q^l and translation vector t^l of the l layer can be calculated by:

$$q^l = \Delta q^l q^{l+1}, \tag{11}$$

$$[0, t^l] = \Delta q^l [0, t^{l+1}](\Delta q^l)^{-1} + [0, \Delta t^l], \tag{12}$$

where pose residuals Δq^l and Δt^l can be obtained by the similar process in the coarsest layer following Eq. (9) and Eq. (10).

Table 1: Comparison with different odometry networks on the KITTI odometry dataset [13]. t_{rel} and r_{rel} mean the average sequence translational RMSE (%) and the average sequence rotational RMSE (°/100m) respectively on 00-10 subsequences in the length of 100, 200, ..., 800m. The best results are **bold**, and the second best results are underlined. * represents the model is trained on the 00-08 sequences.

Method	00		01		02		03		04		05		06		07		08		09		10		mean (07-10)	
	t_{rel}	r_{rel}	t_{rel}	r_{rel}	t_{rel}	r_{rel}	t_{rel}	r_{rel}	t_{rel}	r_{rel}	t_{rel}	r_{rel}	t_{rel}	r_{rel}	t_{rel}	r_{rel}	t_{rel}	r_{rel}	t_{rel}	r_{rel}	t_{rel}	r_{rel}	t_{rel}	r_{rel}
Visual Odometry Methods:																								
SfMLearner* [70]	21.32	6.19	22.41	2.79	24.10	4.18	12.56	4.52	4.32	3.28	12.99	4.66	15.55	5.58	12.61	6.31	10.66	3.75	11.32	4.07	15.25	4.06	12.46	4.55
MLM-SFM* [49]	2.04	0.48	–	–	1.50	0.35	3.37	**0.21**	1.43	0.23	2.19	0.38	2.09	0.81	–	–	2.37	0.44	1.76	0.47	2.12	0.85	2.08	0.59
DFVO* [65]	2.01	0.79	61.17	18.96	2.46	0.79	3.27	0.89	0.79	0.56	1.50	0.74	1.95	0.76	2.28	1.16	2.11	0.74	3.21	0.59	2.89	0.97	2.62	0.87
Cho *et al.* * [6]	1.77	0.79	64.38	16.87	2.62	0.74	3.06	0.89	0.65	0.55	1.31	0.74	1.60	0.56	1.06	0.67	2.28	0.76	2.66	0.53	2.95	0.95	2.24	0.73
LiDAR Odometry Methods:																								
LO-Net [26]	1.47	0.72	1.36	0.47	1.52	0.71	1.03	0.66	0.51	0.65	1.04	0.69	0.71	0.50	1.70	0.89	2.12	0.77	1.37	0.58	1.80	0.93	1.75	0.79
PWCLO [55]	0.89	0.43	1.11	0.42	1.87	0.76	1.42	0.92	1.15	0.94	1.34	0.71	0.60	0.38	1.16	1.00	1.68	0.72	0.88	0.46	2.14	0.71	1.47	0.72
DELO [1]	1.43	0.81	2.19	0.57	1.48	0.52	1.38	1.10	2.45	1.70	1.27	0.64	0.83	0.35	0.58	0.41	1.36	0.64	1.23	0.57	1.53	0.90	1.18	0.63
TransLO [30]	0.85	0.38	1.16	0.45	0.88	0.34	1.00	0.71	0.34	0.18	0.63	0.41	0.73	0.31	0.55	0.43	1.29	0.50	0.95	0.46	1.18	0.61	0.99	0.50
EfficientLO [54]	**0.80**	0.37	0.91	0.40	0.94	0.32	**0.51**	0.43	0.38	0.30	0.57	0.33	0.36	0.23	**0.37**	**0.26**	1.22	0.48	0.87	0.38	0.91	0.50	0.86	**0.41**
Multimodal Odometry Methods:																								
An *et al.* * [3]	2.53	0.79	3.76	0.80	3.95	1.05	2.75	1.39	1.81	1.48	3.49	0.79	1.84	0.83	3.27	1.51	2.75	1.61	3.70	1.83	4.65	0.51	3.59	1.37
H-VLO* [4]	1.75	0.62	43.2	0.46	2.32	0.60	2.52	0.47	0.73	0.36	0.85	0.35	0.75	0.30	0.79	0.48	1.35	**0.38**	1.89	**0.34**	1.39	0.52	1.36	0.43
Ours	**0.80**	**0.35**	**0.85**	**0.33**	**0.81**	**0.29**	0.59	0.36	**0.26**	**0.13**	**0.41**	**0.23**	**0.33**	**0.17**	0.46	0.33	**1.09**	0.44	**0.85**	0.36	**0.88**	**0.46**	**0.82**	**0.41**

3.6 Loss Function

Our network outputs q^l and t^l from four layers will be involved to calculate the supervised loss $\mathcal{L}^l$ [54, 55]. The training loss function of l-th layer is:

$$\mathcal{L}^l = \left\| t_{gt} - t^l \right\| exp(-k_x) + k_x + \left\| q_{gt} - q^l \right\|_2 exp(-k_q) + k_q, \tag{13}$$

where t_{gt} and q_{gt} are the ground truth translation and quaternion, respectively. k_x and k_q are the learnable scalars to scale the loss. $\|\cdot\|$ and $\|\cdot\|_2$ are the L_1 and L_2 norm, respectively. Then, the total training loss is:

$$\mathcal{L} = \sum_{l=1}^{L} \alpha^l \mathcal{L}^l, \tag{14}$$

where L is the total number of layers (set as 4), and α^l is a hyperparameter representing the weight of l layer.

4 Experiment

4.1 KITTI Odometry Dataset

We evaluate our DVLO on the KITTI odometry dataset [13], which is a widely used benchmark for the evaluation of odometry and SLAM system. The dataset consists of 22 sequences of LiDAR point clouds and their corresponding stereo images. In this paper, we only use the monocular left camera image for the fusion with the LiDAR sensor. Since the ground truth pose (trajectory) is only available for sequences 00-10, we utilize these sequences for training and testing.

Table 2: Comparison with traditional visual-LiDAR odometry on KITTI 00-10 sequences. Our DVLO is trained on 00-06 sequences. The best results for each sequence are **bold**, and the second best results are underlined.

Method	00 t_{rel}	01 t_{rel}	02 t_{rel}	03 t_{rel}	04 t_{rel}	05 t_{rel}	06 t_{rel}	07 t_{rel}	08 t_{rel}	09 t_{rel}	10 t_{rel}	Mean(00-10) t_{rel}
V-LOAM [67]	–	–	–	–	–	–	–	–	–	1.74	1.01	1.38
DVL-SLAM [46]	0.93	1.47	1.11	0.92	0.67	0.82	0.92	1.26	1.32	**0.66**	**0.70**	0.98
PL-LOAM [16]	0.99	1.87	1.38	0.65	0.42	0.72	0.61	0.56	1.27	1.06	0.83	0.94
Ours	**0.80**	**0.85**	**0.81**	**0.59**	**0.26**	**0.41**	**0.33**	**0.46**	**1.09**	0.85	0.88	**0.67**

Table 3: Comparison with learning-based multi-modal odometry on KITTI 09-10 sequences. Our DVLO is trained on 00-06 sequences while other models are trained on 00-08 sequences. The best results are **bold**, and the second best results are underlined.

Method	Modalities	09		10		Mean (09-10)	
		t_{rel}	r_{rel}	t_{rel}	r_{rel}	t_{rel}	r_{rel}
Self-VLO [24]	visual+LiDAR	2.58	1.13	2.67	1.28	2.62	1.21
H-VLO [4]	visual+LiDAR	1.89	**0.34**	1.39	0.52	1.64	0.43
SelfVIO [2]	visual+inertial	1.95	1.15	1.81	1.30	1.88	1.23
VIOLearner [44]	visual+inertial	1.82	1.08	1.74	1.38	1.78	1.23
Ours	visual+LiDAR	**0.85**	0.36	**0.88**	**0.46**	**0.87**	**0.41**

4.2 Implementation Details

Data Preprocessing. We directly input all LiDAR points without downsampling. The projected pseudo image size is set in line with the range of the LiDAR sensor as 64×1800. We pad the camera images to a uniform size of 384×1280. Since there is a large spatial range difference between the camera and LiDAR, we design a fusion mask to indicate which point can be fused with the image.
Parameters. Experiments are conducted on an NVIDIA RTX 4090 GPU with PyTorch 1.10.1. We use Adam optimizer with $\beta_1 = 0.9$, $\beta_2 = 0.999$. The initial learning rate is set to 0.001 and exponentially decays every 200000 steps until 0.00001. Batch size is 8. α^l for four layers are 1.6, 0.8, 0.4, and 0.2. Initial values of learnable parameters k_x and k_q are set as 0.0 and -2.5, respectively.
Evaluation Metrics. We follow protocols of PWCLO [55] to evaluate our method with two metrics: (1) Average sequence translational RMSE (%). (2) Average sequence rotational RMSE (°/100m).

4.3 Quantitative Results

Comparison with Visual/LiDAR Odometry. We compare our method with some representative visual odometry (VO) or LiDAR odometry (LO) networks for the comprehensive comparison. Following the settings in [55], we train our model on 00-06 sequences. Quantitative results on the KITTI dataset are listed in Tab. 1, which shows that our DVLO outperforms all these works on most sequences. Compared with deep visual odometry, *e.g.*, DFVO [65] and Cho *et al.* [6], our method's mean errors t_{rel} and r_{rel} on sequence 07-10 have a 63.4% and a 43.8% decline, respectively. Notably, even though these VO methods are mostly trained on larger data (00-08), our method still outperforms them by a large

Table 4: Average inference time of different multi-modal odometry methods on the sequence 07-10 of KITTI odometry dataset.

Method	DV-LOAM [57]	PL-LOAM [16]	OKVIS-S [23]	Shu *et al.* [47]	Ours
Inference Time	167 ms	200 ms	143 ms	100ms	98.5 ms

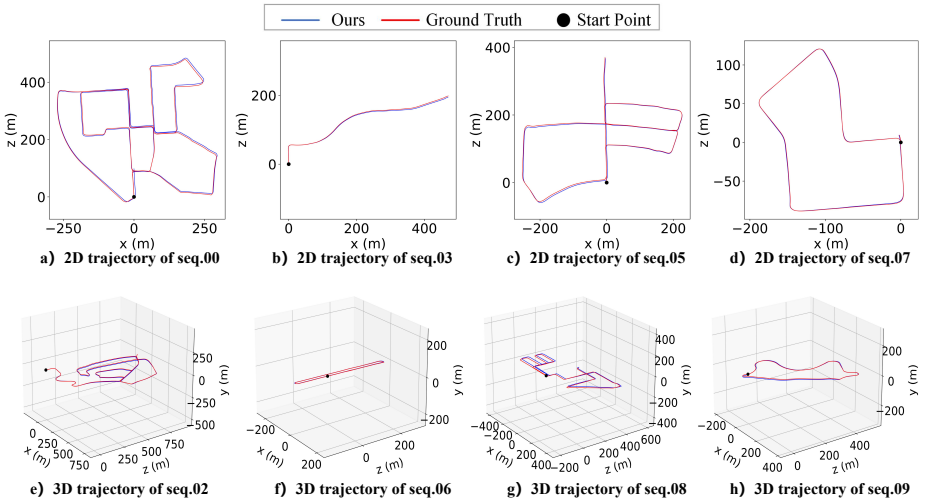


Fig. 4: Trajectory of our estimated pose. This figure shows both 2D and 3D trajectories of our network and also the ground truth one on the KITTI dataset.

margin. Compared with deep LiDAR odometry, our DVLO even outperforms the recent SOTA method EfficientLO [54] on most sequences. Compared with EfficientLO, our method has a competitive 0.41 °/100m rotation error. Moreover, our mean translation error t_{rel} on testing sequences has a 4.9% decline compared with theirs. The experiment results prove the effectiveness and great potential for our visual-LiDAR fusion design.

Comparison with Traditional Multi-Modal Odometry. We compare the performance between our method and previous traditional multi-modal odometry works on the whole KITTI sequences (00-10). The results are shown in Tab. 2, which demonstrates that our DVLO outperforms all these works on most sequences. Compared with PL-LOAM [16], our method's mean translation error t_{rel} on sequence 00-10 has a 28.7% decline.

Comparison with Learning-based Multi-Modal Odometry. Because most deep multi-modal fusion odometry methods are trained on the 00-08 sequences and tested on the 09-10 sequences, we also compare the performance between our DVLO and other learning-based multi-modal odometry works on the 09-10 sequences. The results are shown in Tab. 1 and Tab. 3. Notably, even though our model is only trained on the 00-06 sequences, our method still outperforms H-VLO on most sequences where our method has a 47.0% lower t_{rel} and 2.3% lower r_{rel}, which proves the superiority of our proposed fusion strategy.

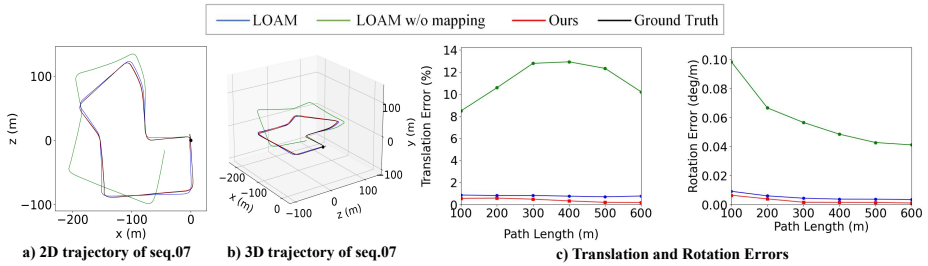


Fig. 5: Trajectory results of LOAM and ours on the KITTI sequence 07 with ground truth. Our performance is better than LOAM both without and with mapping.



Fig. 6: Visualization of our designed local clustering-based fusion mechanism within a certain cluster. Red points indicate the 2D positions of cluster centers. The yellow regions are clustered pixels around each center.

4.4 Runtime Analysis

Efficiency is another extremely significant factor in real-time SLAM systems. As shown in Tab. 4, we compare the runtime of our DVLO with other multi-modal odometry methods. Since the LiDAR points in the KITTI dataset are captured at a 10Hz frequency, previous multi-modal methods [16, 23, 47, 57] rarely satisfy the real-time application requirements (under 100 ms). However, our method has only 98.5 ms inference time, which has the potential for real-time application.

4.5 Visualization Results.

In this section, we visualize 2D and 3D trajectories based on our estimated pose in Fig. 4. The figure shows that our odometry can well track the trajectory of the ground truth. We also conduct experiments to compare trajectory accuracy and estimation errors between the classical method LOAM [66] and ours. Visualization results are shown in Fig. 5. Even though our designed odometry is only the front end of the SLAM system without mapping, our method achieves better localization performance than LOAM with mapping.

Table 5: Comparison with previous scene flow estimation works on the "val" split of the FlyingThings3D subset [39]. "RGB" and "XYZ" denote the image and point cloud respectively. The best results are **bold**.

Method	Reference	Input	2D Metrics		3D Metrics	
			EPE_{2D}	ACC_{1px}	EPE_{3D}	$ACC_{.05}$
FlowNet2 [18]	CVPR'17	RGB	5.05	72.8 %	-	-
PWC-Net [50]	CVPR'18	RGB	6.55	64.3 %	-	-
RAFT [52]	ECCV'20	RGB	3.12	81.1 %	-	-
FlowNet3D [36]	CVPR'19	XYZ	-	-	0.214	18.2 %
PointPWC [63]	ECCV'20	XYZ	-	-	0.195	-
OGSF-Net [42]	CVPR'21	XYZ	-	-	0.163	-
CamLiFlow [28]	CVPR'22	RGB+XYZ	2.18	84.3 %	0.061	85.6 %
DELFlow [43]	ICCV'23	RGB+XYZ	2.02	85.9 %	0.058	86.7 %
CamLiRAFT [29]	TPAMI'23	RGB+XYZ	1.73	87.5 %	0.049	88.4 %
Ours	—	RGB+XYZ	**1.69**	**87.6 %**	**0.048**	**88.6 %**

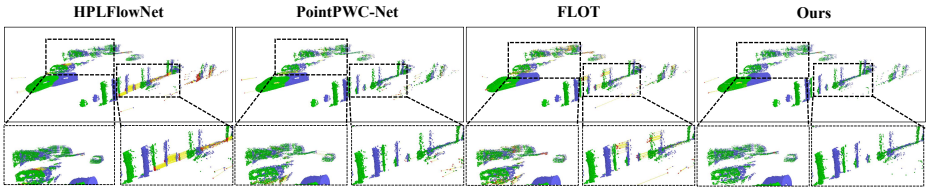


Fig. 7: Visualization of our estimated scene flow. Blue points are the source points. Green and red points respectively indicate the correct and wrong estimated target ones.

To illustrate the clustering mechanism in our designed local fusion module, we also visualize the clustered pixels around specific cluster centers. As in Fig. 6, pixels with similar texture information (yellow regions) are accurately clustered by the point-wise cosine similarity calculation with the cluster centers (red dots).

4.6 Generalization to Scene Flow Estimation Task

It is worth noting that our design can serve as a generic fusion module, which generalizes well to other tasks. Here, we extend our fusion module to the scene flow estimation task [11,12,20,29,32,36,68]. As shown in Tab. 5, our method surpasses all recent SOTA multi-modal scene flow networks on the FlyingThings3D dataset [39] in terms of both 2D and 3D metrics. Our method even consistently outperforms CamLiRAFT [29], which is specially designed for the multi-modal scene flow task. The experiment results demonstrate the strong generalization and universal application capabilities of our method. We also visualize the estimated flow in Fig. 7.

4.7 Ablation Study

In this section, extensive ablation studies are conducted to assess the significance of our designed components.

Without Local Fuser. We remove the Local Fuser module from our network and directly fuse the image and point features using the Global Fuser module.

Table 6: Significance of Local Fuser (LoF) and Global Fuser (GoF) in our local-to-global fusion network. The best results for each sequence are **bold.**

LoF	GoF	07		08		09		10		Mean (07-10)	
		t_{rel}	r_{rel}	t_{rel}	r_{rel}	t_{rel}	r_{rel}	t_{rel}	r_{rel}	t_{rel}	r_{rel}
√		0.48	**0.34**	1.11	0.50	1.11	0.50	1.02	0.54	0.93	0.47
	√	0.68	0.45	1.30	0.52	1.00	0.49	1.02	0.55	1.00	0.50
√	√	**0.46**	**0.34**	**1.09**	**0.44**	**0.85**	**0.36**	**0.88**	**0.48**	**0.82**	**0.41**

Table 7: Ablation studies of different local fusion strategies. Our clustering-based local fusion strategy can achieve the highest accuracy with half inference time compared with attention-based ones. The best results in accuracy and efficiency are **bold.**

Method	07		08		09		10		Mean (07-10)		Inference Time
	t_{rel}	r_{rel}	t_{rel}	r_{rel}	t_{rel}	r_{rel}	t_{rel}	r_{rel}	t_{rel}	r_{rel}	
Attention-based [37]	**0.44**	**0.34**	1.14	0.57	1.25	0.48	1.00	0.49	0.96	0.47	183.76 ms
Convolution-based [17]	0.50	0.37	1.25	0.50	1.14	0.58	1.06	0.56	0.99	0.50	**87.24 ms**
Clustering-based	0.46	**0.34**	**1.09**	**0.44**	**0.85**	**0.36**	**0.88**	**0.48**	**0.82**	**0.41**	98.50 ms

The results in Tab. 6 show that the performance of our model drops significantly without Local Fuser. This demonstrates the importance of the Local Fuser module since local point-to-pixel correspondences can merge more fine-grained features from different modalities.

Without Global Fuser. We remove the Global Fuser module and directly use the local fused features for pose estimation. Results in Tab. 6 demonstrate that the limited receptive field of the Local Fuser module can not enable sufficient global information interaction, which leads to about 13.4% higher t_{rel} and 14.6% higher r_{rel}. The global modeling ability of Global Fuser can help to recognize outliers that are harmful to pose regression [31].

Local Fusion Strategy. We compare the performance of different local fusion strategies. The results in Tab. 7 show that our clustering-based local fusion strategy outperforms the convolution-based [17] and attention-based [37] strategy in accuracy. Also, clustering-based fusion has a gratifying efficiency, where the total inference time is slightly higher than convolution-based fusion and half of attention-based fusion methods.

5 Conclusion

In this paper, we propose a novel local-to-global fusion network with the bi-directional structure alignment for visual-LiDAR odometry. A clustering-based local fusion module is designed to provide fine-grained multi-modal feature exchange. Furthermore, an adaptive global fusion is designed to achieve global information interaction. Comprehensive experiments show that our DVLO achieves state-of-the-art performance in terms of both accuracy and efficiency. Our fusion module can also serve as a rather generic fusion strategy, which generalizes well onto the multi-modal scene flow estimation task. We leave the generalization experiments on more multi-modal tasks for future work.

Acknowledgements

This work was supported in part by the Natural Science Foundation of China under Grant 62225309, 62073222, 62361166632, and U21A20480.

References

1. Ali, S.A., Aouada, D., Reis, G., Stricker, D.: Delo: Deep evidential lidar odometry using partial optimal transport. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 4517–4526 (2023)
2. Almalioglu, Y., Turan, M., Saputra, M.R.U., de Gusmão, P.P., Markham, A., Trigoni, N.: Selfvio: Self-supervised deep monocular visual–inertial odometry and depth estimation. Neural Networks **150**, 119–136 (2022)
3. An, Y., Shi, J., Gu, D., Liu, Q.: Visual-lidar slam based on unsupervised multi-channel deep neural networks. Cognitive Computation **14**(4), 1496–1508 (2022)
4. Aydemir, E., Fetic, N., Unel, M.: H-vlo: hybrid lidar-camera fusion for self-supervised odometry. In: 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 3302–3307. IEEE (2022)
5. Chen, S., Xu, H., Li, R., Liu, G., Fu, C.W., Liu, S.: Sira-pcr: Sim-to-real adaptation for 3d point cloud registration. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 14394–14405 (2023)
6. Cho, H.M., Kim, E.: Dynamic object-aware visual odometry (vo) estimation based on optical flow matching. IEEE Access **11**, 11642–11651 (2023)
7. Deng, J., Wu, Q., Chen, X., Xia, S., Sun, Z., Liu, G., Yu, W., Pei, L.: Nerf-loam: Neural implicit representation for large-scale incremental lidar odometry and mapping. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 8218–8227 (2023)
8. Deng, T., Chen, Y., Zhang, L., Yang, J., Yuan, S., Wang, D., Chen, W.: Compact 3d gaussian splatting for dense visual slam. arXiv preprint arXiv:2403.11247 (2024)
9. Deng, T., Shen, G., Qin, T., Wang, J., Zhao, W., Wang, J., Wang, D., Chen, W.: Plgslam: Progressive neural scene represenation with local to global bundle adjustment. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 19657–19666 (2024)
10. Deng, T., Xie, H., Wang, J., Chen, W.: Long-term visual simultaneous localization and mapping: Using a bayesian persistence filter-based global map prediction. IEEE Robotics & Automation Magazine **30**(1), 36–49 (2023)
11. Ding, F., Palffy, A., Gavrila, D.M., Lu, C.X.: Hidden gems: 4d radar scene flow learning using cross-modal supervision. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9340–9349 (2023)
12. Ding, F., Pan, Z., Deng, Y., Deng, J., Lu, C.X.: Self-supervised scene flow estimation with 4-d automotive radar. IEEE Robotics and Automation Letters **7**(3), 8233–8240 (2022)
13. Geiger, A., Lenz, P., Stiller, C., Urtasun, R.: Vision meets robotics: The kitti dataset. The International Journal of Robotics Research **32**(11), 1231–1237 (2013)
14. Geiger, A., Lenz, P., Urtasun, R.: Are we ready for autonomous driving? the kitti vision benchmark suite. In: 2012 IEEE conference on computer vision and pattern recognition. pp. 3354–3361 (2012)
15. Graeter, J., Wilczynski, A., Lauer, M.: Limo: Lidar-monocular visual odometry. In: 2018 IEEE/RSJ international conference on intelligent robots and systems (IROS). pp. 7872–7879. IEEE (2018)

16. Huang, S.S., Ma, Z.Y., Mu, T.J., Fu, H., Hu, S.M.: Lidar-monocular visual odometry using point and line features. In: 2020 IEEE International Conference on Robotics and Automation (ICRA). pp. 1091–1097. IEEE (2020)
17. Huang, T., Liu, Z., Chen, X., Bai, X.: Epnet: Enhancing point features with image semantics for 3d object detection. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XV 16. pp. 35–52. Springer (2020)
18. Ilg, E., Mayer, N., Saikia, T., Keuper, M., Dosovitskiy, A., Brox, T.: Flownet 2.0: Evolution of optical flow estimation with deep networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2462–2470 (2017)
19. Jiang, C., Du, D., Liu, J., Zhu, S., Liu, Z., Ma, Z., Liang, Z., Zhou, J.: Neurogauss4d-pci: 4d neural fields and gaussian deformation fields for point cloud interpolation. arXiv preprint arXiv:2405.14241 (2024)
20. Jiang, C., Wang, G., Liu, J., Wang, H., Ma, Z., Liu, Z., Liang, Z., Shan, Y., Du, D.: 3dsflabelling: Boosting 3d scene flow estimation by pseudo auto-labelling. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 15173–15183 (2024)
21. Kendall, A., Grimes, M., Cipolla, R.: Posenet: A convolutional network for real-time 6-dof camera relocalization. In: Proceedings of the IEEE international conference on computer vision. pp. 2938–2946 (2015)
22. Konda, K.R., Memisevic, R.: Learning visual odometry with a convolutional network. VISAPP (1) **2015**, 486–490 (2015)
23. Leutenegger, S., Furgale, P., Rabaud, V., Chli, M., Konolige, K., Siegwart, R.: Keyframe-based visual-inertial slam using nonlinear optimization. Proceedings of Robotis Science and Systems (RSS) 2013 (2013)
24. Li, B., Hu, M., Wang, S., Wang, L., Gong, X.: Self-supervised visual-lidar odometry with flip consistency. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 3844–3852 (2021)
25. Li, B., Wang, S., Ye, H., Gong, X., Xiang, Z.: Cross-modal knowledge distillation for depth privileged monocular visual odometry. IEEE Robotics and Automation Letters **7**(3), 6171–6178 (2022)
26. Li, Q., Chen, S., Wang, C., Li, X., Wen, C., Cheng, M., Li, J.: Lo-net: Deep real-time lidar odometry. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8473–8482 (2019)
27. Li, X., Ma, T., Hou, Y., Shi, B., Yang, Y., Liu, Y., Wu, X., Chen, Q., Li, Y., Qiao, Y., et al.: Logonet: Towards accurate 3d object detection with local-to-global cross-modal fusion. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 17524–17534 (2023)
28. Liu, H., Lu, T., Xu, Y., Liu, J., Li, W., Chen, L.: Camliflow: Bidirectional camera-lidar fusion for joint optical flow and scene flow estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5791–5801 (2022)
29. Liu, H., Lu, T., Xu, Y., Liu, J., Wang, L.: Learning optical flow and scene flow with bidirectional camera-lidar fusion. IEEE Transactions on Pattern Analysis and Machine Intelligence (2023)
30. Liu, J., Wang, G., Jiang, C., Liu, Z., Wang, H.: Translo: A window-based masked point transformer framework for large-scale lidar odometry. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 37, pp. 1683–1691 (2023)

31. Liu, J., Wang, G., Liu, Z., Jiang, C., Pollefeys, M., Wang, H.: Regformer: An efficient projection-aware transformer network for large-scale point cloud registration. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 8451–8460 (October 2023)
32. Liu, J., Wang, G., Ye, W., Jiang, C., Han, J., Liu, Z., Zhang, G., Du, D., Wang, H.: Difflow3d: Toward robust uncertainty-aware scene flow estimation with iterative diffusion-based refinement. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 15109–15119 (2024)
33. Liu, Q., Chen, N., Liu, Z., Wang, H.: Toward learning-based visuomotor navigation with neural radiance fields. IEEE Transactions on Industrial Informatics **20**(6), 8907–8916 (2024). https://doi.org/10.1109/TII.2024.3378829
34. Liu, Q., Cui, X., Liu, Z., Wang, H.: Boosting explore-exploit behavior for navigating vehicle by maintaining informative topological frontier. IEEE Transactions on Intelligent Vehicles pp. 1–11 (2024). https://doi.org/10.1109/TIV.2024.3415734
35. Liu, Q., Wang, G., Liu, Z., Wang, H.: Visuomotor navigation for embodied robots with spatial memory and semantic reasoning cognition. IEEE Transactions on Neural Networks and Learning Systems pp. 1–12 (2024). https://doi.org/10.1109/TNNLS.2024.3418857
36. Liu, X., Qi, C.R., Guibas, L.J.: Flownet3d: Learning scene flow in 3d point clouds. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 529–537 (2019)
37. Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B.: Swin transformer: Hierarchical vision transformer using shifted windows. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 10012–10022 (2021)
38. Ma, X., Zhou, Y., Wang, H., Qin, C., Sun, B., Liu, C., Fu, Y.: Image as set of points. In: International Conference on Learning Representations (ICLR) (2023)
39. Mayer, N., Ilg, E., Hausser, P., Fischer, P., Cremers, D., Dosovitskiy, A., Brox, T.: A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4040–4048 (2016)
40. Naumann, J., Xu, B., Leutenegger, S., Zuo, X.: Nerf-vo: Real-time sparse visual odometry with neural radiance fields. arXiv preprint arXiv:2312.13471 (2023)
41. Nicolai, A., Skeele, R., Eriksen, C., Hollinger, G.A.: Deep learning for laser based odometry estimation. In: RSS workshop Limits and Potentials of Deep Learning in Robotics. vol. 184, p. 1 (2016)
42. Ouyang, B., Raviv, D.: Occlusion guided scene flow estimation on 3d point clouds. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2805–2814 (2021)
43. Peng, C., Wang, G., Lo, X.W., Wu, X., Xu, C., Tomizuka, M., Zhan, W., Wang, H.: Delflow: Dense efficient learning of scene flow for large-scale point clouds. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 16901–16910 (2023)
44. Shamwell, E.J., Lindgren, K., Leung, S., Nothwang, W.D.: Unsupervised deep visual-inertial odometry with online error correction for rgb-d imagery. IEEE transactions on pattern analysis and machine intelligence **42**(10), 2478–2493 (2019)
45. Shan, J., Li, Y., Xie, T., Wang, H.: Enerf-slam: A dense endoscopic slam with neural implicit representation. IEEE Transactions on Medical Robotics and Bionics (2024)
46. Shin, Y.S., Park, Y.S., Kim, A.: Dvl-slam: Sparse depth enhanced direct visual-lidar slam. Autonomous Robots **44**(2), 115–130 (2020)

47. Shu, C., Luo, Y.: Multi-modal feature constraint based tightly coupled monocular visual-lidar odometry and mapping. IEEE Transactions on Intelligent Vehicles (2022)
48. Shubodh, S., Omama, M., Zaidi, H., Parihar, U.S., Krishna, M.: Lip-loc: Lidar image pretraining for cross-modal localization. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 948–957 (2024)
49. Song, S., Chandraker, M., Guest, C.C.: High accuracy monocular sfm and scale correction for autonomous driving. IEEE transactions on pattern analysis and machine intelligence **38**(4), 730–743 (2015)
50. Sun, D., Yang, X., Liu, M.Y., Kautz, J.: Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 8934–8943 (2018)
51. Sun, L., Ding, G., Qiu, Y., Yoshiyasu, Y., Kanehiro, F.: Transfusionodom: Transformer-based lidar-inertial fusion odometry estimation. IEEE Sensors Journal (2023)
52. Teed, Z., Deng, J.: Raft: Recurrent all-pairs field transforms for optical flow. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16. pp. 402–419. Springer (2020)
53. Valente, M., Joly, C., de La Fortelle, A.: Deep sensor fusion for real-time odometry estimation. In: 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 6679–6685. IEEE (2019)
54. Wang, G., Wu, X., Jiang, S., Liu, Z., Wang, H.: Efficient 3d deep lidar odometry. IEEE Transactions on Pattern Analysis and Machine Intelligence **45**(5), 5749–5765 (2022)
55. Wang, G., Wu, X., Liu, Z., Wang, H.: Pwclo-net: Deep lidar odometry in 3d point clouds using hierarchical embedding mask optimization. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 15910–15919 (2021)
56. Wang, S., Clark, R., Wen, H., Trigoni, N.: Deepvo: Towards end-to-end visual odometry with deep recurrent convolutional neural networks. In: 2017 IEEE international conference on robotics and automation (ICRA). pp. 2043–2050. IEEE (2017)
57. Wang, W., Liu, J., Wang, C., Luo, B., Zhang, C.: Dv-loam: Direct visual lidar odometry and mapping. Remote Sensing **13**(16), 3340 (2021)
58. Wang, W., Saputra, M.R.U., Zhao, P., Gusmao, P., Yang, B., Chen, C., Markham, A., Trigoni, N.: Deeppco: End-to-end point cloud odometry through deep parallel neural network. In: 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 3248–3254. IEEE (2019)
59. Wang, X., Wang, Y., Ye, J., Wang, Z., Sun, F., Liu, P., Wang, L., Sun, K., Wang, X., He, B.: Animatabledreamer: Text-guided non-rigid 3d model generation and reconstruction with canonical score distillation. arXiv preprint arXiv:2312.03795 (2023)
60. Wang, Y., Wang, X., Chen, Z., Wang, Z., Sun, F., Zhu, J.: Vidu4d: Single generated video to high-fidelity 4d reconstruction with dynamic gaussian surfels. arXiv preprint arXiv:2405.16822 (2024)
61. Wu, W., Wang, G., Deng, T., Aegidius, S., Shanks, S., Modugno, V., Kanoulas, D., Wang, H.: Dvn-slam: Dynamic visual neural slam based on local-global encoding. arXiv preprint arXiv:2403.11776 (2024)
62. Wu, W., Wang, Q., Wang, G., Wang, J., Zhao, T., Liu, Y., Gao, D., Liu, Z., Wang, H.: Emie-map: Large-scale road surface reconstruction based on explicit mesh and implicit encoding. arXiv preprint arXiv:2403.11789 (2024)

63. Wu, W., Wang, Z.Y., Li, Z., Liu, W., Fuxin, L.: Pointpwc-net: Cost volume on point clouds for (self-) supervised scene flow estimation. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part V 16. pp. 88–107. Springer (2020)
64. Yuan, Z., Wang, Q., Cheng, K., Hao, T., Yang, X.: Sdv-loam: Semi-direct visual-lidar odometry and mapping. IEEE Transactions on Pattern Analysis and Machine Intelligence (2023)
65. Zhan, H., Weerasekera, C.S., Bian, J.W., Garg, R., Reid, I.: Df-vo: What should be learnt for visual odometry? arXiv preprint arXiv:2103.00933 (2021)
66. Zhang, J., Singh, S.: Loam: Lidar odometry and mapping in real-time. In: Robotics: Science and systems. vol. 2, pp. 1–9. Berkeley, CA (2014)
67. Zhang, J., Singh, S.: Visual-lidar odometry and mapping: Low-drift, robust, and fast. In: 2015 IEEE International Conference on Robotics and Automation (ICRA). pp. 2174–2181. IEEE (2015)
68. Zhang, Q., Yang, Y., Fang, H., Geng, R., Jensfelt, P.: Deflow: Decoder of scene flow network in autonomous driving. arXiv preprint arXiv:2401.16122 (2024)
69. Zhang, Y.X., Gui, J., Cong, X., Gong, X., Tao, W.: A comprehensive survey and taxonomy on point cloud registration based on deep learning. arXiv preprint arXiv:2404.13830 (2024)
70. Zhou, T., Brown, M., Snavely, N., Lowe, D.G.: Unsupervised learning of depth and ego-motion from video. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1851–1858 (2017)
71. Zhu, S., Qin, R., Wang, G., Liu, J., Wang, H.: Semgauss-slam: Dense semantic gaussian splatting slam. arXiv preprint arXiv:2403.07494 (2024)
72. Zhu, S., Wang, G., Blum, H., Liu, J., Song, L., Pollefeys, M., Wang, H.: Sni-slam: Semantic neural implicit slam. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 21167–21177 (2024)
73. Zhuoins, G., Lu, S., Xiong, L., Zhouins, H., Zheng, L., Zhou, M.: 4drvo-net: Deep 4d radar–visual odometry using multi-modal and multi-scale adaptive fusion. IEEE Transactions on Intelligent Vehicles (2023)