2024 IEEE International Conference on Robotics and Automation (ICRA 2024)
May 13-17, 2024. Yokohama, Japan

Lite-SVO: Towards A Lightweight Self-Supervised Semantic Visual Odometry Exploiting Multi-Feature Sharing Architecture

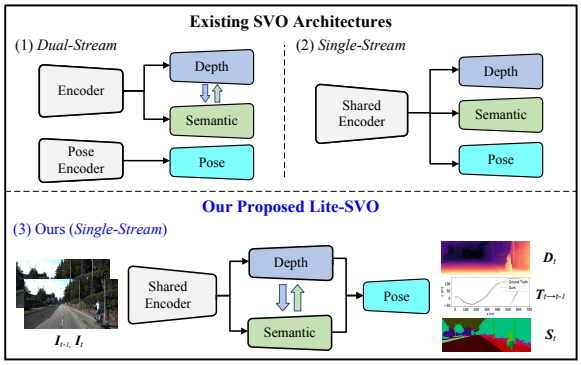
Wenhui Wei[1,2], Jiantao Li[1,2], Kaizhu Huang[3], Jiadong Li[2], Xin Liu[2], Yangfan Zhou[2,*]

***Abstract*— Not relying on ground-truth data for training, self-supervised semantic visual odometry (SVO) has recently gained considerable attention. Within self-supervised SVO, feature representation inconsistency between semantic/depth and pose tasks presents a significant challenge, as it may disrupt cross-task feature representations and lead to notable performance degradation. Regrettably, existing self-supervised SVO lacks an effective solution to address this obstacle, for either overlooking this issue or exploiting a too heavy architecture. In response to this challenge, we propose a groundbreaking solution within the *Single-Stream* architecture, known as Lite-SVO, which is a lightweight yet efficient multi-feature sharing architecture. Lite-SVO is designed to bolster self-supervised SVO, facilitating its adoption on edge devices without compromising accuracy and performance. The crucial innovation lies in the multi-feature sharing architecture, which fuses the semantic and depth maps as pose features, thus significantly reducing the model complexity and boosting the speed in edge devices. Built upon the novel feature sharing framework, Lite-SVO further optimizes the feature sharing representation to improve the performance. Specifically, a cross-feature sharing module alleviates the impact of object boundary in depth estimation, while a multi-feature sharing module focuses on extracting and fusing spatial features to enhance pose estimation. Experimental results demonstrate that our method is at least 84.46% faster than the state-of-the-art *Single-Stream* approaches, and excitingly, our method's pose accuracy is about 79.83% higher than theirs.**

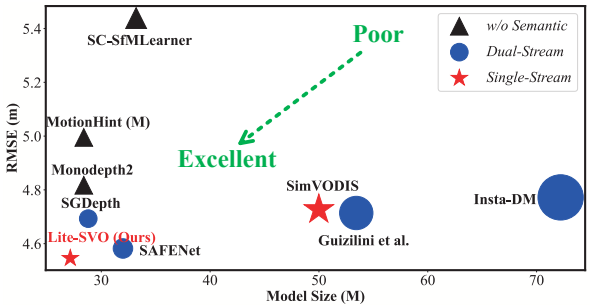
I. INTRODUCTION

Semantic Visual Odometry (SVO) stands at the forefront of visual perception, offering a broader scope of applications and superior performance compared to traditional VO. By harnessing semantic information, SVO transcends the limitations of traditional VO, bolstering the agent's situational awareness of surrounding environment [1]. Remarkably, in autonomous systems within real-world scenarios, self-supervised SVO approaches [2], [3], [4], [5], [6] have emerged as a dominant paradigm, surpassing the efficacy of supervised SVO approaches [7], [8]. This pronounced shift in methodology holds particular significance in critical domains such as autonomous driving and unmanned aerial vehicles (UAVs) [9], as it effectively mitigates the need for costly and time-intensive ground truth labels, thereby enabling a more efficient and agile system development process.

[1] School of Nano-Tech and Nano-Bionics, University of Science and Technology of China, Hefei 230026, China
[2] Suzhou Institute of Nano-Tech and Nano-Bionics (SINANO), Chinese Academy of Sciences, Suzhou 215123, China {whwei2022, jtlli2023, jdli2009, xliu2018, yfzhou2020}@sinano.ac.cn
[3] Data Science Research Center, Duke Kunshan University, Kunshan 215316, China kaizhu.huang@dukekunshan.edu.cn
*Corresponding author: Yangfan Zhou



(a) Existing SVO architectures *vs.* Lite-SVO architecture.



(b) Performance comparison between Lite-SVO and recent superior methods. *Note that the legend size represents the size of the model parameters.*

Fig. 1. Workflow (a) and performance (b) comparison of our proposed Lite-SVO with existing SVO architectures. Our proposed Lite-SVO achieves state-of-the-art performance in both accuracy and model size.

In general, existing self-supervised SVO methods can be categorized into two main architectures: *Dual-Stream* and *Single-Stream*, as illustrated in Fig. 1 (a). *Dual-Stream* architectures [2], [3], [4] leverage semantic features to guide the depth estimation process, which can enhance the accuracy and robustness of self-supervised SVO. Though these methods achieve promising accuracy, the improvement may mainly be attributed to the increased model complexity. In particular, these methods employ two separate encoders to handle pose and depth/semantic tasks; this results in significantly larger overhead in terms of both memory and computational demands of the computing platform. On the other hand, *Single-Stream* architectures [5], [6] attempt to engage a shared encoder to alleviate the tasks overhead in *Dual-Stream* architectures. However, these efforts often overlook compatibility between single-frame (depth/semantic tasks) feature representation and continuous-frame (pose task) feature representation, consequently compromising the

performance of SVO.

To address the challenge of feature representation inconsistency in the existing *Single-Stream* architecture, we present a novel self-supervised SVO method called Lite-SVO, which achieves compelling performance in terms of both accuracy and speed. Specifically, we introduce a novel multi-feature sharing architecture that employs the fusion of depth and semantic maps as input for pose estimation, in contrast to using image features in the existing architectures, thus promoting the feature representation consistency of *Single-Stream* architecture. Moreover, we enhance the task collaboration of the multi-feature sharing architecture. In particular, to tackle the challenge of the blurred object boundaries in depth information, we propose a novel cross-feature sharing module between depth and semantic maps. It utilizes a cross-attention mechanism to share information, while adopting an efficient bidimensional feedforward network to boost the spatial-wise representation of depth and semantic information. Additionally, to prevent the degradation of depth and semantic fusion representations of multi-feature sharing architecture, we develop a simple yet effective multi-feature sharing module to foster better task coordination which can harness the complementary information from depth and semantic maps.

In summary, our major contributions are three-fold:

- We introduce a novel and efficient multi-feature sharing architecture termed Lite-SVO, which serves as a compelling solution for implementing lightweight self-supervised SVO.
- We design lightweight and efficient feature sharing mechanism based modules to capture and represent spatial-wise semantic and depth features, culminating in a substantial enhancement of SVO performance.
- Lite-SVO demonstrates superior accuracy on the KITTI dataset and generalization on the AirDOS-Shibuya dataset in comparison to competitive larger models. Notably, it achieves the state-of-the-art performance both in accuracy and model complexity (see Fig. 1 (b)).

II. Related Works

A. Learning-based SVO Methods

With the advancement of deep neural networks, learning-based SVO has demonstrated remarkable performance. Generally, learning-based SVO methods can be classified into two categories: supervised-based SVO methods and self-supervised based SVO methods.

For supervised-based SVO methods, ClusterVO [7] introduces semantic information to improve probabilistic association. Moreover, DAVO [8] utilizes semantic information to guide the attention mechanism, enhancing pose performance. However, these supervised methods demand large quantities of ground-truth pose trajectories for training, resulting in a high cost associated with data collection and annotation.

Contrastively, self-supervised based SVO methods have gained significant attention as they reduce the need for ground-truth labels. Recent works such as [2] and FSRE-Depth [3] could improve SVO performance by generating semantic information to guide depth estimation. However, these methods typically adopt a heavy *Dual-Stream* architecture, posing challenges for their implementation on edge platforms with limited resources. Recently, SimVODIS [5] and SimVODIS++ [6] utilize a more concise and efficient *Single-Stream* architecture by integrating the semantic task into VO. Unfortunately, these methods neglect task collaboration, which is however crucially important for *Single-Stream* architecture SVO.

Different from the existing methods, we propose a novel multi-feature sharing architecture to significantly enhance features collaboration of *Single-Stream* architecture.

B. Neural Attention Mechanism

Neural attention mechanisms have attracted much attention owing to their exceptional performance in various tasks. In general, attention mechanism can be divided into two types: Transformer-based attention mechanism, and CNN-based attention mechanism.

Transformer-based attention mechanism is effective in capturing global dependencies within the input feature maps. [10] designs a self-attention scheme that leverages information from different representation sub-spaces by employing multi-head attention. Moreover, cross-attention schemes have been utilized to extract features across heterogeneous representations, such as image, speech, and text [11], [12], [13].

CNN-based attention mechanism is commonly used in various computer vision tasks. In this mechanism, attention weights are computed using convolutional operations. A notable example is the Convolutional Block Attention Module (CBAM) [14], which introduces an efficient attention scheme by fusing the spatial and channel attention mechanisms.

In this work, we introduce Lite-SVO, a novel approach that leverages both a cross-feature sharing module and a multi-feature sharing module to substantially enhance the feature representation between depth and semantic tasks.

III. Proposed Method

A. Architecture Overview

This work is dedicated to devising an efficient architecture that effectively addresses semantic, depth, and pose tasks simultaneously. The proposed architecture overview is presented in Fig. 2. It comprises four main components: 1) the Semantic-Depth Shared encoder for receiving consecutive images and extracting image features; 2) the Semantic-Depth decoder for estimating the semantic maps $\boldsymbol{S}_t$ and depth maps $\boldsymbol{D}_t$; 3) the Multi-Feature Sharing (M-FS) module for fusing the semantic and depth maps; and 4) the Pose decoder for estimating pose $\boldsymbol{T}_{t\to t-1}$.

Aiming to improve the task feature representation consistency of existing *Single-Stream* SVO methods, we design a novel multi-feature sharing architecture. Fusing the semantic and depth features as inputs for pose estimation, our framework not only significantly promotes the feature representation consistency, but also leads to high inference speed. Built upon our new architecture, we could effectively address the issue of lacking feature sharing representation

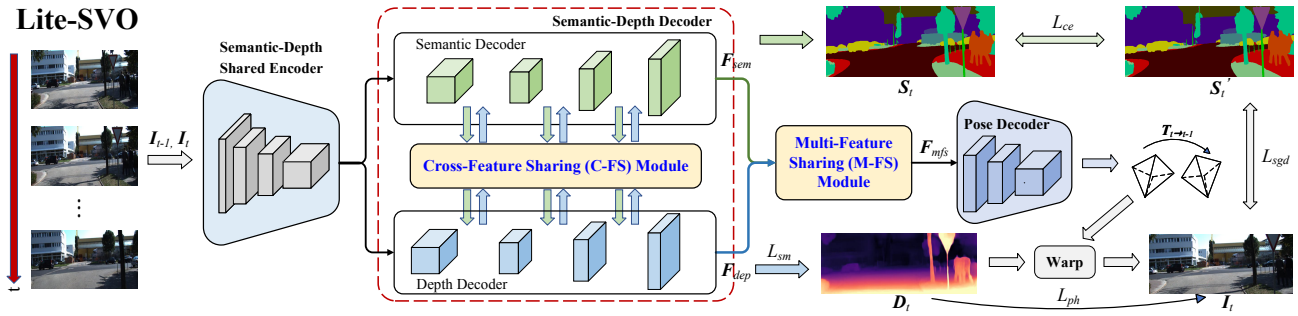
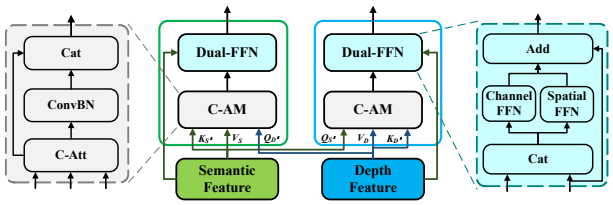


Fig. 2. Overview of our proposed Lite-SVO, which enjoys an innovative multi-feature sharing architecture. Given an image sequence $\{\boldsymbol{I}_{t-1}, \boldsymbol{I}_t\}$, this architecture fuses semantic maps $\boldsymbol{F}_{sem}$ and depth maps $\boldsymbol{F}_{dep}$ from the Semantic-Depth decoder as input features for pose estimation $\boldsymbol{T}_{t\to t-1}$.



(a) C-AM (b) C-FS Module (c) Dual-FFN

Fig. 3. Illustration of the workflow of the C-FS module. (a) shows the structure of C-AM; (b) presents the complete architecture of the C-FS module; and (c) illustrates the structure of Dual-FFN.

between depth and semantic estimation within SVO. We design the effective Cross-Feature Sharing (C-FS) module to improve depth-semantic interaction via an enhanced spatial features network. In addition, recognizing the significance of spatial information in pose estimation, we present the novel M-FS module to extract and fuse spatial-wise semantic and depth features for the pose decoder.

B. Cross-Feature Sharing (C-FS) Module

To emphasize the significance of cross-task representations between depth and semantic information, we propose the C-FS module, as illustrated in Fig. 3. The C-FS module comprises two essential components: Cross-Attention Module (C-AM) and Dual Feed Forward Network (Dual-FFN).

1) C-AM: C-AM effectively promotes feature sharing between semantics and depth by utilizing cross-attention mechanism. The structure of C-AM is shown in Fig. 3 (a). First, the cross-attention mechanism is computed as follows:

$$C-Att(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V}) = Cat(\boldsymbol{H}_1,\boldsymbol{H}_2,...,\boldsymbol{H}_n)\boldsymbol{W}, \quad (1)$$

where $\boldsymbol{Q},\boldsymbol{K}$ and $\boldsymbol{V}$ represent queries, keys, and values, respectively, while $\boldsymbol{W}$ signifies weight matrices. $\boldsymbol{H}_i$ denotes the result of attention mechanism, formulated as follows:

$$\boldsymbol{H}_i = Att(\boldsymbol{Q}_i,\boldsymbol{K}_i,\boldsymbol{V}_i) = softmax(\frac{\boldsymbol{Q}_i\boldsymbol{K}_i^T}{\sqrt{d_k}})\boldsymbol{V}_i, \quad (2)$$

where d_k means the dimension of the keys vectors.

Then, given the attention mechanism lacks the ability to distinguish the position information of the input features, we introduce a spatial positional encoding process [15] for the input $\boldsymbol{X}$. Thus, C-AM can be modeled as follows:

$$\boldsymbol{F}_{cam} = Cat\big(\boldsymbol{X}, ConvBN(C-Att(\boldsymbol{Q}_{y'},\boldsymbol{K}_{x'},\boldsymbol{V}_x))\big), \quad (3)$$

where $ConvBN$ refers to the convolution followed by batch normalization. The term y' denotes the addition of the positional encodings $\boldsymbol{p}_y$ and the input features $\boldsymbol{Y}$, while x' denotes the result obtained by adding the positional encodings $\boldsymbol{p}_x$ to the input features $\boldsymbol{X}$.

2) Dual-FFN: Many existing FFN architectures [10], [11] limit the channel and spatial information, resulting in incomplete features representation. To address this issue, we propose the Dual-FFN module, as shown in Fig. 3. (c). Dual-FFN consists of the Spatial FFN branch and the Channel FFN branch, thus enhancing the spatial features representation.

To elaborate, the Spatial FFN branch generates spatial attention maps by convolution, thus encoding where to emphasize. It can be computed as:

$$\boldsymbol{Sp}_{out} = Conv\big(ELU(ConvBN(\boldsymbol{F}_{in}))\big), \quad (4)$$

where ELU [16] serves as the non-linear activation function, effectively preventing neurons from becoming inactive compared to ReLU [17]. $\boldsymbol{F}_{in}$ signifies the concatenation of $\boldsymbol{F}_{cam}$ and $\boldsymbol{X}_{in}$.

Then, the Channel FFN branch aggregates the feature map in each channel representation, which is modeled as follows:

$$\boldsymbol{F}_{max} = ELU\big(FC(MaxPool(\boldsymbol{F}_{in}))\big), \quad (5)$$

$$\boldsymbol{F}_{avg} = ELU\big(FC(AvgPool(\boldsymbol{F}_{in}))\big), \quad (6)$$

$$\boldsymbol{Ch}_{out} = FC(\boldsymbol{F}_{max}) + FC(\boldsymbol{F}_{avg}), \quad (7)$$

where $MaxPool$ and $AvgPool$ denote the operators of max pooling and average pooling, respectively. FC represents full connection layer.

Finally, the output of the Dual-FFN, denoted as $\boldsymbol{F}_{dual}$, which can be expressed as:

$$\boldsymbol{F}_{dual} = \boldsymbol{X}_{in} + \boldsymbol{Sp}_{out} + \boldsymbol{Ch}_{out}, \quad (8)$$

where $\boldsymbol{F}_{dual}$ represents semantic or depth features integrated into the Semantic-Depth decoder layer. Overall, this module enables a comprehensive representation of spatial-wise features, leading to enhanced cross-feature representation within the CFS module.

C. Multi-Feature Sharing (M-FS) Module

To mitigate information mismatch resulting from directly fusing semantic and depth features, we propose the Multi-Feature Sharing (M-FS) module, as depicted in Fig. 4. The M-FS module is specifically designed to extract and fuse

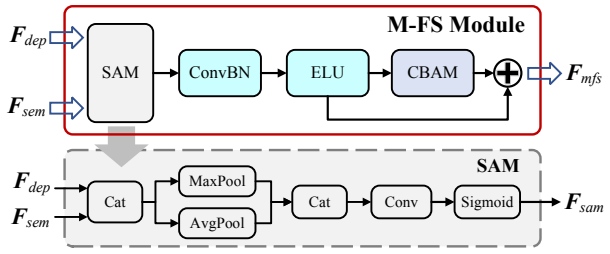


Fig. 4. Illustration of the M-FS module, which efficiently extracts and fuses the spatial-wise semantic and depth features.

spatial-wise semantic and depth feature maps, which are subsequently utilized as input features for the pose decoder.

First, the M-FS module focuses on extracting spatial features from semantic and depth maps by employing the Spatial Attention Module (SAM). SAM effectively captures spatial information from the input features, which is expressed as:

$$\boldsymbol{F}_{cat} = Cat(\boldsymbol{F}_{dep}, \boldsymbol{F}_{sem}), \tag{9}$$

$$\boldsymbol{F}_{pool} = Cat\big(MaxPool(\boldsymbol{F}_{cat}), AvgPool(\boldsymbol{F}_{cat})\big), \tag{10}$$

$$\boldsymbol{F}_{sam} = Sigmoid\big(Conv(\boldsymbol{F}_{pool})\big), \tag{11}$$

where $Sigmoid$ represents the non-linear activation function.

Then, the M-FS module performs non-linear activation on the result of SAM, which can be defined as:

$$\boldsymbol{F}_{inter} = ELU\big(ConvBN(\boldsymbol{F}_{sam})\big). \tag{12}$$

Finally, the M-FS module generates the output results $\boldsymbol{F}_{mfs}$ by fusing feature maps:

$$\boldsymbol{F}_{mfs} = Add(\boldsymbol{F}_{inter}, \boldsymbol{F}_{cbam}), \tag{13}$$

where $\boldsymbol{F}_{cbam}$ means the output of $\boldsymbol{F}_{inter}$ after utilizing CBAM [14]. In essence, the M-FS module mitigates multi-feature inconsistency and promotes spatial feature extraction.

D. Loss Functions

Our comprehensive loss function integrates four distinct components as outlined below:

$$L_{total} = \lambda_1 L_{ph} + \lambda_2 L_{sm} + \lambda_3 L_{ce} + \lambda_4 L_{sgd}, \tag{14}$$

where $\lambda_1, \lambda_2, \lambda_3$ and λ_4 denote the balanced weights.

First, L_{ph} represents the photometric consistency loss for implementing self-supervised training through a view-synthesis approach [22]. Then, L_{sm} means the depth smoothness loss [25], which is applied to ensure alignment with the edges of objects in depth maps. Next, L_{ce} denotes the cross-entropy loss, utilized to enforce semantic segmentation constraints. Notably, we follow the commonly used settings [4], [3], which utilize an off-the-shelf segmentation model [26] to generate labels $\boldsymbol{S}'_{\boldsymbol{t}}$. Finally, L_{sgd} represents semantic-guided depth loss [3] to rectify inaccurate depth estimations along object edges by leveraging semantic features.

IV. Experiments

A. Datasets

1) KITTI: We conducted our experiments on the widely adopted KITTI dataset [27], which serves as a benchmark for VO tasks. It includes diverse urban and highway driving sequences, which present numerous challenges for VO, such as dynamic scenes and complex lighting conditions.

2) AirDOS-Shibuya: We conduct generalization analysis on the AirDOS-Shibuya dataset [28], which offers intricate and complex environments by simulating the bustling road intersection at Shibuya, Tokyo. Lite-SVO is pre-trained on KITTI before being directly applied to the test sequences.

B. Implement Details

1) Hyperparameters: We select the Shared encoder and the Pose decoder, as in [3], [22]. We resize the original image into a resolution of 192×640, and apply the common data augmentation techniques [22]. Furthermore, we configure the batch size to be 4 and employ a learning rate of 1e-4. The training process runs for 20 epochs, and the learning rate decays by a factor of 0.1 after 15 epochs. The Adam optimizer [31] is utilized for optimization. Moreover, we follow [3] to use a common loss parameter setting, i.e., $\{\lambda_1 = 1, \lambda_2 = 0.001, \lambda_3 = 0.3, \lambda_4 = 0.1\}$.

2) Evaluation Metrics: For evaluating pose estimation performance, we report error using well-established metrics [32], which include RMSE (m), Rel.trans. (%) and Rel.rot. (deg/m). To evaluate depth estimation performance, we assess accuracy and error through widely recognized metrics [33], where are Abs Rel, Sq Rel, RMSE, RMSE log, $\delta < 1.25$, $\delta < 1.25^2$ and $\delta < 1.25^3$. Moreover, we evaluate semantic segmentation performance using mIoU (%).

TABLE I

Pose Performance Comparison with the State-of-the-Art Methods on the KITTI Dataset

	Method	Sem	Seq. 09			Seq. 10		
			RMSE (m)	Rel. trans. (%)	Rel. rot. (deg/m)	RMSE (m)	Rel. trans. (%)	Rel. rot. (deg/m)
Sup.	ESP-VO [18]	✗	-	-	-	-	9.77	0.102
	DeepV2D [19]	✗	79.06	8.71	0.037	48.49	12.81	0.083
	DAVO [8]	✓	-	-	-	-	5.37	0.016
Self-Sup.	SC-SfMLearner [20]	✗	77.79	19.15	0.068	67.34	40.40	0.177
	GeoNet [21]	✗	158.45	28.72	0.098	43.04	23.90	0.090
	Monodepth2 [22]	✗	68.18	14.84	0.033	20.46	**7.730**	0.034
	MotionHint (M2)[1] [23]	✗	54.46	11.56	0.026	15.52	10.09	0.039
	Guizilini *et al.* [2]	✓	-	**6.881**	0.024	-	9.125	0.031
	Insta-DM [24]	✓	-	8.600	0.029	-	9.200	0.045
	SimVODIS [5]	✓	186.16	67.61	0.279	79.26	48.16	0.252
	Lite-SVO (Ours)	✓	**33.82**	7.200	**0.015**	**15.45**	8.495	**0.024**

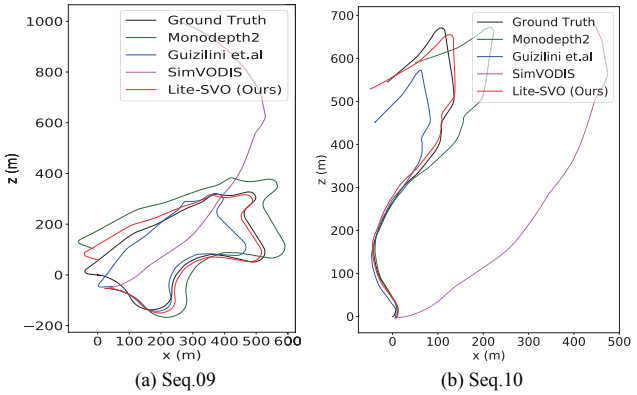
[1] *MotionHint (M2) means that the selection of the Monodepth2-based component in this method.*

TABLE II

DEPTH PERFORMANCE COMPARISON WITH THE STATE-OF-THE-ART METHODS ON THE KITTI DATASET

Method	Sem	Error Metric				Accuracy Metric		
		AbsRel ↓	SqRel ↓	RMSE ↓	RMSE Log ↓	< 1.25 ↑	$< 1.25^2$ ↑	$< 1.25^3$ ↑
SC-SfMLearner [20]	✗	0.137	1.089	5.439	0.217	0.830	0.942	0.975
Monodepth2 [22]	✗	0.114	0.864	4.817	0.192	0.875	0.959	0.981
MotionHint (M2) [23]	✗	0.125	0.932	4.995	0.205	0.846	0.953	0.979
Wang *et al.* [29]	✗	0.130	0.978	5.129	0.208	0.836	0.949	0.981
Guizilini *et al.*[2] [2]	✓	0.117	0.854	4.714	0.191	0.873	**0.963**	0.981
SGDepth [30]	✓	0.113	0.835	4.693	0.191	**0.879**	0.961	0.981
Insta-DM [24]	✓	0.112	**0.777**	4.772	0.191	0.872	0.959	0.982
SAFENet [4]	✓	0.112	0.788	4.582	0.187	0.878	**0.963**	0.983
SimVODIS [5]	✓	0.123	0.797	4.727	0.193	0.877	0.960	**0.984**
Lite-SVO (Ours)	✓	**0.111**	0.786	**4.576**	**0.185**	**0.879**	**0.963**	0.983

[2] *We opt for a widely adopted version, which consists of a ResNet-18 backbone and without two-stage training method for the sake of fairness.*



(a) Seq.09 (b) Seq.10

Fig. 5. Qualitative results of pose estimation on Seq.09 and Seq.10. Our method shows superior pose estimation performance to the other methods.

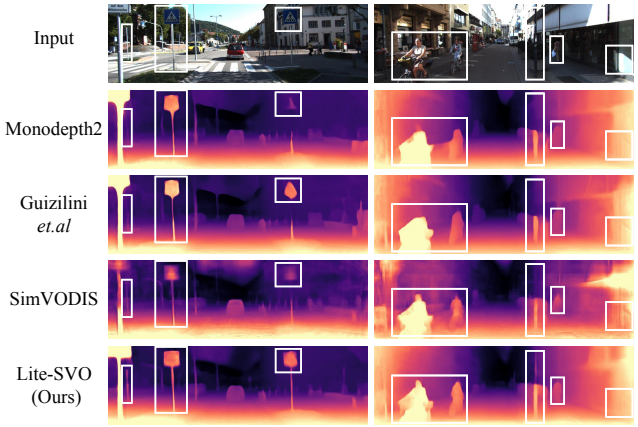


Fig. 6. Qualitative results of depth estimation on KITTI. The highlighted boxes demonstrate the outstanding performance of our method in depth estimation, particularly in capturing finer details.

C. Experiments Results

In this section, we present extensive experimental validation of the proposed Lite-SVO. We compare Lite-SVO with state-of-the-art methods on multiple datasets. Furthermore, we perform complexity and timing evaluations to demonstrate the efficiency of Lite-SVO compared to the other methods.

1) Evaluation Performance:

a) Pose Performance: In the assessment of pose estimation performance, we conducted comprehensive experiments on the widely used KITTI sequences, specifically Seq.09 and Seq.10. Moreover, we evaluated Lite-SVO against both supervised-based (*Sup.*) methods and self-supervised-based (*Self-Sup.*) methods. The results are summarized in Table III-C. It is evident that Lite-SVO outperforms the current the *Self-Sup.* methods by a large margin. Crucially, Lite-SVO showcases a notable improvement of 79.83% in RMSE compared to SimVODIS, establishing its superiority over the state-of-the-art *Single-Stream* method. We also visualize the trajectories of Seq.09 and Seq.10 in Fig. 5. As observed, Lite-SVO indeed exhibits better alignments with the ground-truth trajectories compared to recent advanced methods.

b) Depth Performance: In the evaluation of depth estimation performance, we conduct a comparative analysis involving Lite-SVO and three self-supervised VO categories: 1) *w/o Semantic*, 2) *Dual-Stream*, and 3) *Single-Stream*. The results, as depicted in Table IV-A.2, clearly showcase the superior performance of Lite-SVO across the other methods. Impressively, Lite-SVO demonstrates a remarkable performance enhancement of 9.76% in AbsRel when compared to the previously leading *Single-Stream* architecture, SimVODIS. The qualitative results are also displayed in Fig. 6. It is evident that the proposed Lite-SVO exhibits distinct and precise object boundaries compared to Monodepth2, Guizilini *et al.*, and SimVODIS.

c) Semantic Performance: In the evaluation of semantic estimation performance, we conduct experiments on the KITTI test dataset, following the data split provided by Eigen [33]. The results presented in Table III demonstrate that our proposed Lite-SVO outperforms the current *Dual-Stream* architecture method, achieving a remarkable performance enhancement in semantic estimation accuracy.

TABLE III

SEMANTIC SEGMENTATION RESULTS ON THE KITTI DATASET

Method	Train	mIoU
SGDepth [30]	Cityscapes [34]	51.6
FSRE-Depth [3]	KITTI	56.6
Lite-SVO (Ours)	KITTI	**58.5**

TABLE IV

GENERALIZATION ANALYSIS ON THE AIRDOS-SHIBUYA DATASET

Method	Road Crossing (RMSE)	
	V	VII
Monodepth2 [22]	0.543	0.688
MotionHint (M2) [23]	0.666	0.624
FSRE-Depth [3]	0.736	0.684
Lite-SVO (Ours)	**0.422**	**0.569**

TABLE V

ABLATION STUDY ON THE KITTI DATASET

Method	AbsRel ↓	SqRel ↓	RMSE ↓
Lite-SVO w/o C-FS	0.114	0.799	4.646
Lite-SVO w/o M-FS	0.112	0.803	4.644
Lite-SVO	**0.111**	**0.786**	**4.576**

2) Generalization Analysis: We examine Lite-SVO on the AirDOS-Shibuya dataset to showcase its impressive generalization ability across diverse scenes. As illustrated in Table IV, we conduct a comprehensive comparison of Lite-SVO against the other methods in challenging scenarios such as Road Crossing V and VII. Apparently, these results show that Lite-SVO attains exceptional generalization performance.

3) Ablation Study: To further show the effectiveness of Lite-SVO, we conduct extensive ablation studies on the KITTI dataset. The results are reported in Table V, where we compare two proposed modules, i.e., the C-FS module and the M-FS module. The results indicate that both of our proposed modules are efficient, leading to performance improvements of Lite-SVO. Specifically, the C-FS module improves spatial-wise feature sharing between semantics and depth, while the M-FS module enhances spatial-wise feature for pose estimation. These findings underscore the significance of these modules in improving the performance of *Single-Stream* SVO methods.

4) Complexity and Timing Evaluation:

a) Complexity Evaluation: Lite-SVO's model complexity is compared with recent advanced methods on an NVIDIA 3080 GPU. As presented in Table VI, Lite-SVO exhibits the lowest parameter number when compared to the other methods, and thus facilitates its applications on edge devices. Though SimVODIS is also a *Single-Stream* method, our proposed Lite-SVO can reduce its parameters by 45.69% whilst the accuracy is remarkably increased by 79.83% in pose estimation.

TABLE VII

TIMING ANALYSIS

Method	Sem	Jetson Xavier NX	
		FLOPs (G)	Speed (s)
SGDepth [30]	✓	35.73	0.279
SimVODIS [5]	✓	380.33	1.583
Lite-SVO (Ours)	✓	**25.99**	**0.246**

b) Timing Evaluation: The FLOPs (floating point of operations) and inference speed are computed on an NVIDIA Jetson Xavier NX. In comparison with SGDepth [30] and SimVODIS [5], the results in Table VII demonstrate that Lite-SVO achieves a satisfactory speed, making it highly feasible for deployment on edge devices. Notably, Lite-SVO outperforms SimVODIS not only in inference speed, exhibiting a remarkable 84.46% improvement, but also exhibits satisfactory accuracy performance. These results further verify the advantages of Lite-SVO.

V. CONCLUSION

In this work, we propose a lightweight and efficient SVO called Lite-SVO, which tackles the existing *Single-Stream* SVO problem of multi-task features inconsistency by designing a novel multi-feature sharing architecture. Moreover, Lite-SVO further enhances tasks interaction within the multi-feature sharing architecture. The designed C-FS module improves depth and semantic feature sharing, while the proposed M-FS module captures and fuses the spatial-wise depth and semantic information, resulting in a significant performance boost for Lite-SVO. Extensive experiments show that Lite-SVO not only achieves outstanding performance, but also demands fewer computational resources, when compared to the other state-of-the-art SVO methods.

ACKNOWLEDGMENT

This work was partially supported by Suzhou Foreign Experts Project under grant No. E290010201, Jiangsu Funding Program for Excellent Postdoctoral Talent under grant No.2023ZB547, National Key Research and Development Program of China under grant No.2021YFB3201600, Suzhou Science and Technology Program under grant No.SSD2023001, and National Natural Science Foundation of China under grant No.62074159.

TABLE VI

MODEL COMPLEXITY ANALYSIS

Method	Sem	Params. (M)					
		Encoder	P-Encoder	Depth	Semantics	Pose	Total
SC-SfMLearner [20]	✗	-	-	80.88	-	1.60	82.48
Monodepth2 [22]	✗	11.70	11.70	3.15	-	1.84	28.39
Guizilini *et al.* [2]	✓	11.70	11.70	3.15	25.57	1.31	53.43
Insta-DM [24]	✓	11.70	11.70	3.15	44.34	1.32	72.21
SGDepth [30]	✓	12.36	11.19	1.97	1.97	1.31	28.80
SAFENet [4]	✓	11.70	11.70	6.75		1.84	31.99
SimVODIS [5]	✓	25.57		2.66	18.77	2.99	49.99
Lite-SVO (Ours)	✓	11.70		11.87		1.84+1.74	**27.15**

REFERENCES

[1] C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. Reid, and J. J. Leonard, "Past, Present, and Future of Simultaneous Localization and Mapping: Toward the Robust-Perception Age," *IEEE Transactions on Robotics*, vol. 32, no. 6, pp. 1309–1332, 2016.

[2] V. Guizilini, R. Hou, J. Li, R. Ambrus, and A. Gaidon, "Semantically-Guided Representation Learning for Self-Supervised Monocular Depth," *CoRR*, vol. abs/2002.12319, 2020. [Online]. Available: https://arxiv.org/abs/2002.12319

[3] H. Jung, E. Park, and S. Yoo, "Fine-Grained Semantics-Aware Representation Enhancement for Self-Supervised Monocular Depth Estimation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2021, pp. 12 642–12 652.

[4] J. Choi, D. Jung, D. Lee, and C. Kim, "Self-supervised Monocular Depth Estimation with Semantic-aware Feature Extraction," *CoRR*, vol. abs/2010.02893, 2020. [Online]. Available: https://arxiv.org/abs/2010.02893

[5] U.-H. Kim, S.-H. Kim, and J.-H. Kim, "SimVODIS: Simultaneous Visual Odometry, Object Detection, and Instance Segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 1, pp. 428–441, 2022.

[6] U.-H. Kim, S.-H. Kim, and J.-H. Kim, "SimVODIS++: Neural Semantic Visual Odometry in Dynamic Environments," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 4244–4251, 2022.

[7] J. Huang, S. Yang, T.-J. Mu, and S.-M. Hu, "ClusterVO: Clustering Moving Instances and Estimating Visual Odometry for Self and Surroundings," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[8] X.-Y. Kuo, C. Liu, K.-C. Lin, and C.-Y. Lee, "Dynamic Attention-Based Visual Odometry," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2020.

[9] J. Dai, X. Gong, Y. Li, J. Wang, and M. Wei, "Self-Supervised Deep Visual Odometry Based on Geometric Attention Model," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 3, pp. 3157–3166, 2023.

[10] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention Is All You Need," 2017.

[11] X. Chen, B. Yan, J. Zhu, D. Wang, X. Yang, and H. Lu, "Transformer Tracking," in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 8122–8131.

[12] X. Wei, T. Zhang, Y. Li, Y. Zhang, and F. Wu, "Multi-Modality Cross Attention Network for Image and Sentence Matching," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 10 938–10 947.

[13] J. Chen, W. Rao, Z. Wang, Z. Wu, Y. Wang, T. Yu, S. Shang, and H. Meng, "Speech Enhancement with Fullband-Subband Cross-Attention Network," in *Proc. Interspeech 2022*, 2022, pp. 976–980.

[14] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "CBAM: Convolutional Block Attention Module," in *Computer Vision – ECCV 2018: 15th European Conference, Munich, Germany, September 8–14, 2018, Proceedings, Part VII*. Berlin, Heidelberg: Springer-Verlag, 2018, p. 3–19.

[15] A. Ali, H. Touvron, M. Caron, P. Bojanowski, M. Douze, A. Joulin, I. Laptev, N. Neverova, G. Synnaeve, J. Verbeek, and H. Jegou, "XCiT: Cross-Covariance Image Transformers," in *Advances in Neural Information Processing Systems*, vol. 34. Curran Associates, Inc., 2021, pp. 20 014–20 027.

[16] D.-A. Clevert, T. Unterthiner, and S. Hochreiter, "Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs)," in *ICLR (Poster)*, 2016. [Online]. Available: http://arxiv.org/abs/1511.07289

[17] A. F. Agarap, "Deep Learning using Rectified Linear Units (ReLU)," *ArXiv*, vol. abs/1803.08375, 2018. [Online]. Available: https://api.semanticscholar.org/CorpusID:4090379

[18] S. Wang, R. Clark, H. Wen, and N. Trigoni, "End-to-end, sequence-to-sequence probabilistic visual odometry through deep neural networks," *The International Journal of Robotics Research*, vol. 37, no. 4-5, pp. 513–542, 2018. [Online]. Available: https://doi.org/10.1177/0278364917734298

[19] Z. Teed and J. Deng, "DeepV2D: Video to Depth with Differentiable Structure from Motion," in *International Conference on Learning Representations*, 2020. [Online]. Available: https://openreview.net/forum?id=HJeO7RNKPr

[20] J.-W. Bian, Z. Li, N. Wang, H. Zhan, C. Shen, M.-M. Cheng, and I. Reid, "Unsupervised Scale-Consistent Depth and Ego-Motion Learning from Monocular Video," in *Proceedings of the 33rd International Conference on Neural Information Processing Systems*. Red Hook, NY, USA: Curran Associates Inc., 2019.

[21] Z. Yin and J. Shi, "GeoNet: Unsupervised Learning of Dense Depth, Optical Flow and Camera Pose," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 1983–1992.

[22] C. Godard, O. Mac Aodha, M. Firman, and G. J. Brostow, "Digging Into Self-Supervised Monocular Depth Estimation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

[23] C. Wang, Y.-P. Wang, and D. Manocha, "MotionHint: Self-Supervised Monocular Visual Odometry with Motion Constraints," in *2022 International Conference on Robotics and Automation (ICRA)*, 2022, pp. 1265–1272.

[24] S. Lee, S. Im, S. Lin, and I. S. Kweon, "Learning Monocular Depth in Dynamic Scenes via Instance-Aware Projection Consistency," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 3, 2021, pp. 1863–1872.

[25] C. Godard, O. M. Aodha, and G. J. Brostow, "Unsupervised Monocular Depth Estimation with Left-Right Consistency," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul 2017. [Online]. Available: http://dx.doi.org/10.1109/cvpr.2017.699

[26] Y. Zhu, K. Sapra, F. A. Reda, K. J. Shih, S. Newsam, A. Tao, and B. Catanzaro, "Improving Semantic Segmentation via Video Propagation and Label Relaxation," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 8848–8857.

[27] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision Meets Robotics: The KITTI Dataset," *Int. J. Rob. Res.*, vol. 32, no. 11, p. 1231–1237, sep 2013. [Online]. Available: https://doi.org/10.1177/0278364913491297

[28] Y. Qiu, C. Wang, W. Wang, M. Henein, and S. Scherer, "AirDOS: Dynamic SLAM benefits from Articulated Objects," in *2022 International Conference on Robotics and Automation (ICRA)*, 2022, pp. 8047–8053.

[29] Z. Wang, X. Dai, Z. Guo, C. Huang, and H. Zhang, "Unsupervised Monocular Depth Estimation With Channel and Spatial Attention," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–11, 2022.

[30] M. Klingner, J.-A. Termöhlen, J. Mikolajczyk, and T. Fingscheidt, "Self-Supervised Monocular Depth Estimation: Solving the Dynamic Object Problem by Semantic Guidance," in *European Conference on Computer Vision (ECCV)*, 2020.

[31] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," *CoRR*, vol. abs/1412.6980, 2014.

[32] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A benchmark for the evaluation of RGB-D SLAM systems," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012, pp. 573–580.

[33] D. Eigen, C. Puhrsch, and R. Fergus, "Depth Map Prediction from a Single Image Using a Multi-Scale Deep Network," in *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, ser. NIPS'14. Cambridge, MA, USA: MIT Press, 2014, p. 2366–2374.

[34] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The Cityscapes Dataset for Semantic Urban Scene Understanding," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 3213–3223.