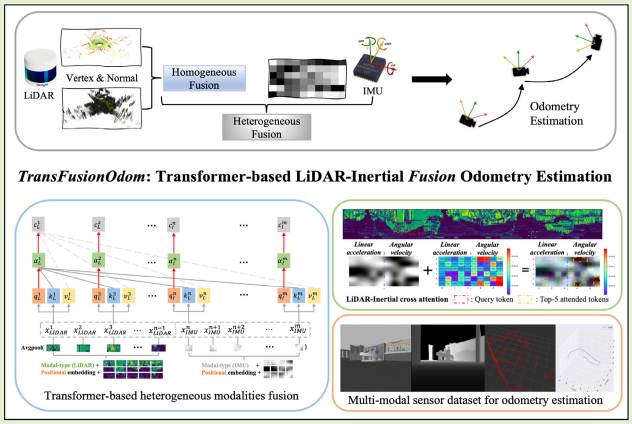
# TransFusionOdom: Transformer-Based LiDAR-Inertial Fusion Odometry Estimation

Leyuan Sun, Guanqun Ding, Yue Qiu, Yusuke Yoshiyasu, *Member, IEEE*, and Fumio Kanehiro, *Member, IEEE*

***Abstract*—Multimodal fusion of sensors is a commonly used approach to enhance the performance of odometry estimation, which is also a fundamental module for mobile robots. Recently, learning-based approaches garner the attention in this field, due to their robust nonhand-crafted designs. However, the question of *How to perform fusion among different modalities in a supervised sensor fusion odometry estimation task?* is one of the challenging issues still remaining. Some simple operations, such as elementwise summation and concatenation, are not capable of assigning adaptive attentional weights to incorporate different modalities efficiently, which makes it difficult to achieve competitive odometry results. Besides, the Transformer architecture has shown potential for multimodal fusion tasks, particularly in the domains of vision with language. In this work, we propose an end-to-end supervised Transformer-based LiDAR-Inertial fusion framework (namely TransFusionOdom) for odometry estimation. The multiattention fusion module demonstrates different fusion approaches for homogeneous and heterogeneous modalities to address the overfitting problem that can arise from blindly increasing the complexity of the model. Additionally, to interpret the learning process of the Transformer-based multimodal interactions, a general visualization approach is introduced to illustrate the interactions between modalities. Moreover, exhaustive ablation studies evaluate different multimodal fusion strategies to verify the performance of the proposed fusion strategy. A synthetic multimodal dataset is made public to validate the generalization ability of the proposed fusion strategy, which also works for other combinations of different modalities. The quantitative and qualitative odometry evaluations on the KITTI dataset verify that the proposed TransFusionOdom can achieve superior performance compared with other learning-based related works.**




***Index Terms*— Attention mechanisms, LiDAR-inertial odometry (LIO), multimodal learning, sensor data fusion, transformer.**

Manuscript received 6 July 2023; revised 25 July 2023; accepted 25 July 2023. Date of publication 10 August 2023; date of current version 14 September 2023. This work was supported in part by a research project grant from Japan Science and Technology (JST) SPRING Fellowship Program under Grant JPMJSP2124 and in part by Japan Society for the Promotion of Science (JSPS) KAKENHI in Japan under Grant 23H03426. The associate editor coordinating the review of this article and approving it for publication was Dr. Guofa Li. *(Corresponding author: Leyuan Sun.)*

Leyuan Sun is with the CNRS-AIST Joint Robotics Laboratory (JRL), IRL, and the Computer Vision Research Team, Artificial Intelligence Research Center (AIRC), National Institute of Advanced Industrial Science and Technology (AIST), Tsukuba 305-8560, Japan (e-mail: son.leyuansun@aist.go.jp).

Guanqun Ding is with the Digital Architecture Research Center (DigiARC), National Institute of Advanced Industrial Science and Technology (AIST), Tokyo 135-0064, Japan (e-mail: guanqun.ding@aist.go.jp).

Yue Qiu and Yusuke Yoshiyasu are with the Computer Vision Research Team, Artificial Intelligence Research Center (AIRC), National Institute of Advanced Industrial Science and Technology (AIST), Tsukuba 305-8560, Japan (e-mail: qiu.yue@aist.go.jp; yusuke-yoshiyasu@aist.go.jp).

Fumio Kanehiro is with the CNRS-AIST Joint Robotics Laboratory (JRL), IRL, National Institute of Advanced Industrial Science and Technology (AIST), Tsukuba 305-8560, Japan, and also with the Department of Intelligent and Mechanical Interaction Systems, Graduate School of Science and Technology, University of Tsukuba, Tsukuba 305-0006, Japan (e-mail: f-kanehiro@aist.go.jp).

Digital Object Identifier 10.1109/JSEN.2023.3302401

## I. Introduction

SENSOR fusion is a popular topic in robotics for several decades. In robotics, two types of sensors are commonly used: exteroceptive sensors and proprioceptive sensors. Exteroceptive sensors, such as cameras, LiDAR, ultrasonic sensors, and radar, provide rich and surrounding-sensitive information, but with a low frequency of around 30 Hz. However, proprioceptive sensors, including inertial measurement units (IMUs), wheel odometers, and joint encoders, can estimate their own state at a high frequency of around 200 Hz, but they tend to drift over time. Hence, combining these two types of sensors is implemented in a wide range of robotics tasks.

In addition, sensor data fusion can be categorized into two categories: homogeneous fusion and heterogeneous fusion. Homogeneous fusion deals with modalities that have the same shape and a certain correspondence between each element, such as RGB with depth. However, heterogeneous fusion does not have the same correspondence between modalities, such as RGB and point cloud. Compared with the fusion between homogeneous modalities, heterogeneous fusion presents more challenges [11], [12] because homogeneous modalities are naturally aligned featurewise.

In the context of our task, odometry estimation involves the computation of an object's position and orientation by utilizing sensor measurements. Within the sensors field, odometry estimation plays a vital role in precise localization, mapping, and tracking applications by utilizing different modalities of sensor data. It contributes to advancements in robotics [1], [8], [10], autonomous vehicles [13], [14], [15], and augmented reality technologies [16], [17]. In particular, visual-inertial odometry (VIO) [1], [6], [18] and LiDAR-inertial odometry (LIO) [5], [19], [20] are commonly used combinations of sensor fusion for odometry estimation in the localization systems for mobile robots.

Sensor fusion technology has been dominated by filter-based approaches such as Kalman filter (KF) and extended KF (EKF) in the past several decades. However, the accuracy of filter-based methods is limited by linearization errors [9]. Particularly in the field of LIO, fusion approaches can be categorized into loosely coupled and tightly coupled methods (see Section II-B), depending on how raw measurements are integrated as constraints for optimization. Recently, numerous studies have confirmed the superiority of data-driven learning-based solutions over traditional solutions in odometry estimation such as those proposed by Tu et al. [1], Iwaszczuk et al. [5], Clark et al. [7], and Chen et al. [21]. Compared with the existing approaches, the majority of the existing learning-based sensor fusion odometry methods are based on convolutional neural network (CNN)-recurrent neural network (RNN) frameworks, which can be inefficient due to the inability to parallelize the RNN component during training and inference [1], [22]. Furthermore, the 3-D raw point cloud data from LiDAR sensors is noisier, sparser, and more irregular compared to RGB images used in visual odometry (VO) tasks, making it more challenging to directly apply convolutional processing [15]. Meanwhile, Transformers [23] have been used for fusing different sensor data in the field of VO, showing good performance in accuracy and robustness [1], [2], but Transformer architecture has not been explored in LIO yet.

However, our motivation comes from the problem that naively increasing the complexity of Transformer-based fusion networks and handling multiple modalities together can lead to the overfitting problem, especially considering that LiDAR data have various formats such as vertex and normal maps [14], [15] generated from raw 3-D point clouds. Moreover, although Transformer-based fusion has achieved good performance, how the modalities are aggregated inside the Transformer is rarely investigated and interpreted.

To tackle these challenges mentioned above, we propose an end-to-end Transformer-based multimodal fusion network for odometry estimation (namely TransFusionOdom). The overview of the proposed TransFusionOdom is shown in Fig. 1. The multiattention fusion approach, which combines the soft mask attention fusion (SMAF) and Transformer, is designed to fuse a mixture of homogeneous and heterogeneous sensor data, while avoiding the overfitting problem that occurs if the complexity of the Transformer-based fusion network is overly increased [24]. To achieve homogeneous data fusion between LiDAR's vertex and normal

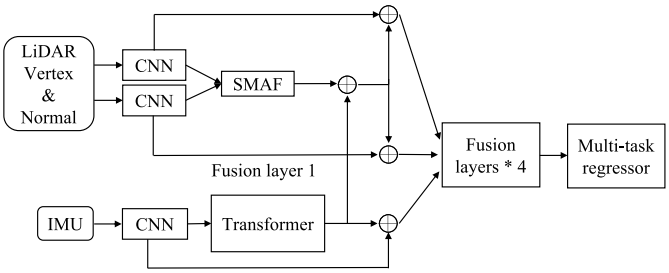



Fig. 1. Simplified overview of proposed TransFusionOdom, inputs are LiDAR raw point cloud and IMU measurements, outputs are six-DoF pose including translation and orientation.

estimation, we implement the SMAF [3], [25], [26]. This allows the adaptive weights assignment on aligned features between homogeneous modalities using a smaller number of network parameters than Transformers, avoiding overfitting even when training the model with a limited amount of data. For heterogeneous data fusion between LiDAR and IMU, we introduce the Transformer [23] encoder architecture. Through the visualization of attentions, the incorporation between heterogeneous modalities is illustrated.

Although learning-based frameworks offer several advantages (see Section II-A), *How should we perform fusion among different modalities in a supervised sensor fusion odometry estimation task?* is still one of the key issues that remains, which we introduce and comprehensively evaluate in this study. Additionally, we extend a proposed fusion strategy to different combinations of modalities to evaluate generalization on our proposed synthetic dataset. From a global perspective, the objective is to develop a fusion module in the field of learning-based multimodal fusion, which serves as a general component for tasks that utilize multiple modalities as input.

The main contributions of this article can be summarized as follows.

1) To the best of our knowledge, TransFusionOdom is the first end-to-end Transformer-based multimodal fusion network for LIO estimation task, which achieves the superior performance on KITTI dataset than previous learning-based approaches. The multiattention fusion module is proposed to achieve the adaptive weights learning for the fusion between homogeneous (vertex and normal map of LiDAR) and heterogeneous modalities (integrated LiDAR and IMU measurement).
2) Exhaustive ablation studies are conducted to evaluate different fusion strategies for a mix of homogeneous and heterogeneous modalities fusion task. In addition, a general visualization approach is introduced to demonstrate the interactions between two modalities inside the Transformer architecture, which could enhance the interpretability of Transformer-based multimodal fusion framework.
3) A synthetic multimodal dataset is publicly available,[1] which is utilized to evaluate the generalization ability of

[1]https://github.com/RakugenSon/Multi-modal-dataset-for-odometry-estimation

TABLE I
RELATED WORKS ON SUPERVISED SENSOR FUSION FOR ODOMETRY ESTIMATION

| Supervised fusion odometry estimation | Modalities | Fusion strategy | Fusion position | Representation of rotation | Distance function |
|---|---|---|---|---|---|
| TransFusionOdom (ours) | LiDAR+IMU | SMAF + Transformer-based | Multi-layer fusion | Euler angle + se(3) | L2 |
| EMA-VIO [1] | RGB+IMU | Transformer-based | Middle fusion | Euler angle | L2 |
| AFT-VO [2] | Multi RGBs | Transformer-based | Middle fusion | Not explain | L2 |
| Chen *et al.* [3] | RGB+IMU | Soft and Hard mask | Middle fusion | Quaternion | L1 |
| Son *et al.* [4] | LiDAR+IMU | Soft mask | Late fusion | Axis-angle | L2 |
| DeepLIO [5] | LiDAR+IMU | Soft mask | Middle fusion | Quaternion + se(3) | L2 |
| ATVIO [6] | RGB+IMU | Cross-domain attention mask | Middle fusion | Euler angle | Others |
| VINet [7] | RGB+IMU | Concat. | Middle fusion | Quaternion + se(3) | L1 |
| Li *et al.* [8] | Laser scan + IMU | Concat. | Middle fusion | 1D angle | L2 |
| VIIONet [9] | RGB+IMU image | Concat. | Middle fusion | Not explain | Gaussian Process Regression |
| HVIOnet [10] | RGB+IMU | Concat. | Middle fusion | Not explain | L2 |

the proposed fusion strategy with different combinations of modalities. This dataset also facilitates easy testing of other fusion algorithms and enables transfer learning within the community.

## II. RELATED WORK

In this section, we provide a brief introduction to geometry-based and learning-based odometry estimation. Since our work primarily focuses on sensor fusion, we discuss traditional sensor fusion approaches in robotics, as well as the categorization of existing learning-based supervised multimodal fusion for odometry estimation, which is presented in Table I.

### A. Geometry-Based and Learning-Based Odometry Estimation

Under ideal situations, traditional geometry-based or handcrafted systems are capable of achieving accurate and robust localization results. Examples include VO systems like ORB-SLAM2 [27], VIO systems like VINS-Mono [18], LiDAR Odometry (LO) systems like LOAM [28], and LIO systems like LIO-SAM [19]. However, in real-world scenarios, the presence of sensor data noise, challenges in illumination, texture-less environments, and dynamic objects are inevitable, which can negatively impact the reliability of traditional geometry-based approaches. With the rapid development of data-driven learning-based approaches in various fields, the advantages of learning-based localization systems can be summarized based on the following three aspects [29].

1) Learning-based approaches can harness the power of highly expressive deep neural networks as universal model, enabling them to automatically discover task-relevant feature representations for challenging situations.
2) Learning-based methods empower spatial and temporal intelligence systems to acquire knowledge from past experiences and actively utilize new information.
3) It has the capacity to effectively utilize the expanding quantity of sensor data for a large dataset and computational power using parallel CUDA technology.

### B. Traditional Sensor Fusion in Robotics

LO can be discussed from the perspective of point cloud registration algorithms, which usually involve some approaches of scan-to-scan or scan-to-local-map registration, such as iterative closest point (ICP) [30] and generalized-ICP (GICP) [31]. Our study mainly focuses on the fusion strategy of LIO, which is generally classified into loosely coupled and tightly coupled methods in the traditional sensor fusion field.

*1) Loosely-Coupled LIO:* Loosely coupled LIO achieves point cloud distortion correction by fusing IMU data. IMU observations can also provide an initial pose estimation for point cloud registration in IMU-aided LOAM [28] and LeGO-LOAM [32], and even gravity direction observations. The final result can be directly output as the point cloud registration results or obtained by fusing the IMU integration results (predictions) with point cloud registration results (observations) using filtering techniques, such as [33]. However, in loosely coupled LIO, the fusion of LiDAR and IMU is limited to the result level, without taking into account the intrinsic constraints between these two types of observations. Consequently, if one of the sensors experiences estimation failure, it can lead to system deterioration and even divergence, which ultimately affects the precision and robustness of the system [34].

*2) Tightly Coupled LIO:* In the tightly coupled LIO, the internal constraints between LiDAR and IMU observations are fully considered, and they mutually influence each other to determine the final estimations. Theoretically, tightly coupled LIO can handle degradation scenarios that cannot be addressed by the previous approaches, such as long tunnel environments or highly dynamic motion conditions. The core of tightly coupled LIO lies in the design of the state estimator, which can be achieved through graph optimization (sliding/time window optimization) techniques such as LIO-SAM [19] and D-LIOM [35], or filtering methods like iterative EKF (IEKF), for example, FAST-LIO [36] and FAST-LIO2 [37].

The advantage of sliding window optimization lies in its ability to jointly estimate the state variables at multiple time steps, resulting in higher accuracy. However, this approach comes at the cost of low computational efficiency. In the early stages, LIO based on sliding window optimization could not guarantee real-time performance [38]. However, IEKF offers high computational efficiency with good real-time performance but it is limited by the number of observations and the early marginalization of the initial state variables, resulting in lower accuracy compared to sliding window optimization.

Furthermore, there is an issue [34] with pose graph optimization in LIO. When incorrect odometry estimation is added as erroneous constraints in the pose graph, their negative effects cannot be eliminated. In general, these two different tightly coupled approaches represent a trade-off between accuracy and efficiency.

Recently, tightly coupled LiDAR-inertial-VO (LIVO) has been developed for accuracy and robust localization, such as FAST-LIVO [39], $R^2$LIVE [40] and CamVox [41]. The former two systems contain the sub-LIO and sub-VIO, and CamVox is to adapt livox LiDAR to the ORB-SLAM2 [27], VINS-Mono [18], and LOAM [28].

Compared with learning-based approaches, traditional sensor fusion approaches face difficulties in being as flexible as learning-based approaches to assign adaptive weights to each sensor in real-time using attention mechanisms [29], making it challenging to represent the reliability of each sensor.

### C. Learning-Based Multimodal Fusion for Odometry Estimation

Because there is a limited number of works related to learning-based LIO, and our main focus is on investigating different fusion strategies in learning-based approaches, Table I presents some related works on supervised sensor fusion odometry estimation. These works are not only limited to LIO but also include VIO and other types of sensors. We have categorized these related works based on their fusion strategies and positions for further comprehensive comparisons.

*1) How to Fuse Multiple Modalities:* Under the category of fusion strategies, there are concatenate-based approaches such as VINet [7], Li et al.'s method [8], VIIONet [9], and HVIOnet [10]. These methods directly concatenate the features from different modalities, which means that the weight of each sensor is constant and equal.

However, Chen et al.'s [3] proposed method, Son et al.'s [4] developed method, DeepLIO [5], and ATVIO [6] use the soft mask-based method, which relies on multilayer perceptron (MLP) to learn the weights of each sensor. These methods are similar to self-attention in PointLoc [25]. The learnable mask could remove outliers by giving low weight, and the adaptive reliability of each sensor during fusion improves the accuracy and robustness of odometry estimation through attentional mechanisms.

However, the ability of the simple MLP-based attentional mask architecture is not sufficient to handle challenging situations in the real world, such as reflection ground and overcast days [1]. Inspired by ViLT [42], the Transformer [23] architecture has shown impressive performance in the field of multimodal fusion, not only limited to odometry estimation tasks but also in navigation [43], semantic segmentation, and object detection tasks [44]. In EMA-VIO [1] and AFT-VO [2], the Transformer architecture is used to fuse multiple modalities, and through challenging real-world experiments, it has shown higher accuracy and robustness than some soft mask-based approaches. But these works did not consider the effect of fusion position, and the Transformer is used as a black box without interpretability to explain how two modalities interact and fusion inside the Transformer architecture.

*2) Where to Conduct Fusion:* Another important issue that needs to be taken into consideration is the fusion position. In Table I, early fusion means fusing source modalities and then feeding them into the backbone for feature extraction. Middle fusion means using different backbones and fusing them before feeding into one regressor. If we feed into different regressors and then fuse them, it is defined as late fusion. Channel Exchange (CE) [11] and MLF-VO [45] have confirmed that multilayer fusion is better than the previous three fusion positions. However, since the Transformer is a data-hungry model [46] compared to CNN-based models, implementing the Transformer with multilayer fusion requires taking the overfitting issue into consideration, which we discuss in the ablation study Section IV-C.

*3) Strength of Transformer for Multimodal Fusion:* Although there are already many multimodal fusion tasks that deploy Transformer architecture, the strengths of Transformer for multimodal learning is still an open question [22]. Based on literature and surveys, these main points are summarized.

1) Transformers possess an inherent global aggregation nature, allowing them to perceive nonlocal patterns.
2) Leveraging their large model capacity, Transformer models excel in handling challenging domain gaps between different modalities more effectively.
3) Due to their parallel computation in both training and inference, Transformers exhibit improved training and inference efficiency compared to RNN-based models (LSTM and GRU), making them well-suited for modeling time-series and sequence modalities [1].
4) Tokenization enables Transformers to flexibly organize multimodal inputs, enhancing their flexibility in handling different types of data.
5) Transformers possess the ability to encode implicit knowledge inside of different modalities, and also its multihead mechanism introduces multiple modeling abilities that further improve the expressive capacity.

## III. Methodology

In this section, we illustrate the proposed framework TransFusionOdom in detail, the network architecture is shown in Fig. 2. This framework includes *Multimodal* input (LiDAR and IMU), *Multiscale* modality tokens, *Multiattention* fusion (SMAF and Transformer-encoder), *Multilayer* fusion, and *Multitask* regressor. The subscripts $L$ and $I$ of all the variables represent LiDAR- and IMU-related information. The superscripts $l_{t,t\in[1,4]}$ represent the features at different layers.

### A. LiDAR and IMU Data Preprocessing

The input modalities are LiDAR 3-D point cloud and IMU signal. Since we use ResNet34 and ResNet18 [47] CNN-based backbones to extract features, it is necessary to project a 3-D point cloud onto a 2-D plane $\mathbb{R}^3 \Rightarrow \mathbb{R}^2$, and convert the IMU signal $\mathbb{R}^6$ to image as well. At every timestamp $t$, the consecutive LiDAR point cloud $\boldsymbol{P}_t$ and $\boldsymbol{P}_{t+1}$, along with all IMU measurements taken between them $\boldsymbol{I}_I^t$ are fed into the TransFusionOdom as input.

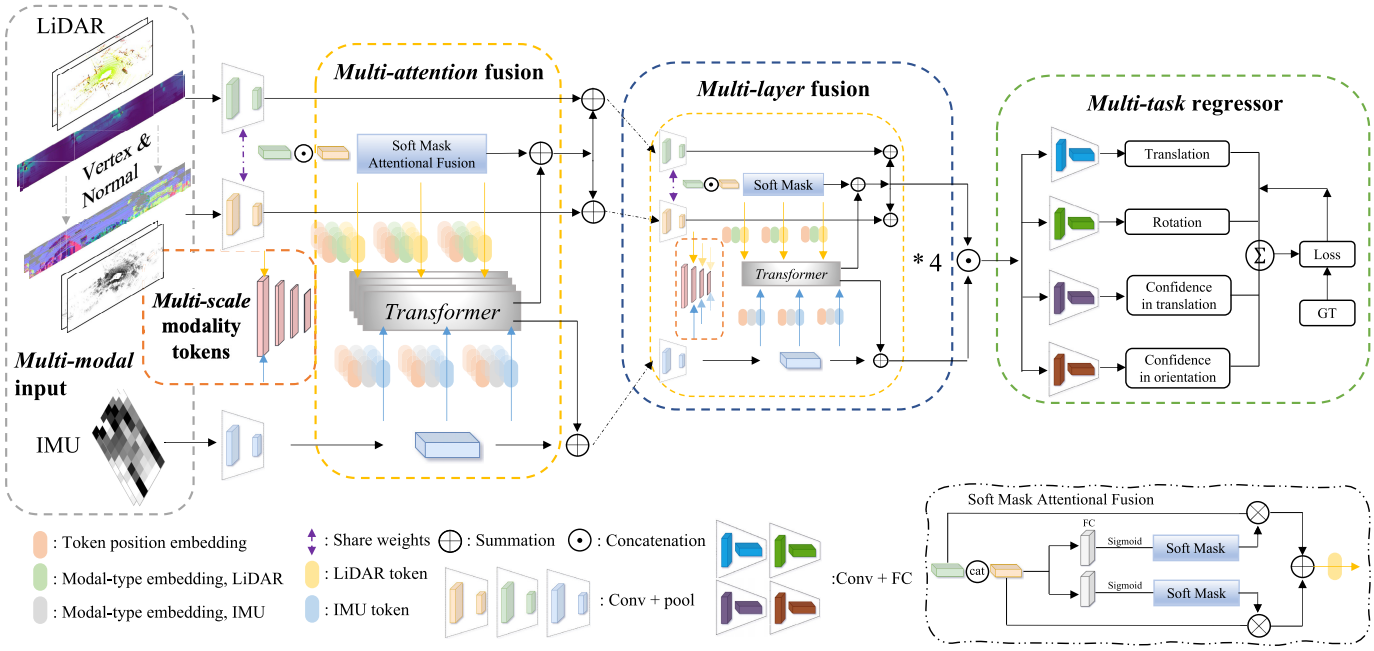



Fig. 2. Network architecture of the proposed TransFusionOdom. Given input data from LiDAR and IMU sensors, TransFusionOdom estimates the six-DoF pose (translation and orientation) as an LIO system. ***Multimodal input***: The LiDAR modality consists of vertex and normal maps, and the IMU modality is converted to a signal feature map. ***Multiattention fusion***: Fusion between homogeneous modalities within LiDAR is achieved using the SMAF approach, while fusion between heterogeneous modalities (LiDAR and IMU) is based on the Transformer. ***Multilayer fusion and Multiscale modality tokens***: Fusion is performed at multiple stages with multiscale feature maps. ***Multitask regressor***: Uncertainty is utilized as learnable weights to balance the error between translation and rotation.

*1) Vertex and Normal Map of LiDAR:* Each LiDAR point cloud $\boldsymbol{p}_t = (p_x^t, p_y^t, p_z^t)$ is projected to 2-D $(u, v)$ through a spherical projection method proposed in the studies of RangeNet [48], DeepLO [15], DeepLIO [5], and UnDeepLIO [49]. The projection can be calculated as follows to obtain the vertex maps $\boldsymbol{V}$:

$$\begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} \left(f_u - \arctan\left(\frac{p_y}{p_x}\right)\right) \Big/ \eta_u \\ \left(f_v - \arctan\left(\frac{p_z}{d}\right)\right) \Big/ \eta_v \end{pmatrix} \tag{1}$$

where $d = (p_x^2 + p_y^2 + p_z^2)^{1/2}$ denotes depth, $f_u$ and $f_v$ represent the horizontal and vertical angle in maximum. $\eta_u$ and $\eta_v$ are the resolutions of pixel representation in horizontal and vertical, respectively. This mapping somehow circumvents the problem that point clouds are cluttered and disordered [49].

The normal vector is useful for point cloud registration [50]. Like related works [5], [15], [49], [51], we implement the normal maps $\boldsymbol{N}$, which have the correspondent relation with the vertex map $\boldsymbol{V}$ in image coordinates ($\boldsymbol{v}_p \Rightarrow \boldsymbol{n}_p$). Each normal vector of point $\boldsymbol{p}$ is given as shown below

$$\boldsymbol{n}_p = \sum_{\boldsymbol{p}_i \in \mathcal{P}} w_{p_0,p} \left(\boldsymbol{v}_{p_0} - \boldsymbol{v}_p\right) \times w_{p_1,p} \left(\boldsymbol{v}_{p_1} - \boldsymbol{v}_p\right) \tag{2}$$

where $d(\cdot)$ is the range value and $w_{1,2} = e^{-0.5|d(\boldsymbol{v}_1) - d(\boldsymbol{v}_2)|}$ is a predefined weight, $\boldsymbol{p}$ denotes the center point of the four-point neighborhood $\mathcal{P}$ in up/right/down/left directions.

*2) IMU Signal Image:* The mainstream approach to process IMU data is to use the RNN LSTM, which is good at modeling sequential data, as proposed in Son et al. [4], UnDeepLIO [49], DeepLIO [5], Li et al. [8], Chen et al. [3], etc. However, LSTM has limitations in parallel computation [1] compared with CNN-based networks. Additionally, Weytjens and De Weerdt [52] explained that CNN-based methods are more robust than LSTM and require less time to learn the model.

Inspired by some human action recognition tasks [53], [54], we convert the IMU signal to images. This preprocessing is also necessary since we implement a multilayer fusion strategy, unlike CNN-based backbones, LSTM does not have intermediate outputs and cannot be parallelized during training and inference [1], [22]. Before feeding the data into the TransFusionOdom framework, we conduct denoising of the IMU signal because Brossard et al. [55] stated that denoising IMU bias and noise could improve the accuracy of state estimation. Similar to VIIONet [9], we apply the Savitzky-Golay filter [56] to filter the IMU high-frequency noise.

We extract the linear acceleration and angular velocity in $x$-, $y$-, and $z$-axis, as shown in Fig. 3. Because the frequency of IMU is higher than LiDAR, we assume there are $\gamma$ IMU measurements between $\boldsymbol{I}_L^t$ and $\boldsymbol{I}_L^{t+1}$. The raw IMU signal image $\boldsymbol{I}_I^{t,t+1}$ between timestamp $[t, t+1]$ is generated as follows:

$$\boldsymbol{I}_I^{t,t+1} = \begin{bmatrix} \boldsymbol{I}_0 \\ \boldsymbol{I}_1 \\ \boldsymbol{I}_2 \\ \cdots \\ \boldsymbol{I}_\gamma \end{bmatrix} = \begin{bmatrix} a_x^0, a_y^0, a_z^0, w_x^0, w_y^0, w_z^0 \\ a_x^1, a_y^1, a_z^1, w_x^1, w_y^1, w_z^1 \\ a_x^2, a_y^2, a_z^2, w_x^2, w_y^2, w_z^2 \\ \cdots \\ a_x^\gamma, a_y^\gamma, a_z^\gamma, w_x^\gamma, w_y^\gamma, w_z^\gamma \end{bmatrix}. \tag{3}$$

To avoid excessive information compression, particularly in the width dimension, when conducting multilayer fusion

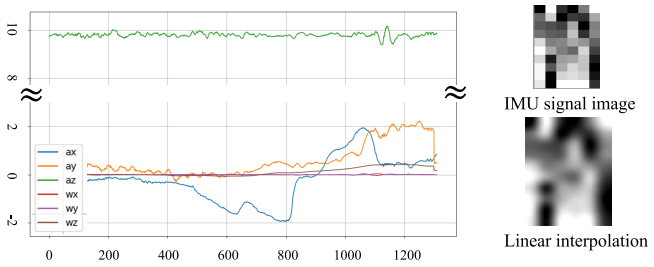



Fig. 3. Raw IMU signal plotted in linear acceleration and angular velocity, IMU signal image resized with linear interpolation.

using intermediate outputs of the ResNet model with the LiDAR modality, we normalize and resize the original IMU signal image, which is $\in \mathbb{R}^{p\times 6\times 3}$, to $(H_I, W_I)$. We conducted the experiments to compare the performance using learnable layer to enlarge the image and a fixed linear interpolation. The results show that the latter solution is better, possibly because the learnable layer has more uncertainty in extracting representative features at the following steps.

### B. Multiattention Fusion for a Mix of Homogeneous and Heterogeneous Modalities

In this section, we introduce the multiattention fusion module in detail, including SMAF and Transformer. The motivation behind employing different approaches for a mix of homogeneous and heterogeneous fusion is the quadratic computational complexity of self-attention in the number of tokens [57]. This can lead to an overfitting problem [58] with a limited number of datasets, especially when using a multilayer fusion strategy.

*1) Soft-Mask Attentional Fusion for Homogeneous Modalities:* After the initial ResNet layer, two backbones share the same weights to extract common features from the vertex and normal maps ($\boldsymbol{V}_p$ and $\boldsymbol{N}_p$). This weight sharing can help compress the size of the model. Because $\boldsymbol{V}_p$ and $\boldsymbol{N}_p$ have the same shape, a certain corresponding relationship exists between each element during preprocessing. The fusion of $\boldsymbol{V}_p$ and $\boldsymbol{N}_p$ is defined as a homogeneous multimodal fusion [11].

We conduct the SMAF during each layer of ResNet, which is an MLP-based learnable mask. Similar attention mechanisms are introduced in [6], [21], [25], and [59]. Here, $\boldsymbol{v}_p^{l_1}$ and $\boldsymbol{n}_p^{l_1}$ are the outputs of the first layer in ResNet34, MLP is used to convert features generated by concatenation $[\boldsymbol{v};\boldsymbol{n}]$ to mask vector $\boldsymbol{M}_v^{l_1}$ and $\boldsymbol{M}_n^{l_1}$, Sigmoid is used to reweight the mask to the range of $[0, 1]$. The whole processing is automatically parameterized by a network as follows:

$$\begin{aligned}\boldsymbol{M}_v^{l_1} &= \text{Sigmoid}\left(\text{MLP}_v\left[\boldsymbol{v}_p^{l_1};\boldsymbol{n}_p^{l_1}\right]\right)\\ \boldsymbol{M}_n^{l_1} &= \text{Sigmoid}\left(\text{MLP}_n\left[\boldsymbol{v}_p^{l_1};\boldsymbol{n}_p^{l_1}\right]\right).\end{aligned} \tag{4}$$

Utilizing the mask vectors, the input homogeneous modalities vertex and normal maps are reweighed by elementwise multiplication $\otimes$. This SMAF $\tau_{\text{soft}}(\boldsymbol{v},\boldsymbol{n})$ operation conducts at layer $t$ of ResNet, and is modeled as

$$\tau_{\text{soft}}\left(\boldsymbol{v}_p^{l_t},\boldsymbol{n}_p^{l_t}\right) = \boldsymbol{v}_p^{l_t}\otimes\boldsymbol{M}_v^{l_t} + \boldsymbol{n}_p^{l_t}\otimes\boldsymbol{M}_n^{l_t}. \tag{5}$$

*2) Transformer-Based Fusion for Heterogeneous Modalities:* The output of SMAF $\tau_{\text{soft}}(\boldsymbol{v}_p^{l_t},\boldsymbol{n}_p^{l_t})$ needs to be fused with the IMU modality $\boldsymbol{i}_p^{l_t}$ after each layer of ResNet, which is a heterogeneous multimodal fusion task. In this article, we utilize the Transformer to perform heterogeneous data fusion. Unlike LSTM, Transformer can perform parallel computation efficiently, which is the reason that we do not conduct RNN for feature extraction and regression like EMA-VIO [1]. Although, it is still necessary to manage the size of tokens for maintaining the efficiency and size of the entire model.

Different from ViT [57] and ViLT [42], which use image patches as token, we apply the average pooling operation to downsample the image patch for reducing the computational burden and obtain a set $\in \mathbb{R}^{H\times W\times C}$, which includes $[x_L^1, x_L^2, x_{Li}^3, \ldots, x_L^{n-1}]$ and $[x_I^n, x_I^{n+1}, x_I^{n+2}, \ldots, x_I^m]$. In this set, each element is treated as a token. We avoid the disorder and clutter of point clouds by positional encoding $\boldsymbol{x}_L^{\text{pos}}/\boldsymbol{x}_I^{\text{pos}} \in \mathbb{R}^{H\times W\times C}$, in addition to projecting a raw point cloud to image. We also apply the modal-type embedding $\boldsymbol{l}^{\text{type}}/\boldsymbol{i}^{\text{type}} \in \mathbb{R}^W$ to give prior knowledge about which token belongs to which modality. The modal-type embedding is generated by a learnable linear layer. The ability of differentiation between different sources could improve the performance of the Transformer, which has been validated in AFT-VO [2] and ViLT [42]. The set sequence, positional embedding, and modal-type embedding of each modality integrate together by elementwise summation as follows:

$$\bar{\boldsymbol{x}}_L = \left[x_L^1, x_L^2, x_{Li}^3, \ldots, x_L^{n-1}\right] + \boldsymbol{x}_L^{\text{pos}} \tag{6}$$

$$\bar{\boldsymbol{x}}_I = \left[x_I^n, x_I^{n+1}, x_I^{n+2}, \ldots, x_I^m\right] + \boldsymbol{x}_I^{\text{pos}} \tag{7}$$

$$\boldsymbol{G}^{\text{in}} = \left[\bar{\boldsymbol{x}}_L + \boldsymbol{l}^{\text{type}}; \bar{\boldsymbol{x}}_I + \boldsymbol{i}^{\text{type}}\right]. \tag{8}$$

The input to the Transformer-encoder is $\boldsymbol{G}^{\text{in}} \in \mathbb{R}^{M\times D_f}$, each token is a feature vector with a dimension of $D_f$. The query $\boldsymbol{Q}$, key $\boldsymbol{K}$, and value $\boldsymbol{V}$ are generated by linear transformation with $\boldsymbol{M}^q \in \mathbb{R}^{D_f\times D_q}$, $\boldsymbol{M}^k \in \mathbb{R}^{D_f\times D_k}$, and $\boldsymbol{M}^v \in \mathbb{R}^{D_f\times D_v}$, respectively,

$$\boldsymbol{Q} = \boldsymbol{G}^{\text{in}}\boldsymbol{M}^q, \quad \boldsymbol{K} = \boldsymbol{G}^{\text{in}}\boldsymbol{M}^k, \quad \boldsymbol{V} = \boldsymbol{G}^{\text{in}}\boldsymbol{M}^v. \tag{9}$$

Then the attention mechanism shown in Fig. 4 inside the Transformer-encoder is calculated by the following formulas:

$$\boldsymbol{\alpha}_{L,I} = \frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{D_k}} \tag{10}$$

$$\boldsymbol{C}_{L,I} = \text{softmax}(\boldsymbol{\alpha}_{L,I})\boldsymbol{V} \tag{11}$$

$$\boldsymbol{G}^{\text{out}} = \text{MLP}(\boldsymbol{C}_{L,I}) + \boldsymbol{G}^{\text{in}} \tag{12}$$

where $\boldsymbol{G}^{\text{out}}$ is the same shape with $\boldsymbol{G}^{\text{in}}$. There are several layers which are applied in the original Transformer-encoder, the multihead attention generates parallel $\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}$ and involves concatenating the attention value of $\boldsymbol{C}_{L,I}$.

The output of the fusion, $\boldsymbol{G}^{\text{out}}$, is up-sampled to recover to the original resolution through bilinear interpolation, which is the dimension of each layer's output from ResNet. After upsampling, elementwise summation is used to integrate the output with existing feature maps as residual learning [47] to prevent gradient degradation.

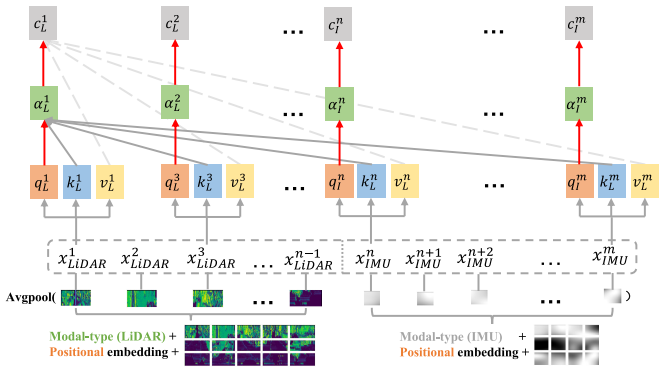



Fig. 4. Transformer-based fusion between LiDAR and IMU tokens.

In contrast to TransFuser [60], which scales tokens at a fixed shape of $H_0 = H_1 = H_2 = H_n$ and $W_0 = W_1 = W_2 = W_n$ for each layer, we propose a multiscale Transformer token fusion strategy where the dimensions of the tokens are gradually scaled to correspond with the shape of each layer in ResNet. Hence, this approach provides the different resolutions of tokens for multilayer fusion. The benefits of multiscale Transformer have been highlighted in many works [61], [62], [63], [64], which leverage the coarse-to-fine concept of the classic image pyramid architecture [65].

### C. Interpretable Multimodal Fusion Inside of Transformer

In this section, we demonstrate the interpretation of interactions between two modalities $\bar{\boldsymbol{x}}_L$ and $\bar{\boldsymbol{x}}_I$ inside the Transformer-encoder. Through the generation of the attention matrix $\boldsymbol{\alpha}_{L,I} \in \mathbb{R}^{m\times m} = [\alpha_L^1; \alpha_L^2; \ldots; \alpha_I^m]$, it can be split into four components which represent different self and cross-attentions. If the query and key belong to the same modality, such as $(\boldsymbol{x}_L^n \dagger \boldsymbol{x}_L^{n'})$ and $(\boldsymbol{x}_I^n \dagger \boldsymbol{x}_I^{n'})$, where $\dagger$ denotes the process of using the query to search the key, we define it as self-attention. Otherwise, $(\boldsymbol{x}_L^n \dagger \boldsymbol{x}_I^{n'})$ is defined as cross-attention. Based on this definition, in Fig. 5(a), the top-left of $\boldsymbol{\alpha}_{L,I}$ represents LiDAR self-attention, the top-right represents LiDAR-to-IMU cross-attention, the bottom-left represents IMU-to-LiDAR cross-attention, and the bottom-right represents IMU self-attention.

Each column in $\boldsymbol{\alpha}_{L,I}$ is the attention weights of one query token in two modalities. We can reshape the weights to their original resolution of corresponding modality. The relationship between query token and attended token with highest score could be observed as shown in Fig. 5(b). The detailed visualization results are discussed in Section IV-B.

### D. Multitask Regression

This section is the extension of our previous work CertainOdom [14]. In the supervised 6-D pose regression field, the objective is to predict the 6-D pose vector $\boldsymbol{p}$ including translation and rotation $\boldsymbol{x}$ and $\boldsymbol{q}$

$$\boldsymbol{p} = [\boldsymbol{x}, \boldsymbol{q}]. \tag{13}$$

The baseline method proposed in PoseNet [66] and employed in DeepVO [67], ATVIO [6], VINet [7], and

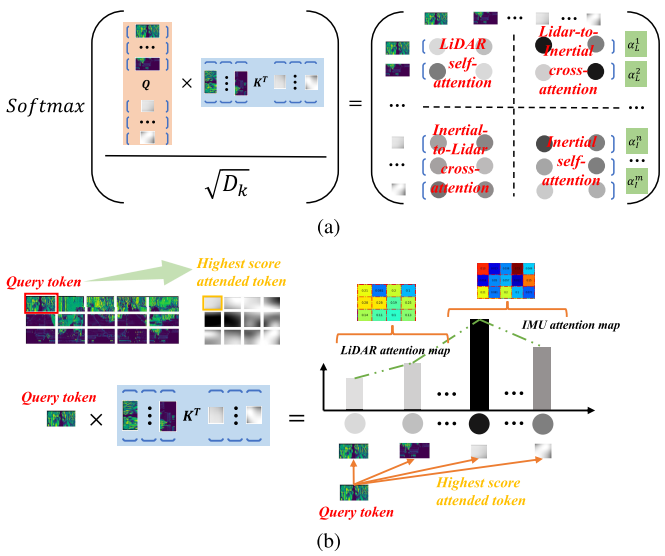



Fig. 5. Interpretation of interactions between LiDAR and IMU inside the Transformer. (a) Generation of attention matrix and the definition of self/cross attention component. (b) Generation of attention map and the relationship between query tokens and attended tokens with highest score.

SelectFusion [21] is as follows:

$$\text{loss} = \|\boldsymbol{x} - \hat{\boldsymbol{x}}\|_2 + \beta \left\| \frac{\boldsymbol{q}}{\|\boldsymbol{q}\|} - \hat{\boldsymbol{q}} \right\|_2 \tag{14}$$

where L2 loss is used as the distance function to calculate the error between prediction and ground truth $[\hat{\boldsymbol{x}}, \hat{\boldsymbol{q}}]$, the manually tuned hyperparameter $\beta$ is used to weigh the error between translation and rotation.

To overcome the problems that come from hyperparameters, DeepLO [15], Lo-Net [51], and MS-Transformer [68] introduce the following loss function:

$$\text{loss} = L_x \exp(-s_x) + s_x + L_q \exp(-s_q) + s_q \tag{15}$$

where $L_{x,q}$ denotes the distance function, $s_x$ and $s_q$ are the learned parameters to balance the error between translation and rotation.

The main contribution of our previous work CertainOdom [14] is to leverage the uncertainty which regressed from multidecoders as multitask learning to weigh the error from translation and rotation automatically. The supervised uncertainty estimation proposed by Kendall and Gal [69] is as follows:

$$\text{loss} = \frac{1}{N}\sum_{i=1}^{N} \frac{1}{2\sigma(x_i)^2} \|y_i - f(x_i)\|^2 + \frac{1}{2}\log\sigma(x_i)^2 \tag{16}$$

where $\sigma$ is the predicted aleatoric uncertainty for the input $x$; $y_i$ and $f(x_i)$ denote the ground truth and prediction with regard to input $x_i$, respectively.

Besides, similar to CertainOdom [14], DeepLIO [5], VINet [7], EMA-VIO [1], and what Zou et al. proposed [70], the frame-to-frame (f2f) constraint represents the relative transformation between each frame in one sliding window and frame-to-global (f2g) is an absolute pose denotes each the transformation from each frame to initial frame.

Regarding the selection of rotation representation, VINet [7] and DeepLIO [5] incorporate the Lie algebra se(3) and Lie group SE(3) into the f2f and f2g constraints, respectively, which yield better performance than using a unique representation of rotation. The definitions are as follows:

$$\mathbf{se(3)} = \left\{ \boldsymbol{\xi}^{\wedge} = \begin{bmatrix} \boldsymbol{\rho}^{\wedge} & \boldsymbol{\Phi} \\ \mathbf{0} & 0 \end{bmatrix} \in \mathbb{R}^{4\times4} \mid \boldsymbol{\xi} = \begin{bmatrix} \boldsymbol{\rho} \\ \Phi \end{bmatrix} \in \mathbb{R}^{6} \right\} \tag{17}$$

$$\mathbf{SE(3)} = \left\{ \mathbf{T} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0} & 1 \end{bmatrix} \in \mathbf{R}^{4\times4} \mid \mathbf{R} \in \mathbf{SO(3)}, \mathbf{t} \in \mathbb{R}^{3} \right\}. \tag{18}$$

However, it is difficult to directly regress the rotation matrix $\mathbf{R} \in \mathbf{SO(3)}$, $\mathbf{R} \in \mathbb{R}^{3\times3}$, since it needs to enforce their special orthogonal properties [71]. There are three options for the conversion of rotation matrix, as we listed in Table I, Euler angle, quaternion and axis angle. Different from our previous work, we conduct the exhaustive experiments to obtain the best combination between different representations of rotation and distance functions in Section IV-D.

The predictions for f2f and f2g are represented as $\boldsymbol{y}_i^f = (\boldsymbol{\phi}_i^f, \boldsymbol{\rho}_i^f)$, and $\boldsymbol{y}_i^g = (\boldsymbol{t}_i^g, \boldsymbol{\varphi}_i^g)$, where $\boldsymbol{\varphi}_i^g$ denotes Euler angles. The corresponding ground truths are represented as $\hat{\boldsymbol{y}}i^f$ and $\hat{\boldsymbol{y}}i^g$. Additionally, we formulate the proposed loss $L(\boldsymbol{\theta}, \sigma_x, \sigma_y, \sigma_z, \sigma_{r_x}, \sigma_{r_y}, \sigma_{r_z})$ as a function of our network, where $\boldsymbol{\theta}$ represents the weights of the network and $\boldsymbol{\sigma}(x, y, z, r_x, r_y, r_z)$ denotes the uncertainties in translation and rotation. Combining the supervised baseline loss and uncertainty estimation regression loss, the loss can be calculated as follows:

$$L_x(\boldsymbol{\theta}, \sigma_x) = \sum_i \frac{1}{2}\exp\left(-s_x^i\right) \times \left( \left[ \left\| \phi_{ix}^f - \hat{\phi}_{ix}^f \right\|^2, \left\| t_{ix}^g - \hat{t}_{ix}^g \right\|^2 \right] \right) + \frac{1}{2} s_x^i \tag{19}$$

$$L_r(\boldsymbol{\theta}, \sigma_{r_x}) = \sum_i \frac{1}{2}\exp\left(-s_{r_x}^i\right) \times \left( \left[ \left\| \rho_{ir_x}^f - \hat{\rho}_{ir_x}^g \right\|^2, \left\| \varphi_{ir_x}^f - \hat{\varphi}_{ir_x}^g \right\|^2 \right] \right) + \frac{1}{2} s_{r_x}^i \tag{20}$$

where $\boldsymbol{\rho}_i^f \in \mathbb{R}^3 = (\rho_{ix}^f, \rho_{iy}^f, \rho_{iz}^f)$ represents the translation in the $x$-, $y$-, and $z$-axis, and $\boldsymbol{\varphi}_i^f \in \mathbb{R}^3 = (\varphi_{ir_x}^f, \varphi_{ir_y}^f, \varphi_{ir_z}^f)$ represents the rotation in roll, pitch, and yaw. Following the approach of [69], we use log variance $s_i = \log\sigma_i^2$ for uncertainty implementation. By using learnable uncertainties, the error between translation and orientation can be automatically weighted, rather than using predefined weighting hyperparameters, which is similar to how Kendall et al. [72] use uncertainty to weigh semantic, instance segmentation, and depth estimation tasks in a multitask learning framework. Finally, the joint loss can be computed as follows:

$$L\left(\boldsymbol{\theta}, \sigma_i^{x,y,z,r_x,r_y,r_z}\right) = \sum_i \left( L_x\left(\boldsymbol{\theta}, \sigma_i^x\right) + \ldots + L_{r_z}\left(\boldsymbol{\theta}, \sigma_i^{r_z}\right) \right). \tag{21}$$

## IV. Experiments

The proposed TransFusionOdom is implemented with Pytorch and 250 epochs are trained with NVIDIA Quadro GV100 (32-GB VRAM). The window size of f2f and f2g constraints is 8 and batch size is set to 16. Multilayer fusion is conducted four times after the repeated conventional layers of ResNet34 (3,4,6,3) and ResNet18 (2,2,2,2). The number of multiheads is set to 4. Most of the empirical hyperparameters are referenced from other related works, such as MLF-VO [45] and Transfuser [60].

For the dataset, KITTI [13] is an autonomous driving dataset commonly used as a benchmark for evaluating odometry tasks. We use sequence 00–08 for training and 09 and 10 for validation. As mentioned in [5], since IMU and LiDAR are not synchronized, the number of IMU measurements between two LiDAR frames varies, with 10–13 IMU measurements between two consecutive LiDAR frames. We normalized the IMU measurements and set $\gamma = 10$, as introduced in Section III-A.2. The overfitting problem in trajectory estimation is evaluated on KITTI dataset. All statistical metrics are calculated by the publicly available KITTI evaluation tool.[2] We also publish a synthetic multimodal dataset for odometry estimation based on the Gazebo simulation environment. This dataset can be used to conveniently validate the generalization ability of the proposed fusion strategy on the different combinations of modalities.

A laptop PC is used to conduct inference experiments in order to deploy on mobile platforms, whose configuration is with Intel i9-12900H (up to 5.0-GHz Turbo) and RTX 3080Ti (16-GB VRAM). Since the computational cost is a crucial factor that affects the application of autonomous driving, and fusing different modalities inevitably increases the model size, we analyze the parameter numbers and inference time for different fusion strategies. In summary, the proposed TransFusionOdom meets the real-time requirements on GPU. Compared to implement the vanilla Transformer for fusing all modalities together, our approach reduces the parameter numbers by 55% and nearly doubles the frames per second (frames/s) on GPU, reaching up to 32 frames/s. However, optimizing our model for real-time capability on CPU-only platforms remains a future work.

### A. Positioning Results on KITTI Dataset

We show the positioning results evaluated qualitatively in Fig. 6 and quantitatively in Table II, where the sensor-fusion-related works is categorized into geometry-based and learning-based considering different types of sensors. The results show that our TransFusionOdom outperforms these related works in most of cases, especially in translation as shown in Fig. 6. Compared with the listed geometry-based approaches in VIO, LO, and LIO, the advantage of our proposed approach not only exists in the fusion stage but also in the front-end feature representation modeling ability, which we mentioned in Section II-A. Besides, the improvement compared with EMA-VIO [1], which also employs a Transformer-based approach for fusion similar to ours, is possibly because the multilayer fusion module aggregates the LiDAR and inertial data at different scales [11], [45] and LiDAR provides more information than the camera

[2]https://github.com/LeoQLi/KITTI_odometry_evaluation_tool

TABLE II
POSITIONING RESULTS ON KITTI DATASET

| Type | Geometry-based VIO | | Learning-based VIO | | Geometry-based LO | | Learning-based LO | | | | Geometry-based LIO | | | | Learning-based LIO | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Seq | VINS-Mono [18] | | EMA-VIO [1] | | LeGO-LOAM [73] | | LO-Net [51] | | CertainOdom [14] | | LIO-SAM [20], [74] | | LIO-CSI [20] | | DeepLIO [5] | | TransFusionOdom | |
| | $t_{rel}$(%) | $r_{rel}$ (°) | $t_{rel}$(%) | $r_{rel}$ (°) | $t_{rel}$(%) | $r_{rel}$ (°) | $t_{rel}$(%) | $r_{rel}$ (°) | $t_{rel}$(%) | $r_{rel}$ (°) | $t_{rel}$(%) | $r_{rel}$ (°) | $t_{rel}$(%) | $r_{rel}$ (°) | $t_{rel}$(%) | $r_{rel}$ (°) | $t_{rel}$(%) | $r_{rel}$ (°) |
| 01 | / | / | / | / | 13.4 | 1.02 | 1.36($3^{rd}$) | 0.47($2^{nd}$) | 0.52($2^{nd}$) | 1.26 | 4.38 | 0.96($3^{rd}$) | 6.02 | 1.14 | 5.28 | 1.51 | **0.43(↑17.3%)** | **0.46(↑2.1%)** |
| 05 | / | / | / | / | 1.28 | 0.74 | 1.04 | 0.69($2^{nd}$) | 0.53($2^{nd}$) | 1.29 | 1.19($3^{rd}$) | **0.67** | 1.21 | 0.72 | 1.28 | 0.93 | **0.48(↑9.4%)** | 0.73(↓8.9%) |
| 08 | / | / | / | / | 1.99($3^{rd}$) | 0.94($3^{rd}$) | 2.12 | 0.77($2^{nd}$) | 1.23($2^{nd}$) | 1.52 | 3.88 | 1.67 | 3.28 | 1.72 | 2.33 | 0.94($3^{rd}$) | **0.99(↑19.5%)** | **0.75(↑2.6%)** |
| 09 | 41.4 | 2.41 | 8.86 | 1.54 | 1.97 | 0.98 | 1.37($3^{rd}$) | **0.58** | 0.53($2^{nd}$) | 1.25 | 1.28 | 0.83($3^{rd}$) | 3.77 | 1.81 | 4.4 | 1.21 | **0.49(↑15.5%)** | 0.63(↓8.6%) |
| 10 | 20.35 | 2.73 | 7.46 | 2.26 | 2.21 | 0.92($3^{rd}$) | 1.80($3^{rd}$) | 0.93 | 0.82($2^{nd}$) | 1.06 | 1.31 | **0.76** | 2.57 | 1.60 | 4.0 | 1.51 | **0.72(↑12.2%)** | 0.78(↓2.6%) |
| Avg. (09/10) | 30.91 | 2.57 | 8.07 | 1.90 | 2.09 | 0.95 | 1.59($3^{rd}$) | 0.76($2^{nd}$) | 0.68($2^{nd}$) | 1.16 | 1.30 | 0.79($3^{rd}$) | 2.70 | 1.33 | 4.2 | 1.36 | **0.61(↑10.3%)** | **0.71(↑6.6%)** |

• $t_{rel}$: average sequence translational RMSE (%) on the length of 100m, 200m, ..., 800m.

• $r_{rel}$: average sequence rotational RMSE (°/100m) on the length of 100m, 200m, ..., 800m.

• The positioning results of existed methods are collected from the cited literature, LIO-SAM is from the avg. of [20], [74] and our own implementation results.

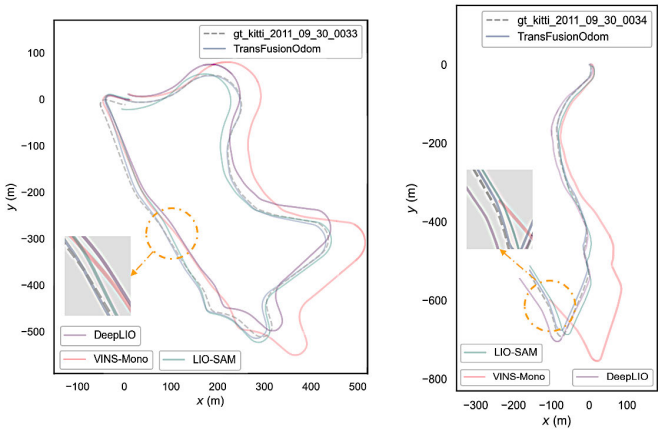



Fig. 6. Trajectory evaluation in KITTI 09 and 10 sequence with related works.

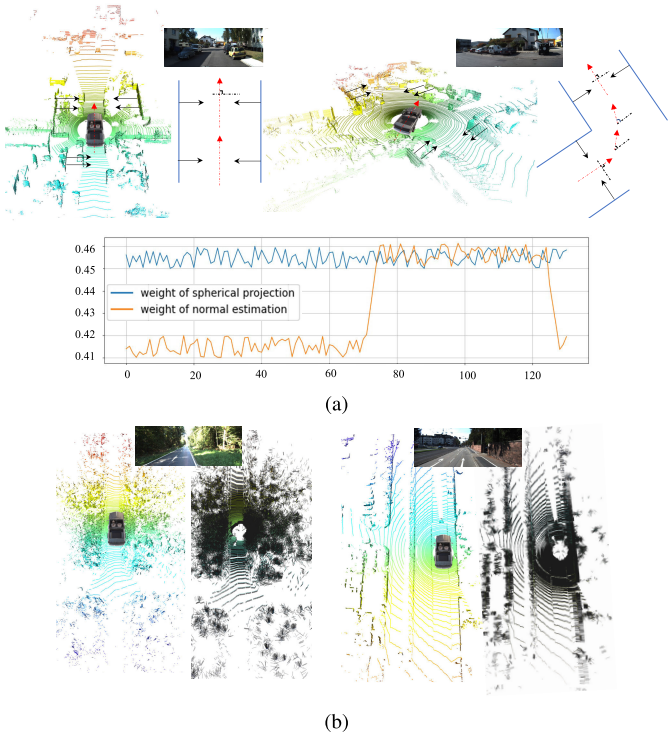



Fig. 7. Visualization of attention between homogeneous modalities. (a) SMAF between vertex spherical projection and normal map during straight and turning, the car starts to turn around 70th and finish in the end. (b) Normal estimation under different (left: forest, right: wall) road situations.

thanks to the field-of-view. It is worth noting that the listed learning-based approaches tend to achieve better performance than geometry-based approaches in VO, LO, and LIO. Furthermore, the learning-based approach of LO-Net [51] even outperforms the geometry-based LIO approaches such as LIO-SAM [19] and LIO-SCI [20]. This demonstrates that the advantage of the learning-based approach with a single sensor is greater than the traditional sensor-fusion approaches. However, we observed that our advantage in rotation is inferior to translation compared to the related works. We believe that the possible reason is highly related to the representation of rotation in learning-based approaches [75], which we discuss in Section IV-D.

## B. Visualization of Interactions Between Multiple Modalities

*1) Attention Inside of Homogeneous Fusion:* According to the introduction in Section III-B, we plot the soft mask weights $M^{l_1}_{\text{vertex}}$ and $M^{l_1}_{\text{normal}}$ in Fig. 7(a). We observe that during turning, the weights of the normal map increased a lot compared to straight driving in $l_{1\sim4}$. The possible reason is that the angle between the driving direction and normal vector changes more during turning than during straight driving. Similar conclusions were mentioned in UnDeepLIO [49], where they only used vertex information to estimate the translation because the change in translation does not affect the normal information. However, we also find out that the normal map is sensitive to road situations, as shown in Fig. 7(b). For example, when the surroundings are in a forest, the normal information is in a highly random condition compared to a wall along the driving direction. Generally, the network gives more attention to changing information instead of static features.

*2) Attention Inside of Heterogeneous Fusion:* Based on the approach we introduced in Section III-C, we first visualize the attention matrix of head 4 from Transformer 1 to 4 as shown in Fig. 8. By observing the position of high-value attended tokens, we can see that in T1, the self-attention domains are more prominent compared to the later Transformer fusion stages. From T2 to T4, the network gradually learns cross-attention. In the final layer T4, the highest values of attended

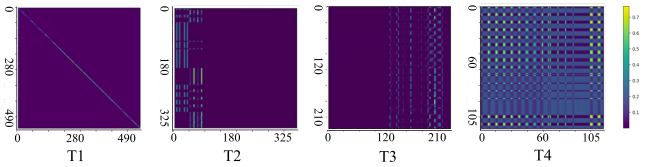



Fig. 8. Attention matrix of T1–T4 in head 4.

TABLE III
CROSS-ATTENTION QUANTITATIVE EVALUATION

| Head | T1 | | T2 | | T3 | | T4 | |
|---|---|---|---|---|---|---|---|---|
| | $L_t$ | $I_t$ | $L_t$ | $I_t$ | $L_t$ | $I_t$ | $L_t$ | $I_t$ |
| 1 | 0 | 0 | 5.7 | 22.3 | 35.6 | 15.6 | 60.7 | 63.5 |
| 2 | 0 | 0 | 7.3 | 25.6 | 42.3 | 23.6 | 65.4 | 67.4 |
| 3 | 0 | 0 | 10.1 | 33.2 | 44.5 | 22.5 | 70.3 | 67.9 |
| 4 | 1.2 | 1.5 | 12.7 | 30.8 | 45.7 | 21.3 | 76.7 | 78.3 |

tokens are distributed uniformly, indicating that there are interactions based on attention weights between the two modalities inside the fusion.

In addition to qualitative results, we conduct the statistical analysis of cross-attention on KITTI dataset 09 sequence as shown in Table III. We present the percentage of tokens ($L_t$: LiDAR tokens, $I_t$: inertial tokens) that have at least three of the top five attended tokens belonging to the other modality in each head of T1–T4. Consistent with the visualization of the attention matrix, T1 shows rare cross-attention. Besides, in each fusion stage, almost all the later heads exhibit more cross-attention weights than their former heads. Also, the value of T4 indicates that our proposed fusion strategy is capable of aggregating information from two modalities.

Additionally, as illustrated in Fig. 5(b), we reshape the array of attended token values and overlap them onto the source modality image, as shown in Fig. 9. We present visualizations of head 4 in T1 for self-attention and T4 which has the most prominent cross-attention. In Fig. 9(a), we select the first patch as the query token, and the highest value of the attended token is located in the same position as the query token or nearby positions. Moreover, the sum of attention values in the query's modality is around 0.75, indicating less attention in the other modalities. Similar conditions exist in the LiDAR self-attention of T1.

In Fig. 9(b), the query token is selected on the left side representing linear acceleration. The high values of the attended tokens are mainly located at the bottom of the road position. In Fig. 9(c), the query token is located in a low-value position of the IMU signal image, and the values of the attended tokens are distributed almost equally. In comparison, in Fig. 9(d), if the query token is selected in a high-value angular velocity position, we observe that the high weights are located on the main road and corners of the building. One possible reason for this is that the geometry shape of the corner contributes more when the car is turning with high angular velocity. To verify this hypothesis, we visualize the LiDAR-to-Inertial cross-attention in Fig. 9(e), by selecting the corner patch as the query token, the high weights of attended tokens are mainly located in the right parts with high angular velocity value positions.

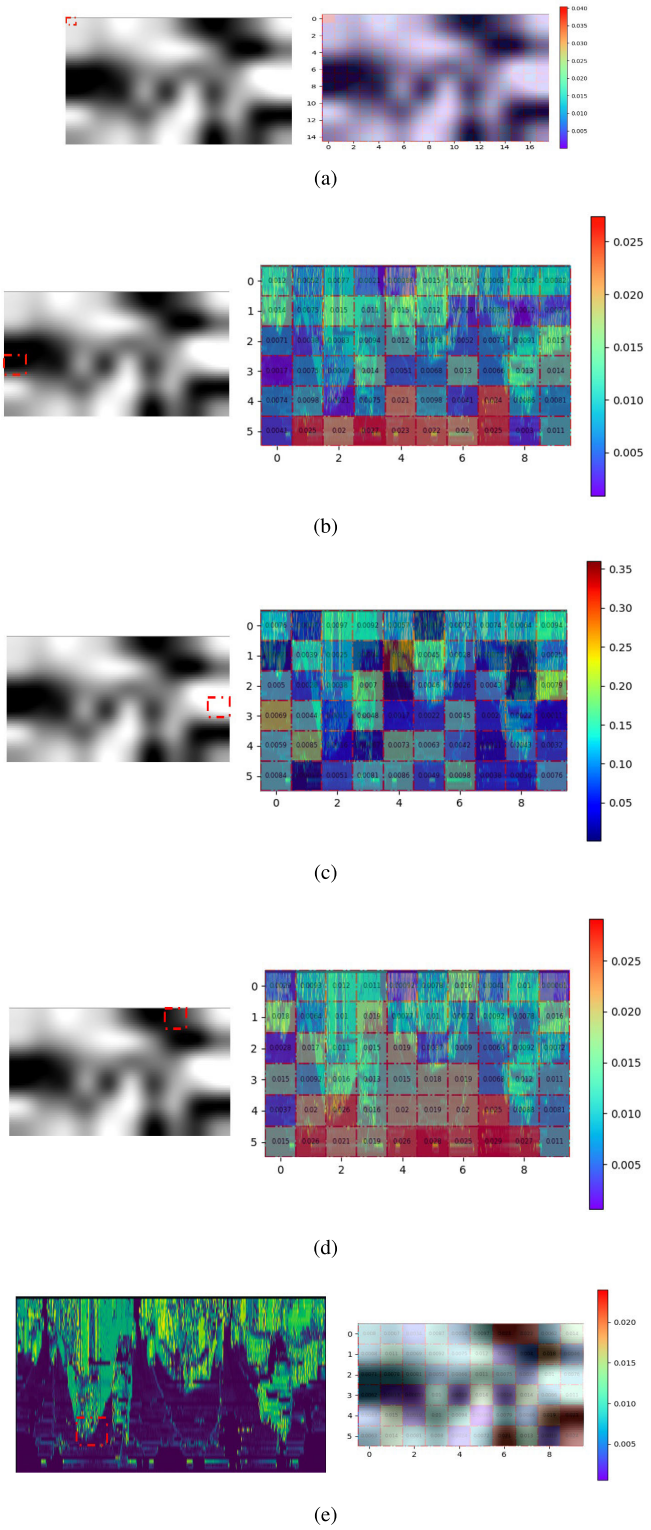



Fig. 9. Visualization of self-attention in T1 of head4 and cross-attention in T4 of head 4, red rectangle is query token, the value of the attended tokens is overlapped on source images. (a) Inertial-to-inertial self-attention, query token: the first patch. (b) Inertial-to-LiDAR cross attention, query token: high linear acceleration value. (c) Inertial-to-LiDAR cross attention, query token: low angular velocity value. (d) Inertial-to-LiDAR cross attention, query token: high angular velocity value. (e) LiDAR-to-Inertial cross attention, query token: corner position.

Through the above visualization, it is shown that our proposed fusion strategy could promote or restrain some specific interactions between two modalities via assigning

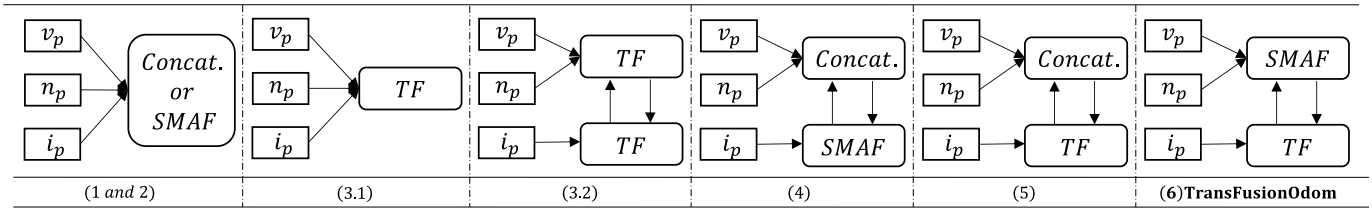



Fig. 10. Different simplified overviews of network architecture in ablation study, vertex $v_p$ and normal $n_p$ are homogeneous modalities, which are heterogeneous with $i_p$ from IMU, TF is short for Transformer.

TABLE IV
ABLATION STUDY RESULTS

| Module / Case | Multi-layer | Multi-task | Multi-scale | Concat. | SMAF | Transformer | Average of KITTI Training | | Average of KITTI Testing | | Model Parameters Number | Inference Time | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | $t_{rel}(\%)$ | $r_{rel}$ (°) | $t_{rel}(\%)$ | $r_{rel}$ (°) | | CPU(ms) | GPU(ms) |
| (1) | ✓ | ✓ | ✓ | ✓ | | | 10.82 | 8.32 | 11.25 | 8.95 | ≈ 36M | 40(25fps) | 12.5(80fps) |
| (2) | ✓ | ✓ | ✓ | | ✓ | | 8.24 | 6.12 | 8.69 | 6.47 | ≈ 40M | 48(21fps) | 15(67fps) |
| (3) | ✓ | ✓ | ✓ | | | ✓ | **0.49** | **0.62** | 3.52 | 4.66 | ≈ **185M** | 270(4fps) | 72(14fps) |
| (4) | ✓ | ✓ | ✓ | ✓ | ✓ | | 4.67 | 4.89 | 4.72 | 4.93 | ≈ 40M | / | / |
| (5) | ✓ | ✓ | ✓ | ✓ | | ✓ | 3.53 | 3.72 | 3.75 | 3.98 | ≈ 80M | / | / |
| **(6) TransFusionOdom** | ✓ | ✓ | ✓ | | ✓ | ✓ | 0.52 | 0.67 | 0.61 | 0.71 | ≈ 84M | 125(8fps) | 31(32fps) |
| (7) | ✓ | ✓ | | | ✓ | ✓ | 2.97 | 2.63 | 3.12 | 2.88 | ≈ 60M | / | / |

• $t_{rel}$: average sequence translational RMSE (%) on the length of 100m, 200m, ..., 800m.

• $r_{rel}$: average sequence rotational RMSE (°/100m) on the length of 100m, 200m, ..., 800m.

adaptive weights, which makes the whole incorporation more effective.

### C. Ablation Study and Overfitting Problem

Since the proposed framework includes many modules, an exhaustive ablation study is necessary. We design the ablation study by separating the modules and evaluating their impact on performance, as shown in Table IV and Fig. 10. The multilayer strategy has been validated in MLF-VO [45] and CE [11], and the multitask regressor module is compared with the baseline 6-D pose regressor in our previous work, CertainOdom [14]. Therefore, we keep these two modules as "open" status in all cases.

The cases from (1) to (5) are designed to test three different fusion approaches: concatenate, SMAF, and Transformer. When selecting two fusion methods among these three as a combination such as (4) and (5), they actually include two opposite ways (e.g., concatenate between $\boldsymbol{v}_p$ and $\boldsymbol{n}_p$ and then SMAF with $\boldsymbol{i}_p$ or inverse SMAF and concatenate). We only show the better results in Table IV. Case (7) is to verify the performance of the multiscale module compared to the proposed TransFusionOdom (6).

The reason we not only list the testing results but also the training dataset results is that we have observed an overfitting problem in case (3). The issue of overfitting in learning-based odometry estimation tasks has rarely been mentioned. As we know, the Transformer-based models are more data-hungry than CNN-based approaches [76]. Additionally, as mentioned before, the model size or complexity of Transformer-based models highly depends on the number and resolution of tokens [57], and there is also a multilayer fusion strategy inside the proposed framework, which increases the number of parameters. The bigger the model, the easier it is to overfit. Considering the above, it is necessary to check whether the proposed solutions and other Transformer-based fusion approaches are a good fit or overfitting model.

In case (3), regardless of whether we use a Transformer to first fuse $\boldsymbol{v}_p$ and $\boldsymbol{n}_p$ as one general LiDAR modality and then deploy another Transformer to fuse it with $\boldsymbol{i}_p$ as (3.2) of Fig. 10, or directly implement one Transformer with $\boldsymbol{v}_p$, $\boldsymbol{n}_p$ and $\boldsymbol{i}_p$ three modalities as (3.1) of Fig. 10, the overfitting problem occurs. As shown in Fig. 11, overfitting can be detected by the loss curves, also the overfitting model performs better than the good-fitted model in training, but in testing, the result is worse than good-fitted model which is not what we expect. To overcome the overfitting problem, we implement a shared weights configuration between the backbones for vertex and normal features in Fig. 2. This way, the network can learn not only the common features but also its size can be reduced to avoid overfitting. Additionally, we could deploy data augmentation [77] to increase the size of the training dataset, but it is outside the scope of this study.

From the statistical results in Table IV, it is clear how the performance of each fusion strategy compares. If we can overcome the overfitting problem with Transformer-based fusion, its performance is much better than that of SMAF and concatenation approaches. In case (7), we set a fixed resolution of input tokens from $l_1$ to $l_4$, which is equal to the $l_4$ resolution in TransfusionOdom (6). The reason for this is that the coarse-to-fine resolution is gradually reduced in multiscale fusion. The higher the fixed resolution in (7), the more likely it is to lead to overfitting. Therefore, multiscale strategy can also help avoid overfitting problems in Transformer-based fusion.

### D. Representations of Rotation and Distance Functions

To the best of our knowledge, there is no definitive conclusion in the odometry estimation task about which rotation representation is the best, as shown in Table I. Unlike other

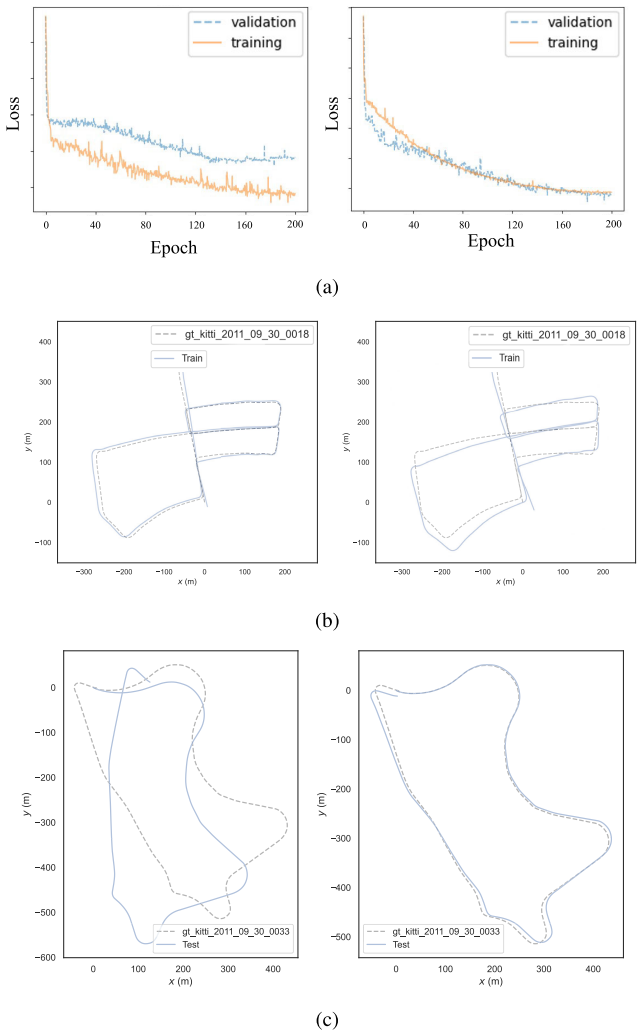



Fig. 11. Left: overfitting model. Right: good-fitted model. (a) Loss curves in training and validation. (b) Trajectory results of training data. (c) Trajectory results of tested data.

6-D pose-related tasks, in odometry estimation, the network predicts the ego-motion between two consecutive input frames, which means the transformation is relatively small. The same situation applies to the selection of the distance function. Similar to DeepLO [15], which evaluates the rotation error of each frame in testing, we consider these two variables and conduct comprehensive experiments with our proposed loss function to select the combination from Table V with the best rotation estimation performance.

The approach for selecting the best option is to calculate the rotation error under the same training and validation conditions. The predicted rotation is calculated between each consecutive frames after the model is finished training, and compared with ground truth value for rotation error. Instead of directly plotting the rotation error as done in DeepLO [15], we use box plot for more detailed observations. Through the box plot in Fig. 12, we can observe the mean rotation error, stability, and outliers of different combinations. The best one with our proposed loss is Euler angle with f2f constraint and $\mathbf{se(3)}$ with f2g, combined with L2 distance function. Based on all results, we find out that Euler angle is more

TABLE V
ALL COMBINATIONS OF REPRESENTATION OF ROTATION WITH L1/2 DISTANCE

| | Rotation combinations with L1 and L2 distance |
|---|---|
| 1 | Both Euler angle |
| 2 | Both Axis-angle |
| 3 | Both Quaternion |
| 4 | Both se3 |
| 5/6 | Euler (f2f) + Axis-angle (f2g) / Opposite |
| 7/8 | Euler (f2f) + quaternion (f2g) / Opposite |
| 9/10 | Euler (f2f) + se3 (f2g) / Opposite |
| 11/12 | Axis-angle (f2f) + quaternion (f2g) / Opposite |
| 13/14 | Axis-angle (f2f) + se3 (f2g) / Opposite |
| 15/16 | Quaternion (f2f) + se3 (f2g) / Opposite |

TABLE VI
GENERALIZATION EXPERIMENTS ON SYNTHETIC MULTIMODAL DATASET

| Modality | LiDAR | RGB | Depth | IMU | Position RMSE (m) | Attitude RMSE (°) |
|---|---|---|---|---|---|---|
| LiDAR-based | ✓ | | | | 10.32 | 8.19 |
| L-(2) | ✓ | ✓ | | | 7.32 | 6.73 |
| L-(3) | ✓ | | ✓ | | 8.94 | 7.59 |
| L-(4) | ✓ | | | ✓ | 4.67 | **2.83** |
| RGB-based | | ✓ | | | 13.47 | 10.76 |
| R-(2) | | ✓ | ✓ | | 9.32 | 8.17 |
| R-(3) | | ✓ | | ✓ | 6.33 | 3.72 |
| R-(4) | | ✓ | ✓ | ✓ | **4.06** | 2.99 |

suitable for relative transformation and $\mathbf{se(3)}$ is good at global transformation, L2 is better than L1 in most situations. However, it is difficult to draw a general conclusion on different designed loss functions and data distributions. Compared with the conclusion we obtain through extensive experiments, the approach used to evaluate all combinations is more useful in case we need to validate with other loss functions for different tasks.

As we discovered in Section IV-A, the improvement in rotation is lower compared to translation. One possible reason is that for 3-D rotations, all representations exhibit discontinuities in the real Euclidean spaces of four or fewer dimensions [75]. Recently, ChiNet [78] and Zhou et al. [79] utilize a 6-D attitude representation proposed by Zhou et al. [75] to facilitate the learning of rotations in their approaches. This representation enables a continuous mapping $\mathbb{R}^6 \in \mathbf{SO(3)}$ for object pose estimation tasks and could serve as a reference for future work in learning-based odometry estimation field.

### E. Gazebo-Based Synthetic Multimodal Dataset

We publish a synthetic multimodal dataset for odometry estimation that was collected in the Gazebo simulator, as shown in Fig. 13. This dataset provides multisensor data, including Velodyne VLP-16 3-D LiDAR, Realsense D435 RGB-D camera, IMU, and can also easily integrate other sensors. The ground truth can directly be obtained from the simulation state. The dataset includes five scenarios/maps, each one is collects multimodal data five times (three for training, two for validation) using random trajectory generation.

TABLE VII
NETWORK ARCHITECTURE OF IMPLEMENTATION

| | (I) SMAF for homogeneous fusion | (II) Transformer for heterogeneous fusion |
|---|---|---|
| (1) Conv2d | inputs are vertex and normal map<br>$kernel\_size = 7^2, stride = 2^2, padding = 3$ | inputs are output of stage (I) and inertial digital map<br>$kernel\_size = 7^2, stride = 2^2, padding = 3$ |
| (2) BatchNorm2d & ReLU | ✓ | ✓ |
| (3) Maxpool2d | $kernel\_size = 3^2, stride = 2^2, padding = 1$ | $kernel\_size = 3^2, stride = 2^2, padding = 1$ |
| (4) ResNet | $ResNet34.layer1(64, 3, 2e-2); output = [\boldsymbol{v}_{l1}; \boldsymbol{n}_{l1}]$<br>$ResNet34.layer2(128, 4, 2e-2); output = [\boldsymbol{v}_{l2}; \boldsymbol{n}_{l2}]$<br>$ResNet34.layer3(256, 6, 2e-2); output = [\boldsymbol{v}_{l3}; \boldsymbol{n}_{l3}]$<br>$ResNet34.layer4(512, 3, 2e-2); output = [\boldsymbol{v}_{l4}; \boldsymbol{n}_{l4}]$ | $ResNet18.layer1(64, 2, 2e-2); output = \boldsymbol{i}_{l1}$<br>$ResNet18.layer2(128, 2, 2e-2); output = \boldsymbol{i}_{l2}$<br>$ResNet18.layer3(256, 2, 2e-2); output = \boldsymbol{i}_{l3}$<br>$ResNet18.layer4(512, 2, 2e-2); output = \boldsymbol{i}_{l2}$ |
| (5) AdaptiveAvgPool2d | / | $Layer1(15, 18), output = [\boldsymbol{i}_{pl1}, \boldsymbol{s}_{pl1}]$<br>$Layer2(12, 15), output = [\boldsymbol{i}_{pl2}, \boldsymbol{s}_{pl2}]$<br>$Layer3(10, 12), output = [\boldsymbol{i}_{pl3}, \boldsymbol{s}_{pl3}]$<br>$Layer4(6, 10), output = [\boldsymbol{i}_{pl4}, \boldsymbol{s}_{pl4}]$ |
| (6) Self and cross attention in Transformer | / | $\forall\, l \in \{l_1, l_2, l_3, l_4\}\ input = [\boldsymbol{i}_{pl}, \boldsymbol{s}_{pl}]$<br>$output = [\boldsymbol{i}_{T_l}, \boldsymbol{s}_{T_l}]$ |
| (7) Interpolate | / | $\forall\, l \in \{l_1, l_2, l_3, l_4\}$<br>$\boldsymbol{s}_{T_l} \rightarrow interpolate(\boldsymbol{s}_l.size(), mode =' bilinear')$<br>$\boldsymbol{i}_{T_l} \rightarrow interpolate(\boldsymbol{i}_l.size(), mode =' bilinear')$ |
| (8) Fusion | $\boldsymbol{s}_l = FC_{\boldsymbol{v}_l}[\boldsymbol{v}_l;\boldsymbol{n}_l]{*}\boldsymbol{v}_l + FC_{\boldsymbol{n}_l}[\boldsymbol{v}_l;\boldsymbol{n}_l]{*}\boldsymbol{n}_l$<br>$\forall\, l \in \{l_1, l_2, l_3, l_4\}$ as input for stage (II) | $\forall\, l \in \{l_1, l_2, l_3, l_4\}$<br>$\boldsymbol{S}_l = \boldsymbol{s}_l + \boldsymbol{s}_{T_l}$<br>$\boldsymbol{I}_l = \boldsymbol{i}_l + \boldsymbol{i}_{T_l}$<br>$output = [\boldsymbol{S}_{l_4}; \boldsymbol{I}_{l_4}]$ for step (9) |
| (9) 6D pose and uncertainty regressor | 2 Linear(1024,3) for translation and rotation + 2 Linear(1024,3) for uncertainty in position and orientation | |

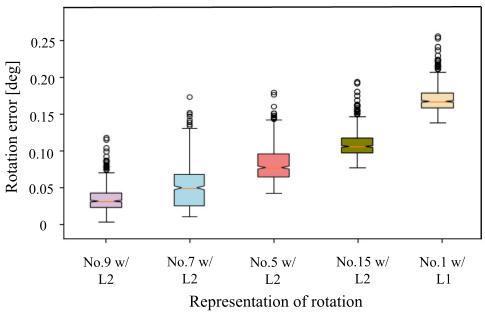



Fig. 12. Box plot about the top five combinations of representation of rotation and distance function using rotation error.

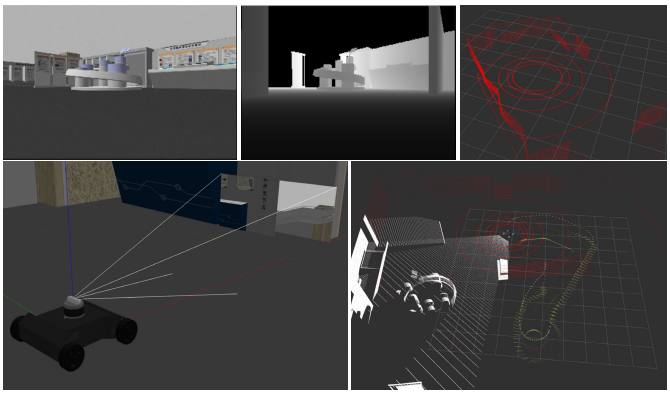


Fig. 13. Publicly available synthetic multimodal dataset for odometry estimation.

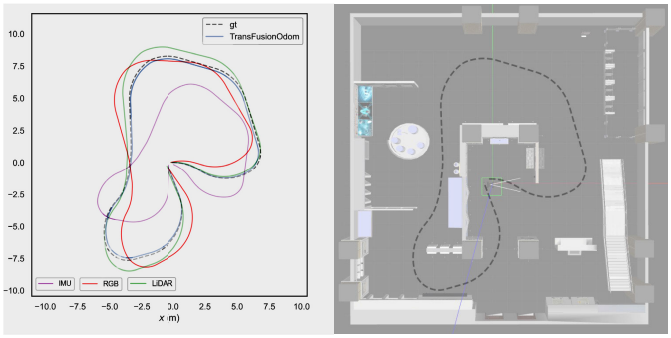



Fig. 14. Comparison between unimodal and TransFusionOdom on synthetic dataset.

In the future, the Gazebo world maps can be replaced and even simulate moving objects. Using this dataset, the algorithms developed in this community can be conveniently tested and can also be used to apply the transfer learning [15] for training initial weights.

We use this dataset to confirm the generalization ability of the proposed fusion strategy to other combinations of modalities. We separate the combinations into LiDAR-based and RGB-based, which are commonly configured for robotics perception and are listed in Table VI. We did not test the combinations with four or three modalities that include LiDAR because, as in TransFusionOdom, LiDAR includes vertex and normal submodalities. Adding more modalities can lead to overfitting problems. All modalities were converted to image type as input.

In Fig. 14, we present a trajectory comparison between unimodal and proposed TransFusionOdom. It turns out that it is difficult to obtain acceptable results by only using inertial data with Transformer architecture compared to using only vision-based modalities such as RGB and LiDAR. Moreover, unlike the common problem in geometry-based solutions, which is the scale ambiguity [9], [10], thanks to the f2f and f2g constraints, the scale problem is not so obvious in the learning-based approach. This similar conclusion was also observed in DeepLIO [5].

In Table VI, there are different combinations of exteroceptive sensors (L-2, L-3, and R-2) and proprioceptive with exteroceptive sensors (L-4, R-3, and R-4). The latter one has a better performance than the former combination, which is the same as our common sense. Besides, we observe that IMU contributes more to attitude than position. In L-4 and R-3, the performance has on average increased by 53% in position and 65% in attitude, respectively. In R-4, RGB and depth are homogeneous modalities fused by SMAF and then integrated with IMU data using Transformer, which obtains the best

performance in position. Generally, these experiments are not designed to test which combination is the best, but to verify that our fusion strategy can generalize to different modalities to achieve better performance than unimodal instead of only LiDAR with IMU in TransFusionOdom. Particularly in RGB-based experiments, we observe that the performance improves with the inclusion of more modalities using our proposed fusion strategy.

## V. Conclusion and Future Work

In this study, we present TransFusionOdom, a Transformer-based supervised end-to-end LIO framework, the network architecture as shown in Table VII. We mainly discussed the performance of different fusion strategies that involve the homogeneous and heterogeneous modalities. We conducted an exhaustive ablation study to check the performance of different fusion strategies. Additionally, we illustrated and discussed the overfitting problem in odometry estimation caused by a Transformer-based fusion network. We demonstrated a general approach to visualize self- and cross attention inside TransFusionOdom, which enables us to interpret how different modalities interact with each other via attention mechanisms. We also collected and made publicly available a synthetic dataset to validate the generalization ability of the proposed fusion strategy on different modalities. The odometry estimation result is evaluated qualitatively and quantitatively on the KITTI dataset which outperforms previous approaches.

Back to the previous question: *How should we perform fusion among different modalities in a supervised sensor fusion odometry estimation task?*. The task we tackle is a mix of homogeneous and heterogeneous modalities fusion. Homogeneous modalities have the characteristic of naturally aligned features at each corresponding position, making the lightweight MLP-based SMAF potentially capable of learning attentions between aligned features deterministically [21]. However, the structural discrepancies between heterogeneous modalities make fusion more challenging. Thanks to the token-to-token attention mechanism inside the Transformer architecture, it becomes possible to extract informative attentions spatially from nonaligned features. Although larger datasets and models have stronger learning abilities, using SMAF and Transformer to fuse a mix of homogeneous and heterogeneous modalities in this study is a trade-off between performance and cost. Finally, combined with a multiscale and layer fusion module, a generic and flexible fusion strategy is developed and validated in the study.

Moreover, vision Transformer suffers from high redundancy by only focusing on local features or self-attention domains in shallow layers [80], as we discussed in the visualization of the attention matrix in the early fusion stage. If we can effectively achieve global context modeling at the early stage of the Transformer-based architecture, we can make the neural network model lightweight, which is beneficial for easy training and real-world applications.

## References


[1] Z. Tu, C. Chen, X. Pan, R. Liu, J. Cui, and J. Mao, "EMA-VIO: Deep visual–inertial odometry with external memory attention," *IEEE Sensors J.*, vol. 22, no. 21, pp. 20877–20885, Nov. 2022.

[2] N. Kaygusuz, O. Mendez, and R. Bowden, "AFT-VO: Asynchronous fusion transformers for multi-view visual odometry estimation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2022, pp. 2402–2408.

[3] C. Chen et al., "Selective sensor fusion for neural visual-inertial odometry," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 10534–10543.

[4] H. Son, B. Lee, and S. Sung, "Synthetic deep neural network design for LiDAR-inertial odometry based on CNN and LSTM," *Int. J. Control, Autom. Syst.*, vol. 19, no. 8, pp. 2859–2868, Aug. 2021.

[5] A. Javanmard-Gh, D. Iwaszczuk, and S. Roth, "DeepLIO: Deep LiDAR inertial sensor fusion for odometry estimation," *ISPRS Ann. Photogramm., Remote Sens. Spatial Inf. Sci.*, vol. V-1-2021, pp. 47–54, Jun. 2021.

[6] L. Liu, G. Li, and T. H. Li, "ATVIO: Attention guided visual-inertial odometry," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Jun. 2021, pp. 4125–4129.

[7] R. Clark, S. Wang, H. Wen, A. Markham, and N. Trigoni, "VINet: Visual-inertial odometry as a sequence-to-sequence learning problem," in *Proc. AAAI Conf. Artif. Intell.*, 2017, vol. 31, no. 1, pp. 3995–4001.

[8] C. Li, S. Wang, Y. Zhuang, and F. Yan, "Deep sensor fusion between 2D laser scanner and IMU for mobile robot localization," *IEEE Sensors J.*, vol. 21, no. 6, pp. 8501–8509, Mar. 2021.

[9] M. F. Aslan, A. Durdu, and K. Sabanci, "Visual-inertial image-odometry network (VIIONet): A Gaussian process regression-based deep architecture proposal for UAV pose estimation," *Measurement*, vol. 194, May 2022, Art. no. 111030.

[10] M. F. Aslan, A. Durdu, A. Yusefi, and A. Yilmaz, "HVIOnet: A deep learning based hybrid visual–inertial odometry approach for unmanned aerial system position estimation," *Neural Netw.*, vol. 155, pp. 461–474, Nov. 2022.

[11] Y. Wang, W. Huang, F. Sun, T. Xu, Y. Rong, and J. Huang, "Deep multimodal fusion by channel exchanging," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 4835–4845.

[12] R. He, Y. Li, X. Wu, L. Song, Z. Chai, and X. Wei, "Coupled adversarial learning for semi-supervised heterogeneous face recognition," *Pattern Recognit.*, vol. 110, Feb. 2021, Art. no. 107618.

[13] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? The KITTI vision benchmark suite," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 3354–3361.

[14] L. Sun, G. Ding, Y. Yoshiyasu, and F. Kanehiro, "CertainOdom: Uncertainty weighted multi-task learning model for LiDAR odometry estimation," in *Proc. IEEE Int. Conf. Robot. Biomimetics (ROBIO)*, Dec. 2022, pp. 121–128.

[15] Y. Cho, G. Kim, and A. Kim, "DeepLO: Geometry-aware deep LiDAR odometry," 2019, *arXiv:1902.10562*.

[16] T. Schöps, J. Engel, and D. Cremers, "Semi-dense visual odometry for AR on a smartphone," in *Proc. IEEE Int. Symp. Mixed Augmented Reality (ISMAR)*, Sep. 2014, pp. 145–150.

[17] M. Cao, L. Zheng, W. Jia, and X. Liu, "Fast monocular visual odometry for augmented reality on smartphones," *IEEE Consum. Electron. Mag.*, early access, May 8, 2020, doi: 10.1109/MCE.2020.2993086.

[18] T. Qin, P. Li, and S. Shen, "VINS-Mono: A robust and versatile monocular visual-inertial state estimator," *IEEE Trans. Robot.*, vol. 34, no. 4, pp. 1004–1020, Aug. 2018.

[19] T. Shan, B. Englot, D. Meyers, W. Wang, C. Ratti, and D. Rus, "LIO-SAM: Tightly-coupled LiDAR inertial odometry via smoothing and mapping," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2020, pp. 5135–5142.

[20] G. Wang, S. Gao, H. Ding, H. Zhang, and H. Cai, "LIO-CSI: LiDAR inertial odometry with loop closure combined with semantic information," *PLoS ONE*, vol. 16, no. 12, Dec. 2021, Art. no. e0261053.

[21] C. Chen, S. Rosa, C. X. Lu, B. Wang, N. Trigoni, and A. Markham, "Learning selective sensor fusion for state estimation," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Jun. 3, 2022, doi: 10.1109/TNNLS.2022.3176677.

[22] P. Xu, X. Zhu, and D. A. Clifton, "Multimodal learning with transformers: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, early access, May 11, 2023, doi: 10.1109/TPAMI.2023.3275156.

[23] A. Vaswani et al., "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–11.

[24] B. Li et al., "DropKey for vision transformer," in *Proc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 22700–22709.

[25] W. Wang et al., "PointLoc: Deep pose regressor for LiDAR point cloud localization," *IEEE Sensors J.*, vol. 22, no. 1, pp. 959–968, Jan. 2022.

[26] R. Gao, X. Xiao, W. Xing, C. Li, and L. Liu, "Unsupervised learning of monocular depth and ego-motion in outdoor/indoor environments," *IEEE Internet Things J.*, vol. 9, no. 17, pp. 16247–16258, Sep. 2022.

[27] R. Mur-Artal and J. D. Tardós, "ORB-SLAM2: An open-source SLAM system for monocular, stereo, and RGB-D cameras," *IEEE Trans. Robot.*, vol. 33, no. 5, pp. 1255–1262, Oct. 2017.

[28] J. Zhang and S. Singh, "LOAM: LiDAR odometry and mapping in real-time," in *Proc. Robot., Sci. Syst.*, Berkeley, CA, USA, 2014, vol. 2, no. 9, pp. 1–9.

[29] C. Chen, B. Wang, C. Xiaoxuan Lu, N. Trigoni, and A. Markham, "A survey on deep learning for localization and mapping: Towards the age of spatial machine intelligence," 2020, *arXiv:2006.12567*.

[30] P. J. Besl and N. D. McKay, "A method for registration of 3-D shapes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 14, no. 2, pp. 239–256, Feb. 1992.

[31] A. V. Segal, D. Haehnel, and S. Thrun, "Generalized-ICP," in *Proc. Robot. Sci., Syst.*, Seattle, WA, USA, Jun. 2009, vol. 2, no. 4, p. 435.

[32] T. Shan and B. Englot, "LeGO-LOAM: Lightweight and ground-optimized LiDAR odometry and mapping on variable terrain," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2018, pp. 4758–4765.

[33] S. Zhao, Z. Fang, H. Li, and S. Scherer, "A robust laser-inertial odometry and mapping method for large-scale highway environments," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Nov. 2019, pp. 1285–1292.

[34] C. Wang, Z. Cao, J. Li, J. Yu, and S. Wang, "Hierarchical distribution-based tightly-coupled LiDAR inertial odometry," *IEEE Trans. Intell. Vehicles*, early access, May 5, 2023, doi: 10.1109/TIV.2023.3273288.

[35] Z. Wang, L. Zhang, Y. Shen, and Y. Zhou, "D-LIOM: Tightly-coupled direct LiDAR-inertial odometry and mapping," *IEEE Trans. Multimedia*, early access, Apr. 19, 2022, doi: 10.1109/TMM.2022.3168423.

[36] W. Xu and F. Zhang, "FAST-LIO: A fast, robust LiDAR-inertial odometry package by tightly-coupled iterated Kalman filter," *IEEE Robot. Autom. Lett.*, vol. 6, no. 2, pp. 3317–3324, Apr. 2021.

[37] W. Xu, Y. Cai, D. He, J. Lin, and F. Zhang, "FAST-LIO2: Fast direct LiDAR-inertial odometry," *IEEE Trans. Robot.*, vol. 38, no. 4, pp. 2053–2073, Aug. 2022.

[38] H. Ye, Y. Chen, and M. Liu, "Tightly coupled 3D LiDAR inertial odometry and mapping," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2019, pp. 3144–3150.

[39] C. Zheng, Q. Zhu, W. Xu, X. Liu, Q. Guo, and F. Zhang, "FAST-LIVO: Fast and tightly-coupled sparse-direct LiDAR-Inertial-Visual Odometry," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2022, pp. 4003–4009.

[40] J. Lin, C. Zheng, W. Xu, and F. Zhang, "$R^2$ LIVE: A robust, real-time, LiDAR-inertial-visual tightly-coupled state estimator and mapping," *IEEE Robot. Autom. Lett.*, vol. 6, no. 4, pp. 7469–7476, Oct. 2021.

[41] Y. Zhu, C. Zheng, C. Yuan, X. Huang, and X. Hong, "CamVox: A low-cost and accurate LiDAR-assisted visual SLAM system," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2021, pp. 5049–5055.

[42] W. Kim, B. Son, and I. Kim, "ViLT: Vision-and-language transformer without convolution or region supervision," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2021, pp. 5583–5594.

[43] R. Fukushima, K. Ota, A. Kanezaki, Y. Sasaki, and Y. Yoshiyasu, "Object memory transformer for object goal navigation," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2022, pp. 11288–11294.

[44] Y. Wang, X. Chen, L. Cao, W. Huang, F. Sun, and Y. Wang, "Multimodal token fusion for vision transformers," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 12176–12185.

[45] Z. Jiang, H. Taira, N. Miyashita, and M. Okutomi, "Self-supervised ego-motion estimation based on multi-layer fusion of RGB and inferred depth," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2022, pp. 7605–7611.

[46] A. Hassani, S. Walton, N. Shah, A. Abuduweili, J. Li, and H. Shi, "Escaping the big data paradigm with compact transformers," 2021, *arXiv:2104.05704*.

[47] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[48] A. Milioto, I. Vizzo, J. Behley, and C. Stachniss, "RangeNet++: Fast and accurate LiDAR semantic segmentation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Nov. 2019, pp. 4213–4220.

[49] Y. Tu and J. Xie, "UnDeepLIO: Unsupervised deep LiDAR-inertial odometry," in *Proc. Asian Conf. Pattern Recognit.* Cham, Switzerland: Springer, 2022, pp. 189–202.

[50] X. Zhan, Y. Cai, H. Li, Y. Li, and P. He, "A point cloud registration algorithm based on normal vector and particle swarm optimization," *Meas. Control*, vol. 53, nos. 3–4, pp. 265–275, Mar. 2020.

[51] Q. Li et al., "LO-Net: Deep real-time LiDAR odometry," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 8465–8474.

[52] H. Weytjens and J. De Weerdt, "Process outcome prediction: CNN vs. LSTM (with attention)," in *Proc. Bus. Process Manage. Workshops (BPM)*. Seville, Spain: Springer, Sep. 2020, pp. 321–333.

[53] Z. Ahmad and N. Khan, "Human action recognition using deep multilevel multimodal ($M^2$) fusion of depth and inertial sensors," *IEEE Sensors J.*, vol. 20, no. 3, pp. 1445–1455, Feb. 2020.

[54] Z. Ahmad and N. Khan, "CNN-based multistage gated average fusion (MGAF) for human action recognition using depth and inertial sensors," *IEEE Sensors J.*, vol. 21, no. 3, pp. 3623–3634, Feb. 2021.

[55] M. Brossard, S. Bonnabel, and A. Barrau, "Denoising IMU gyroscopes with deep learning for open-loop attitude estimation," *IEEE Robot. Autom. Lett.*, vol. 5, no. 3, pp. 4796–4803, Jul. 2020.

[56] A. Savitzky and M. J. E. Golay, "Smoothing and differentiation of data by simplified least squares procedures." *Anal. Chem.*, vol. 36, no. 8, pp. 1627–1639, Jul. 1964.

[57] A. Dosovitskiy et al., "An image is worth 16 × 16 words: Transformers for image recognition at scale," 2020, *arXiv:2010.11929*.

[58] Z. Xia, X. Pan, S. Song, L. E. Li, and G. Huang, "Vision transformer with deformable attention," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 4784–4793.

[59] Y. Cho, G. Kim, and A. Kim, "Unsupervised geometry-aware deep LiDAR odometry," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2020, pp. 2145–2152.

[60] A. Prakash, K. Chitta, and A. Geiger, "Multi-modal fusion transformer for end-to-end autonomous driving," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 7073–7083.

[61] P. Zhang et al., "Multi-Scale Vision Longformer: A new vision transformer for high-resolution image encoding," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 2978–2988.

[62] C. R. Chen, Q. Fan, and R. Panda, "CrossViT: Cross-attention multi-scale vision transformer for image classification," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 347–356.

[63] J. Gu et al., "Multi-scale high-resolution vision transformer for semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 12084–12093.

[64] M. Chen et al., "CF-ViT: A general coarse-to-fine method for vision transformer," 2022, *arXiv:2203.03821*.

[65] B.-W. Jeon, G.-I. Lee, S.-H. Lee, and R.-H. Park, "Coarse-to-fine frame interpolation for frame rate up-conversion using pyramid structure," *IEEE Trans. Consum. Electron.*, vol. 49, no. 3, pp. 499–508, Aug. 2003.

[66] A. Kendall, M. Grimes, and R. Cipolla, "PoseNet: A convolutional network for real-time 6-DOF camera relocalization," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 2938–2946.

[67] S. Wang, R. Clark, H. Wen, and N. Trigoni, "DeepVO: Towards end-to-end visual odometry with deep recurrent convolutional neural networks," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2017, pp. 2043–2050.

[68] Y. Shavit, R. Ferens, and Y. Keller, "Learning multi-scene absolute pose regression with transformers," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 2713–2722.

[69] A. Kendall and Y. Gal, "What uncertainties do we need in Bayesian deep learning for computer vision?" in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–11.

[70] Y. Zou, P. Ji, Q.-H. Tran, J.-B. Huang, and M. Chandraker, "Learning monocular visual odometry via self-supervised long-term modeling," in *Proc. 16th Eur. Conf. Comput. Vis. ECCV*. Glasgow, U.K.: Springer, Aug. 2020, pp. 710–727.

[71] A. Gabas, Y. Yoshiyasu, R. P. Singh, R. Sagawa, and E. Yoshida, "APE: A more practical approach to 6-DoF pose estimation," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2020, pp. 3164–3168.

[72] R. Cipolla, Y. Gal, and A. Kendall, "Multi-task learning using uncertainty to weigh losses for scene geometry and semantics," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7482–7491.

[73] M. Yokozuka, K. Koide, S. Oishi, and A. Banno, "LiTAMIN2: Ultra light LiDAR-based SLAM using geometric approximation applied with KL-divergence," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2021, pp. 11619–11625.

[74] G. Wang, X. Gao, T. Zhang, Q. Xu, and W. Zhou, "LiDAR odometry and mapping based on neighborhood information constraints for rugged terrain," *Remote Sens.*, vol. 14, no. 20, p. 5229, Oct. 2022.
[75] Y. Zhou, C. Barnes, J. Lu, J. Yang, and H. Li, "On the continuity of rotation representations in neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 5738–5746.
[76] Y. Liu, E. Sangineto, W. Bi, N. Sebe, B. Lepri, and M. Nadai, "Efficient training of visual transformers with small datasets," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 34, 2021, pp. 23818–23830.
[77] W. Zhang and I. Vaidya, "MixUp training leads to reduced overfitting and improved calibration for the transformer architecture," 2021, *arXiv:2102.11402*.
[78] D. Rondao, N. Aouf, and M. A. Richardson, "ChiNet: Deep recurrent convolutional learning for multimodal spacecraft pose estimation," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 59, no. 2, pp. 937–949, Apr. 2023.
[79] Y. Zhou, J. Xiao, Y. Zhou, and G. Loianno, "Multi-robot collaborative perception with graph neural networks," *IEEE Robot. Autom. Lett.*, vol. 7, no. 2, pp. 2289–2296, Apr. 2022.
[80] H. Huang, X. Zhou, J. Cao, R. He, and T. Tan, "Vision transformer with super token sampling," in *Proc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 22690–22699.

**Leyuan Sun** received the B.Eng. degree from Jiangsu University, Zhenjiang, China, in 2015, and the M.Eng. and Ph.D. degrees from the University of Tsukuba, Tsukuba, Japan, in 2020 and 2023, respectively.

From April 2018 to March 2023, he was a member of the CNRS-AIST Joint Robotics Laboratory (JRL), IRL, National Institute of Advanced Industrial Science and Technology (AIST), Tsukuba, where he currently holds a postdoctoral position with the Computer Vision Research Team. His research interests include robotics, computer vision, SLAM, virtual reality teleoperation, and object goal navigation.

Dr. Sun serves as a Reviewer for the IEEE ROBOTICS AND AUTOMATION LETTERS and *Advanced Robotics*.

**Guanqun Ding** received the M.Eng. degree from the Jiangxi University of Finance and Economics, Nanchang, China, in 2019, and the Ph.D. degree from the University of Tsukuba, Tsukuba, Japan, in 2023.

Since February 2021, he is a member of the Digital Architecture Research Center (DigiARC), National Institute of Advanced Industrial Science and Technology (AIST), Tokyo, Japan, where he currently holds a postdoctoral position with DigiARC. His research interests include robotics, odometry, computer vision, saliency detection, and visual attention.

Dr. Ding serves as a Reviewer for the IEEE TRANSACTIONS ON MULTIMEDIA.

**Yue Qiu** received the B.E. degree from the Hefei University of Technology, Hefei, China, in 2014, and the M.S. and Ph.D. degrees from the University of Tsukuba, Tsukuba, Japan, in 2018 and 2021, respectively.

She is currently a Researcher with the Computer Vision Research Team, Artificial Intelligence Research Center (AIRC), National Institute of Advanced Industrial Science and Technology (AIST), Tsukuba. Her research interests include computer vision, 3-D vision, vision and language, and natural language processing.

Dr. Qiu serves as a Reviewer for top-tier conferences in computer vision and artificial intelligence, such as Computer Vision and Pattern Recognition (CVPR), International Conference on Computer Vision (ICCV), European Conference on Computer Vision (ECCV), and Conference on Neural Information Processing Systems (NeurIPS).

**Yusuke Yoshiyasu** (Member, IEEE) received the Ph.D. degree from Keio University, Tokyo, Japan, in March 2013.

He was a Visiting Scholar with the Leonidas Guibas Laboratory, Computer Science Department, Stanford University, Stanford, CA, USA, from 2015 to 2016. He is a Senior Research Scientist with the National Institute of Advanced Industrial Science and Technology (AIST), Tsukuba, Japan, where he is a member of the Computer Vision Research Team, AI Research Center. He is also an Adjunct Member of CNRS-AIST Joint Robotics Laboratory (JRL). His research interests include shape analysis, computer vision, robot vision, and machine learning.

**Fumio Kanehiro** (Member, IEEE) received the B.E., M.E., and Ph.D. degrees in engineering from the University of Tokyo, Tokyo, Japan, in 1994, 1996, and 1999, respectively.

He was a Research Fellow of the Japan Society for the Promotion of Science from 1998 to 1999. In 2000, he joined the Electrotechnical Laboratory, Agency of Industrial Science and Technology, Ministry of Industrial Science and Technology (AIST-MITI), later reorganized as the National Institute of Advanced Industrial Science and Technology (AIST), Tsukuba, Japan. In April 2007, he was a Visiting Researcher at the Laboratory for Analysis and Architecture of Systems-French National Center for Scientific Research (LAAS-CNRS), Toulouse, France, for one year and three months. He is currently the Director of CNRS-AIST Joint Robotics Laboratory (JRL), IRL, AIST and the Professor of Department of Intelligent and Mechanical Interaction Systems, Graduate School of Science and Technology, University of Tsukuba. His research interests include the software platform development and whole body motion planning of the humanoid robot.