IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2024, Yokohama, Japan. Cite as RA-L paper.

Attentive Multimodal Fusion for Optical and Scene Flow

Youjie Zhou[1], Guofeng Mei[2], Yiming Wang[2], Fabio Poiesi[2], Yi Wan[1]

***Abstract*—This paper presents an investigation into the estimation of optical and scene flow using RGBD information in scenarios where the RGB modality is affected by noise or captured in dark environments. Existing methods typically rely solely on RGB images or fuse the modalities at later stages, which can result in lower accuracy when the RGB information is unreliable. To address this issue, we propose a novel deep neural network approach named FusionRAFT, which enables early-stage information fusion between sensor modalities (RGB and depth). Our approach incorporates self- and cross-attention layers at different network levels to construct informative features that leverage the strengths of both modalities. Through comparative experiments, we demonstrate that our approach outperforms recent methods in terms of performance on the synthetic dataset Flyingthings3D, as well as the generalization on the real-world dataset KITTI. We illustrate that our approach exhibits improved robustness in the presence of noise and low-lighting conditions that affect the RGB images. We release the code, models and dataset at https://github.com/jiesico/FusionRAFT.**

***Index Terms*—Optical and scene flow, multimodal fusion, self- and cross-attention.**

I. Introduction

OPTICAL flow algorithms are essential for determining the motion of objects or regions within images between consecutive video frames. They generate a 2D vector field that describes the apparent movement of pixels over time. In contrast, scene flow focuses on estimating the pixel-level 3D motion in stereo or RGBD video frames [1]. These algorithms have wide applications in robotics [2], [3] and surveillance [4], [5]. Computing optical flow becomes particularly challenging in environments with non-informative textures or when scenes are captured under low-lighting conditions. To address these difficulties, deep learning methods have emerged as effective solutions for optical flow estimation, formulating the problem as an energy minimization task [6]–[9]. Deep learning-based optical flow approaches have demonstrated significant improvements over traditional methods [10]–[12].

Several approaches utilize the computation of a correlation volume in the visible spectrum (RGB) to estimate the optical flow between two frames [6], [10], [11]. The correlation volume captures inter-frame similarity by taking the dot product of the corresponding convolutional feature vectors and can be generated through an end-to-end deep network. This deep network can be designed to minimize an underlying energy function. However, relying solely on RGB information can be limited in scenes affected by motion blurs, non-informative textures, or low illumination conditions. To address this limitation, some approaches have incorporated multimodal information. For example, depth or point cloud data can provide an alternative representation of the underlying scene structure. This multimodal information can be integrated through *late fusion*, where feature vectors are combined without intermediate information exchange [1], [13], or through exchanging information between branches while sacrificing the independence of the single-modality representation [12].

In this paper, we present a novel multimodal fusion approach, named FusionRAFT, for optical and scene flow estimation, specifically designed to handle data captured in noisy or low-lighting conditions, for example those that can be encountered in search and rescue applications [14]. Our approach introduces three key components to address these challenges. Firstly, we propose a feature-level fusion technique that seamlessly blends RGB and depth information using a shared loss function. Secondly, we introduce a self-attention mechanism that enhances the expressiveness of feature vectors by dynamically balancing the importance of features within each individual modality. Lastly, we incorporate an optimized cross-attention module that facilitates information exchange and balance between RGB and depth modalities. We integrate these new modules within RAFT [10] and RAFT-3D [1], using an application-oriented data augmentation strategy to learn robust feature representations that make optical and scene flow estimation effective in complex environments. We conduct extensive evaluations on standard optical and scene flow benchmarks, as well as on two new settings that we introduce to assess robustness against photometric noise and challenging illumination conditions. Our method achieves state-of-the-art performance on the synthetic dataset FlyingThings3D [15] and demonstrates superior generalization capabilities on the real-world dataset KITTI [16] without fine-tuning.

II. Related work

We provide a comprehensive analysis of the recent progress in optical flow estimation using deep learning, followed by an in-depth investigation into the integration of multimodal fusion techniques for improving flow estimation performance.

Optical flow. FlowNet [6] pioneered the use of deep neural networks to estimate optical flow as a supervised learning

This work was supported by the China government project (2019JZZY010112), (2020JMRH0202), (YC-KYXM-07-2021) and by the PNRR project FAIR - Future AI Research (PE00000013), under the NRRP MUR program funded by the NextGenerationEU.

[1]Youjie Zhou and Yi Wan are with the School of Mechanical Engineering, Shandong University, China, and the Key Laboratory of High Efficiency and Clean Mechanical Manufacture of Ministry of Education, Shandong University, China. 202020511@mail.sdu.edu.cn,<wanyi>@sdu.edu.cn. Yi Wan is the corresponding author.

[2]Guofeng Mei, Yiming Wang, and Fabio Poiesi are with Fondazione Bruno Kessler, Italy <gmei,ywang,poiesi>@fbk.eu.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2024, Yokohama, Japan. Cite as RA-L paper.

task. FlowNet learns features across scales and abstraction levels to determine pixel correspondences. FlowNet inspired FlowNet2.0 [7], PWC-Net [17], MaskFlowNet [18] and LiteFlowNet3 [19]. FlowNet2.0 presents a warping operation and a method for stacking multiple networks through this operation [7]. PWC-Net utilizes pyramidal processing, warping, and a cost volume approach to improve both the size and accuracy of optical flow models [17]. MaskFlowNet incorporates an asymmetric occlusion-aware feature matching module, which learns to filter out occluded regions through feature warping without the need for explicit supervision [18]. LiteFlowNet3 tackles the challenge of estimating optical flow in the presence of partially occluded or homogeneous regions by using an adaptive affine transformation and a confidence map that identifies unreliable flow [19]. The confidence map is used to guide the generation of transformation parameters.

RAFT [10] is a per-pixel feature extraction approach that constructs multi-scale 4D correlation volumes for each pixel pair, and updates the flow field iteratively through a recurrent unit. Like FlowNet, RAFT has inspired GMA [20] and CRAFT [11]. GMA addresses occlusions by modeling image self-similarities by using a global motion aggregation module, a transformer-based approach for finding long-range dependencies between pixels in the first image, and a global aggregation of the corresponding motion features. CRAFT aims to estimate the large motion displacements through a semantic smoothing transformer layer that integrates the features of one image and a cross-attention layer that replaces the original dot-product operator for correlation used in RAFT. Unlike these approaches, we tackle the problem of estimating optical flow in situations of unreliable RGB information, such as noises and scarce illuminations, by appropriately fusing multiple modalities through self and cross attention within feature extraction layers.

Multimodal fusion. Multimodal fusion can be performed at various stages: early-, mid-, and late-fusion. Inspired by multimodal fusion, information from other modalities, such as depth or point cloud data, is introduced and integrated using multimodal fusion methods for optical and scene flow.

1) RGB + Point Cloud Data.: DeepLiDARFlow [13] exhibits improved performance in challenging conditions, such as reflective surfaces, poor illumination, and shadows. Images and point clouds are processed by using multi-scale feature pyramid networks. Late-fusion based on differentiable confidence volumes produces the fused features. CamLiFlow [12] improves upon DeepLiDARFlow by fusing dense image features and sparse point features more effectively. Instead of late-fusion, CamLiFlow adopts a multi-stage, bidirectional fusion strategy, in which the two modalities are learned in separate branches using modality-specific architectures. CamLiRAFT [21] further improves the performance based on the RAFT [10] framework, leading to superior results compared to CamLiFlow [12]. Our method differs from previous methods in that it ensures the independence of each modality through the use of two separate branches and balances the information between the modalities through multi-stage information exchange.

2) RGB + depth: RAFT-3D [1] extends RAFT to estimate both optical and scene flow from RGBD data. RGB images serve as inputs to the feature network, where a 4D correlation volume is constructed and a soft grouping of pixels into rigid objects is formed with the aid of depth information. Unlike RAFT [10], RAFT-3D employs late-fusion with the depth information and the RGB features in the prediction module, improving the stability of flow prediction. However, RAFT's feature extraction method may not sufficiently capture the rich 3D structural information. To address this, our approach employs early-fusion, in which features are extracted from both RGB and depth information, enabling stable estimation even in cases where RGB information is unreliable.

III. Our approach

We present a Multimodal Feature Fusion (MFF) Encoder that performs early fusion of RGB and depth modalities to improve the estimation of both optical and scene flow under noisy or poor lighting conditions. Our encoder is flexible and can be integrated into flow estimation frameworks by replacing their original feature encoder. To achieve this, we employ self-attention, cross-attention, and Multimodal Transfer Module (MMTM) [22]. We extract low-level features from each modality and improve their expressivity using self-attention. Cross-attention enables the network to attend to the most informative modality. MMTM is used to further fuse the attended features that are computed from the two modalities. Fig. 1(a) shows the architecture of our encoder.

A. Multimodal Feature Fusion Encoder

The Multimodal Feature Fusion Encoder takes a pair of consecutive RGBD frames $(P^t,\ P^{t+1})$ at time t as input. Each frame $P^t = \{I^t, Z^t\}$ is composed of a RGB image I^t and a depth image Z^t.

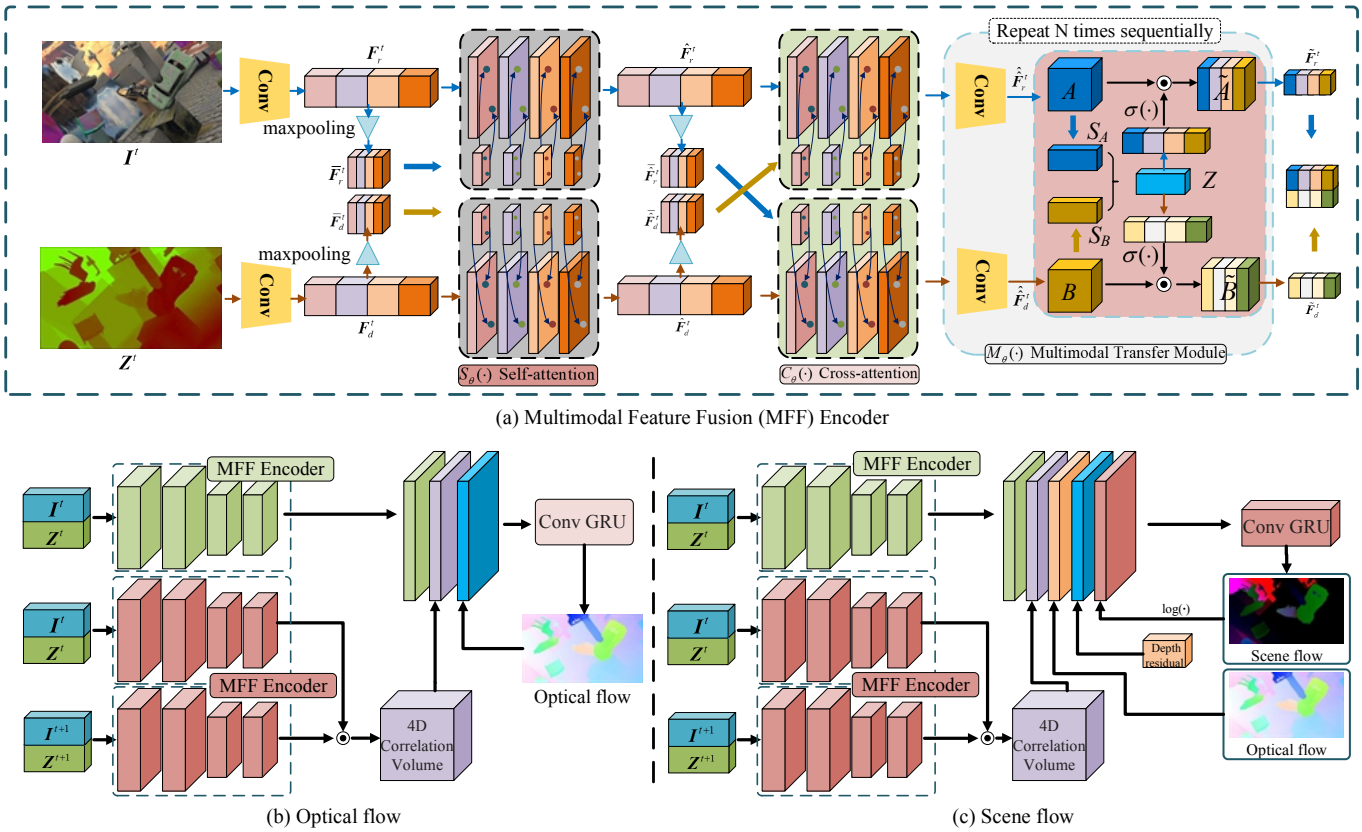
We first obtain low-level features $\boldsymbol{F}_r^t \in \mathbb{R}^{W\times H\times D}$ and $\boldsymbol{F}_d^t \in \mathbb{R}^{W\times H\times D}$ from each modality with convolutional blocks, where we use the subscript r to represent the RGB branch and d for the depth branch (Fig. 1(a)). D is the feature dimension and $W \times H$ is the resolution of the features.

Self-attention. The local features $\boldsymbol{F}_r^t$ and $\boldsymbol{F}_d^t$ are obtained with convolutions that have limited receptive fields, therefore we model global structures by establishing long-range dependencies through a self-attention module ($S_\theta(\cdot)$ in Fig. 1(a)). To mitigate the high computational cost of self-attention, we downsample $\boldsymbol{F}_r^t \in \mathbb{R}^{N\times D}$ and $\boldsymbol{F}_d^t \in \mathbb{R}^{N\times D}$ to obtain $\bar{\boldsymbol{F}}_r^t$ and $\bar{\boldsymbol{F}}_d^t$ via 3×3 and 5×5 max-pooling layers. With these downsampled features, we can use a multi-attention layer with four parallel attention heads to process $\boldsymbol{F}_r^t$ and $\bar{\boldsymbol{F}}_r^t$ (or $\boldsymbol{F}_d^t$ and $\bar{\boldsymbol{F}}_d^t$) in parallel and get $\hat{\boldsymbol{F}}_k^t$:

$$\begin{aligned}\hat{\boldsymbol{F}}_k^t &\leftarrow S_\theta(\boldsymbol{F}_k^t, \bar{\boldsymbol{F}}_k^t)\\ &= \boldsymbol{F}_k^t + \mathrm{MLP}\left(\sigma\left(\boldsymbol{W}_{Q_s}^t \boldsymbol{F}_k^t \left(\boldsymbol{W}_{K_s}^t \bar{\boldsymbol{F}}_k^t\right)^\top / \sqrt{D}\right) \boldsymbol{W}_{V_s}^t \bar{\boldsymbol{F}}_k^t\right),\end{aligned} \tag{1}$$

where $k \in \{r, d\}$ and σ is the *softmax* function. D is the feature dimension. $\boldsymbol{W}_{Q_s}^t \in \mathbb{R}^{N\times D}, \boldsymbol{W}_{K_s}^t \in \mathbb{R}^{J\times D}$ and $\boldsymbol{W}_{V_s}^t \in \mathbb{R}^{J\times D}$ are the query, key and value matrices, where $N = W\times H$, $J = (W\times H)/(3\times 5)$. MLP$(\cdot)$ denotes a three-layer fully connected network with instance normalization [23] and ReLU [24] activation after the first two layers.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2024, Yokohama, Japan. Cite as RA-L paper.



(a) Multimodal Feature Fusion (MFF) Encoder

(b) Optical flow

(c) Scene flow

Fig. 1. Block diagram of FusionRAFT. (a) Our encoder architecture: RGB and depth frames are taken as inputs. The encoder network is a two-branch network with a transformer (self-attention plus cross-attention) and a Multimodal Transfer Module. (b) Optical flow and (c) scene flow architectures. Two consecutive RGBD frames are taken as inputs by the MFF for the feature encoder, and the first RGBD frame is taken as input by the MFF for the context encoder.

Cross-attention. We promote information exchange between the two modalities via cross-attention, which we implement through the network $C_\theta(\cdot)$ (Fig. 1(a)). Attention signals from one modality (e.g. RGB) emphasize the features of another modality (e.g. depth), and vice versa. Given the self-attended features $\hat{\boldsymbol{F}}_r^t \in \mathbb{R}^{N\times D}$ and $\hat{\boldsymbol{F}}_d^t \in \mathbb{R}^{N\times D}$, we also adopt two downsampled networks max-pooling (3×3), and max-pooling (5×5) to generate the downsampled image feature map $\bar{\hat{\boldsymbol{F}}}_r^t$ (or $\bar{\hat{\boldsymbol{F}}}_d^t$). We denote the transformed features as $\hat{\hat{\boldsymbol{F}}}_r^t \in \mathbb{R}^{N\times D}$ and $\hat{\hat{\boldsymbol{F}}}_d^t \in \mathbb{R}^{N\times D}$ attained by cross-attention via

$$\begin{aligned}\hat{\hat{\boldsymbol{F}}}_r^t &\leftarrow C_\theta(\hat{\boldsymbol{F}}_r^t, \bar{\hat{\boldsymbol{F}}}_d^t)\\ &= \hat{\boldsymbol{F}}_r^t + \text{MLP}\left(\sigma\left(\boldsymbol{W}_{Q_c}^t \hat{\boldsymbol{F}}_r^t \left(\boldsymbol{W}_{K_c}^t \bar{\hat{\boldsymbol{F}}}_d^t\right)^\top / \sqrt{D}\right)\boldsymbol{W}_{V_c}^t \bar{\hat{\boldsymbol{F}}}_d^t\right),\end{aligned} \tag{2}$$

where $W_{Q_c}^t \in \mathbb{R}^{N\times D}, W_{K_c}^t \in \mathbb{R}^{J\times D}$ and $W_{V_c}^t \in \mathbb{R}^{J\times D}$ are the query, key and value matrices. This cross-attention block is also applied in the reverse direction so that information flows in both directions, i.e., RGB→depth and depth→RGB.

Multimodal Transfer Module. Because our architecture operates with multimodal information, we further promote information exchange between modalities after attention. Let $M_\theta(\cdot)$ be the Multimodal Transfer Module [22] we use to improve the balance between RGB and depth information (Fig. 1(a)). Let $\hat{\hat{\boldsymbol{F}}}_r^t \in \mathbb{R}^{N\times D_M}$ and $\hat{\hat{\boldsymbol{F}}}_d^t \in \mathbb{R}^{N\times D_M}$ be the input multimodal features to MMTM, and $\tilde{\boldsymbol{F}}_r^t \in \mathbb{R}^{N\times D_M}$ and $\tilde{\boldsymbol{F}}_d^t \in \mathbb{R}^{N\times D_M}$ be the respective outputs. MMTM first squeezes the feature vectors into $S_{\tilde{\boldsymbol{F}}_r^t}$ and $S_{\tilde{\boldsymbol{F}}_d^t}$ via a global average pooling. MMTM then maps these tensors to a joint representation Z through concatenation and a fully-connected layer. Based on Z, MMTM finally balances RGB and depth information by gating the channel-wise features:

$$\begin{aligned}S_{\tilde{\boldsymbol{F}}_k^t} &= \frac{1}{\Pi_{t=1}^K N_k}\sum_{n_1,\cdots,n_K}\hat{\hat{\boldsymbol{F}}}_k^t(n_1,\cdots,n_K),\\ Z &= \mathbf{W}[S_{\tilde{\boldsymbol{F}}_r^t}, S_{\tilde{\boldsymbol{F}}_d^t}] + b\\ \tilde{\boldsymbol{F}}_k^t &= 2\sigma(\mathbf{W}_{\tilde{\boldsymbol{F}}_k^t} Z)\odot \hat{\hat{\boldsymbol{F}}}_k^t,\end{aligned} \tag{3}$$

where $[\cdot,\cdot]$ is the concatenation operator and $k\in\{r,d\}$. N_k represents the spatial dimensions of $\hat{\hat{\boldsymbol{F}}}_k^t$ and D_M represents the number of channels of the features. $\mathbf{W}\in\mathbb{R}^{D_Z\times 2D_M}$, $\mathbf{W}_{\tilde{\boldsymbol{F}}_k^i}\in\mathbb{R}^{D_M\times D_Z}$ are the weights, and $b\in\mathbb{R}^{D_Z}$ are the biases of the fully connected layers.

B. Optical and scene flow estimation

The inputs of optical and scene flow estimation are the feature vectors $[\tilde{\boldsymbol{F}}_r^t, \tilde{\boldsymbol{F}}_d^t]$ and $[\tilde{\boldsymbol{F}}_r^{t+1}, \tilde{\boldsymbol{F}}_d^{t+1}]$. By calculating the dot product of feature vectors between the inputs, a 4D correlation volume $\mathbf{C}$ is generated:

$$\begin{aligned}&fnet(P^t) = [\tilde{\boldsymbol{F}}_r^t, \tilde{\boldsymbol{F}}_d^t] = [M_\theta(C_\theta(S_\theta(I^t), S_\theta(Z^t)))],\\ &\mathbf{C}(P^t, P^{t+1}) = \langle fnet(P^t), fnet(P^{t+1})\rangle.\end{aligned} \tag{4}$$

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2024, Yokohama, Japan. Cite as RA-L paper.

where $\langle \cdot, \cdot \rangle$ is the dot product operator. A four-layer pyramid $\{\mathbf{C}_1, \mathbf{C}_2, \mathbf{C}_3, \mathbf{C}_4\}$ is generated by reducing the last two dimensions of the correlation volume through pooling with kernels of size 1, 2, 4, and 8.

We compute 4D correlation volumes to estimate optical and scene flow [1], [10]. Through $\{\mathbf{C}_1, \mathbf{C}_2, \mathbf{C}_3, \mathbf{C}_4\}$, we iteratively estimate the dense displacement field $\{\mathbf{f}^1_{est}, \mathbf{f}^2_{est}, ..., \mathbf{f}^M_{est}\}$ with M iterations to update the optical and scene flow. We train our network by computing the loss between the estimated flow and the ground-truth flow $\mathbf{f}_{gt}$ as

$$\mathcal{L} = \sum_{k=1}^{M} \gamma^{M-k} \left\| \mathbf{f}^k_{est} - \mathbf{f}_{gt} \right\|_1, \tag{5}$$

where as the iteration k increases, the weight per loss term exponentially increases with a base γ. Fig. 1(b,c) show how our Multimodal Feature Fusion Encoder is integrated in RAFT and RAFT-3D to estimate the optical flow and the scene flow, respectively. Our module can be integrated seamlessly and does not require any modification to RAFT and RAFT-3D's modules after the 4D correlation volume computation.

IV. Experiments

We compare FusionRAFT against state-of-the-art approaches on the FlyingThings3D [15] and KITTI [16] datasets. We design two experimental settings to mimic corrupted RGB images and poor lighting condition scenarios. We also evaluate on data we acquired with a RGBD sensor in various lighting conditions. We report both quantitative and qualitative results, and carry out ablation studies.

A. Experimental setup

Datasets. FlyingThings3D [15] is split into *clean* and *final* sets containing dynamic synthetic scenes. The former is composed of 27K RGBD images including changing lighting and shading effects, while the latter is an augmented version of the former with simulated challenging motions and blurs. Each set contains train and test splits. Previous methods [1], [10], [12] exclude samples containing fast-moving objects during the evaluation. However, as such visual challenges is of interests to our problem, we use the *whole* training set of FlyingThings3D and sample 1K RGBD image pairs from the *whole* test set for the evaluation. KITTI consists of real-world scenes captured from vehicles in urban scenarios. Because the original dataset does not provide depth data, we use the disparity estimated by GA-Net [25] as in [1]. We exploit KITTI to assess the ability of our model and the compared ones in generalizing from synthetic to real data, without training or finetuning using any of the KITTI's sequences. We use the training set of KITTI as our evaluation set since KITTI's test set is not publicly available. To further validate the performance of FusionRAFT in real-world scenarios, we collect an RGBD dataset using a Realsense D415 camera in an indoor office with moving people under three lighting setups, named Bright, Dimmed, and Dark. The Bright setting features bright lighting, where the moving objects are clearly visible. The Dimmed setting features dimmed lighting, where the moving objects can be observed with a lower visual quality. The Dark setting features very low lighting where the moving objects can be barely seen. We only qualitatively evaluate this dataset because we could not produce optical flow ground truth.

Evaluation metrics. We quantify the optical and scene flow results using conventional evaluation metrics [1], [10], [11]: for the optical flow we use $\mathrm{AEPE}_{\mathrm{2D}}$(pixel), $\mathrm{ACC}_{\mathrm{1px}}(\%)$ and $\mathrm{Fl}^{\mathrm{all}}_{\mathrm{2D}}(\%)$, for the scene flow we use $\mathrm{AEPE}_{\mathrm{3D}}$(m), $\mathrm{ACC}_{\mathrm{0.05m}}(\%)$, $\mathrm{ACC}_{\mathrm{0.10m}}(\%)$ and $\mathrm{Fl}^{\mathrm{all}}_{\mathrm{3D}}(\%)$. $\mathrm{AEPE}_{\mathrm{2D}}$ measures the average end-point error (EPE) [10], which is an average value of all the 2D flow errors. $\mathrm{AEPE}^{\mathrm{epe<100}}_{\mathrm{2D}}$ measures the average end-point error (EPE) among the 2D flow errors that are less than 100 pixels. $\mathrm{AEPE}_{\mathrm{3D}}$ is the average of euclidean distance (EPE for 3D) between the ground-truth 3D scene flow and the predicted results. $\mathrm{AEPE}^{\mathrm{epe<1}}_{\mathrm{3D}}$ measures the average end-point error (EPE) among the 3D flow errors that are less than 1 meter. $\mathrm{ACC}_{\mathrm{1px}}$ [1] measures the portion of errors that are within a threshold of one pixel. $\mathrm{ACC}_{\mathrm{0.05m}}$ [1] measures the portion of errors that are within a threshold of 0.05 meters, while $\mathrm{ACC}_{\mathrm{0.10m}}$ [1] measures the portion of errors that are within a threshold of 0.10 meters. $\mathrm{MEAN}_{\mathrm{AEPE}}$ and $\mathrm{MEAN}_{\mathrm{ACC}}$ are the average values of $\mathrm{AEPE}^{\mathrm{all}}_{\mathrm{2D}}$ and $\mathrm{ACC}_{\mathrm{1px}}$, respectively, calculated over FlyingThings3D-clean and FlyingThings3D-final. $\mathrm{Fl}^{\mathrm{all}}_{\mathrm{2D}}$ [11] is the percentage of outlier pixels whose end-point error is > 3 pixels or 5% of the ground-truth flow magnitude. $\mathrm{Fl}^{\mathrm{all}}_{\mathrm{3D}}$ [26] is the percentage of outlier pixels whose 3D Euclidean distance between the ground-truth 3D scene flow and the predicted one is > 0.3 m or 5% of the ground-truth flow magnitude.

Evaluation settings. Environments with poor light conditions lead to weak texture information that can compromise the stability of feature representation. Also additive Gaussian noises can affect optical and scene flow estimation. To assess the robustness, we design three experimental settings on the public FlyingThings3D and KITTI datasets: *Standard*: we use the original version of the dataset; *AGN*: we apply Additive Gaussian Noise on RGB images; *Dark*: we darken RGB images. In AGN we randomly sample noise values (α) from a normal distribution centered in zero with a standard deviation equal to 35. In Dark we divide pixel values by a random factor $\beta \sim \mathrm{U}(\{1, 2, \cdots, 9\})$.

Implementation details. We implemented FusionRAFT in PyTorch with all modules initialized with random weights. We train our network for 100K iterations with the batch size of 6 on 3 Nvidia 3090 GPUs. During training, we set the initial learning rate at $1.25 \cdot 10^{-4}$ and use linear decay. We apply MMTM sequentially with N = 3 times as suggested in the original paper [22]. We set γ=0.8 in Eq. (5) as in RAFT [10].

B. Comparisons

We compare FusionRAFT against RGB methods for 2D optical flow estimation, i.e. RAFT [10], GMA [20], CRAFT [11], and Separable flow [27], and against methods for 3D scene flow estimation, i.e. RAFT-3D [1] and CamLiRAFT [21]. See Sec. II for the description of these methods.

1) Quantitative results: Tab. I (top) reports optical flow results in Standard setting. FusionRAFT-2D outperforms GMA by $+1.56\%$ and $+1.55\%$ in terms of $\mathrm{ACC}_{\mathrm{1px}}$, and $+0.91$

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2024, Yokohama, Japan. Cite as RA-L paper.

TABLE I

OPTICAL FLOW ESTIMATION IN THE STANDARD, AGN, AND DARK SETTINGS ON FLYINGTHINGS3D-CLEAN, FLYINGTHINGS3D-FINAL, AND KITTI. ALL MODELS ARE TRAINED WITH FLYINGTHINGS3D, WITHOUT FINE-TUNING ON KITTI. BOLD FONT INDICATES THE BEST-PERFORMING METHOD.

	Method	Input	FlyingThings3D-clean			FlyingThings3D-final			KITTI-Train	
			$\mathrm{ACC}_{1\mathrm{px}}$	$\mathrm{AEPE}_{2\mathrm{D}}^{\mathrm{epe}<100}$	$\mathrm{AEPE}_{2\mathrm{D}}^{\mathrm{all}}$	$\mathrm{ACC}_{1\mathrm{px}}$	$\mathrm{AEPE}_{2\mathrm{D}}^{\mathrm{epe}<100}$	$\mathrm{AEPE}_{2\mathrm{D}}^{\mathrm{all}}$	$\mathrm{AEPE}_{2\mathrm{D}}^{\mathrm{all}}$	$\mathrm{Fl}_{2\mathrm{D}}^{\mathrm{all}}$
Standard setting	RAFT [10]	RGB	77.06	2.65	4.69	76.91	2.67	4.39	6.76	20.99
	GMA [20]	RGB	78.81	2.58	4.43	78.66	2.57	4.20	6.10	20.47
	Separable flow [27]	RGB	75.39	2.88	4.57	75.29	2.84	4.29	6.40	20.66
	CRAFT [11]	RGB	77.90	2.79	4.85	77.70	2.77	4.66	6.82	21.95
	FusionRAFT-2D	RGBD	**80.37**	**2.17**	**3.52**	**80.21**	**2.22**	**3.42**	**5.49**	**18.05**
	RAFT-3D [1]	RGBD	86.01	1.79	3.58	85.97	1.76	3.57	5.91	17.80
	CamLiRAFT [21]	RGB+LiDAR	83.59	1.81	3.03	83.26	1.80	2.84	4.84	14.76
	FusionRAFT-3D	RGBD	**87.45**	**1.57**	**2.58**	**87.39**	**1.58**	**2.69**	**4.70**	**12.36**
AGN setting	RAFT [10]	RGB	71.42	2.98	4.89	71.01	2.96	4.64	7.23	23.45
	GMA [20]	RGB	72.63	2.99	5.23	72.20	2.94	5.24	7.01	23.52
	Separable flow [27]	RGB	68.95	3.15	5.32	68.57	3.18	5.23	8.26	25.79
	CRAFT [11]	RGB	73.30	2.86	4.65	72.81	2.88	4.67	7.45	23.65
	FusionRAFT-2D	RGBD	**77.24**	**2.24**	**3.50**	**76.77**	**2.30**	**3.38**	**5.47**	**19.15**
	RAFT-3D [1]	RGBD	84.59	1.79	3.14	84.26	1.84	3.21	5.50	17.94
	CamLiRAFT [21]	RGB+LiDAR	76.98	2.23	3.98	76.31	2.33	3.71	5.26	16.98
	FusionRAFT-3D	RGBD	**86.75**	**1.55**	**2.63**	**86.71**	**1.53**	**2.60**	**4.53**	**11.57**
Dark setting	RAFT [10]	RGB	60.26	4.00	8.15	60.36	4.01	7.85	11.75	31.81
	GMA [20]	RGB	63.36	4.50	9.96	62.10	4.62	10.34	9.65	27.87
	Separable flow [27]	RGB	68.20	4.84	7.96	68.03	4.86	7.79	10.09	28.41
	CRAFT [11]	RGB	70.07	4.84	8.46	69.77	4.87	8.44	11.10	29.47
	FusionRAFT-2D	RGBD	**76.70**	**2.39**	**3.65**	**76.57**	**2.38**	**3.66**	**8.29**	**23.87**
	RAFT-3D [1]	RGBD	81.03	2.20	3.78	80.96	2.20	3.56	15.14	32.08
	CamLiRAFT [21]	RGB+LiDAR	74.80	2.54	4.54	74.73	2.64	4.11	7.44	**16.97**
	FusionRAFT-3D	RGBD	**87.11**	**1.55**	**2.91**	**87.03**	**1.58**	**2.84**	**7.26**	20.07

TABLE II

SCENE FLOW ESTIMATION IN THE STANDARD, AGN, AND DARK SETTINGS ON FLYINGTHINGS3D-CLEAN, FLYINGTHINGS3D-FINAL, AND KITTI. ALL MODELS ARE TRAINED WITH FLYINGTHINGS3D, WITHOUT FINE-TUNING ON KITTI. BOLD FONT INDICATES THE BEST-PERFORMING METHOD.

Method	Setting	FlyingThings3D-clean				FlyingThings3D-final				KITTI-Train	
		$\mathrm{ACC}_{0.05\mathrm{m}}$	$\mathrm{ACC}_{0.10\mathrm{m}}$	$\mathrm{AEPE}_{3\mathrm{D}}^{\mathrm{epe}<1}$	$\mathrm{AEPE}_{3\mathrm{D}}^{\mathrm{all}}$	$\mathrm{ACC}_{0.05\mathrm{m}}$	$\mathrm{ACC}_{0.10\mathrm{m}}$	$\mathrm{AEPE}_{3\mathrm{D}}^{\mathrm{epe}<1}$	$\mathrm{AEPE}_{3\mathrm{D}}^{\mathrm{all}}$	$\mathrm{AEPE}_{3\mathrm{D}}^{\mathrm{all}}$	$\mathrm{Fl}_{3\mathrm{D}}^{\mathrm{all}}$
RAFT-3D [1]	Standard	74.01	81.22	0.064	0.186	74.25	81.43	0.064	0.180	0.136	5.20
CamLiRAFT [21]	Standard	76.83	**87.98**	**0.049**	0.104	**76.87**	**88.20**	**0.049**	0.102	**0.121**	7.13
FusionRAFT-3D	Standard	**77.04**	83.74	0.056	**0.100**	76.80	83.58	0.057	**0.101**	0.134	**4.90**
RAFT-3D [1]	AGN	75.05	81.77	0.065	0.193	74.75	81.56	0.066	0.144	0.134	5.36
CamLiRAFT [21]	AGN	73.83	**86.59**	**0.055**	0.116	73.38	**86.31**	**0.055**	0.126	**0.122**	7.81
FusionRAFT-3D	AGN	**76.60**	82.71	0.061	**0.104**	**76.35**	82.55	0.062	**0.107**	0.134	**5.52**
RAFT-3D [1]	Dark	71.21	79.33	0.071	0.203	71.00	79.24	0.072	0.145	0.145	9.39
CamLiRAFT [21]	Dark	66.70	82.24	0.068	**0.141**	66.65	82.01	0.069	0.127	0.171	9.39
FusionRAFT-3D	Dark	**76.72**	**83.62**	**0.057**	0.175	**76.58**	**83.51**	**0.057**	**0.112**	**0.136**	**6.20**

and $+0.78$ in terms of $\mathrm{AEPE}_{2\mathrm{D}}^{\mathrm{all}}$ in FlyingThings3D-clean and FlyingThings3D-final, respectively. FusionRAFT-3D outperforms RAFT-3D by $+1.44\%$ and $+1.42\%$ in terms of $\mathrm{ACC}_{1\mathrm{px}}$, and $+1.00$ and $+0.88$ in terms of $\mathrm{AEPE}_{2\mathrm{D}}^{\mathrm{all}}$. While RAFT-3D extracts features only from RGB images, our MFF encoder extracts features from both RGB and depth, producing more informative internal representations. FusionRAFT-3D outperforms CamLiRAFT by $+3.86\%$ and $+4.13\%$ in terms of $\mathrm{ACC}_{1\mathrm{px}}$, and $+0.45$ and $+0.15$ in terms of $\mathrm{AEPE}_{2\mathrm{D}}^{\mathrm{all}}$.

Tab. I (middle) reports optical flow results in AGN setting. FusionRAFT-2D outperforms CRAFT by $+3.94\%$ and $+3.96\%$ in terms of $\mathrm{ACC}_{1\mathrm{px}}$, and $+1.15$ and $+1.29$ in terms of $\mathrm{AEPE}_{2\mathrm{D}}^{\mathrm{all}}$ in FlyingThings3D-clean and FlyingThings3D-final, respectively. FusionRAFT-3D outperforms RAFT-3D by $+2.16\%$ and $+2.45\%$ in in terms of $\mathrm{ACC}_{1\mathrm{px}}$ and $+0.51$ and $+0.61$ in terms of $\mathrm{AEPE}_{2\mathrm{D}}^{\mathrm{all}}$. $\mathrm{AEPE}_{2\mathrm{D}}^{\mathrm{all}}$ of RAFT-3D is lower than that of the Standard setting. This is because $\mathrm{AEPE}_{2\mathrm{D}}^{\mathrm{all}}$ computes the average of all errors, and the average is known to be sensitive to outliers. In fact by computing the median (less sensitive to outliers), the performance of RAFT-3D in AGN setting is worse than the Standard setting: e.g. RAFT-3D in FlyingThings3D-clean achieves a median $\mathrm{EPE}_{2\mathrm{D}}^{\mathrm{all}}$ of 0.127 and 0.130 in the Standard and AGN settings, respectively.

Tab. I (bottom) reports optical flow results in Dark setting. RGB methods perform worse than the previous settings, whereas our FusionRAFT methods outperform RGB methods, RAFT-3D, and CamLiRAFT. Tab. I also reports $\mathrm{AEPE}_{2\mathrm{D}}^{\mathrm{all}}$ and $\mathrm{Fl}_{2\mathrm{D}}^{\mathrm{all}}$ on KITTI without fine-tuning the models. Although CamLiRAFT demonstrates a better generalization capability than RAFT-3D, FusionRAFT outperforms almost all the other methods in all three settings, demonstrating its robustness and adaptability in real-world scenarios.

Tab. II reports scene flow results. FusionRAFT-3D outperforms RAFT-3D on both FlyingThings3D-clean and FlyingThings3D-final. CamLiRAFT scores on par with FusionRAFT-3D in all three settings. FusionRAFT-3D performs better in terms of $\mathrm{ACC}_{0.05\mathrm{m}}$, while CamLiRAFT performs better in terms of $\mathrm{ACC}_{0.10\mathrm{m}}$. This suggests that

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2024, Yokohama, Japan. Cite as RA-L paper.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2024, Yokohama, Japan. Cite as RA-L paper.

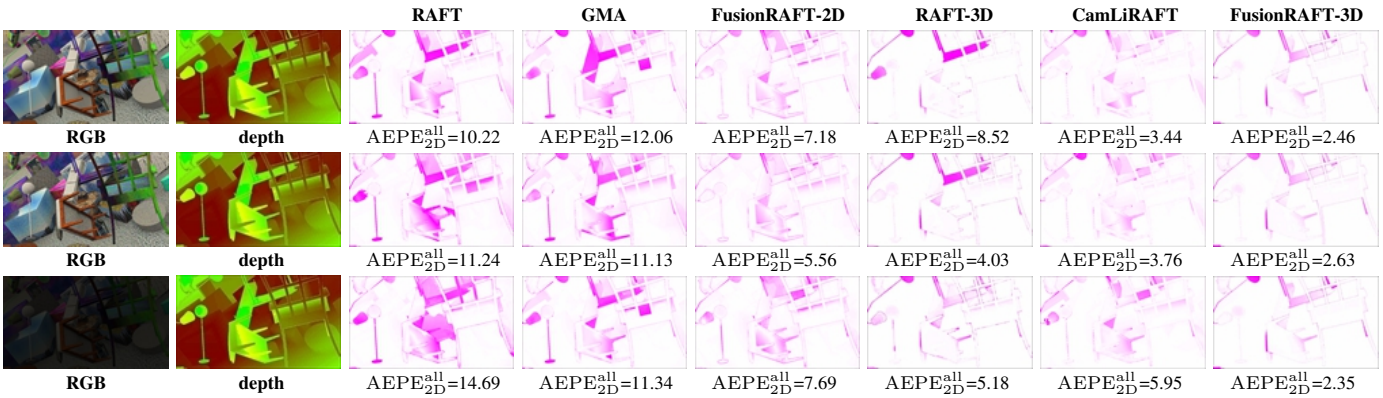


Fig. 2. Examples of optical flow estimation error in the FlyingThings3D-clean dataset. The more vivid the magenta, the higher the error. FusionRAFT-2D method handles optical flow estimation better than RGB-based methods, while FusionRAFT-3D method outperforms RAFT-3D with a smaller AEPE. Best viewed in color.

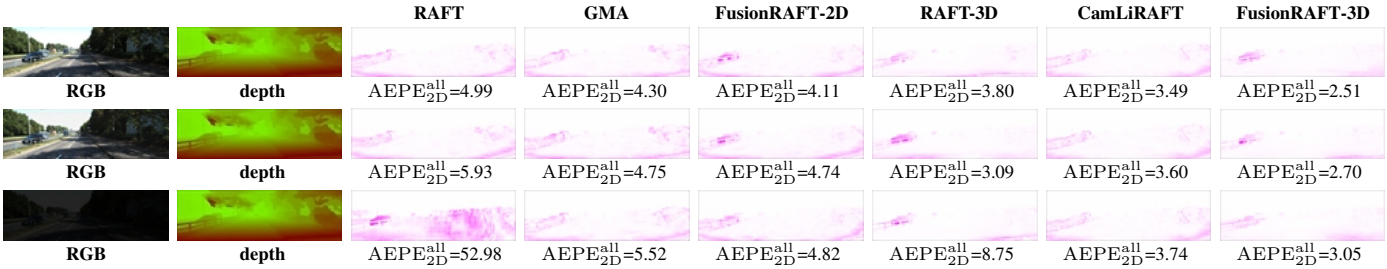


Fig. 3. Examples of optical flow estimation error in the KITTI dataset. The more vivid the magenta, the higher the error. FusionRAFT-2D method handles optical flow estimation better than all RGB-based methods. FusionRAFT-3D method outperforms RAFT-3D with a smaller AEPE. Best viewed in color.

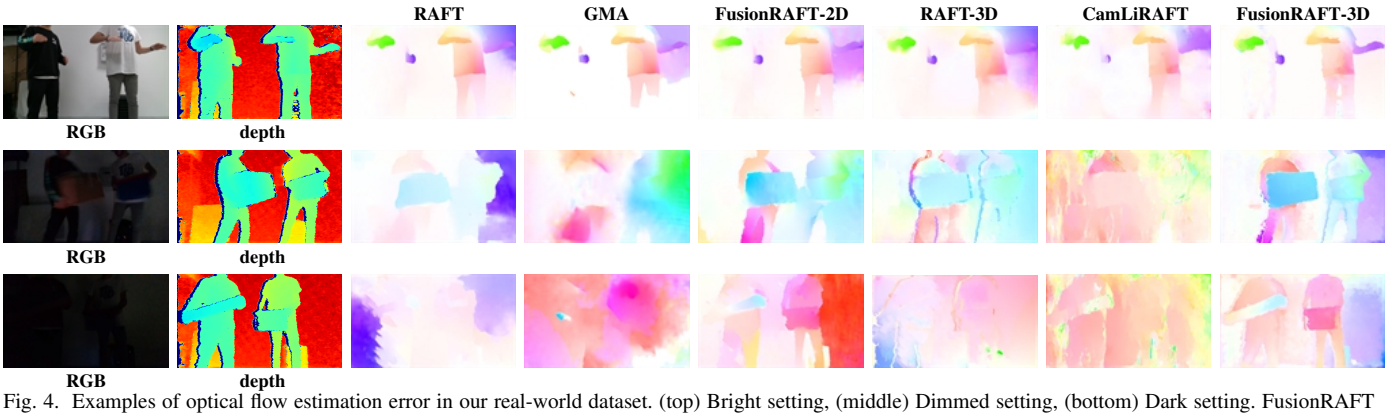


Fig. 4. Examples of optical flow estimation error in our real-world dataset. (top) Bright setting, (middle) Dimmed setting, (bottom) Dark setting. FusionRAFT method can handle also the Dark setting (see sharper flow boundaries). Best viewed in color.

FusionRAFT-3D produces more small flow errors than CamLiRAFT. In general, FusionRAFT-3D performs stably across all three settings, while the performance of RAFT-3D and CamLiRAFT degrades in the AGN and Dark settings. On KITTI, FusionRAFT-3D is the best-performing method on all three settings in terms of Fl$^{all}_{3D}$. In terms of AEPE$^{all}_{3D}$, CamLiRAFT performs better in the Standard and AGN settings, while FusionRAFT-3D scores the best in the Dark setting.

2) Qualitative results: We provide examples of qualitative optical flow results indicated with their corresponding AEPE$^{all}_{2D}$. We visualize the errors with respect to the ground-truth: the stronger the magenta, the higher the error. Fig. 2 and Fig. 3 show the results of optical flow errors on FlyingThings3D and KITTI, respectively. Both FusionRAFT-2D and FusionRAFT-3D consistently produce smaller AEPE$^{all}_{2D}$ values than the other methods, which can also be visually verified with less magenta areas produced by our models. Fig. 4 shows the flow estimation on our acquired indoor dataset with RAFT, GMA, RAFT-3D, CamLiRAFT, and FusionRAFT. In the Bright setting (top), all compared methods produce good-quality results. In the Dimmed setting (middle), RAFT, GMA, and CamLiRAFT show low-quality results, which we can observe from the poor edges produced by the moving objects. In the Dark setting, FusionRAFT is the only method that produces results where the moving objects are distinguishable.

C. Ablation study

Tab. III reports the ablation study on self-attention (SA), cross-attention (CA), and Multimodal Transfer Module (MMTM) on the FlyingThings3D dataset in both Standard and Dark settings. Overall, we can observe that all the components we added provide an incremental contribution to improve

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2024, Yokohama, Japan. Cite as RA-L paper.

TABLE III
ABLATION STUDY IN STANDARD AND DARK SETTINGS ON FLYINGTHINGS3D. $\mathrm{MEAN}_{\mathrm{ACC}}$ AND $\mathrm{MEAN}_{\mathrm{AEPE}}$ ARE THE MEAN OF $\mathrm{ACC}_{1\mathrm{px}}$ AND $\mathrm{AEPE}_{\mathrm{2D}}^{\mathrm{all}}$, RESPECTIVELY, ON FLYINGTHINGS3D-CLEAN AND FLYINGTHINGS3D-FINAL. ✓(RGB): SELF-ATTENTION IS PERFORMED ON THE RGB.

	Exp	RGB	depth	SA	CA	Fusion	FlyingThings3D-clean		FlyingThings3D-final		$\mathrm{MEAN}_{\mathrm{ACC}}$	$\mathrm{MEAN}_{\mathrm{AEPE}}$
							$\mathrm{ACC}_{1\mathrm{px}}$	$\mathrm{AEPE}_{\mathrm{2D}}^{\mathrm{all}}$	$\mathrm{ACC}_{1\mathrm{px}}$	$\mathrm{AEPE}_{\mathrm{2D}}^{\mathrm{all}}$		
Standard setting	1	✓	-	-	-	-	77.06	4.69	76.91	4.39	76.99	4.54
	2	✓	-	✓	-	-	77.81	4.50	77.72	4.26	77.77	4.38
	3	✓	✓	-	-	concat	79.21	3.74	79.06	3.71	79.14	3.73
	4	✓	✓	-	-	MMTM	79.43	3.79	79.27	3.69	79.35	3.74
	5	✓	✓	✓(RGB)	-	concat	79.35	3.72	79.18	3.66	79.27	3.69
	6	✓	✓	✓	-	concat	79.71	3.81	79.55	3.63	79.63	3.72
	7	✓	✓	✓	-	MMTM	78.97	3.73	78.77	3.65	78.87	3.69
	8	✓	✓	✓	✓	concat	80.14	3.56	79.95	3.53	80.05	3.55
	9	✓	✓	✓	✓	MMTM	80.37	3.52	80.21	3.42	80.29	3.47
Dark setting	10	✓	-	-	-	-	60.26	8.15	60.36	7.85	60.31	8.00
	11	✓	-	✓	-	-	67.92	8.01	67.80	7.77	67.86	7.89
	12	✓	✓	-	-	concat	75.33	4.06	75.17	4.05	75.25	4.06
	13	✓	✓	-	-	MMTM	75.56	3.97	75.40	3.99	75.48	3.98
	14	✓	✓	✓(RGB)	-	concat	75.47	3.95	75.32	3.94	75.40	3.95
	15	✓	✓	✓	-	concat	75.69	3.96	75.57	3.90	75.63	3.93
	16	✓	✓	✓	-	MMTM	75.75	3.81	75.60	3.76	75.68	3.79
	17	✓	✓	✓	✓	concat	76.43	3.79	76.26	3.72	76.35	3.76
	18	✓	✓	✓	✓	MMTM	76.70	3.65	76.57	3.66	76.64	3.66

the quality of the output optical flow compared to the RGB baseline. SA and CA consistently improve performance (see Exp 3 vs 6 vs 8, 4 vs 7 vs 9, and similarly for the Dark setting). The SA applied to both depth and RGB is better than applying it to the RGB branch only (see Exp 5 vs 6 for the Standard setting, and 14 vs 15 for the Dark setting). MMTM fusion consistently outperforms the simple concatenation of RGB and depth branches in the Dark setting (see Exp 12 vs 13, 15 vs 16, 17 vs 18). There is one case in the Standard setting where this last does not occur (see Exp 6 vs 7). In general, SA focuses on intra-modality relationships while CA focuses on inter-modality relationships. MMTM further exchanges information across modalities at a deeper level. The best performance is achieved when all the modules are activated.

D. Computation analysis

We measure the number of parameters, Floating-Point Operations (FLOPs), and inference time of all compared methods using FlyingThings3D. We conducted the experiments with a Nvidia 3090 GPU (24G) and I9-10900 CPUs, and reported the results in Tab. IV. Despite FusionRAFT-2D has the second-largest number of parameters, its number of FLOPs and inference time are in-between the other methods for optical flow estimation. The inference time of FusionRAFT-3D is slightly higher than that of CamLiRAFT, although our number of parameters is one order of magnitude larger than CamLiRAFT. From the per-component analysis of FusionRAFT-2D in Tab. V, we can observe that Self-attention and Cross-attention have a higher computational cost than MMTM and the two-branch encoder. The most time-consuming component is *Others* which includes all the other modules to compute the optical flow.

TABLE IV
COMPARATIVE COMPUTATIONAL ANALYSIS BY USING 960×540-SIZED IMAGES ON A NVIDIA 3090.

Models	Params [M]	FLOPs [T]	Inference time [ms]
RAFT [10]	5.31	0.78	99
GMA [20]	5.88	0.59	87
Separable flow [27]	8.35	0.50	639
CRAFT [11]	6.31	0.99	302
FusionRAFT-2D	8.13	0.68	219
RAFT-3D [1]	44.50	0.51	277
CamLiRAFT [21]	8.41	0.67	312
FusionRAFT-3D	86.32	0.76	398

TABLE V
ABLATIONS OF COMPUTATIONAL PERFORMANCE ON FUSIONRAFT-2D BY USING 960×540-SIZED IMAGES ON A NVIDIA 3090.

Models	Params [M]	FLOPs [T]	Inference time [ms]
Image encoder	2.33	0.11	13
Depth encoder	2.33	0.11	13
Self-attention	0.16	0.06	63
Cross-attention	0.09	0.04	43
MMTM	0.24	2M	1
Others	2.98	0.36	86
Total	8.13	0.68	219

V. CONCLUSIONS

We presented FusionRAFT, a novel approach for optical and scene flow estimation. FusionRAFT improves feature extraction with an early-fusion Multimodal Feature Fusion (MFF) Encoder. MFF attends to informative features and enables information exchange within and across modalities by using self-attention, cross-attention, and the Multimodal Transfer Module. Through experimental validation, we showed that FusionRAFT generates more stable and informative feature descriptions by exploiting the different modalities. FusionRAFT scores state-of-the-art results in Standard setting, but also in our newly introduced AGN and Dark settings where RGB information is corrupted. Future research directions may include the integration of FusionRAFT in robotic systems for autonomous navigation.

REFERENCES

[1] Z. Teed and J. Deng, "Raft-3d: Scene flow using rigidmotion embeddings," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2021.

[2] T. Zhang, H. Zhang, Y. Li, Y. Nakamura, and L. Zhang, "FlowFusion: Dynamic Dense RGB-D SLAM Based on Optical Flow," in *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, Paris, Fr, June 2020.

[3] G. de Croon, C. De-Wagter, and T. Seidl, "Enhancing optical-flow-based control by learning visual appearance cues for flying robots," *Nature Machine Intelligence*, vol. 3, no. 1, pp. 33–41, 2021.

[4] L. Sevilla-Lara, Y. Liao, F. Guney, V. Jampani, A. Geiger, and M. J. Black, "On the integration of optical flow and action recognition," in *Proceedings of German Conference on Pattern Recognition (GCPR)*, Stuttgart, Ger, October 2018.

[5] S. Sun, Z. Kuang, L. Sheng, W. Ouyang, and W. Zhang, "Optical flow guided feature: A fast and robust motion representation for video action recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Utah, US, June 2018.

[6] A. Dosovitskiy, P. Fischer, , E. Ilg, P. Hausser, C. Hazirbas, V. Golkov, P. der Smagt, D. Cremers, and T. Brox, "FlowNet: Learning Optical Flow with Convolutional Networks," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Santiago, Chile, Dec 2015.

[7] E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, and T. Brox, "Flownet 2.0: Evolution of optical flow estimation with deep networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Hawaii, US, July 2017.

[8] C. Bailer, B. Taetz, and D. Stricker, "Flow fields: Dense correspondence fields for highly accurate large displacement optical flow estimation," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Santiago, Chi, August 2015.

[9] R. Schuster, C. Bailer, O. Wasenm, and D. Stricker, "Flowfields++: Accurate optical flow correspondences meet robust interpolation," in *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, Athens, GR, October 2018.

[10] Z. Teed and J. Deng, "Raft: Recurrent all-pairs field transforms for optical flow," in *Proceedings of the European conference on computer vision European (ECCV)*, August 2020.

[11] X. Sui, S. Li, X. Geng, Y. Wu, X. Xu, Y. Liu, R. Goh, and H. Zhu, "CRAFT: Cross-Attentional Flow Transformer for Robust Optical Flow," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Louisiana, US, June 2022.

[12] H. Liu, T. Lu, Y. Xu, J. Liu, W. Li, and L. Chen, "CamLiFlow: Bidirectional Camera-LiDAR Fusion for Joint Optical Flow and Scene Flow Estimation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, New Orleans, US, June 2022.

[13] R. Rishav, R. Battrawy, R. Schuster, O. Wasenmuller, and D. Stricker, "DeepLiDARFlow: A Deep Learning Architecture For Scene Flow Estimation Using Monocular Camera and Sparse LiDAR," in *Proceedings of IEEE International Conference on Intelligent Robots and Systems (IROS)*, Las Vegas, US, June 2020.

[14] R. Murphy, J. Kravitz, S. Stover, and R. Shoureshi, "Mobile robots in mine rescue and recovery," *IEEE Robotics & Automation Magazine*, 2009.

[15] N. Mayer, E. Ilg, P. Hausser, and P. Fischer, "A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, US, June 2016.

[16] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The kitti dataset," *The International Journal of Robotics Research*, vol. 32, no. 11, pp. 1231–1237, 2013.

[17] D. Sun, X. Yang, M. Liu, and J. Kautz, "PWC-Net: CNNs for optical flow using pyramid, warping, and cost volume," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Utah, US, June 2018.

[18] S. Zhao, Y. Sheng, Y. Dong, E. Chang, and Y. Xu, "Maskflownet: Asymmetric feature matching with learnable occlusion mask," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[19] T. Hui and C. Change, "LiteFlowNet3: Resolving Correspondence Ambiguity for More Accurate Optical Flow Estimation," in *Proceedings of the European conference on computer vision European (ECCV)*, August 2020.

[20] S. Jiang, D. Campbell, Y. Lu, H. Li, and R. Hartley, "Learning to estimate hidden motions with global motion aggregation," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2021.

[21] H. Liu, T. Lu, Y. Xu, J. Liu, and L. Wang, "Learning optical flow and scene flow with bidirectional camera-lidar fusion," *arXiv preprint arXiv:2303.12017*, 2023.

[22] H. Reza, A. Shaban, M. L. Iuzzolino, and K. Koishida, "MMTM: Multimodal Transfer Module for CNN Fusion," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[23] D. Ulyanov, A. Vedaldi, and V. Lempitsky, "Instance normalization: The missing ingredient for fast stylization," *arXiv preprint arXiv:1607.08022*, 2016.

[24] B. Xu, N. Wang, T. Chen, and M. Li, "Empirical evaluation of rectified activations in convolutional network," *arXiv*, 2015.

[25] F. Zhang, V. Prisacariu, R. Yang, and P. Torr, "GA-Net: Guided Aggregation Net for End-To-End Stereo Matching," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, US, June 2019.

[26] Z. Wang, S. Li, H. Howard-Jenkins, V. Prisacariu, and M. Chen, "Flownet3d++: Geometric losses for deep scene flow estimation," in *Proceedings of IEEE Winter Conference on Applications of Computer Vision (WACV)*, Waikoloa, US, March 2020.

[27] F. Zhang, O. J-Woodford, V. Prisacariu, and P. Torr, "Separable flow: Learning motion cost volumes for optical flow estimation," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, March 2021.