

ROD: RGB-Only Fast and Efficient Off-road Freespace Detection

Tong Sun^{1,2}, Hongliang Ye³, Jilin Mei^{1,†}, Liang Chen¹, Fangzhou Zhao¹, Leiqiang Zong⁴, Yu Hu^{1,5,†}

Abstract—Off-road freespace detection is more challenging than on-road scenarios because of the blurred boundaries of traversable areas. Previous state-of-the-art (SOTA) methods employ multi-modal fusion of RGB images and LiDAR data. However, due to the significant increase in inference time when calculating surface normal maps from LiDAR data, multi-modal methods are not suitable for real-time applications, particularly in real-world scenarios where higher FPS is required compared to slow navigation. This paper presents a novel RGB-only approach for off-road freespace detection, named ROD, eliminating the reliance on LiDAR data and its computational demands. Specifically, we utilize a pre-trained Vision Transformer (ViT) to extract rich features from RGB images. Additionally, we design a lightweight yet efficient decoder, which together improve both precision and inference speed. ROD establishes a new SOTA on ORFD and RELLIS-3D datasets, as well as an inference speed of 50 FPS, significantly outperforming prior models. Our code will be available at https://github.com/STLIFE97/offroad_roadseg.

I. INTRODUCTION

In recent years, autonomous driving is attracted growing attention and research. Freespace detection, a binary classification segmentation task, plays a fundamental role in autonomous driving systems by delineating navigable areas crucial for vehicle planning and control. The majority of research are mainly focused on urban on-road scenarios, characterized by well-defined features such as lanes and traffic signs [1][4]. In contrast, research on off-road scenarios receives less attention [5]. These off-road scenarios present a higher degree of complexity and diversity, with the freespace being less distinct [2]. Vehicles are required to traverse a variety of terrains, including grasslands, sandy areas, icy grounds, snowy regions, and muddy terrains. As illustrated in Fig. 1(a), the boundaries of traversable areas in off-road scenarios are blurred.

To cope with the above difficulties, previous SOTA methods predominantly rely on the multi-modal fusion of RGB imagery with LiDAR data [3][6]. While RGB images focus more on surface color, texture, and other visual information, LiDAR data focuses more on distance, depth, and position [7]. Multi-modal fusion can help these two modalities complement each

[†]This work was supported by National Natural Science Foundation of China under Grant No.U23B2034, No.62203424, and No.62176250; and in part by the Innovation Program of Institute of Computing Technology, Chinese Academy of Sciences under Grant No. 2024000112.

¹Research Center for Intelligent Computing Systems, Institute of Computing Technology, Chinese Academy of Sciences, Beijing, 100190, China.

²University of Chinese Academy of Sciences, Beijing, 100190, China.

³Astronomical Computing Research Center, Zhejiang Lab

⁴Beijing Special Vehicle Academy, Beijing, 100072, China.

⁵School of Computer Science and Technology, University of Chinese Academy of Sciences, Beijing, 100190, China.

[†]Correspondence: Jilin Mei, Yu Hu, {meijilin, huyu}@ict.ac.cn

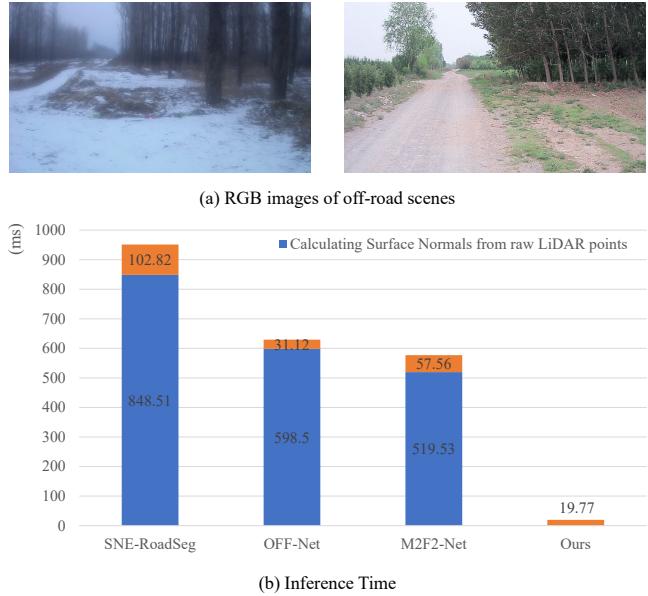


Fig. 1. (a) provides examples of RGB images in off-road scenes, where the boundaries of traversable areas are not clearly defined. (b) illustrates the inference time for SNE-RoadSeg [1], OFF-Net [2], the prior SOTA method M2F2-Net[3] and ours. Multi-modal fusion based methods do not meet real-time requirements due to the computational steps involved in generating surface normal maps.

other to achieve better performance, and thus multi-modal methods typically achieve higher accuracy [8]. Nonetheless, most methods for fusing LiDAR data require the calculation of surface normal maps [3], which is a time-consuming process. This huge computational overhead significantly affects the inference speed, making the algorithm less viable for real-time applications on vehicles. For high-speed autonomous systems such as off-road vehicles or drones operating in dynamic environments, achieving high frame rates is critical to ensure timely decision-making and collision avoidance. As shown in Fig. 1(b), the inference time for the M2F2-Net [3] is predominantly consumed by the generation of surface normal maps from LiDAR data, accounting for 90.02% of the total inference time. Such latency severely limits the practical deployment of LiDAR-based methods in scenarios requiring rapid response. Beyond computational constraints, LiDAR sensors also impose significant hardware costs and energy consumption.

To reduce reliance on surface normal map, this study proposes a novel approach that utilizes a pre-trained ViT model to extract features only from RGB images, thereby significantly enhancing the inference speed. With the development of ViT [9], the performance of large vision models [10][11]

improves as more data becomes available for pre-training. As shown in Fig. 2, the ViT encoder, which is employed for feature extraction, is kept frozen. Only the simple yet powerful decoder is trained. The RGB images are fed into the ViT encoder to generate an image embedding and to extract features from the latent layers of the transformer encoder blocks. These features are then used for subsequent fusion, and finally fed into the decoder for feature integration and prediction, culminating in the generation of the final prediction mask.

Our contribution can be summarized as follows:

- We conduct an investigation into the factors affecting inference speed in freespace detection models and propose a novel method, ROD. ROD only utilizes RGB data. This method surpasses prior multi-modal fusion methods in terms of both accuracy and inference speed.
- ROD integrates a pre-trained ViT model into the off-road freespace detection task, and design a powerful decoder that effectively merges image embeddings with latent features.
- The proposed ROD achieves performance on the ORFD and RELLIS-3D Dataset, with 98.3% F1-score and 96.7% IoU on the ORFD dataset, and 97.1% F1-score and 97.8% Accuracy on the RELLIS-3D dataset. Additionally, ROD achieves an inference speed of 50 FPS, comfortably meeting real-time requirements.

The remainder of this paper is organized as follows: Section 2 reviews related work, Section 3 details our model, Section 4 presents experiments, and Section 5 concludes the study.

II. RELATED WORKS

A. Freespace Detection

Freespace detection involves assigning a label to each pixel in traversable regions, which is categorized under semantic segmentation. The majority of research in this area focuses on on-road scenarios [1][4]. With the development of deep learning, freespace detection has expanded to include both single-modal and multi-modal approaches.

Single-modal methods utilize either RGB images or LiDAR data. CNNs are widely used for RGB data, as studies [12][13] demonstrate their effectiveness in freespace detection. [12] develops a driving scene generator to augment training data, while [13] introduces the RPP(Residual network with Pyramid Pooling) model, combining full convolutional networks with residual and pyramid pooling. Within LiDAR-based methods, the most approaches employ occupancy grids and scene flow to delineate obstacles. MotionNet [14] adopts a voxel-based representation, and PointMotionNet [15] opts for a point-based representation. Nevertheless, single-modal models that rely on either LiDAR or RGB images are limited by the data type they process. This limitation reduces the breadth of information available to the model, therefore, leading to lower accuracy compared to multi-modal systems [16].

The multi-modal methods for freespace detection fuse RGB images with LiDAR data. PETRv2 [17] and CVT [7] utilize multi-modal camera fusion to create Bird's Eye View

(BEV) images for road segmentation. BiFNet [18] introduces a bi-directional fusion network for integrating point cloud images with BEVs. RoadSeg [19] proposes a method that leverages LiDAR data for road segmentation. SNE-RoadSeg [1] enhances segmentation performance by fusing surface normal maps with RGB image features. Its successor, SNE-RoadSeg+ [4], further refines the accuracy of surface normal estimation and improves the network's performance and speed. Building on these advancements, M2F2-Net [3] and RoadFormer [6] achieve the SOTA performance in freespace detection.

However, the computation of surface normal maps from LiDAR data in multi-modal methods is time-consuming, resulting in an inability to meet real-time requirements. Therefore, this paper aims to find a method that utilizes only RGB images to improve inference speed and preserve precision.

B. Applications of Vision Transformer

The Vision Transformer (ViT) [9] adapts the transformer, usually for NLP, to image classification. Its streamlined design allows scalability and, with sufficient pre-training data, can surpass CNNs in transferability across tasks [9]. This paves the way for the development of large-scale models. These models are successfully applied to an array of visual tasks, further substantiating ViT's prowess in feature extraction. For instance, ViT demonstrates exceptional performance in segmentation applications [20][21][22], image inpainting [23], image editing [24], object detection [10], image captioning [25], object tracking [26][27], and 3D object reconstruction [28], among others.

Therefore, this paper attempts to integrate pre-train ViT model with freespace detection. However, a challenge with ViT is their large model size, which presents significant challenges for real-time applications, recent developments offer solutions. Lightweight ViT models such as MobileSAM [29], FastSAM [30] are proposed, they increase the inference speed but decrease the accuracy. Subsequently EfficientSAM [11] introduces a method called SAM-leveraged masked image pretraining (SAMI), which results in a lightweight ViT encoder. With less than 5% parameter of the original SAM, the performance drop is minimal, making it feasible for application in autonomous driving.

Thus, this study adopts the pre-trained ViT encoder from EfficientSAM for freespace detection. Leveraging the ViT encoder's prowess in feature extraction, we are able to achieve high accuracy using only RGB images.

C. Off-road Dataset

Compared to a large number of on-road datasets, off-road datasets supporting freespace detection are few. The RUGD dataset [5] is designed for semantic understanding in off-road scenarios, including mountain trails, streams, parks, villages and so on. The RELLIS-3D [31] dataset is derived from RUGD and includes unique terrain like puddles. In addition, RELLIS-3D includes 3D LiDAR annotations. [2] proposed the ORFD dataset, which is an off-road freespace

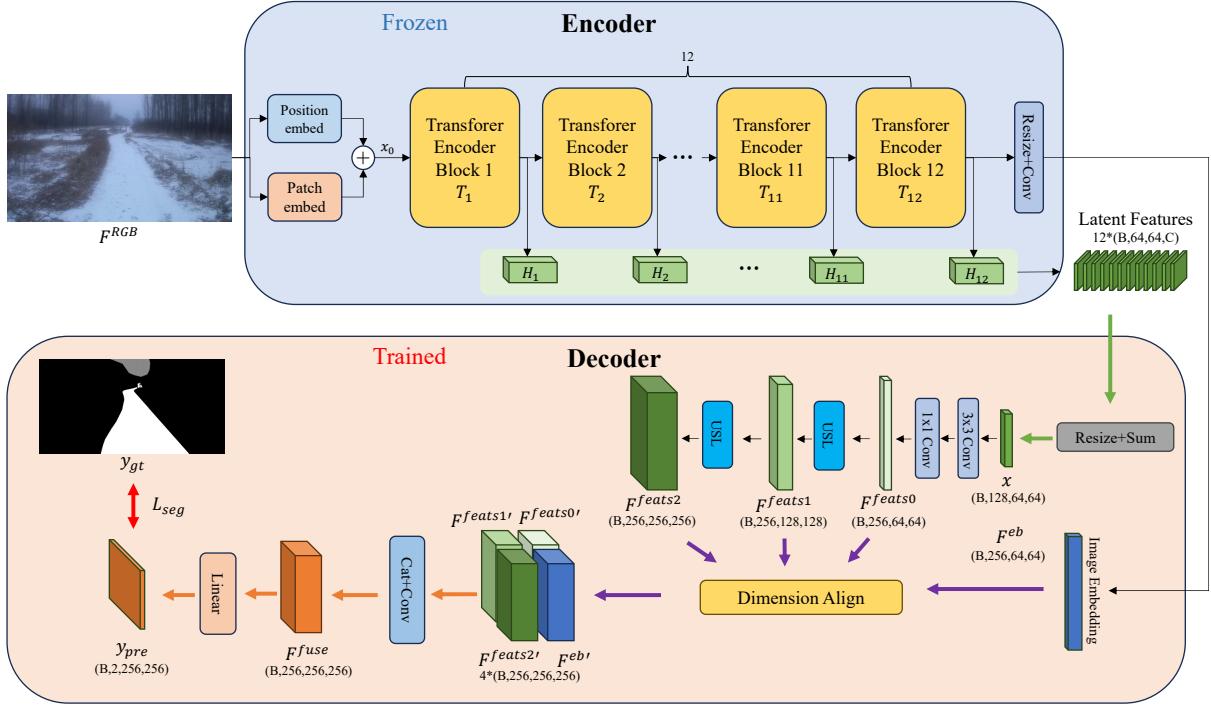


Fig. 2. The detailed architecture of the network proposed in this paper. (a) the frozen encoder part takes RGB data as input and processes it through 12 layers of transformer encoder block T_i to obtain the image embedding F^{eb} and latent features H_i . The number of channels C in H_i depends on the pre-trained model used, specifically, C=1280 for ViT-H, C=1024 for ViT-L, C=768 for ViT-B, C=384 for ViT-S and C=192 for ViT-T. (b) a simple but powerful decoder is designed to fuse the H_i features and obtain features at different scales, then it outputs the prediction mask y_{pre} .

detection dataset. The dataset was collected under different scenarios (woodland, farmland, grassland, and countryside), different weather conditions (sunny, rainy, foggy and snowy), and different lighting conditions (bright light, daylight, dusk, and darkness). The proposed method will be tested on the RELLIS-3D [31] and ORFD [2] datasets.

III. METHODOLOGY

A. Problem Definition

Freespace detection is a pixel-level classification problem, also known as semantic segmentation. The network f classifies each pixel or point to determine if it is passable. Given one RGB image frame F^{RGB} and its ground-truth mask y_{gt} with pixel-level annotations, the goal is to optimize the model parameters of f using the cross-entropy loss L_{seg} :

$$L_{seg} = - \sum_{H,W} y_{gt} \log(f(F^{RGB})) \quad (1)$$

where H and W represent the dimensions of the RGB image.

B. Overall Network Structure

The network architecture, as detailed in Fig. 2, is composed of two parts: the frozen encoder, which utilizes a pre-trained ViT-S model [11] to extract features from RGB images; and the trained decoder, which fuses these features to produce the prediction mask and updates its parameters using cross-entropy loss.

The RGB image F^{RGB} is utilized by the patch embed and position embed modules to generate positional and image

patch encodings. These encodings are summed and passed through a 12-layer transformer encoder block T_i , obtaining 12 features from the latent layers H_i and an image embedding F^{eb} . In the subsequent decode, the 12 feature maps H_i are merged to obtain the x . The channel dimension of x is expanded to create F^{feats0} , which is upsampled to produce F^{feats1} . F^{feats1} is then upsampled again to generate F^{feats2} . The features F^{feats0} , F^{feats1} , F^{feats2} , and F^{eb} are aligned and fused based on the dimensions of F^{feats2} to obtain F^{fuse} . Lastly, F^{fuse} is input to a linear layer to obtain the final prediction mask y_{pre} .

C. Architecture of the Encoder

The ViT encoder is utilized to extract high-performance features from only RGB images. A lightweight encoder, pre-trained by EfficientSAM [11], is employed. The encoder's details are shown in Fig. 2. The encoder processes the input F^{RGB} , and subsequently, it generates image embeddings F^{eb} and a set of latent features H_i .

Firstly, the RGB image F^{RGB} is processed by the Patch Embed module (*patch*), which segments it into a series of 16x16 pixel patches as embedded representations; meanwhile, F^{RGB} is fed to the Position Embed module (*pos*) to obtain positional embeddings and ensure that the positional embeddings match the dimensions of the input image. The resulting outputs are combined to form the initial feature vector x_0 , as detailed by the following equation:

$$x_0 = pos(F^{RGB}) + patch(F^{RGB}) \quad (2)$$

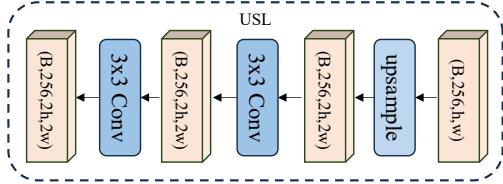


Fig. 3. The specific structure of the upsample layers (USL), where h and w represent the dimensions of the feature map.

The ViT encoder sets the depth of transformer encoder blocks T_i to 12 layers ($i = 12$). x_0 is fed into the first block T_1 . Subsequently, each block T_i receives the output from the preceding block as its input, with the formula as follows:

$$H_i = \begin{cases} T_i(x_0) & \text{if } i = 1 \\ T_i(H_{i-1}) & \text{if } 1 < i \leq 12 \end{cases} \quad (3)$$

The ViT encoder processes data through 12 transformer blocks, resulting in 12 feature maps H_i . The last feature map H_{12} input into *Resize + Conv* module, undergoes a dimension permute, and is passed through two convolutional layers (1×1 and 3×3 kernels) to produce the final image embedding F^{eb} , sized (B, 256, 64, 64).

D. Architecture of the Decoder

To balance precision and inference speed, we design a simple yet powerful seg decoder to predict segmentation masks for freespace detection. The architecture of the decoder is detailed in Fig. 2.

Firstly, we permute the dimensions of the 12 latent features H_i and downsampling them to (B, 128, 64, 64). Then summing these features to $\sum_{i=1}^{12} H_i$, and inputting the $\sum_{i=1}^{12} H_i$ through two consecutive convolutional layers, each with a 3×3 kernel, then the outputs is merged back with $\sum_{i=1}^{12} H_i$, creating a residual connection, and obtaining the summed features x . The formula is as follows:

$$x = \text{Conv}(\text{Conv}(\sum_{i=1}^{12} H_i)) + \sum_{i=1}^{12} H_i \quad (4)$$

Fig. 3 illustrates the specific structures of the upsample layers (USL). The strategy consists of a sequence of two convolutional layers followed by a residual connection to retain finer details during the feature map's expansion. x is passed through a 1×1 convolution, which doubles the channel count from 128 to 256, obtaining the feature F^{feats0} . The features F^{feats0} are input into the upsample layers (USL) for bilinear upsample followed by two convolutional layers with a kernel size of 3×3 , expanding the feature dimensions to 128×128 from 64×64 . The output is merged with the upsampled F^{feats0} to produce F^{feats1} . This process is replicated on F^{feats1} , scaling the features from 128×128 to 256×256 , to yield the feature F^{feats2} . The formula are as follows:

$$F^{feats0} = \text{Conv}(x) \quad (5)$$

$$F^{feats1} = \text{USL}(F^{feats0}) + \text{upsample}(F^{feats0}) \quad (6)$$

$$F^{feats2} = \text{USL}(F^{feats1}) + \text{upsample}(F^{feats1}) \quad (7)$$

Subsequently, the image embedding F^{eb} and the feature maps F^{feats0} , F^{feats1} , and F^{feats2} are upsampled to match dimensions of (B, 256, 256, 256) through bilinear interpolation, resulting in $F^{eb'}$, $F^{feats0'}$, $F^{feats1'}$, and $F^{feats2'}$. The formula is as follows:

$$\begin{aligned} F^{eb'}, F^{feats0'}, F^{feats1'}, F^{feats2'} \\ = \text{interpolate}(F^{eb}, F^{feats0}, F^{feats1}, F^{feats2}) \end{aligned} \quad (8)$$

Ultimately, the feature maps $F^{eb'}$, $F^{feats0'}$, $F^{feats1'}$, and $F^{feats2'}$ are concatenated to the dimensions (B, 4x256, 256, 256). This output is then processed by a 1×1 convolutional layer to downscale the channel number from 4x256 to 256, resulting F^{fuse} . Subsequently, a linear layer is applied to F^{fuse} to produce the final predicted mask y_{pre} .

IV. EXPERIMENT

A. Experimental Settings

Our method is evaluated on two datasets: the ORFD dataset [2], which provides 12,198 pairs of RGB images and LiDAR point clouds; and the RELLIS-3D dataset [31], a large-scale dataset focusing on freespace detection in off-road scenarios. The ORFD dataset contains 8,398 training samples, 1,245 validation samples, and 2,555 test samples, with image resolutions of 1280×720 . In contrast, RELLIS-3D is a multi-modal dataset collected in off-road scenarios, comprising 13,556 LiDAR scans and 6,235 images. It includes 3,302 training samples, 983 validation samples, and 1,672 test samples, with images at a resolution of 1920×1200 .

For testing on the ORFD dataset [2], this paper uses the evaluation metrics from the M2F2-Net [3], including IoU, Precision, Recall, Accuracy, F1_score, and FPS.

For testing on the RELLIS-3D dataset [31], this paper refers to the evaluation metrics from self-supervisions-only [39], utilizing Precision (Pre), Recall (Rec), Accuracy (Acc), and F1_score as the assessment criteria.

This study employs the AdamW optimizer with its default parameters, initiating the learning rate at 1e-3. We adopt 'poly' decay policy for the learning rate, utilizing a decay power of 0.9. Batch size is set to 8. All experiments are conducted on a single NVIDIA RTX A6000 GPU.

B. Comparisons on ORFD Dataset

Comparisons on the ORFD dataset are presented in TABLE I. We compare our approach with various methods, including U-Net [32], DeepLabV3Plus [33] (DLV3+-MNet and DLV3+-R101), FuseNet [34], MFNet [35], RTFNet [36], SNE-RoadSeg [1], OFF-Net [2], M2F2-Net [3], and RoadFormer [6]. Compared to methods that only use RGB [32][33], surpasses DLV3+-R101 with a 15.9% improvement in F1_score and a 26.7% improvement in IoU. In methods

TABLE I
COMPARISON TO OTHER METHODS ON THE TESTING SET OF ORFD

Method	Modality	Year	Accuracy	Precision	Recall	F1_score	IoU	Inference Time (ms)		FPS(total)
								CSN*	Model Inference	
U-Net [32]	RGB	2015	0.959	0.637	0.537	0.583	0.411	0	225.23	4.44
DLV3+-MNet [33]	RGB	2018	0.976	0.743	0.847	0.792	0.655	0	14.77	67.70
DLV3+-R101 [33]	RGB	2018	0.980	0.781	0.871	0.824	0.700	0	25.81	38.73
FuseNet [34]	RGB+LiDAR	2017	0.874	0.745	0.852	0.795	0.660	-	-	-
MFNet [35]	RGB+LiDAR	2017	-	0.896	0.903	0.899	0.817	-	-	-
RTFNet [36]	RGB+LiDAR	2019	-	0.842	0.967	0.900	0.818	-	-	-
SNE-RoadSeg [1]	RGB+LiDAR	2020	0.938	0.867	0.927	0.896	0.812	848.51	102.82	1.05
OFF-Net [2]	RGB+LiDAR	2022	0.945	0.866	0.943	0.903	0.823	598.50	31.12	1.58
M2F2-Net [3]	RGB+LiDAR	2023	0.981	0.973	0.955	0.964	0.931	519.53	57.56	1.73
RoadFormer [6]	RGB+LiDAR	2023	0.979	0.951	0.972	0.961	0.925	-	-	-
Ours	RGB	2024	0.991	0.984	0.983	0.983	0.967	0	19.77	50.56

* Calculating Surface Normal(CSN) from raw LiDAR points.

– The necessary code related to this is not provided in related paper.

TABLE II
COMPARISON TO OTHER METHODS ON THE TESTING SET OF RELLIS-3D

Method	Modality	Year	Accuracy	Precision	Recall	F1_score
U-Net [32]	RGB	2015	0.977	0.809	0.856	0.832
DLV3+-MNet [33]	RGB	2018	0.966	0.570	0.856	0.832
DLV3+-R101 [33]	RGB	2018	0.966	0.586	0.428	0.495
Real-NVP [37]	RGB+LiDAR	2020	0.5625	0.5710	0.9742	0.7001
AE Based [38]	RGB+LiDAR	2022	0.7348	0.7079	0.9181	0.7437
Self-Supervisions Only [39]	RGB+LiDAR	2023	0.9036	0.9164	0.8508	0.8622
M2F2-Net [3]	RGB+LiDAR	2023	0.955	0.925	0.964	0.944
Ours	RGB	2024	0.9786	0.9702	0.9721	0.9712

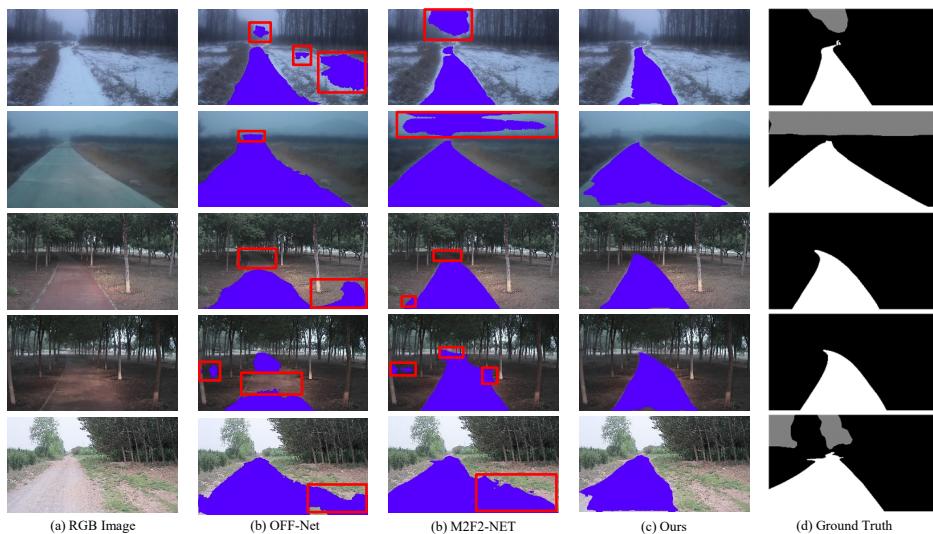


Fig. 4. Qualitative results of OFF-Net [2], M2F2-Net [3] and our method on ORFD dataset [2]. The red boxes are the area where OFF-Net and M2F2-Net predicts incorrectly but ours predicts correctly.

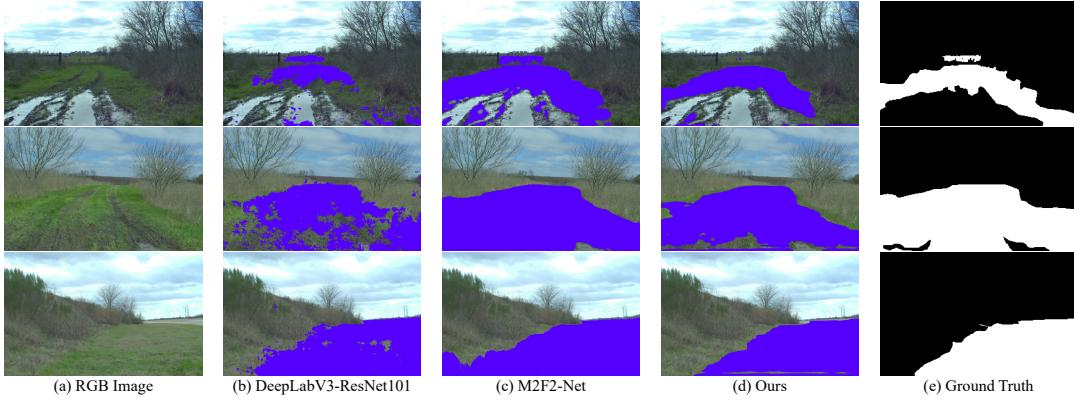


Fig. 5. Qualitative results of DLV3+-R101 [33], M2F2-Net [3] and our method on RELLIS-3D dataset [2].

TABLE III
ABLATION STUDY ON ViT ENCODER SELECTION

ViT Encoder	Params (M)	IoU	F1_score	FPS
ViT-H	632	0.937	0.968	2.54
ViT-L	308	0.933	0.965	4.58
ViT-B	91	0.931	0.964	13.20
ViT-T	9.8	0.927	0.962	46.34
ViT-S	25	0.967	0.983	50.56

¹ ViT-H is the pre-trained model sam_vit_h_4b8939 in SAM [10];

² ViT-L is the pre-trained model sam_vit_l_0b3195 in SAM [10];

³ ViT-B is the pre-trained model sam_vit_b_01ec64 in SAM [10];

⁴ ViT-T is the pre-trained model efficient.sam_vitt in EfficientSAM [11];

⁵ ViT-S is the pre-trained model efficient.sam_vits in EfficientSAM [11].

that fuse RGB with LiDAR, M2F2-Net and RoadFormer are the previous SOTA methods for off-road freespace detection. Our method, which utilizes only the RGB images, surpasses M2F2-Net with a 0.6% improvement in F1_score and a 1.1% improvement in IoU. In terms of inference speed, our method, by avoiding the calculation of surface normal maps, is 25 times faster than M2F2-Net.

As shown in Fig. 4, we visualize the prediction results of our method and compare them with those of OFF-Net [2] and M2F2-Net [3] on the ORFD dataset. By comparing the predictions with Ground Truth, it becomes clear that our method outperforms OFF-Net and M2F2-Net. While all models excel in conditions with ample sunlight and clear weather. However, OFF-Net and M2F2-Net face several challenges in low light and adverse weather, including misidentifying the sky as ground, overlooking shaded passable roads, and incorrectly classifying roadside grass as part of the road.

C. Comparisons on RELLIS-3D Dataset

Comparisons on the RELLIS-3D dataset are detailed in Table II. We compare our approach with various methods,

including U-Net [32], DeepLabV3Plus [33] (DLV3+-MNet and DLV3+-R101), Real-NVP [37], AE Based [38], Self-Supervisions Only [39] and M2F2-Net [3]. The results indicate that our method outperforms these methods, achieving an 2.2% improvement in Accuracy and a 2.7% improvement in the F1_score.

As shown in Fig. 5, we visualize the prediction results of our method and compare them with those of DeepLabV3Plus-ResNet101 [33] (DLV3+-R101) and M2F2-Net [3] on the RELLIS-3D dataset. By comparing the predictions with Ground Truth, it becomes clear that our method outperforms DLV3+-R101 and M2F2-Net.

D. Ablation Studies

To determine the most suitable Vision Transformer (ViT) as the encoder for our feature extraction module, we evaluate ViT-H, ViT-L and ViT-B from SAM [10], as well as ViT-T and ViT-S from EfficientSAM [11], on the ORFD dataset [2]. Our evaluations, as presented in Table III, indicates that ViT-H's performance, with a speed of only 3 FPS, could not meet the criteria for real-time applications. According to [11], instance segmentation models using ViT-T are 14% faster than those based on ViT-S. However, this paper only utilizes the encoder parts of ViT-T and ViT-S, both of which have 12 blocks and output 12 hidden features for subsequent feature fusion, thus their inference speeds are roughly comparable. Consequently, we choose ViT-S as the encoder for subsequent experimental comparisons.

V. CONCLUSIONS

This paper introduces a fast and efficient method for off-road freespace detection. By employing a pre-trained ViT as the encoder, our approach effectively extracts features from RGB data alone, which enhances accuracy and significantly increases the model's inference speed by eliminating the need for surface normal map calculations. Additionally, we design a seg decoder that merges the encoder's latent features with image embeddings, ensuring greater detail retention and higher accuracy. Our method achieves SOTA performance on both the ORFD and RELLIS-3D datasets. Moreover, it performs inference at 50 FPS, significantly outperforming previous methods in terms of speed.

REFERENCES

- [1] R. Fan, H. Wang, P. Cai, and M. Liu, "Sne-roadseg: Incorporating surface normal information into semantic segmentation for accurate freespace detection," in *European Conference on Computer Vision*. Springer, 2020, pp. 340–356.
- [2] C. Min, W. Jiang, D. Zhao, J. Xu, L. Xiao, Y. Nie, and B. Dai, "Orfd: A dataset and benchmark for off-road freespace detection," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 2532–2538.
- [3] H. Ye, J. Mei, and Y. Hu, "M2f2-net: Multi-modal feature fusion for unstructured off-road freespace detection," in *2023 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2023, pp. 1–7.
- [4] H. Wang, R. Fan, P. Cai, and M. Liu, "Sne-roadseg+: Rethinking depth-normal translation and deep supervision for freespace detection," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021, pp. 1140–1145.
- [5] M. Wigness, S. Eum, J. G. Rogers, D. Han, and H. Kwon, "A rugged dataset for autonomous navigation and visual perception in unstructured outdoor environments," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 5000–5007.
- [6] J. Li, Y. Zhang, P. Yun, G. Zhou, Q. Chen, and R. Fan, "Roadformer: Duplex transformer for rgb-normal semantic road scene parsing," *arXiv preprint arXiv:2309.10356*, 2023.
- [7] Y. Wang, L. Zhu, S. Huang, T. Hui, X. Li, F. Wang, and S. Liu, "Cross-modality domain adaptation for freespace detection: A simple yet effective baseline," in *Proceedings of the 30th ACM International Conference on Multimedia*, 2022, pp. 4031–4042.
- [8] K. Huang, B. Shi, X. Li, X. Li, S. Huang, and Y. Li, "Multi-modal sensor fusion for auto driving perception: A survey," *arXiv preprint arXiv:2202.02703*, 2022.
- [9] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.
- [10] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo, P. Dollár, and R. Girshick, "Segment anything," *arXiv:2304.02643*, 2023.
- [11] Y. Xiong, B. Varadarajan, L. Wu, X. Xiang, F. Xiao, C. Zhu, X. Dai, D. Wang, F. Sun, F. Iandola *et al.*, "Efficientsam: Leveraged masked image pretraining for efficient segment anything," *arXiv preprint arXiv:2312.00863*, 2023.
- [12] R. Fan, H. Wang, P. Cai, J. Wu, M. J. Bucus, L. Qiao, and M. Liu, "Learning collision-free space detection from stereo images: Homography matrix brings better data augmentation," *IEEE/ASME Transactions on Mechatronics*, vol. 27, no. 1, pp. 225–233, 2021.
- [13] X. Liu and Z. Deng, "Segmentation of drivable road using deep fully convolutional residual network with pyramid pooling," *Cognitive Computation*, vol. 10, pp. 272–281, 2018.
- [14] P. Wu, S. Chen, and D. N. Metaxas, "Motionnet: Joint perception and motion prediction for autonomous driving based on bird's eye view maps," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 11 385–11 395.
- [15] J. Wang, X. Li, A. Sullivan, L. Abbott, and S. Chen, "Pointmotionnet: Point-wise motion learning for large-scale lidar point clouds sequences," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 4419–4428.
- [16] Y. Huang, C. Du, Z. Xue, X. Chen, H. Zhao, and L. Huang, "What makes multi-modal learning better than single (provably)," *Advances in Neural Information Processing Systems*, vol. 34, pp. 10 944–10 956, 2021.
- [17] Y. Liu, J. Yan, F. Jia, S. Li, A. Gao, T. Wang, and X. Zhang, "PetrV2: A unified framework for 3d perception from multi-camera images," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 3262–3272.
- [18] H. Li, Y. Chen, Q. Zhang, and D. Zhao, "Bifnet: Bidirectional fusion network for road segmentation," *IEEE transactions on cybernetics*, vol. 52, no. 9, pp. 8617–8628, 2021.
- [19] E. Capellier, F. Davoine, V. Cherfaoui, and Y. Li, "Fusion of neural networks, for lidar-based evidential road mapping," *Journal of Field Robotics*, vol. 38, no. 5, pp. 727–758, 2021.
- [20] J. Cen, Y. Wu, K. Wang, X. Li, J. Yang, Y. Pei, L. Kong, Z. Liu, and Q. Chen, "Sad: Segment any rgbd," *arXiv preprint arXiv:2305.14207*, 2023.
- [21] T. Chen, L. Zhu, C. Deng, R. Cao, Y. Wang, S. Zhang, Z. Li, L. Sun, Y. Zang, and P. Mao, "Sam-adapter: Adapting segment anything in underperformed scenes," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 3367–3375.
- [22] R. Deng, C. Cui, Q. Liu, T. Yao, L. W. Remedios, S. Bao, B. A. Landman, L. E. Wheless, L. A. Coburn, K. T. Wilson *et al.*, "Segment anything model (sam) for digital pathology: Assess zero-shot segmentation on whole slide imaging," *arXiv preprint arXiv:2304.04155*, 2023.
- [23] T. Yu, R. Feng, R. Feng, J. Liu, X. Jin, W. Zeng, and Z. Chen, "Inpaint anything: Segment anything meets image inpainting," *arXiv preprint arXiv:2304.06790*, 2023.
- [24] S. Gao, Z. Lin, X. Xie, P. Zhou, M.-M. Cheng, and S. Yan, "Editanything: Empowering unparalleled flexibility in image editing and generation," in *Proceedings of the 31st ACM International Conference on Multimedia*, 2023, pp. 9414–9416.
- [25] T. Wang, J. Zhang, J. Fei, Y. Ge, H. Zheng, Y. Tang, Z. Li, M. Gao, S. Zhao, Y. Shan, and F. Zheng, "Caption anything: Interactive image description with diverse multimodal controls," *arXiv preprint arXiv:2305.02677*, 2023.
- [26] Y. Cheng, L. Li, Y. Xu, X. Li, Z. Yang, W. Wang, and Y. Yang, "Segment and track anything," *arXiv preprint arXiv:2305.06558*, 2023.
- [27] J. Yang, M. Gao, Z. Li, S. Gao, F. Wang, and F. Zheng, "Track anything: Segment anything meets videos," *arXiv preprint arXiv:2304.11968*, 2023.
- [28] Q. Shen, X. Yang, and X. Wang, "Anything-3d: Towards single-view anything reconstruction in the wild," *arXiv preprint arXiv:2304.10261*, 2023.
- [29] C. Zhang, D. Han, Y. Qiao, J. U. Kim, S.-H. Bae, S. Lee, and C. S. Hong, "Faster segment anything: Towards lightweight sam for mobile applications," *arXiv preprint arXiv:2306.14289*, 2023.
- [30] X. Zhao, W. Ding, Y. An, Y. Du, T. Yu, M. Li, M. Tang, and J. Wang, "Fast segment anything," *arXiv preprint arXiv:2306.12156*, 2023.
- [31] P. Jiang, P. Osteen, M. Wigness, and S. Saripalli, "Rellis-3d dataset: Data, benchmarks and analysis," in *2021 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2021, pp. 1110–1116.
- [32] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical image computing and computer-assisted intervention—MICCAI 2015: 18th international conference, Munich, Germany, October 5–9, 2015, proceedings, part III 18*. Springer, 2015, pp. 234–241.
- [33] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 801–818.
- [34] C. Hazirbas, L. Ma, C. Domokos, and D. Cremers, "Fusenet: Incorporating depth into semantic segmentation via fusion-based cnn architecture," in *Computer Vision—ACCV 2016: 13th Asian Conference on Computer Vision, Taipei, Taiwan, November 20–24, 2016, Revised Selected Papers, Part I 13*. Springer, 2017, pp. 213–228.
- [35] Q. Ha, K. Watanabe, T. Karasawa, Y. Ushiku, and T. Harada, "Mfnet: Towards real-time semantic segmentation for autonomous vehicles with multi-spectral scenes," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2017, pp. 5108–5115.
- [36] Y. Sun, W. Zuo, and M. Liu, "Rtfnet: Rgb-thermal fusion network for semantic segmentation of urban scenes," *IEEE Robotics and Automation Letters*, vol. 4, no. 3, pp. 2576–2583, 2019.
- [37] L. Wellhausen, R. Ranftl, and M. Hutter, "Safe robot navigation via multi-modal anomaly detection," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 1326–1333, 2020.
- [38] R. Schmid, D. Atha, F. Schöller, S. Dey, S. Fakoorian, K. Otsu, B. Ridge, M. Bjelonic, L. Wellhausen, M. Hutter *et al.*, "Self-supervised traversability prediction by learning to reconstruct safe terrain," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 12 419–12 425.
- [39] J. Seo, S. Sim, and I. Shim, "Learning off-road terrain traversability with self-supervisions only," *IEEE Robotics and Automation Letters*, 2023.