CGI2018 manuscript No.
(will be inserted by the editor)

Multi-Feature Fusion Enhanced Monocular Depth Estimation with Boundary Awareness

Abstract Self-supervised monocular depth estimation has opened up exciting possibilities for practical applications, including scene understanding, object detection, and autonomous driving, without the need for expensive depth annotations. However, traditional methods for single-image depth estimation encounter limitations in photometric loss due to a lack of geometric constraints, reliance on pixel-level intensity or color differences, and the assumption of perfect photometric consistency, leading to errors in challenging conditions and resulting in overly smooth depth maps with insufficient capture of object boundaries and depth transitions. To tackle these challenges, we propose MFFENet, which leverages multi-level semantic and boundary-aware features to improve depth estimation accuracy. MFFENet extracts multi-level semantic features using our modified HRFormer approach. These features are then fed into our decoder and enhanced using attention mechanisms to enrich the boundary information generated by Laplacian pyramid residuals. To mitigate the weakening of semantic features during convolution processes, we introduce a feature-enhanced combination strategy. We also integrate the DeconvUp module to improve the restoration of depth map boundaries. We introduce a boundary loss that enforces constraints between object boundaries. We propose an extended evaluation method that utilizes Laplacian pyramid residuals to evaluate boundary depth. Extensive evaluations on the KITTI, Cityscape, and Make3D datasets demonstrate the superior performance of MFFENet compared to state-of-the-art models in monocular depth estimation.

Keywords Self-supervised monocular depth estimation · Laplacian pyramid residuals · boundary depth · multi-level semantic

1 Introduction

Depth estimation is a fundamental task in computer vision with applications in domains such as 3D reconstruction, autonomous driving, and virtual reality. Traditional methods rely on stereo matching algorithms [13] to generate a disparity map from paired images, but their complexity and time-consuming nature limit their applicability.

In recent years, the success of deep neural networks has led to the development of learning-based methods for monocular depth estimation. Monocular depth estimation from a single image is cost-effective and easy to implement, making it highly attractive. There are two main categories: supervised [7,18] and unsupervised learning. While supervised methods achieve impressive results, they require extensive ground truth depth data, which is challenging and expensive to obtain. In contrast, self-supervised methods, which can be trained using stereo image pairs [29,22] or sequential images from monocular videos [38,25,12], do not require labeled data.

Many self-supervised methods in depth estimation rely on Structure from Motion (SfM) to generate supervision signals, which are effective only for pixels adhering to static scene and self-motion assumptions. These methods heavily rely on photometric loss and smoothness constraints, which have limitations. These limitations stem from a lack of explicit geometric constraints and heavy reliance on pixel-level intensity or color differences. These methods also assume perfect photometric consistency between rendered images based on the estimated depth map and the ground truth images, leading to errors in the presence of occlusions, textureless regions, or challenging lighting conditions. Consequently, these issues result in depth maps that are excessively smooth at object boundaries and an inabil-

ity to accurately capture depth transitions between objects. Previous research has attempted to address these challenges by incorporating external semantic segmentation and optical flow into the monocular depth estimation process [16,17,40]. While these methods, which utilize external semantic segmentation networks, have shown promising results, they require semantic labels for training and can impose computational burdens.

Recent research suggests that changing the backbone of monocular depth estimation networks can improve accuracy. While ResNet is commonly used as the backbone in depth networks [9,10], alternative backbones like HRNet [37,12] and PackNet have been introduced. However, these studies mainly focus on employing more powerful CNN architectures, resulting in deeper and more complex models. It should be noted that CNNs face challenges in capturing global contextual information from images, leading to performance limitations for networks with CNN backbones. To address this limitation, researchers have explored the application of Vision Transformer (ViT) [6] in monocular depth estimation networks. ViT, known for capturing global contextual information and exhibiting good performance in tasks like semantic segmentation, has been utilized in monocular depth estimation networks in studies such as [34,1,36]. However, these networks have not specifically addressed the issue of depth blur at boundaries caused by photometric consistency loss.

In this paper, we propose MFFENet, a novel self-supervised monocular depth estimation framework that leverages multi-level semantic and boundary-aware features to enhance depth estimation accuracy. Inspired by DIFFNet [37], we modify HRFormer to extract multi-level semantic features. These features are then incorporated into our decoder and enhanced using attention mechanisms to enrich boundary information obtained from Laplacian pyramid residuals. To address the challenge of semantic feature weakening during convolution processes, we introduce a feature-enhanced combination strategy. Additionally, we propose the DeconvUp module, which together with feature-enhanced combination strategy facilitates the recovery of depth map boundaries. To enforce boundary constraints, we introduce a boundary loss.

We extensively evaluate the performance of MFFENet on multiple datasets. Specifically, we demonstrate its accuracy in depth estimation by comparing it to state-of-the-art methods on the KITTI dataset. We also showcase its excellent depth estimation performance on the Cityscape dataset, highlighting its broad applicability. Furthermore, we assess the generalization capability of MFFENet by comparing it with other monocular depth estimation methods on the Make3D dataset. Our results demonstrate the superior generalization ability of MFFENet, outperforming competing models in terms of depth estimation performance. We propose an extended method for evaluating boundary depth and demonstrate the effectiveness of our method on boundary depth. Our contributions include:

1. We propose MFFENet, a novel self-supervised monocular depth estimation framework that utilizes an improved HRFormer as the encoder and effectively leverages Laplacian pyramid residuals to address inaccurate object boundary depth predictions caused by photometric loss.
2. We introduce a feature-enhanced combination strategy and DeconvUp module. Both approaches respectively enhance the semantic information and boundary details of the features to generate accurate depth maps.
3. We propose a boundary loss function that combines boundary-relevant pixels extracted from the Laplacian pyramid residuals with the berHu loss, explicitly improving the model's ability to constrain object boundaries.
4. We propose an additional evaluation method that allows for a assessment of depth estimation at boundaries, providing a more comprehensive evaluation of MFFENet and other methods.

2 Related Work

2.1 Self-Supervised Monocular Depth Estimation

The field of monocular depth estimation has seen a surge in self-supervised approaches, driven by the limitations of labeled training data. Initially, methods like Monodepth [9] relied on stereo images for training. Inspired by the classic algorithm SfM, Zhou et al. [38] introduced a pioneering self-supervised framework. They incorporated depth estimation and pose estimation networks, utilizing monocular videos for training, thereby reducing the requirements and costs associated with self-supervised depth estimation.

To bridge the performance disparity between monocular self-supervised and stereo self-supervised methods, Godard et al. introduced Monodepth2 [10]. Monodepth2 incorporates multi-scale estimation, per-pixel minimum reprojection loss, and self-masked fixed pixels, surpassing numerous supervised methods and establishing itself as a widely adopted baseline framework. It has laid the groundwork for several subsequent self-supervised works [14,39,37].

Johnston et al. [14] proposed enhancing depth estimation further by incorporating self-attention mecha-

nisms and discrete disparity volume modules into self-supervised networks, aiming to improve robustness and sharpness. Additionally, PackNet [12] introduced 3D convolutions in encoding and decoding layers, enabling better processing of fine-grained details for improved depth estimation. Tosi et al. [29] explored self-distillation, utilizing the SGM algorithm [13] to generate accurate pseudo-labels as supervision signals, thereby enhancing prediction accuracy in monocular depth estimation.

Existing methods frequently overlook the limitations of the photometric consistency loss. In contrast, we propose a novel architecture capable of addressing inaccurate depth boundaries resulting from this loss.

2.2 Semantic Network and Boundary Awareness

For depth estimation, the integration of multiple tasks, such as semantic segmentation, has proven effective in addressing various challenges. Kendall et al. [16] demonstrated the benefits of multi-task learning in visual models compared to training separate models. By leveraging dense semantic information obtained from semantic segmentation, prior knowledge can be applied to select scenes consistent with known information, aiding monocular depth estimation. This has led to the emergence of methods that combine depth estimation and semantic segmentation networks.

Choi et al. [4] incorporated additional segmentation networks to enhance prediction accuracy. These methods utilize independent semantic segmentation models to identify pixels that violate scene assumptions and avoid compromising photometric loss, as demonstrated by Klingner et al. [17]. Another approach, such as Chen et al. [2], involves incorporating semantic segmentation cues into self-supervised depth estimation. This guides the network in learning semantic-rich features and sharing contextual information using a shared encoder. Zhu et al. [40] employed edge segmentation to explicitly align depth edges with semantic edges, while Jung et al. [15] proposed a semantic-guided triple loss to enhance depth maps that align with semantic boundaries. Chen et al. [3] further improved upon the work of Jung et al. [15] to enhance its effectiveness. However, these methods require semantic labels or additional networks for training, which restricts their practical applicability and diminishes the benefits of self-supervised learning.

Sun et al. [27] introduced an edge-aware loss that samples point pairs around image edges and combines them with pseudo-depth and relative normality losses to constrain object boundaries. However, they rely on an additional network to obtain pseudo-depth, which adds complexity to the system.

To address these issues, we propose a novel approach that builds upon the insights from previous work. We adopt HRFormer [33] as the encoder backbone, leveraging its inherent semantic information. This eliminates the need for semantic labels while effectively enhancing the encoder's capacity to extract crucial feature information from images. HRFormer combines the advantages of both CNN and Vision Transformer, achieving superior performance with fewer parameters. By incorporating HRFormer and addressing the limitations emphasized in [27], we aim to overcome the need for semantic labels or additional networks, making our approach more practical and benefiting from the advantages of self-supervised learning.

2.3 Transformer-Based Monocular Depth Estimation

Recent advancements in depth estimation have explored the utilization of Transformer-based architectures, showing promising outcomes. For instance, Zhao et al. [36] introduced MonoViT, which employed MPViT [19] as the encoder and achieved state-of-the-art (SOTA) accuracy. However, the parallel block design in MonoViT resulted in a large model parameter size. Bae et al. [1] proposed a hybrid architecture, but its combination with ViT led to high computational complexity. While these VIT-based models have demonstrated remarkable results, the large number of VIT parameters limits their training time and inference speed. Therefore, it is crucial to address this limitation by developing a VIT model with a reduced parameter size. Compared to other VIT models, HRFormer offers a compact parameter size. Our MFFENet-tiny encoder, based on HRFormer, has a mere 2.45M parameters.

In terms of reducing the network parameter size for VIT-based depth estimation, Lite-Mono [34] achieved a smaller parameter size (3.1M parameters), but its accuracy was unsatisfactory. However, in our experiments, we demonstrate that our MFFENet-tiny (4.5M parameters) not only outperforms Lite-Mono [34] in terms of accuracy but also surpasses its enhanced version, Lite-Mono-8m [34](8.7M parameters). This highlights the effectiveness of our proposed model in achieving a balance between model compactness and accuracy.

3 Method

In this section, we present our proposed MFFENet framework, which incorporates an improved HRFormer as the encoder, leverages the effectiveness of Laplacian pyramid residuals, and integrates our feature-enhanced

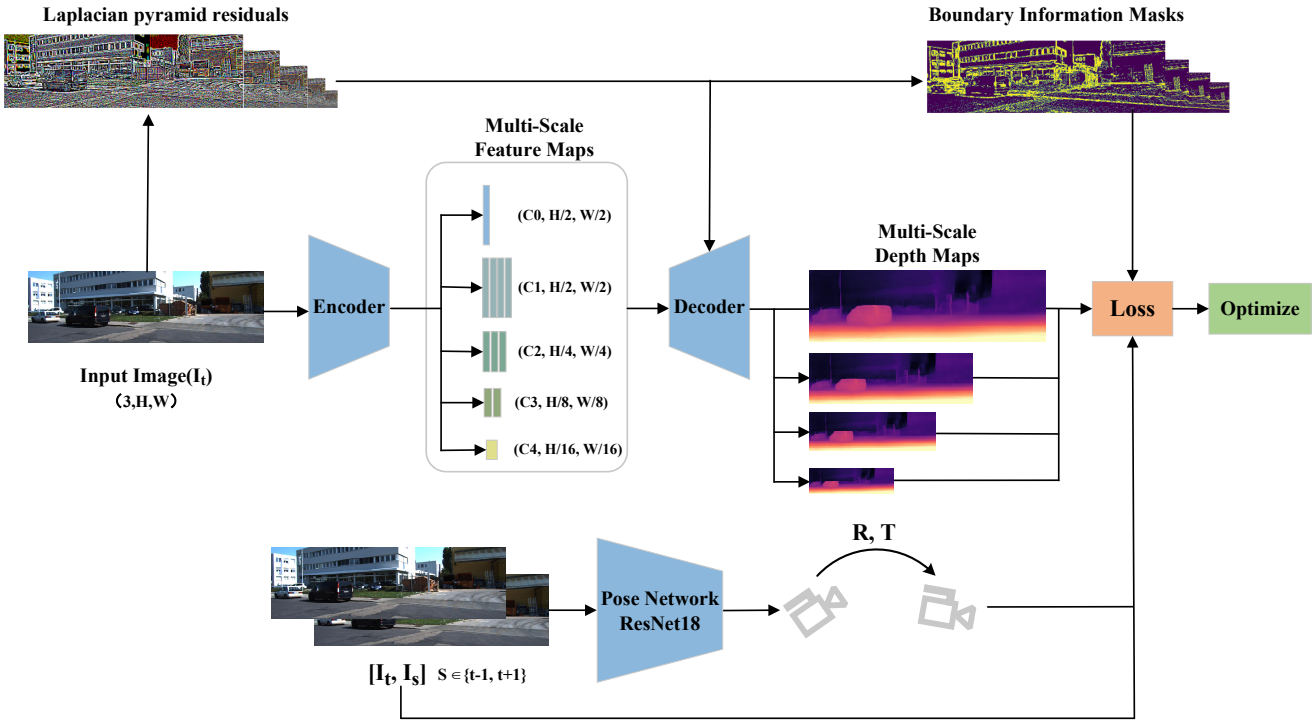

Fig. 1 The architecture of our MFFENet framework. The input image is processed by our encoder, generating multi-scale feature maps at $\frac{1}{2}$, $\frac{1}{4}$, $\frac{1}{8}$, and $\frac{1}{16}$ resolutions of the input image (see Fig. 2). These feature maps are then fed into our depth decoder (see Fig. 3), producing four multiscale depth maps. Concurrently, the input adjacent frames are used by PoseNet to compute the 6-DOF relative pose. Finally, the loss is calculated by upsampling each output to match the input image resolution. For details on Laplacian Pyramid Residuals and Boundary Information Masks, please refer to Section 3.2 and Section 3.3, respectively.

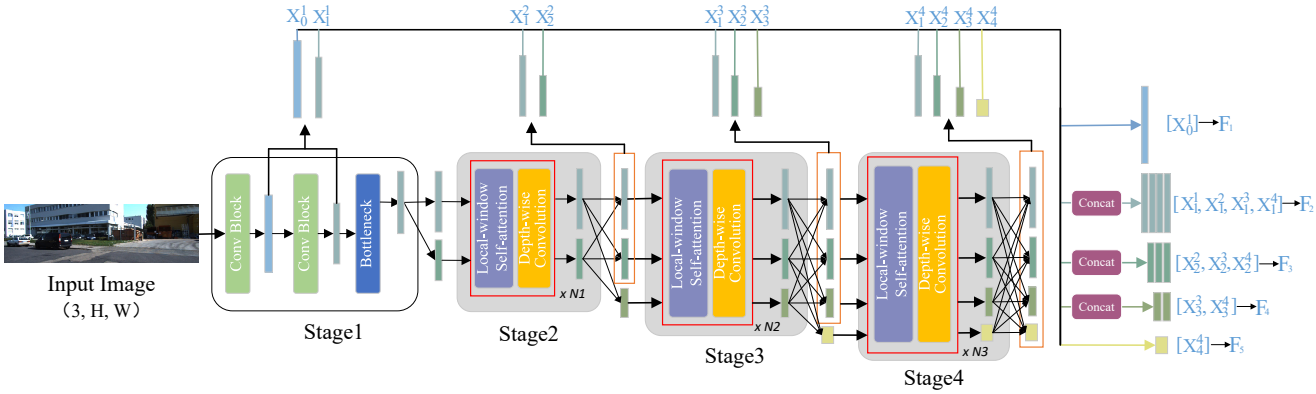

Fig. 2 Encoder architecture of MFFENet. $N1$ to $N3$ represent the number of successive local window self-attention and depth-wise convolution layers stacked at each stage, respectively. "Concat" denotes the concatenation of feature maps.

combination strategy and DeconvUp module. We demonstrate the effectiveness of modifying the HRFormer architecture with our proposed approach. We discuss the purpose and significance of each module in the MFFENet encoder. Also, we detail the integration of Laplacian pyramid residuals in the loss function and describe the self-supervised training approach employed by our framework. The architecture of MFFENet is illustrated in Fig. 1.

3.1 Depth Encoder

We recognize the strong performance of DIFFNet's multi-stage internal feature fusion mechanism in monocular depth estimation. However, its encoder utilizes a CNN network architecture, which may suffer from limited capture of global information in depth estimation tasks [36]. Therefore, our approach aims to enhance HRFormer by incorporating the multi-stage internal feature fusion concept from DIFFNet. In the specific implementation,

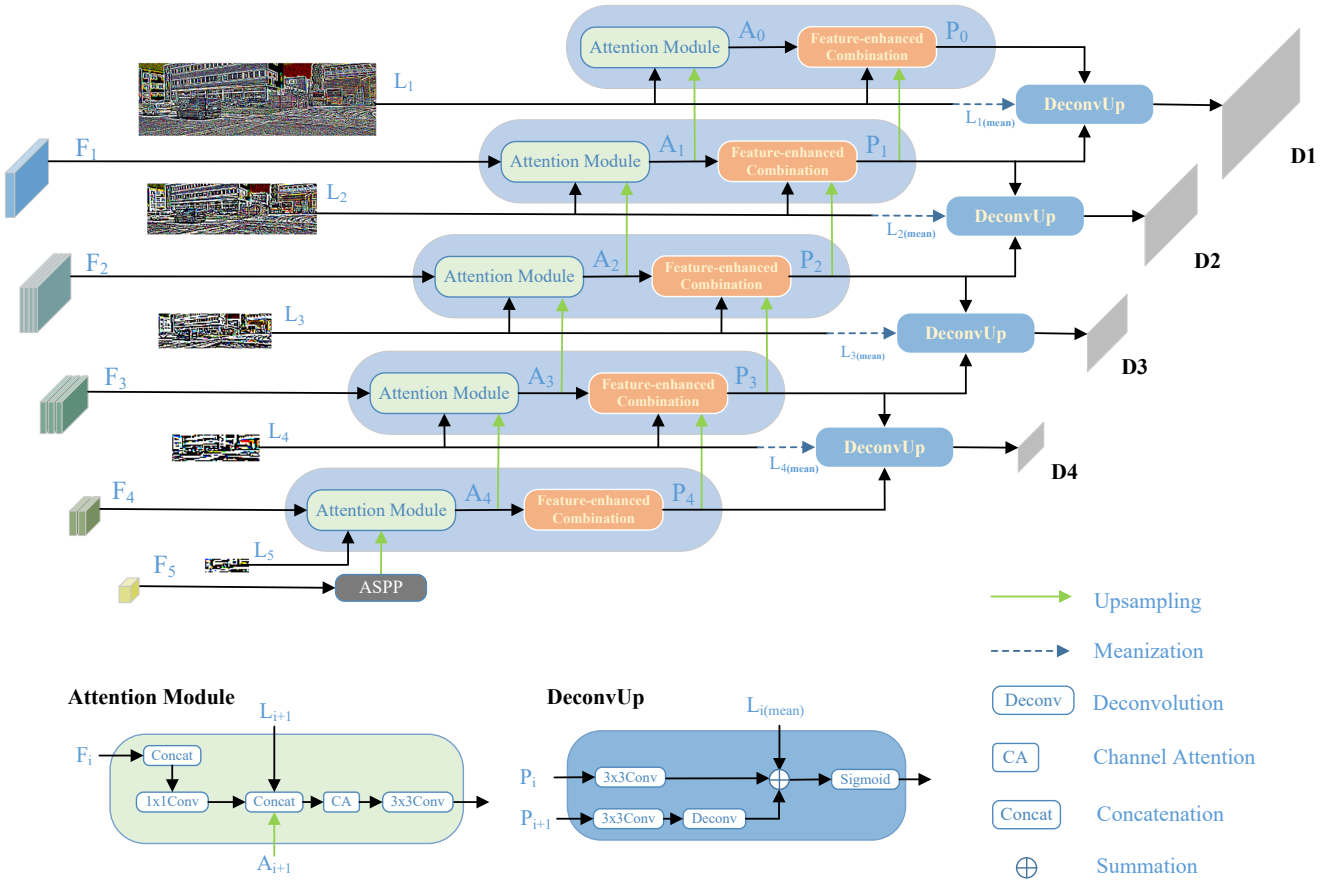

Fig. 3 MFFENet Decoder: The decoder in MFFENet uses feature maps from the encoder and combines them with the input image's Laplacian pyramid residuals, and then generates the final depth map using the attention module, feature-enhanced combination strategy, and DeconvUp module respectively.

we reconsider the feature fusion approach to ensure better integration with HRFormer.

As shown in Fig. 2, MFFENet encoder fuses features from all stages before decoding (for detailed encoder architecture, please refer to the supplementary material Section E). The feature map of the final output of the encoder is defined as:

$$F_i = \begin{cases} X_0^1, & i = 1, \\ [X_{i-1}^j], & j = (i-1)....4, \\ X_4^4, & i = 5, \end{cases} \tag{1}$$

where the concatenation layer is denoted as $[\cdot]$, and X_i^j represents the feature map of each stage (for details on the encoder feature fusion process, please see supplementary material Section G). Different colored arrows in Fig. 2 depict the concatenation of feature maps with the same resolution, denoted as F_i. This method stands apart from other transformer-based approaches. We visualize features from the DIFFNet encoder, HRFormer, and our encoder. As demonstrated in Fig. 4, the features generated by our encoder exhibit clearer details in contrast to those output by the DIFFNet encoder and HRFormer. Our ablation experiments in Section 5.3 illustrate that our proposed encoder effectively enhances the accuracy of monocular depth estimation compared to the original HRFormer.

3.2 Depth Decoder

Inspired by [37,26] and drawing on the ideas of ResNet, we propose a decoder based on attention mechanism and Laplacian pyramid residuals. This decoder is combined with our encoder to form the U-Net architecture. Each step of our decoder is designed with the consideration of addressing the issue of depth map blurring at the boundaries. The decoder consists of attention modules, a feature enhancement combination strategy, and the DeconvUp module, as depicted in Fig. 3. We utilize the feature maps F_i obtained from the encoder, where the subscript $i \in \{1, 2, 3, 4, 5\}$ denotes different scales, along with the corresponding Laplacian pyramid residuals L_i of the input image. These serve as inputs to the attention module, which generates A_i.

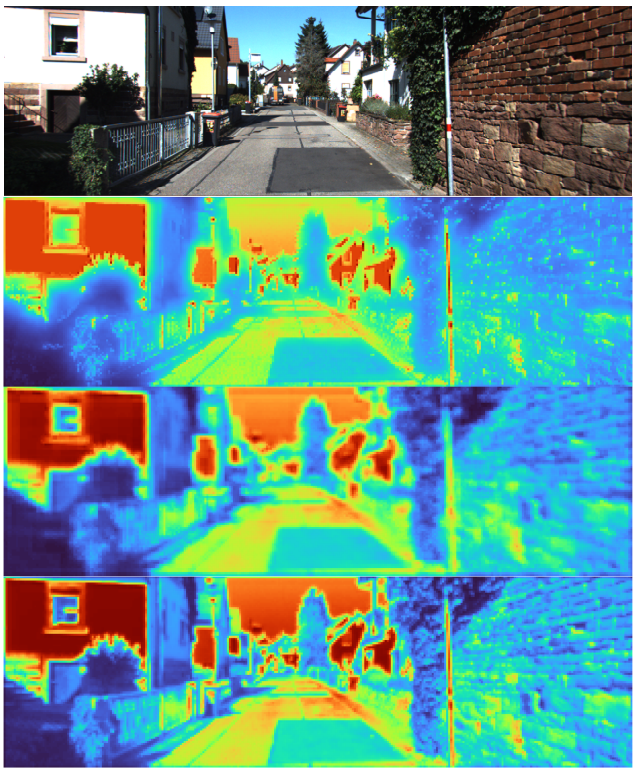

Fig. 4 Visualization of feature maps generated by the encoder. **First row**: Source RGB image. **Second row**: Feature map output by the DIFFNet encoder. **Third row**: Feature map output by HRFormer. **Last row**: Feature map output by our encoder.

Subsequently, we employ a feature-enhanced combination strategy to enrich features and obtain P_i. Finally, the DeconvUp modules combine features P_i and P_{i+1} to generate the depth map D_i. Detailed information about the decoder's structure can be found in the supplementary material Section E.

Attention Module. We incorporate the Laplacian pyramid residuals to fuse with the feature maps in each module of the decoder to add boundary information. Specifically, we utilize F_i, L_{i+1}, and previously generated A_{i+1} as inputs to the attention module, employing the channel attention. Zhou et al. [37] have observed that channel attention produces better results in monocular depth estimation. Therefore, we also employ channel attention. The calculation process of the attention module is illustrated as follows:

$$A_i = \begin{cases} \phi([F_i], Up(A_{i+1}), L_{i+1}), & i = 1,2,3,4 \\ \phi(Up(A_1), L_1), & i = 0 \end{cases} \tag{2}$$

where $\phi()$ denotes a sequence of convolutions or a combination of convolution and attention mechanisms and $[\cdot]$ indicates the concatenation layer. The Laplacian pyramid residuals of the input image is calculated as: $L_i = I_i - Up(I_{i+1})$. Here, I_i corresponds to downsampling the original input color image (at a ratio of $(\frac{1}{2})^{i-1}$). The function $Up()$ performs up-sampling (at a ratio of 2). Our decoder differs from [37,26]. Compared to [37], we incorporate Laplacian pyramid residuals, and in contrast to [26], we utilize channel attention mechanism.

Feature-enhanced Combination Strategy. We consider that the semantic information of A_i may be weakened during a series of convolutional processes, which hinders the accurate prediction of depth boundaries and the generation of precise depth maps. Based on the concept of residual blocks, we adopt this strategy to recombine A_i, resulting in the generation of feature P_i with enhanced semantic information. Additionally, we reintroduce the Laplacian pyramid residual L_i to enhance the boundary information in the network. As a result, five feature maps are generated at all scales, as follows:

$$P_i = \begin{cases} C1([A_4]), & i = 4, \\ C1([A_i, Up(P_{i+1}), L_{i+1}]), & i = 0,1,2,3 \end{cases} \tag{3}$$

where $C1()$ denotes a 1×1 convolution layer. To demonstrate the effectiveness of our strategy, we visualize the first three feature maps of the attention module and the first three feature maps of the feature enhancement combination strategy. As shown in Fig. 5, compared to the blurry contours in A_2, P_2 already captures object outline information from the input image, such as the signage on the right. As we ascend through the hierarchy, it becomes evident in P_0 that the object outline information becomes more abundant, and the boundaries become clearer.

DeconvUp Module. To improve the upsampling of feature maps at the boundaries, we utilize deconvolution to upsample the feature P_{i+1} and enhance its boundary information, and then sum with P_i. To incorporate the Laplacian pyramid residuals, we average them($3\times W\times H$ to $1\times W\times H$). The computation process of the DeconvUp module is illustrated as follows:

$$D_i = \sigma(S[C(P_{i-1}), \rho(C(P_i)), L_{i(mean)}]), i = 1,2,3,4, \tag{4}$$

where $\rho()$ represents a deconvolution operator, $C()$ denotes a 3×3 convolution layer, $S[\cdot]$ represents the sum operator, $\sigma()$ is a sigmoid activation function, and $L_{i(mean)}$ refers to the Laplacian pyramid residuals of the input image averaged over the channel dimension, D_i represents the final output depth map. Finally, the decoder outputs complete depth maps at resolutions of 1, $\frac{1}{2}$, $\frac{1}{4}$, and $\frac{1}{8}$.

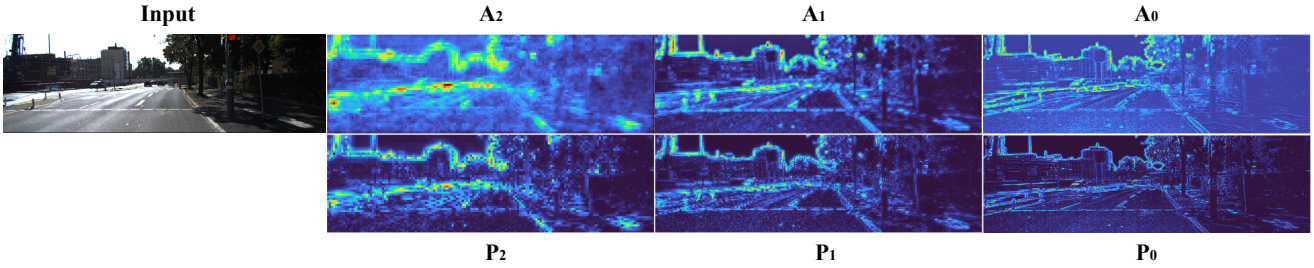

Fig. 5 Visualization of Feature Maps. The first column is the input image, the first row (except the first column) is the feature map obtained from the attention module, and the second row is the feature map obtained from the feature augmentation combination strategy.

3.3 Boundary Loss

Our work uncovers the latent value of Laplacian pyramid residuals in input images and introduces novel contributions in their utilization. We leverage these residuals not only in the decoder architecture but also in the loss function, exploiting their boundary information property. To explicitly constrain the depth of borders between objects, we compute the average of the channels from the Laplacian pyramid residual image, resulting in the transformation into a grayscale image. Next, to apply it in the loss function, we calculate the average value of the grayscale image as a reference and employ a discriminator to identify pixels with values exceeding this reference, generating a binary mask image $M(M \in \{0,1\})$. This binary mask image, depicted at the bottom of Fig. 6, identifies pixels with boundary information in the RGB image. To further refine the training of boundary depth, we combine the binary mask image with the berHu loss, defined as:

$$L_b = M \begin{cases} |I_t - I'_t|, & if\ |I_t - I'_t| \le c, \\ \frac{|I_t - I'_t|^2 - c^2}{2c}, & Otherwise, \end{cases} \tag{5}$$

where I_t represents the target frame, I'_t denotes the composite frame, and c is calculated as $\delta \cdot \max(|I_t - I'_t|)$ with δ set to 0.2. Our ablation experiments in Section 5.3 demonstrate the effectiveness of our proposed boundary loss for our model. The impact of boundary loss in MFFENet can be viewed in supplementary material Section C.

4 Training

Similar to most self-supervised monocular depth estimation methods that utilize Structure from Motion (SfM) training, we adopt a similar approach. In our method, the target frame is denoted as I_t, and we select the source frame as I_s, where s corresponds to $t-1$ or $t+1$. To compute the photometric loss, we reproject I_s to reconstruct I_t by simultaneously training a depth network and a pose network. In addition to the commonly used photometric loss in existing methods, we also introduce our proposed boundary loss into the overall loss function. The depth network predicts the depth map $d(I_t)$ using I_t as the input image, while the pose network predicts the 6-degree-of-freedom (6-DOF) relative camera pose $T_{t\to s}$ between the target frame and the source frame. The reprojected image is defined as:

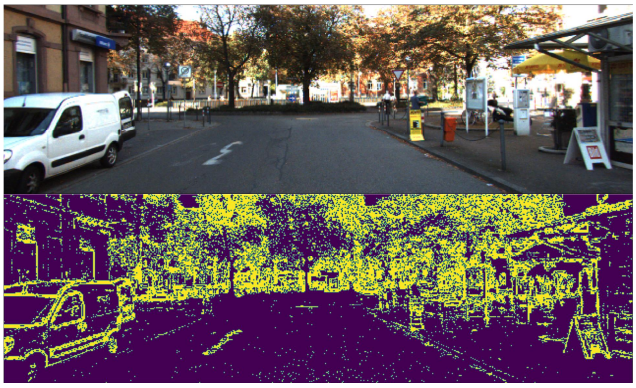

Fig. 6 Examples of boundary information masks generated by the Laplacian pyramid residuals of the input image. **Top**: Source RGB image. **Bottom**: Boundary information binary mask map, highlighting pixels containing boundary information (indicated by yellow points).

$$I_{s\to t} = I_s[W(K, T_{t\to s}, \theta)], \tag{6}$$

where K is known as camera intrinsics. $W()$ represents the transformation and projection operations applied to the 3D point cloud θ of I_t, which is defined in Equation 7. Here, $P(I_t)$ denotes the homogeneous coordinate of the pixel in I_t. The sampling operator $[\cdot]$ is used to sample the source images I_s using bilinear interpolation, resulting in the reprojected image $I_{s\to t}$.

$$\theta = d(I_t) \cdot K^{-1} \cdot P(I_t), \tag{7}$$

The photometric error, L_{phot}, between $I_{s\to t}$ and I_t, where $s \in \{t-1, t+1\}$, is defined as the combination of structural similarity (SSIM) and L1 error:

$$L_{phot}(I_t, I_{s\to t}) = \alpha \frac{1 - SSIM(I_t, I_{s\to t})}{2} + (1-\alpha)\,|I_t - I_{s\to t}|\,,$$

(8)

To regularize local smoothness in regions with low image gradient, we introduce an edge-aware smoothness loss:

$$L_{sm} = |\partial_x d_t|\, e^{-|\partial_x I_t|} + |\partial_y d_t|\, e^{-|\partial_y I_t|}, \tag{9}$$

To address occlusion between views and enhance depth estimation accuracy, we employ the minimum reprojection technique and integrate auto-masking. Additionally, we enforce smoothness in the estimated depth map by incorporating the smoothness regularization loss, L_{sm}. The overall loss for self-supervised depth estimation is defined as:

$$L = \mu[\min(L_{phot}(I_t, I_{s\to t}) + \lambda L_b) + \lambda_2 L_{sm}], \tag{10}$$

where λ and λ_2 are the weights for the boundary loss and smoothness regularization terms, respectively. The auto-masking operation, denoted by $\mu[\cdot]$, is applied to filter out inappropriate pixels. This loss function is utilized for the joint training of the depth and pose networks. Implementation details can be seen in supplementary material Section D.

5 Experiments

5.1 Datasets

KITTI Dataset. The KITTI dataset [8] serves as a widely adopted benchmark for stereo and monocular depth estimation tasks. It comprises real-world image data captured in diverse environments, including urban, rural, and highway settings. The RGB images in the dataset have an approximate resolution of 1241×376 pixels, while the corresponding ground truth depth maps are sparse and lack extensive coverage. To prepare the training data for monocular sequences, we employ the data split preprocessing technique introduced by Eigen et al. [7] and Zhou et al. [38], which involves removing static frames. As a result, we obtain 39,810 monocular triplets for training and 4,424 monocular triplets for validation. For evaluating the depth prediction performance, we select 697 images from the dataset as the test set and scale the ground truth depths to the median ground truth scaling following the evaluation protocol described in [38]. Consistent with recent studies, we set the maximum predicted depth for testing evaluation to 80 meters. Additional datasets Cityscape and Make3D can be viewed in supplementary material Section A.

5.2 Evaluation on Datasets

Results on KITTI. We conduct a comprehensive evaluation of our model's depth prediction performance on the KITTI dataset using the metrics proposed by [7]. Table 1 presents the results of our model alongside recent methods. In test scenarios with an input image resolution of 640×192, our model surpasses state-of-the-art methods across multiple metrics, including Abs Rel, SqRel, and RMSE. Notably, even our compact variant, MFFENet-tiny, achieves comparable performance (see supplementary material Section E for details on MFFENet-tiny). Remarkably, MFFENet-tiny comprises only 4.5M parameters.

In our experiments, we compare the performance of MFFENet-tiny with recent models such as Lite-Mono and DIFFNet. Although MFFENet-tiny has a parameter size comparable to Lite-Mono (3.1M parameters), it outperforms Lite-Mono-8m (8.7M parameters) in terms of accuracy. We also present results on stereo video (MS) and high resolution (1024×320). While MFFENet-tiny may not achieve the best performance across all metrics due to parameter constraints, MFFENet-small consistently delivers superior results, demonstrating the effectiveness of our approach in addressing the issue of photometric loss.

A qualitative comparison of our method with Lite-Mono-8m [34], DIFFNet [37], R-MSFM6 [39], Monodepth2 [10], and HR-Depth [21] is depicted in Fig. 7 (further comparisons are available in supplementary material Section H). Our method demonstrates notable strengths, highlighted by the blue dashed regions in the second row of the figure. Specifically, in the first column, our method accurately predicts the tree trunk bifurcation, a task where other methods falter. Moreover, our method captures fine details such as the rearview mirror of the car. In the second and third columns, our method exhibits a superior understanding of object structures. Notably, in challenging scenarios with thin structures (last column), our method surpasses others, which show deformations, divergence, and inaccuracies in the depth estimation of the blue area. These issues in other methods primarily arise from their inadequate focus on object boundaries, an aspect we have successfully addressed. Results on the Cityscape dataset, Make3D dataset, and additional results on the KITTI dataset are provided in supplementary material Section B. Furthermore, several instances of poor depth map results predicted by our method are available for review in Supplementary Material Section F.

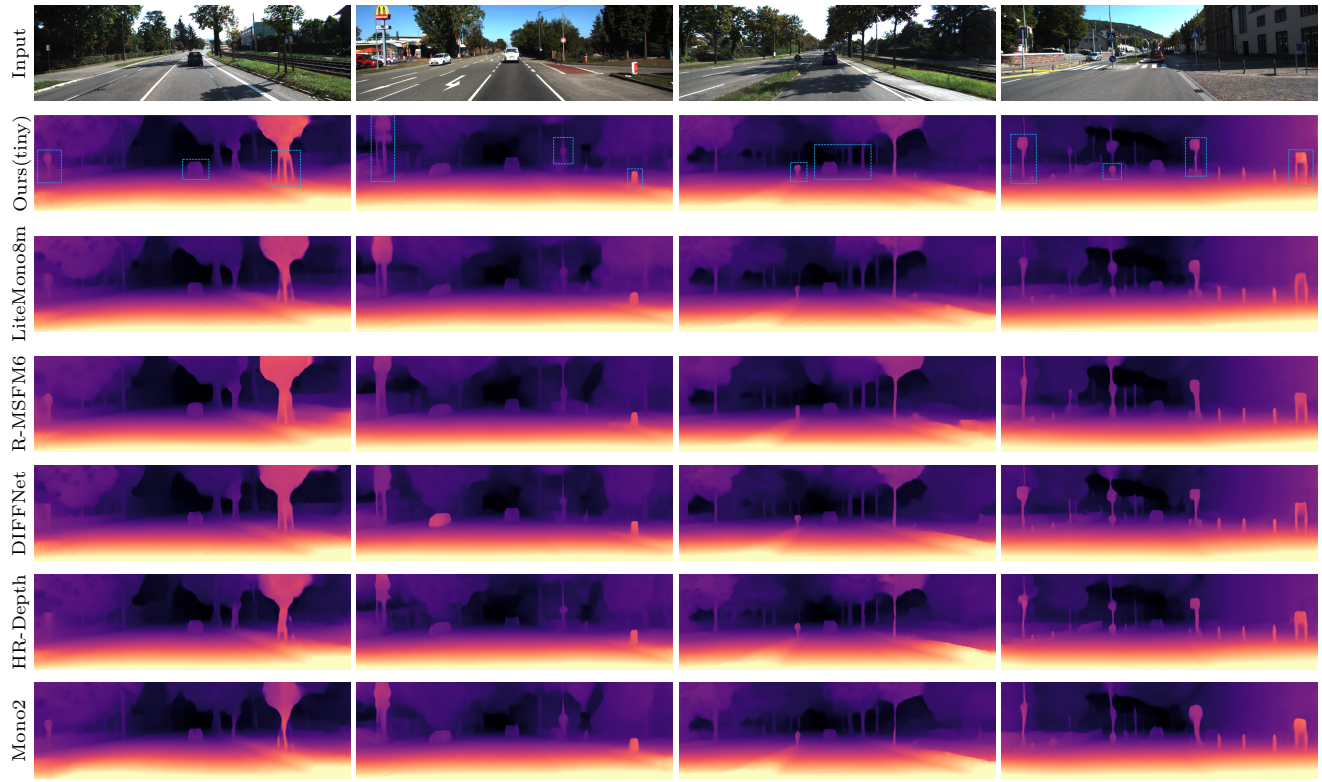

Fig. 7 Qualitative results on the KITTI Eigen split test set. The first row shows the input RGB image. The second row displays the output of our MFFENet, while the subsequent rows show the output of other methods.

5.3 Ablation Study

To validate the performance improvement of our proposed contributions in monocular depth estimation, we conduct ablation experiments on the KITTI dataset. We compare different variants of MFFENet by combining our proposed attention module, feature augmentation combination strategy, deconvUp model, and boundary loss.

The effectiveness of each proposed contribution in enhancing the accuracy of monocular depth estimation is illustrated in the results showcased in Table 2. Combining these contributions in pairs shows a synergistic effect, resulting in further performance improvements. Especially, our feature-enhanced combination strategy plays a crucial role in enriching the feature information.

In Table 3, we conduct ablation experiments on the MFFENet encoder (Section 3.1) to evaluate the efficacy of feature concatenation. Notably, when no feature fusion is performed (Stage selection: fourth stage, representing the direct output of HRFormer), the depth accuracy experiences a significant decrease. As we progressively fuse feature maps from each stage, estimation accuracy improves, peaking when all stage feature maps are fused (Stage selection: full (fourth, third, second, first stage), representing our complete approach). Additional ablation experiments are detailed in the supplementary material Section C.

5.4 Extended Evaluation for Boundaries

To show the effect of the model on the boundary depth estimates. We leverage Laplacian pyramid residuals to enhance the evaluation of boundary depth accuracy on the test set and also facilitate comparisons with other models. By applying the binary mask from Section 3.3, we extract pixels containing boundary information to modify the testing procedure for depth estimation. The results presented in Table 4 indicate that both versions of our model outperform other methods on all metrics, attributed to our method's focus on perceiving depth at boundaries. When employing our proposed components in the testing process, DIFFNet exhibits improvements in multiple metrics compared to its original version, providing further evidence of the efficacy of our contributions.

6 Conclusion

We present MFFENet, a novel self-supervised depth estimation network framework aimed at improving ac-

Table 1 Comparison of our model with other models on the KITTI Benchmark using the Eigen split. M: trained on monocular video sequences. MS: trained on stereo video sequences. Se: trained with semantic labels. The best result for each metric is highlighted in bold.

Method	Train	W×H	The lower is better (↓)				The higher is better (↑)			Params(↓)
			Abs Rel	Sq Rel	RMSE	RMSE log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$	
Wang et al. [31]	M	640×192	0.151	1.257	5.583	0.228	0.810	0.936	0.974	28.1M
Monodepth2 [10]	M	640×192	0.115	0.903	4.863	0.193	0.877	0.959	0.981	14.3M
Klingner et al. [17]	M+Se	640×192	0.113	0.835	4.693	0.191	0.879	0.961	0.981	16.3M
SC-DepthV3 [27]	M	832×256	0.118	0.756	4.709	0.188	0.864	0.960	0.984	14.1M
Choi et al. [4]	M+Se	640×192	0.112	0.788	4.582	0.187	0.878	0.963	0.983	-
PackNet [12]	M	640×192	0.111	0.785	4.601	0.189	0.878	0.960	0.982	128M
HR-Depth [21]	M	640×192	0.109	0.792	4.632	0.185	0.884	0.962	0.983	14.7M
Bae et al. [1]	M	640×192	0.104	0.846	4.580	0.183	0.891	0.962	0.982	23.9M+
FSRE-Depth [15]	M+Se	640×192	0.105	0.722	4.547	0.182	0.886	0.964	**0.984**	24.5M
DIFFNet [37]	M	640×192	0.102	0.764	4.483	0.180	0.896	0.965	0.983	10.8M
Sun et al. [28]	M	640×192	0.117	0.863	4.813	0.192	0.871	0.959	0.982	-
Zhang et al. [35]	M	640×192	0.112	0.856	4.778	0.190	0.880	0.961	0.982	-
MonoViT-tiny [36]	M	640×192	0.102	0.733	4.459	0.177	0.895	0.965	**0.984**	10.3M
Lite-Mono-8M [34]	M	640×192	0.101	0.729	4.454	0.178	0.897	0.965	0.983	8.7M
Ours(tiny)	M	640×192	0.101	0.716	4.356	0.177	0.898	0.966	0.983	**4.5M**
Ours(small)	M	640×192	**0.098**	**0.695**	**4.352**	**0.175**	**0.900**	**0.967**	**0.984**	11.8M
Monodepth2 [10]	MS	640×192	0.106	0.818	4.750	0.196	0.874	0.957	0.979	14.3M
Yang et al. [32]	MS	640×192	0.099	0.763	4.485	0.185	0.885	0.958	0.979	-
HR-Depth [21]	MS	640×192	0.107	0.785	4.612	0.185	0.887	0.962	0.982	14.7M
DIFFNet [37]	MS	640×192	0.101	0.749	4.445	0.179	0.898	0.965	0.983	10.8M
R-MSFM6 [39]	MS	640×192	0.111	0.787	4.625	0.189	0.882	0.961	0.981	**3.8M**
Ours(tiny)	MS	640×192	0.102	0.717	4.372	0.177	0.897	0.966	**0.984**	4.5M
Ours(small)	MS	640×192	**0.096**	**0.707**	**4.371**	**0.175**	**0.906**	**0.967**	**0.984**	11.8M
Monodepth2 [10]	M	1024×320	0.115	0.882	4.701	0.190	0.879	0.961	0.982	14.3M
PackNet [12]	M	1280×384	0.107	0.802	4.538	0.186	0.889	0.962	0.981	128M
Klingner et al. [17]	M+Se	1280×384	0.107	0.768	4.468	0.186	0.891	0.963	0.982	16.3M
Shu et al. [25]	M	1024×320	0.104	0.729	4.481	0.179	0.893	0.965	**0.984**	35.2M
Sun et al. [28]	M	1024×320	0.110	0.791	4.557	0.184	0.887	0.964	0.983	-
Lite-Mono-8M [34]	M	1024×320	0.097	0.710	4.309	0.174	0.905	0.967	**0.984**	8.7M
Ours(tiny)	M	1024×320	0.100	0.725	4.332	0.175	0.902	0.967	**0.984**	**4.5M**
Ours(small)	M	1024×320	**0.095**	**0.687**	**4.232**	**0.172**	**0.910**	**0.969**	**0.984**	11.8M

Table 2 Ablation Studies. AM: Attention Module. FCS: Feature-enhanced Combination Strategy. DeconvUp: DeconvUp Module. Boundary Loss: Proposed boundary loss. The last row represents our complete method.

Method	AM	FCS	DeconvUp	Bounbary Loss	The lower is better (↓)				The higher is better (↑)		
					Abs Rel	Sq Rel	RMSE	RMSE log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Ours		✓	✓	✓	0.103	0.753	4.446	0.179	0.896	**0.966**	**0.983**
	✓		✓	✓	0.103	0.742	4.432	0.178	0.895	**0.966**	**0.983**
	✓	✓		✓	0.103	0.761	4.456	0.179	0.895	**0.966**	**0.983**
	✓	✓	✓		0.102	0.750	4.430	0.178	0.896	**0.966**	**0.983**
	✓	✓	✓	✓	**0.101**	**0.716**	**4.356**	**0.177**	**0.898**	**0.966**	**0.983**

Table 3 Ablation experiment of MFFENet encoder using feature map fusion mechanism. Stage selection refers to the fusion of feature maps from different stages in the MFFENet encoder. The fourth stage represents the original HRFormer architecture.

Method	Stage selection	The lower is better (↓)				The higher is better (↑)		
		Abs Rel	Sq Rel	RMSE	RMSE log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Ours	fourth stage	0.105	0.767	4.510	0.181	0.890	0.964	**0.983**
	fourth, third stage	0.104	0.777	4.494	0.180	0.894	0.964	**0.983**
	fourth, third, second stage	0.102	0.738	4.444	0.179	0.895	0.965	**0.983**
	full(fourth, third, second, first stage)	**0.101**	**0.716**	**4.356**	**0.177**	**0.898**	**0.966**	**0.983**

curacy at object boundaries, a challenge often associated with photometric loss. Our framework employs a modified HRFormer backbone as the encoder, offering superior performance with fewer parameters. Through ablation experiments, we demonstrate the effectiveness of connecting and fusing feature maps from different

Table 4 Ablation experiments using pixels containing boundary information. DIFFNet (Our): DIFFNet with our proposed components. All models are trained on KITTI with an image resolution of 640×192.

Method	The lower is better (↓)				The higher is better (↑)		
	Abs Rel	Sq Rel	RMSE	RMSE log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Monodepth2	0.130	1.247	5.600	0.213	0.855	0.948	0.974
HR-Depth	0.124	1.098	5.351	0.206	0.862	0.950	0.976
Lite-Mono	0.121	1.018	5.227	0.202	0.866	0.952	0.977
Lite-Mono-8m	0.116	0.996	5.141	0.198	0.875	0.954	0.977
DIFFNet	0.117	1.056	5.200	0.201	0.874	0.954	0.976
DIFFNet (Ours)	0.117	1.028	5.188	0.199	0.873	0.954	0.977
Ours(tiny)	0.115	0.998	5.076	0.198	0.876	0.954	0.977
Ours(small)	**0.113**	**0.950**	**5.058**	**0.196**	**0.879**	**0.956**	**0.978**

encoder stages, enhancing MFFENet's overall performance. Additionally, we propose innovative contributions to the decoder, including attention mechanisms and Laplacian pyramid residuals in the attention module, which augment boundary information. Our feature-enhanced combination strategy, combined with the DeconvUp module, synergistically enables the generation of precise depth maps. We introduce a boundary loss leveraging Laplacian pyramid residuals to extract boundary pixels, resulting in improved depth estimation accuracy near object boundaries.

To assess the effectiveness and generalizability of MFFENet, we conducted comprehensive experiments on the KITTI dataset, surpassing the state-of-the-art methods. Ablation studies confirm the positive impact of our contributions on monocular depth estimation. We also showcase good performance of MFFENet on the Cityscape dataset. To validate its robustness and generalizability, we evaluated MFFENet on the Make3D dataset, demonstrating its ability to perform well across different datasets. Our extended evaluation method assesses boundary depth accuracy, affirming the effectiveness of our model in enhancing this aspect.

While successful, it is crucial to acknowledge that MFFENet's encoder, combining convolutional and Transformer networks, introduces heightened computational complexity during both training and inference, compared to fully convolutional models. This underscores the need for future research to explore faster methods for monocular depth estimation. Additionally, integrating architectures related to object detection tasks into the monocular depth estimation framework holds promise for further performance enhancements.

References

1. Bae, J., Moon, S., Im, S.: Deep digging into the generalization of self-supervised monocular depth estimation. In: Proceedings of the AAAI conference on artificial intelligence, vol. 37, pp. 187–196 (2023)
2. Chen, P.Y., Liu, A.H., Liu, Y.C., Wang, Y.C.F.: Towards scene understanding: Unsupervised monocular depth estimation with semantic-aware representation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2624–2632 (2019)
3. Chen, X., Zhang, R., Jiang, J., Wang, Y., Li, G., Li, T.H.: Self-supervised monocular depth estimation: Solving the edge-fattening problem. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 5776–5786 (2023)
4. Choi, J., Jung, D., Lee, D., Kim, C.: Safenet: Self-supervised monocular depth estimation with semantic-aware feature extraction. In: Thirty-fourth Conference on Neural Information Processing Systems, NIPS 2020. NeurIPS (2020)
5. Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., Schiele, B.: The cityscapes dataset for semantic urban scene understanding. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 3213–3223 (2016)
6. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. In: International Conference on Learning Representations (2020)
7. Eigen, D., Puhrsch, C., Fergus, R.: Depth map prediction from a single image using a multi-scale deep network. Advances in neural information processing systems **27** (2014)
8. Geiger, A., Lenz, P., Stiller, C., Urtasun, R.: Vision meets robotics: The kitti dataset. The International Journal of Robotics Research **32**(11), 1231–1237 (2013)
9. Godard, C., Mac Aodha, O., Brostow, G.J.: Unsupervised monocular depth estimation with left-right consistency. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 270–279 (2017)
10. Godard, C., Mac Aodha, O., Firman, M., Brostow, G.J.: Digging into self-supervised monocular depth estimation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 3828–3838 (2019)
11. Gordon, A., Li, H., Jonschkowski, R., Angelova, A.: Depth from videos in the wild: Unsupervised monocular depth learning from unknown cameras. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 8977–8986 (2019)
12. Guizilini, V., Ambrus, R., Pillai, S., Raventos, A., Gaidon, A.: 3d packing for self-supervised monocular depth estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2485–2494 (2020)
13. Hirschmuller, H.: Accurate and efficient stereo processing by semi-global matching and mutual information. In:

2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), vol. 2, pp. 807–814. IEEE (2005)
14. Johnston, A., Carneiro, G.: Self-supervised monocular trained depth estimation using self-attention and discrete disparity volume. In: Proceedings of the ieee/cvf conference on computer vision and pattern recognition, pp. 4756–4765 (2020)
15. Jung, H., Park, E., Yoo, S.: Fine-grained semantics-aware representation enhancement for self-supervised monocular depth estimation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 12,642–12,652 (2021)
16. Kendall, A., Gal, Y., Cipolla, R.: Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 7482–7491 (2018)
17. Klingner, M., Termöhlen, J.A., Mikolajczyk, J., Fingscheidt, T.: Self-supervised monocular depth estimation: Solving the dynamic object problem by semantic guidance. In: European Conference on Computer Vision, pp. 582–600. Springer (2020)
18. Laina, I., Rupprecht, C., Belagiannis, V., Tombari, F., Navab, N.: Deeper depth prediction with fully convolutional residual networks. In: 2016 Fourth international conference on 3D vision (3DV), pp. 239–248. IEEE (2016)
19. Lee, Y., Kim, J., Willette, J., Hwang, S.J.: Mpvit: Multi-path vision transformer for dense prediction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 7287–7296 (2022)
20. Li, H., Gordon, A., Zhao, H., Casser, V., Angelova, A.: Unsupervised monocular depth learning in dynamic scenes. In: Conference on Robot Learning, pp. 1908–1917. PMLR (2021)
21. Lyu, X., Liu, L., Wang, M., Kong, X., Liu, L., Liu, Y., Chen, X., Yuan, Y.: Hr-depth: High resolution self-supervised monocular depth estimation. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, pp. 2294–2301 (2021)
22. Peng, R., Wang, R., Lai, Y., Tang, L., Cai, Y.: Excavating the potential capacity of self-supervised monocular depth estimation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 15,560–15,569 (2021)
23. Petrovai, A., Nedevschi, S.: Exploiting pseudo labels in a self-supervised learning framework for improved monocular depth estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1578–1588 (2022)
24. Saxena, A., Sun, M., Ng, A.Y.: Make3d: Learning 3d scene structure from a single still image. IEEE transactions on pattern analysis and machine intelligence **31**(5), 824–840 (2008)
25. Shu, C., Yu, K., Duan, Z., Yang, K.: Feature-metric loss for self-supervised learning of depth and egomotion. In: European Conference on Computer Vision, pp. 572–588. Springer (2020)
26. Song, M., Lim, S., Kim, W.: Monocular depth estimation using laplacian pyramid-based depth residuals. IEEE transactions on circuits and systems for video technology **31**(11), 4381–4393 (2021)
27. Sun, L., Bian, J.W., Zhan, H., Yin, W., Reid, I., Shen, C.: Sc-depthv3: Robust self-supervised monocular depth estimation for dynamic scenes. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) (2023)
28. Sun, Q., Tang, Y., Zhang, C., Zhao, C., Qian, F., Kurths, J.: Unsupervised estimation of monocular depth and vo in dynamic environments via hybrid masks. IEEE transactions on neural networks and learning systems **33**(5), 2023–2033 (2022)
29. Tosi, F., Aleotti, F., Poggi, M., Mattoccia, S.: Learning monocular depth estimation infusing traditional stereo knowledge. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 9799–9809 (2019)
30. Uhrig, J., Schneider, N., Schneider, L., Franke, U., Brox, T., Geiger, A.: Sparsity invariant cnns. In: 2017 international conference on 3D Vision (3DV), pp. 11–20. IEEE (2017)
31. Wang, C., Buenaposada, J.M., Zhu, R., Lucey, S.: Learning depth from monocular videos using direct methods. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2022–2030 (2018)
32. Yang, N., Stumberg, L.v., Wang, R., Cremers, D.: D3vo: Deep depth, deep pose and deep uncertainty for monocular visual odometry. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1281–1292 (2020)
33. Yuan, Y., Fu, R., Huang, L., Lin, W., Zhang, C., Chen, X., Wang, J.: Hrformer: High-resolution vision transformer for dense predict. Advances in Neural Information Processing Systems **34**, 7281–7293 (2021)
34. Zhang, N., Nex, F., Vosselman, G., Kerle, N.: Lite-mono: A lightweight cnn and transformer architecture for self-supervised monocular depth estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 18,537–18,546 (2023)
35. Zhang, Y., Gong, M., Li, J., Zhang, M., Jiang, F., Zhao, H.: Self-supervised monocular depth estimation with multiscale perception. IEEE transactions on image processing **31**, 3251–3266 (2022)
36. Zhao, C., Zhang, Y., Poggi, M., Tosi, F., Guo, X., Zhu, Z., Huang, G., Tang, Y., Mattoccia, S.: Monovit: Self-supervised monocular depth estimation with a vision transformer. In: International Conference on 3D Vision (2022)
37. Zhou, H., Greenwood, D., Taylor, S.: Self-supervised monocular depth estimation with internal feature fusion. In: British Machine Vision Conference (BMVC) (2021)
38. Zhou, T., Brown, M., Snavely, N., Lowe, D.G.: Unsupervised learning of depth and ego-motion from video. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1851–1858 (2017)
39. Zhou, Z., Fan, X., Shi, P., Xin, Y.: R-msfm: Recurrent multi-scale feature modulation for monocular depth estimating. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 12,777–12,786 (2021)
40. Zhu, S., Brazil, G., Liu, X.: The edge of depth: Explicit constraints between segmentation and depth. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 13,116–13,125 (2020)

Supplementary Material

A Additional Datasets

Cityscape Dataset. We select the Cityscape dataset [5] due to its higher-resolution urban driving images compared to KITTI. Also, Cityscape offers a greater presence of moving objects, presenting a more challenging scenario. This dataset has been underutilized for training and evaluation in depth estimation, providing an opportunity for comparative analysis with previous methods. The training set comprises 69,731 image triplets, while the test set includes 1,525 images. We adhere to the cropping and evaluation scheme outlined in [20] to ensure a consistent benchmark for comparison.

Make3D Dataset. To assess the generalization capability of our model in monocular depth estimation, we employ the Make3D dataset [24] as our test dataset. Make3D is a widely utilized outdoor dataset for evaluating model performance in terms of generalization. We adopt the same data processing methods employed by previous studies to ensure a fair and reliable comparison with other models. Particularly, all models are solely trained on the KITTI dataset, underscoring the transferability of our approach.

B Additional Evaluation

B.1 Comparing on Improved Ground Truth

The evaluation method on KITTI by Eigen et al. [7] utilizes reprojected lidar points to generate ground truth images, yet it does not address occlusions and moving objects. Uhrig et al. [30] introduced enhanced high-quality ground truth depth maps for the KITTI dataset. These improved images are derived from 5 lidar frames and provide better handling of occlusions through stereo images. As a result, 652 (93%) of the 697 original test frames in the Eigen test split [7] are retained. Leveraging these improved ground truth depth maps, we compare our method with others without the need for retraining all models. We employ the same evaluation strategy and metrics for consistency. The results are detailed in Table 8.

B.2 Results on KITTI.

Our method excels in accurately predicting depth details, particularly at object boundaries. Qualitative results comparing our method with others approaches are showcased in Fig. 8. We selected scenes with challenging boundary depths, such as poles, traffic lights, and trees. Both DIFFNet and HR-depth fail to accurately estimate the depth positions between objects in these scenes, which can be attributed to their disregard for the problem caused by photometric loss, resulting in depth divergence. In contrast, our model effectively mitigates this issue, as illustrated by the blue boxes in Fig. 8.

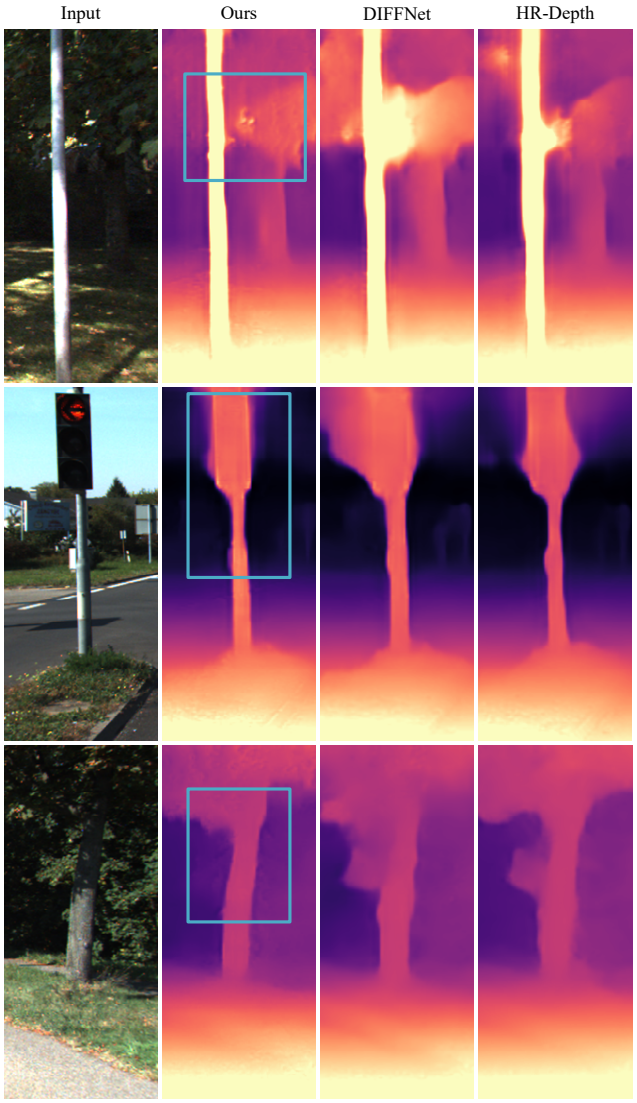

Fig. 8 Boundary depth between objects. The accuracy of predicting depth boundaries is crucial in avoiding blurring and divergence of depth estimates. Our network architecture outperforms previous methods, providing more accurate depth predictions.

A qualitative comparison between our method and boundary-aware models, including FSRE-Depth [15], SC-DepthV3 [27], and TriDepth [3], is presented in Fig. 9. The blue dashed regions in the second row of the figure

highlight the differences between our approach and the other methods. In the first column, our method successfully separates the boundaries of the platform from other objects, while the other methods exhibit varying degrees of boundary blurring. In the second column, we are also able to effectively separate the boundaries between the railing and the trees behind it, whereas FSRE-Depth and TriDepth confuse the front-back relationship. This confusion arises because they are guided by semantic networks that erroneously identify the railing as part of the tree trunk. Meanwhile, SC-DepthV3 exhibits depth divergence along the boundaries, which is influenced by the inaccurate pseudo-depth map for boundary awareness. Our method is not explicitly constrained by semantic networks but implicitly extracts semantic information and utilizes the proposed boundary loss to constrain the boundaries.

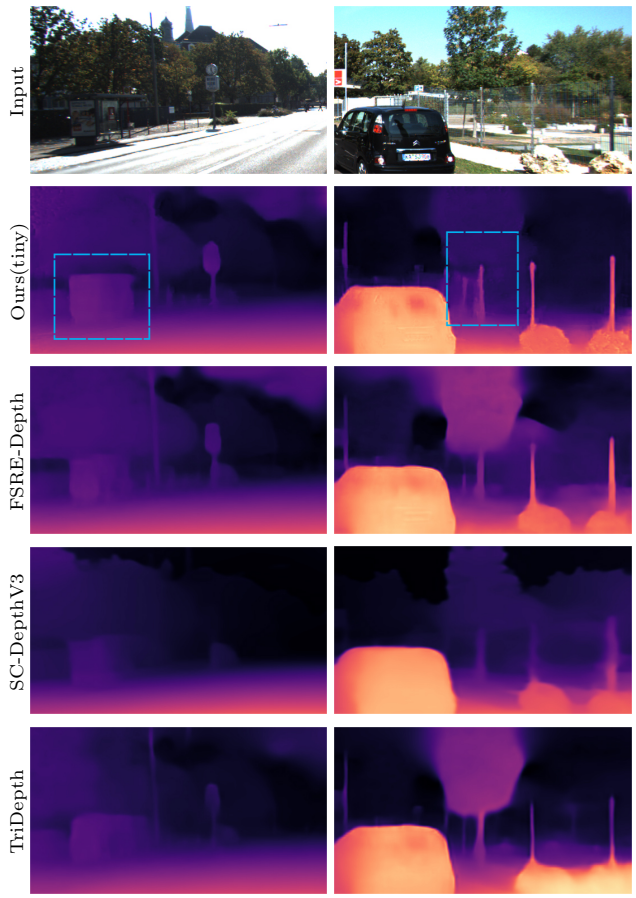

Fig. 9 Qualitative comparison of boundary-aware. The first row is the input image, and the remaining rows are the depth maps obtained by each model.

B.3 Results on Cityscape.

We evaluate MFFENet on the Cityscape dataset [5] and compare it with state-of-the-art models, as shown in Table 5. While SD-SSMDE [23] performs better than our model in terms of SqRel and RMSE, it utilizes a teacher-student network with a ResNet-50 backbone encoder, which has a higher parameter count than our network. On the whole, our network achieves comparable performance to SD-SSMDE in terms of SqRel and RMSE, and outperforms SD-SSMDE in AbsRel and RMSElog. This demonstrates the improvements of our method in boundary depth estimation.

Table 5 Comparison of MFFENet with other methods on the Cityscape dataset [5]. Input image resolution for training and testing is 416×128.

Method	The lower is better (↓)			
	Abs Rel	Sq Rel	RMSE	RMSE log
Monodepth2 [10]	0.129	1.569	6.876	0.187
Gordon et al. [11]	0.127	1.330	6.960	0.195
Li et al. [20]	0.119	1.290	6.980	0.190
SD-SSMDE [23]	0.110	**0.988**	**5.953**	0.165
Ours(tiny)	0.106	1.185	6.216	0.162
Ours(small)	**0.104**	1.086	6.082	**0.160**

B.4 Results on Make3D.

To evaluate our model's ability to generalize, we conduct tests on the Make3D dataset [24]. Following the methodology of prior studies [10,34], we exclusively train our model on the KITTI dataset [8] and assess its generalization performance against other models using the same setup. Table 6 presents a comparison of our two model variants with six other methods, with MFFENet-small achieving the highest performance. Despite the parameter limitations of MFFENet-tiny, it still demonstrates superior generalization compared to existing state-of-the-art methods (Lite-Mono and Lite-Mono-8m [34]) with similar parameter sizes. This is credited to our method's encoder implicitly extracting semantic information, enabling it to capture object structure cues and enhance network generalization. Qualitative examples of our model and other methods on the Make3D dataset are illustrated in Fig. 10. Although MFFENet is not trained on this dataset, it exhibits improved perception of object size, relative position, and depth compared to other models. This is evident from the clear depiction of distant tree trunk size and separation in the blue boxes of the first column, as well as the accurate repre-

sentation of the eave structure in the blue boxes of the second column.

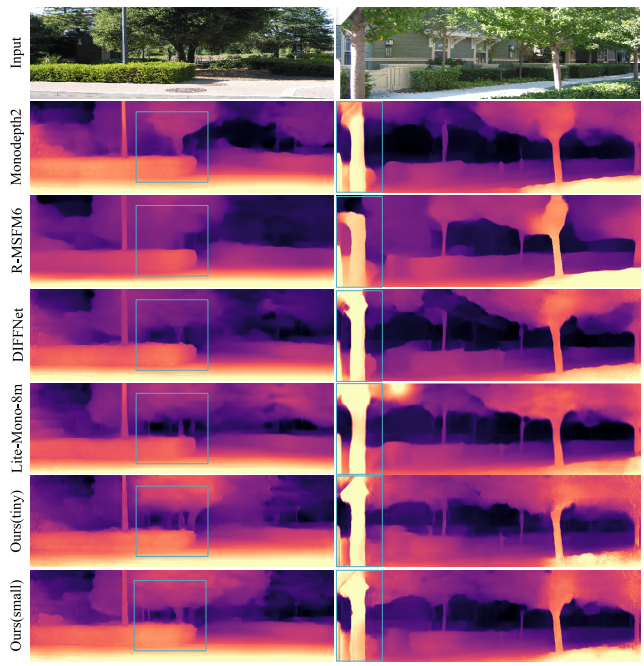

Fig. 10 Qualitative comparison on the Make3D test dataset. MFFENet is compared against monodepth2 [10], R-MSFM [39], DIFFNet [37], and Lite-Mono [34].

Table 6 Comparison of MFFENet with other methods on the Make3D dataset [24]. All models are trained on KITTI [8] using an image resolution of 640×192.

Method	The lower is better (↓)			
	Abs Rel	Sq Rel	RMSE	RMSE log
Wang et al. [31]	0.387	4.720	8.090	0.204
Monodepth2 [10]	0.322	3.589	7.417	0.163
R-MSFM6 [39]	0.334	3.285	7.212	0.169
Lite-Mono [34]	0.305	3.060	6.981	0.158
Lite-Mono-8m [34]	0.309	3.145	7.016	0.158
HR-Depth [21]	0.305	2.944	6.857	0.157
DIFFNet [37]	0.298	2.901	6.753	0.153
Ours(tiny)	0.302	2.964	6.880	0.157
Ours(small)	**0.288**	**2.785**	**6.676**	**0.149**

C Additional Ablation Experiments

To demonstrate the generality of our proposed modules, we investigate their impact on DIFFNet, as shown in Table 7. By incorporating our contribution at the decoder and the bounding loss function into DIFFNet, we observe an improvement in performance. This finding validates our contribution in enhancing the perception of object boundaries in other methods, thereby leading to improved performance. Moreover, by comparing the last row of data in Table 2 and Table 7, we can demonstrate the validity of replacing HRNet with HRFormer in Section 3.1. We observe a significant improvement in the depth estimation performance.

Fig. 13 illustrates a qualitative comparison between DIFFNet with and without our proposed modules. The enhanced DIFFNet demonstrates improved perception of the positional relationship between thin-structured objects and their surroundings, without experiencing depth divergence or blurring, as highlighted by the blue box area.

In addition, we also conduct ablation experiments on boundary loss. The impact of the boundary loss in MFFENet is illustrated in Fig. 12. The second and last rows display the depth maps generated by the full MFFENet and MFFENet without the boundary loss, respectively. It shows that the complete MFFENet, leveraging the boundary loss, effectively avoids significant divergence or disappearance of object depth within the region highlighted by the green box. This outcome highlights the explicit constraints imposed by our boundary loss and demonstrates its contribution to preserving accurate depth information.

D Additional Implementation Details

For the model output, we apply a conversion to obtain depth D from the sigmoid output σ of the last layer: $D = \frac{1}{(a\sigma+b)}$, where $a = 0.1$ and $b = 100$.

During training, our proposed model, MFFENet, is trained on two Nvidia Tesla T4 GPUs for 20 epochs using PyTorch. The KITTI training images are resized to 640×192. We employ the Adam optimizer with default parameters $\beta_1 = 0.9$ and $\beta_2 = 0.999$. The learning rate is initially set to $1e^{-4}$ for the first 14 epochs and then decayed to $1e^{-5}$. In the final self-supervised loss ((Equation 10), we set the SSIM weight α to 0.85, the edge-aware smoothness weight λ_2 to 0.001, and the boundary loss weight λ to 0.001.

Depth Network. MFFENet's encoder (see Section 3.1) is initialized with HRFormer pre-trained on ImageNet. During training, all four outputs of the model are used for photometric loss calculations. But during testing, only the depth map with the highest resolution from the model output is used. The effectiveness of MFFENet is demonstrated in Table 1.

Pose Network. We utilize the pose estimation network proposed by [10], which employs ResNet-18 as the backbone. It takes target frames and source frames as

input and produces the 6-DOF relative pose between I_t and I_s, where $s \in \{t-1, t+1\}$.

E Detail of Tiny Version of MFFENet

In our experiments, we use a tiny version of HRFormer's encoder model with pretrained ImageNet weights as the foundation of MFFENet-tiny. We tailor this tiny HRFormer model to extract only the feature maps it generates at each stage for synthesizing the encoder's output. The extracted feature maps from each stage of HRFormer are exclusively showcased in Table 9 (MFFENet Encoder). For a more detailed network structure, please refer to HRFormer [33]. In the MFFENet-tiny decoder section, we adopt the architecture outlined in Section 3.2 of this paper, with its detailed process outlined in Table 9 (MFFENet decoder). As for the pose model, we utilize the ResNet-18 architecture and pose decoder as defined by Monodepth2 [10].

F Qualitative Comparison of Poor Prediction Results

Fig. 11 presents a qualitative comparison highlighting areas of poorer depth map predictions by our method. The blue boxed areas in the second row indicate regions where our method performs poorly. In the first column, inadequate smoothing of depth on the top board is observed, possibly due to unnecessary boundary information introduced by the Laplacian pyramid residual we utilize. In the second column, inaccurate depth prediction for the person and bicycle is evident. This discrepancy arises from our method's reliance on implicit semantic cues, compared to FSRE-Depth, which incorporates explicit semantic information, leading to degraded predictions in regions with less distinct semantic differentiation. Addressing these issues will be a primary focus of our future work.

G The Specific Process of MFFENet Encoder Feature Fusion

In the first stage of our encoder, we employ two convolutions of size 3×3 with a stride of 1 to downsample the input image of dimensions $3\times H\times W$. This downsampling process yields two feature maps, $C_1\times\frac{H}{2}\times\frac{W}{2}$ and $C_2\times\frac{H}{4}\times\frac{W}{4}$, which we save for future use.

In the second stage, the upsampled feature maps are branched to generate two feature maps: $C_3\times\frac{H}{4}\times\frac{W}{4}$ and $C_4\times\frac{H}{8}\times\frac{W}{8}$. These feature maps serve as inputs to the HRFormer block, which combines continuous local window self-attention and depth convolution. This block updates the feature maps and saves them. Furthermore, we perform convolutional multi-scale fusion to obtain three new feature maps. The resulting third feature map, denoted as $C_5\times\frac{H}{16}\times\frac{W}{16}$, is then passed to the next stage.

The third and fourth stages follow a similar procedure to the second stage, producing a new feature map of $C_6\times\frac{H}{32}\times\frac{W}{32}$. Finally, we concatenate the saved feature maps from each stage, to obtain the final output of the encoder.

H More Qualitative Comparisons

In Fig. 14, we perform additional qualitative comparisons to multiple prior works on the KITTI dataset. Our method can more clearly predict the depth boundaries between objects.

Fig. 11 Qualitative comparison with poor prediction results. The first row is the input image, and the remaining rows are the depth maps obtained by each model.

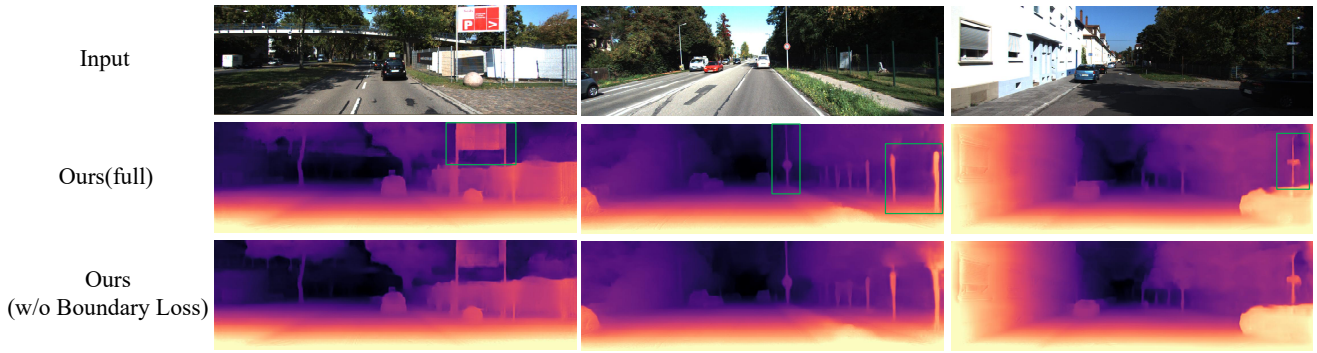

Fig. 12 Impact of our proposed boundary loss. The first row displays the input images. The second row exhibits the depth maps generated by our full model, while the last row showcases the depth maps generated without applying our proposed boundary loss. Variations between the results are emphasized in the green boxes.

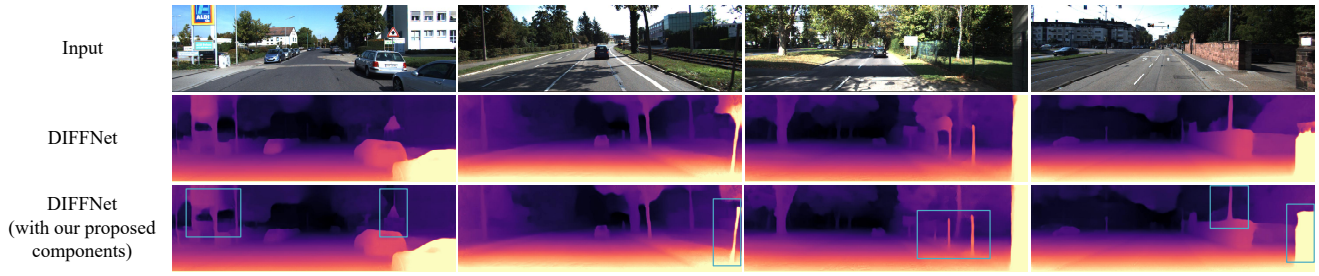

Fig. 13 Qualitative comparison between DIFFNet with our proposed components and the baseline DIFFNet. Our method enhances the quality of depth estimation, as illustrated in these visual comparisons.

Table 7 Ablation experiments utilizing pixels containing boundary information. DIFFNet (Ours): DIFFNet integrated with our proposed components. All models are trained on KITTI with an image resolution of 640×192.

Method	The lower is better (↓)				The higher is better (↑)		
	Abs Rel	Sq Rel	RMSE	RMSE log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
DIFFNet	**0.102**	0.764	4.483	0.180	**0.896**	**0.965**	**0.983**
DIFFNet(Ours)	**0.102**	**0.746**	**4.461**	**0.178**	**0.896**	**0.965**	**0.983**

Table 8 Performance comparison using enhanced KITTI ground truth. Evaluation of various methods on KITTI 2015 dataset utilizing improved ground truth and the Eigen split [7]. The best results for each metric are highlighted in bold. M: Self-supervised monocular supervision.

Method	Train	The lower is better				The higher is better		
		Abs Rel	Sq Rel	RMSE	RMSE log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Zhou[38]	M	0.176	1.532	6.129	0.244	0.758	0.921	0.971
Monodepth2[10]	M	0.090	0.545	3.942	0.137	0.914	0.983	0.995
R-MSFM6[39]	M	0.088	0.492	3.836	0.135	0.915	0.983	0.995
Johnston[14]	M	0.081	0.484	3.716	0.126	0.927	0.985	0.996
FSRE-Depth[15]	M	0.084	0.436	3.740	0.129	0.919	0.985	0.996
Lite-Mono[34]	M	0.082	0.455	3.683	0.127	0.923	0.985	0.996
HR-Depth[21]	M	0.079	0.421	3.603	0.123	0.928	0.987	**0.997**
DIFFNet[37]	M	0.076	0.414	3.492	0.119	0.936	0.988	0.996
Lite-Mono-8m[34]	M	0.077	0.423	3.527	0.119	0.934	0.988	**0.997**
Ours(tiny)	M	0.076	0.397	3.428	0.118	0.935	0.988	**0.997**
Ours(small)	M	**0.074**	**0.374**	**3.368**	**0.114**	**0.938**	**0.990**	**0.997**

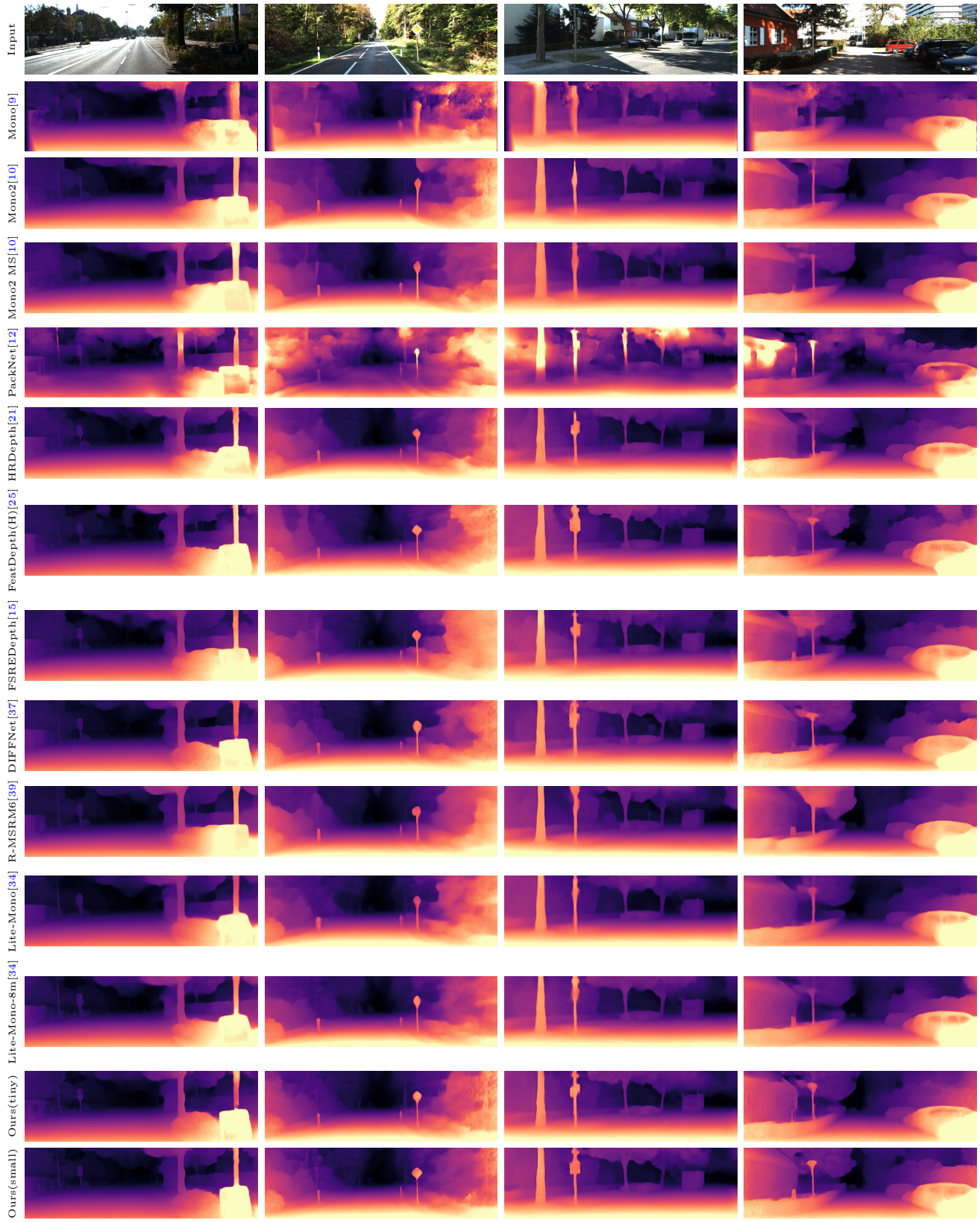

Fig. 14 Additional Qualitative Comparison. Qualitative comparison between our method (last two rows) and other monocular and stereo self-supervised depth estimation techniques. **(H)** denotes training using a dataset with a resolution of 1024×320.

Table 9 Details of MFFENet-Tiny Architecture. Stage denotes the stages of the encoder output. **Output** represents the feature map generated by each stage of the encoder. **K** indicates the kernel size, **S** denotes the stride, **Ch** represents the output channels of each layer, **Dilation** signifies the dilation factor, **Res** denotes the reduction factor relative to the input image, **Input** indicates the input of each layer, **Up()** signifies up-sampling (with a ratio of 2), [·] represents concatenation, and **DeConv** signifies deconvolution.

MFFENet Encoder				
stage	input	output	res	ch
stage1	image	X_0^1	×2	64
		X_1^1	×4	64
stage2	branch1	X_1^2	×4	18
		X_2^2	×8	36
stage3	branch2	X_1^3	×4	18
		X_2^3	×8	36
		X_3^3	×16	72
stage4	branch3	X_1^4	×4	18
		X_2^4	×8	36
		X_3^4	×16	72
		X_4^4	×32	144

MFFENet Decoder							
layer	k	s	ch	dilation	res	input	activation
ASPP	3	1	144	-	×32	$[X_4^4]$	ReLU
Conv1	1	1	144	-	×32	ASPP	-
AM1Conv1	1	1	144	-	×16	$[X_3^3, X_3^4]$	ReLU
AM1CA	-	-	291	-	×16	Up(Conv1)+AM1Conv1+L_5	
AM1Conv2	3	1	256	1	×16	AM1CA	
AM2Conv1	1	1	108	-	×8	$[X_2^2, X_2^3, X_2^4]$	ReLU
AM2CA	-	-	367	-	×8	Up(AM1Conv2)+AM2Conv1+L_4	
AM2Conv2	3	1	128	1	×8	AM2CA	
AM3Conv1	1	1	118	-	×4	$[X_1^1, X_1^2, X_1^3, X_1^4]$	ReLU
AM3CA	-	-	249	-	×4	Up(AM2Conv2)+AM3Conv1+L_3	
AM3Conv2	3	1	64	1	×4	AM3CA	
AM4Conv1	1	1	64	-	×2	$[X_0^1]$	ReLU
AM4CA	-	-	131	-	×2	Up(AM3Conv2)+AM4Conv1+L_2	
AM4Conv2	3	1	32	1	×2	AM4CA	
UpConv1	3	1	16	1	×2	AM4Conv2	ELU
UpConv2	3	1	16	1	×1	Up(UpConv1)+L_1	
FRM1	1	1	128	-	×8	AM2Conv2+Up(AM1Conv2)+L_4	ReLU
FRM2	1	1	64	-	×4	AM3Conv2+Up(FRM1)+L_3	
FRM3	1	1	32	-	×2	AM4Conv2+Up(FRM2)+L_2	
FRM4	1	1	16	-	×1	UpConv2+Up(FRM3)+L_1	
downC1	3	1	1	1	×1	FRM4	-
downC2	3	1	1	1	×2	FRM3	
downC3	3	1	1	1	×4	FRM2	
downC4	3	1	1	1	×8	FRM1	
downC5	3	1	1	1	×16	AM1Conv2	
Disp1	5,3,3	1	1	2,1,1	×1	downC1+DeConv(downC2)+$L_{1(mean)}$	Sigmoid
Disp2	5,3,3	1	1	2,1,1	×2	downC2+DeConv(downC3)+$L_{2(mean)}$	
Disp3	5,3,3	1	1	2,1,1	×4	downC3+DeConv(downC4)+$L_{3(mean)}$	
Disp4	5,3,3	1	1	2,1,1	×8	downC4+DeConv(downC5)+$L_{4(mean)}$	