DFormer++: Improving RGBD Representation Learning for Semantic Segmentation

Bo-Wen Yin, Jiao-Long Cao, Dan Xu, Ming-Ming Cheng, *Senior Member, IEEE*, Qibin Hou, *Member, IEEE*

Abstract—We explore the potential of pretrain-and-finetune manner on the RGB-D semantic segmentation to solve the common mismatch problem in this field. Specifically, we present DFormer++, a novel RGB-D pretrain-and-finetune framework to learn transferable representations for RGB-D semantic segmentation. This paper has two vital innovations. 1) Framework perspective: Different from the existing methods that finetune RGB pretrained backbone to the RGB-D scenes, we pretrain the backbone using image-depth pairs from ImageNet-1K, and hence the model is endowed with the capacity to encode RGB-D representations; 2) Architecture perspective: Our model comprises a sequence of RGB-D attention blocks, which are tailored for encoding both RGB and depth information through a novel attention mechanism. Our DFormer++ avoids the mismatched encoding of the 3D geometry relationships in depth maps by RGB pretrained backbones, which widely lies in previous works but has not been resolved. Meanwhile, the tailored architecture greatly reduces redundant parameters for encoding RGB-D data and achieves efficient and accurate perception. Experimental results show that our DFormer++ achieves new cutting-edge performance on three popular RGB-D semantic segmentation benchmarks. Our code is available at: https://github.com/VCIP-RGBD/DFormer.

Index Terms—Depth; RGB-D representation learning; pretraining-and-finetuning; attention mechanism.

1 Introduction

With the widespread use of 3D sensors, RGB-D data is becoming increasingly available to access. The geometric information within the depth maps is of great benefit for distinguishing instances and context. The RGB-D research is vital for robust high-level scene understanding. Meanwhile, RGB-D data also presents considerable potential in a large number of applications, *e.g.*, SLAM [50], automatic driving [30], and robotics [39]. Therefore, RGB-D research has attracted great attention in the past few years.

Current mainstream RGB-D methods adopt RGB pretrained backbone to process the RGB-D scenes. The left part of Fig. 1 shows the pipeline of this type of approach. As can be seen, two individual RGB pretrained backbones are used to extract the features of the RGB images and depth maps. Interactions between the features of RGB and depth maps are performed between the two backbones. Current state-of-the-art methods [53], [62] have achieved excellent performance on several RGB-D benchmark datasets, but there are still three important issues that cannot be ignored. i) Mismatch between pretraining and finetuning: The backbone in the RGB-D downstream tasks takes an image-depth pair as input, inconsistent with the pretraining process that takes only RGB images as input, causing a huge representation distribution shift; ii) Interference on the backbone: Interactions are densely performed between the RGB branch and depth branch during finetuning, which may destroy the representation distribution when using the RGB pretrained backbone. iii) Redundant parameters and computation cost. The dual backbones in RGB-D networks bring more computational cost compared to the standard RGB methods, which is not efficient. We argue that the main reason for these issues is that depth clues are not considered during pretraining.

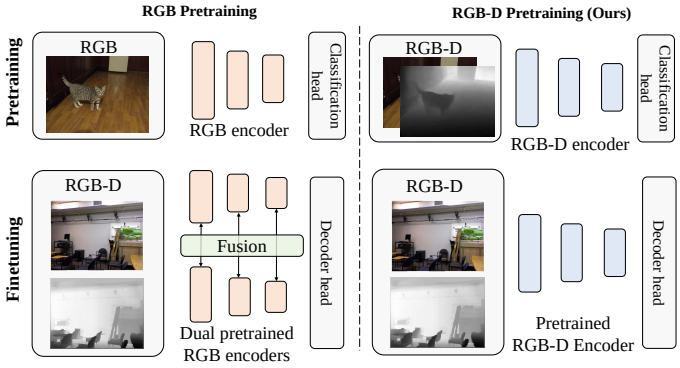


Fig. 1. Comparisons between the existing popular pretrain-and-finetune pipeline and ours for RGB-D segmentation. **RGB pretraining**: Recent mainstream methods adopt two RGB pretrained backbones to separately encode RGB and depth information and fuse them. **RGB-D pretraining**: The RGB-D backbone in DFormer++ learns transferable RGB-D representations during pretraining and then is finetuned for segmentation.

Taking the above issues into account, a straightforward question arises: Is it possible to specifically design an RGB-D pretraining framework to eliminate this gap? This motivates us to present a novel RGB-D pretraining framework in our

- *B.-W. Yin, J.-L. Cao, M.-M. Cheng, Q. Hou are with VCIP, School of Computer Science, Nankai University, Tianjin, China (bowenyin@mail.nankai.edu.cn).*
- *D. Xu is with HKUST.*
- *Q. Hou is the corresponding author (houqb@nankai.edu.cn).*
- *This work was partially funded by National Key Research and Development Project of China (No. 2024YFE0100700), NSFC (No. 62495061, 62522607, 62276145), the Science and Technology Support Program of Tianjin, China (No. 23JCZDJC01050), and the Fundamental Research Funds for the Central Universities (Nankai University) under Grant 070-63253220.*
- *The previous version of this paper has been published in ICLR 2024 [60].*

Manuscript received March 1, 2022; revised August 26, 2022.

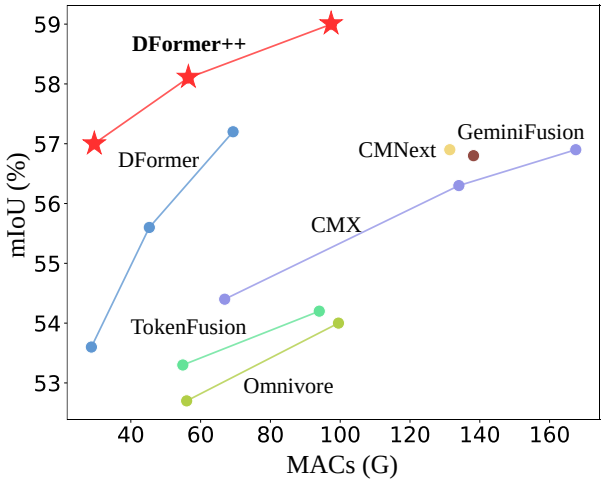


Fig. 2. Performance *vs.* computational cost on the NYU Depth v2 dataset [45]. DFormer++ achieves the state-of-the-art 59.0% mIoU and the best trade-off compared to other methods.

conference version [60], as shown in the right part of Fig. 1. During the pretraining period, we consider taking image-depth pairs[1], rather than relying solely on RGB images, as input and propose to build interactions between RGB and depth features within the building blocks of the encoder. In this way, the inconsistency issue between the mismatch inputs of pretraining and finetuning can be naturally avoided.

In this paper, we further analyze the RGB-D pretraining framework through additional experiments, including investigating the effect of different depth estimation methods for the pretraining and exploring the rationale for incorporating depth information. To alleviate the heavy computation burdens of processing RGB-D scenes, our conference version presents an RGB-D block consisting of global, local, and base modules to construct an efficient and effective RGB-D model. This work improves the hybrid fusion block to a unified depth-guided attention block to construct our DFormer++. We leverage depth features to guide the enhancement of RGB features, enabling simultaneous RGB-D feature fusion, thereby eliminating the need for a separate RGB-only encoding module.

DFormer++ demonstrates further improvements compared to the conference version. We observe that the depth information only needs a small portion of channels to encode. There is no need to use a whole pretrained backbone to extract depth features as done in previous works. Since the feature interactions between RGB images and depth images begin during the pretraining stage, the efficiency of these interactions can be largely improved compared to previous works, as demonstrated in Fig. 2. To track recent developments in the field and make fair and reasonable comparisons, we also introduce and discuss more recent published algorithms.

To validate the effectiveness of DFormer++, we perform extensive experiments on three widely-used RGB-D semantic segmentation benchmarks, *i.e.*, NYU Depth v2 [45], SUN-RGBD [46], and Stanford2D3D [2]. Compared to the conference version, we expand our experiments to include Stanford2D3D [2], a large-scale RGB-D segmentation dataset, to further validate the effectiveness of our method. By integrating a lightweight decoder on top of our pretrained RGB-D backbone, the proposed DFormer++ achieves new state-of-the-art records with less computational cost compared to previous methods. Notably, our largest model, DFormer++-Base, achieves a result of 59.0% mIoU on NYU Depth v2 while maintaining lower computational cost. Meanwhile, our lightweight model, DFormer++-Tiny, achieves 57.0% mIoU on NYU Depth v2 with merely 17.3M parameters and 29.5G MACs. Among all the benchmarks, our approach achieves the best trade-off between segmentation performance and computations compared to other recent models.

To sum up, our main contributions can be summarized as follows:

- We present a novel RGB-D pretraining framework, termed DFormer++, to alleviate the mismatch issue between pretraining and finetuning and avoid the distribution shift.
- We propose a depth guided attention module to fuse the RGB and depth features to provide transferable representations for RGB-D downstream tasks.
- We found that in our framework it is enough to use a small portion of channels to encode the depth information compared to RGB features, an effective way to reduce model size.

A preliminary version of this work appeared in [60]. In this journal version, we make a substantial revision along four axes: (1) Architecture. We replace the three-branch CNN–Transformer hybrid fusion block with a unified depth-guided attention that fuses and encodes within attention, removing the separate RGB-only branch and reducing redundancy while improving the accuracy–efficiency trade-off; (2) Technical details and analyses. We provide the full RGB-D pretraining protocol (data construction, objectives, schedules), systematically compare three decoding manners, and add distribution-shift diagnostics to clarify design choices; (3) Expanded experiments. We evaluate on large-scale and diverse real-world datasets (e.g., Stanford2D3D [2], Cityscapes [14]), test RGB–LiDAR and RGB–Thermal variants, and study pretraining with multiple depth estimators; (4) Updated comparisons and deeper analysis. We incorporate more recent state-of-the-art methods and provide more ablations and visualizations. Overall, the proposed framework provides a more principled fusion mechanism, richer technical exposition, and broader empirical validation than the conference version.

2 Related Work

2.1 RGB-D Semantic Segmentation

Recently, the rise of deep learning technologies (*e.g.*, CNNs [26], and Transformers [33], [48]) has made significant progress in scene parsing [57], [59], [64], one of the most important pursuits of computer vision. The current deep learning methods achieve excellent performance on various scene perception tasks, such as classification [15], [27], [36], [37], semantic segmentation [13], [22] and object

1. For depth maps, we employ a popular depth estimation model [58] to predict the depth map for each RGB image, which we found works well.

detection [11]. However, in the challenging scenes from the real world [34], [47], they still struggle to obtain accurate perception, as they only focus on RGB images that provide them with distinct colors and textures but neglect 3D geometric information. Considering these limitations within visual images, researchers integrate images with depth maps for a more comprehensive and robust understanding of real-world scenes.

Semantic segmentation aims to produce per-pixel category prediction across a given scene. Given the supplementary depth maps, researchers have conducted a lot of investigation in building fusion modules to bridge the RGB and depth features extracted by two parallel pretrained backbones to achieve the interaction and alignment between them. For instance, methods like CMX [61], Sigma [49], and Geminifusion [31] dynamically fuse RGB-D representations in RGB and depth encoders and aggregate them in the decoder. In the RGB-D semantic segmentation task, a series of works introduces the evolution of fusion manners and significantly extends the performance boundaries. Nevertheless, they still face three common issues: (1) mismatching problem in pretraining and finetuning; (2) interference on the pretrained RGB backbone; (3) redundant computational burden caused by the dual backbone architecture.

Another line of methods focuses on the design of operators [6], [8], [51], [56] to encode complementary information from images and depth maps. For example, methods like ShapeConv [6], and SGNet [8] propose depth-aware convolutions, which enable efficient RGB features and 3D spatial information integration to largely enhance the capability of perceiving geometry. These methods are efficient, but the improvement is usually limited due to the insufficient utilization of the 3D geometry information involved in the depth modal. In general, these two mainstream paradigms are not sufficient to process RGB-D scenes efficiently and accurately.

2.2 Multi-Modal Learning

The pretrain-and-finetune paradigm, which has achieved great success in natural language processing and computer vision, has now been extended to the multi-modal domain. The resulting transferable representations have demonstrated exceptional performance across a broad range of downstream tasks. Existing multi-modal learning methods encompass a wide array of modalities, such as image and text [7], [12], [40], [55], [63], text and video [1], text and 3D mesh [65], image, depth, and video [21]. From the modality encoding perspective, these methods can be mainly categorized into two groups, *i.e.*, multi- and joint-encoder ones. Specifically, the multi-encoder methods exploit multiple encoders to independently project the inputs in different modalities into a common space and minimize the distance or perform representation fusion between them. For example, methods like CLIP [40] and VATT [1] employ several individual encoders to embed the representations in different modalities and align them via a contrastive learning strategy. In contrast, the joint-encoder methods simultaneously input different modalities and use a multi-modal encoder based on attention mechanisms to model joint representations. For instance, MultiMAE [3] adopts

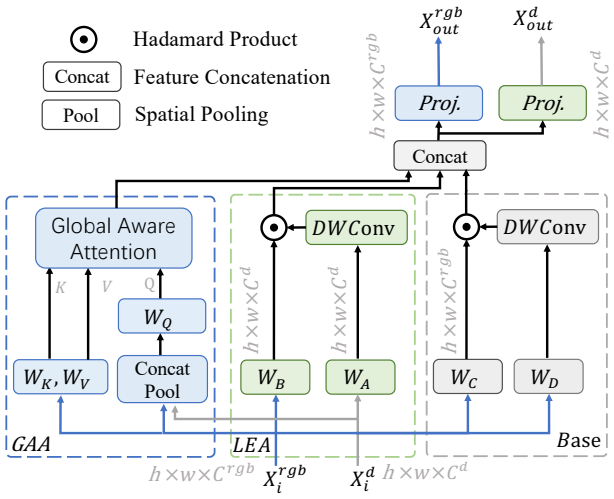


Fig. 3. Detailed structure of the RGB-D block in our conference version.

a unified transformer to encode the tokens with a fixed dimension that are linearly projected from a small subset of randomly sampled multi-modal patches and multiple task-specific decoders to reconstruct their corresponding masked patches by attention mechanisms separately. Earlier works such as ESANet [43], EMSANet [42], and EMSAFormer [18] have explored RGB-D pretraining or multi-task learning for semantic segmentation. However, these approaches typically rely on fully synthetic RGB-D corpora with a pronounced domain gap to real indoor scenes, or pseudo or weak modality pairings or comparatively small-scale, task-specific pretraining regimes that limit generalization.

In this paper, we propose DFormer++, a novel framework that achieves RGB-D representation learning in a pretraining manner to maintain consistency within pretraining and finetuning. We are among the first to encourage the semantic cues from RGB and depth modalities to align together by the explicit supervision signals of classification, yielding transferable representations for RGB-D downstream tasks.

2.3 Multi-Modal Fusion

The process of RGB-D multi-modal fusion involves harnessing 3D geometry information within depth maps to enrich associated details, thereby exceeding the capabilities of their unimodal counterparts. As attention mechanism has achieved great success in various vision tasks [28], [44], recently, using attention-based methods to perform multi-modal fusion has become a new trend. ACNet [29] proposes an Attention Complementary Module (ACM) to enable the multimodal fusion between the RGB and depth branches. CMX [61] designs a Cross-Modal Feature Rectification Module to build connections between the RGB branch and depth branch and deploys a Feature Fusion Module to perform the exchange of long-range contexts before the decoder. GeminiFusion [31] proposes a pixel-wise fusion approach that capitalizes on aligned cross-modal representations. In this paper, we propose a depth guided attention block that performs the multi-modal fusion at local and global regions. This feature enables each token on the feature map to utilize the geometry clues to identify which object each

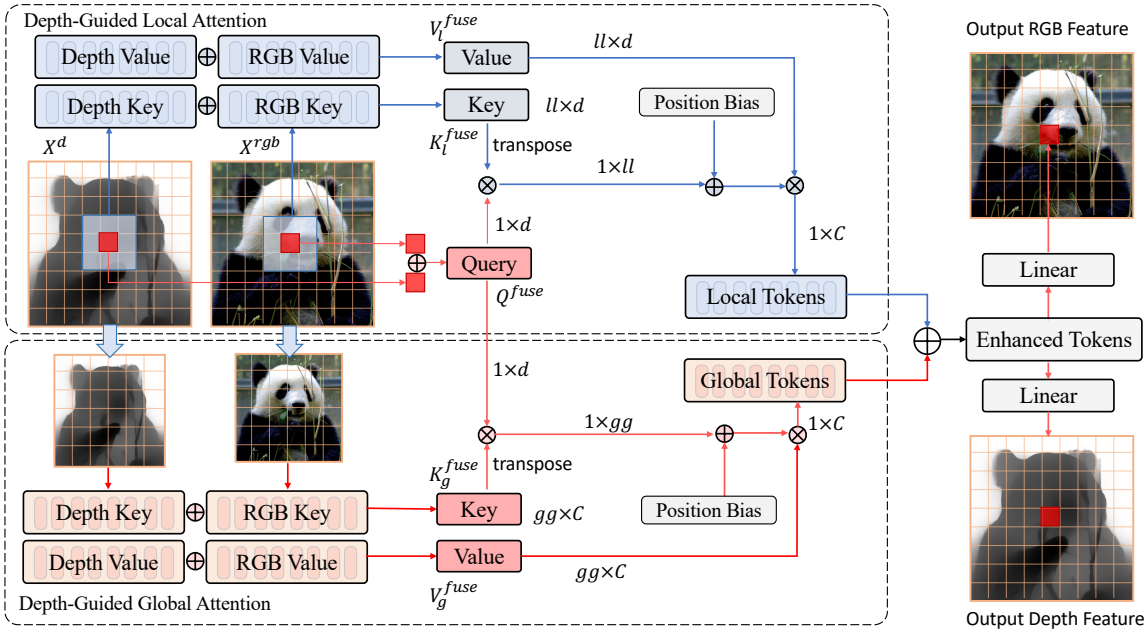


Fig. 4. Diagrammatic details on how to conduct interactions between RGB and depth features. 'l' is the local window size and 'g' is the global pooling size.

pixel belongs to in the local regions and capture the object semantics in the global regions.

3 DFormer++

In this section, we first revisit the building block of our conference version in Sec. 3.1 and then describe the core component of our model, *i.e.*, the proposed depth guided attention mechanism, in Sec. 3.2. Then, we present the overall architecture of our RGB-D encoder in Sec. 3.3. Finally, we present the details of our RGB-D pretraining and how to finetune the pretrained model to downstream tasks in Sec. 3.4 and Sec. 3.5.

3.1 Revisiting DFormer Block

The building block within DFormer is mainly composed of the global awareness attention (GAA) module, the local enhancement attention (LEA) module, and a base module, as shown in Fig. 3. GAA incorporates depth information and aims to enhance the capability of object localization from a global perspective, while LEA adopts a large-kernel convolution to capture the local clues from the depth features, which can refine the details of the RGB representations. Different from the self-attention mechanism [48] that introduces quadratic computation growth as the pixels or tokens increase, the Query (Q) in GAA is down-sampled to a fixed size $k \times k$ and hence the computational complexity can be reduced. So, Q comes from the concatenation of the RGB features X^{rgb} and depth features X^d, while key (K) and value (V) are extracted from RGB features. Based on the generated $Q \in \mathbb{R}^{k \times k \times C}$, $K \in \mathbb{R}^{h \times w \times C}$, and $V \in \mathbb{R}^{h \times w \times C}$, where h and w are the height and width of features in the current stage, C is the channel number, we formulate the GAA as follows:

$$X_{GAA} = \mathrm{UP}(V \cdot \mathrm{Softmax}(\frac{Q^\top K}{\sqrt{C^d}})), \quad (1)$$

where $\mathrm{UP}(\cdot)$ is a bilinear upsampling operation that converts the spatial size from $k \times k$ to $h \times w$.

The LEA module captures local details and is regarded as a supplement to the GAA module. Unlike most previous works that use addition or concatenation to fuse the RGB features and depth features, we conduct a depth-wise convolution with a large kernel on the depth features and use the resulting features as attention weights to reweigh the RGB features via a simple Hadamard product inspired by [27]. The calculation process of can be defined as follows:

$$X_{LEA} = \mathrm{DConv}_{k' \times k'}(\mathrm{Linear}(X^d)) \odot \mathrm{Linear}(X^{rgb}), \quad (2)$$

where $\mathrm{DConv}_{k' \times k'}$ is a depth-wise convolution with kernel size $k^{'} \times k^{'}$ and '$\odot$' is the Hadamard product. To preserve the diverse appearance information, we also build a base module to transform the RGB features X^{rgb} to X_{Base}:

$$X_{Base} = \mathrm{DConv}_{k' \times k'}(\mathrm{Linear}(X^{rgb})) \odot \mathrm{Linear}(X^{rgb}), \quad (3)$$

where the resulting features X_{Base} has the same spatial size as X_{GAA} and X_{LEA}. Finally, these three features are fused together by concatenation and linear projection to generate the output features.

While this RGB-D block shows good performance by leveraging cross-attention for global interaction, convolutions for local modality fusion, and a dedicated RGB branch for preserving color features, its multi-branch design introduces computational redundancy and may limit the full potential of depth-guided feature learning. To address these limitations, we propose a unified depth guided attention that can better achieve local and global interactions. Using depth information as a guide signal, our approach enables the attention mechanism within our DFormer++ to dynamically focus on spatially relevant regions, thus further enhancing overall segmentation accuracy and efficiency.

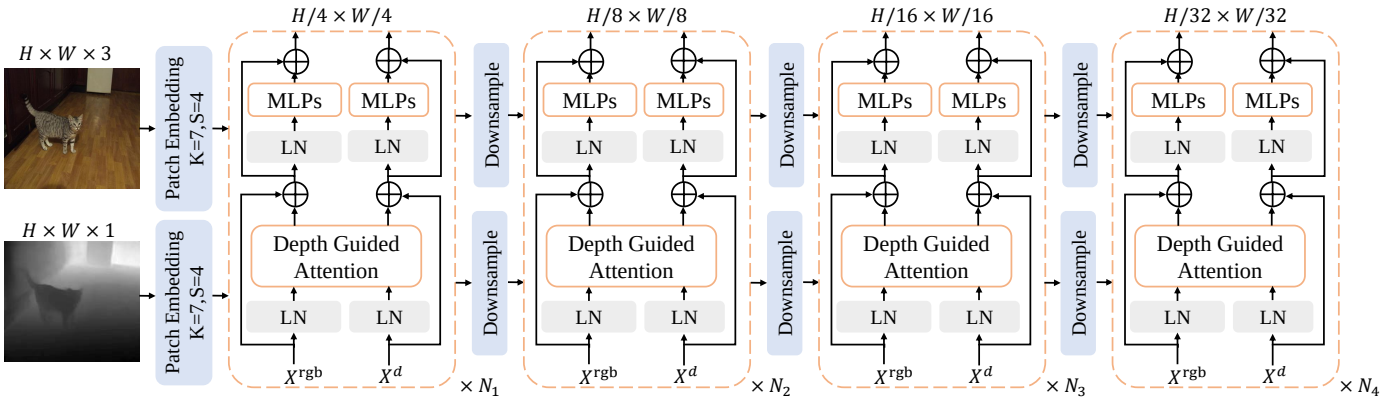


Fig. 5. Overall architecture of the proposed DFormer++. First, we use the pretrained DFormer++ to encode the RGB-D data. Then, the features from the last three stages are concatenated and delivered to a lightweight decoder head for final prediction. Note that only the RGB features from the encoder are used in the decoder.

3.2 Depth Guided Attention

Leveraging the complementary roles of depth at different granularities, our method enables RGB–D interactions between fine-grained local tokens and summarized global tokens within a unified attention block. At the local level, depth provides precise geometric cues that sharpen object boundaries and near-field spatial relations, which is crucial for separating adjacent objects or surfaces with similar appearance but different depths. At the global level, depth captures the overall scene layout and high-level semantics, supplying long-range contextual constraints that disambiguate category assignments. By jointly integrating local geometry and global structure, the proposed fusion better exploits depth as a complementary signal to RGB, thereby improving multi-modal representation quality and yielding stronger scene understanding performance without the need of base modules in the conference version.

Our attention mechanism is composed of a depth guided local attention and a depth guided global attention, as shown in Fig. 4. In each block, given the input RGB features X^{rgb} with C channels and depth features X^d with C^d channels, we now focus on the operations conducted on a single token for the two modalities. In the depth guided local attention module (DGLA), the calculation is performed in a local window with a size of $l \times l$ which is centered at the focused token. We project RGB and depth features to obtain the Q, K, V embeddings for RGB and depth, denoted as Q^{rgb}, Q^d, K^{rgb}, K^d, V^{rgb}, V^d. The local windows in the K and V embeddings are denoted as K_l^{rgb}, K_l^d, V_l^{rgb}, V_l^d, respectively. Linear projections preserve the dimensionality C for RGB features but transform depth features from C^d to C, enabling seamless multi-modal fusion. Here, we empirically found that a simple addition operation performs well for fusing the embeddings from the two modalities. Thus, the fused embeddings are computed as: $Q^{fuse} = Q^{rgb} + Q^d$, $K_l^{fuse} = K_l^{rgb} + K_l^d$, $V_l^{fuse} = V_l^{rgb} + V_l^d$. Based on the generated Q^{fuse}, K_l^{fuse} and V_l^{fuse}, we formulate the DGLA as follows:

$$X_{DGLA} = V_l^{fuse} \cdot \text{Softmax}(\frac{{Q^{fuse}}^\top K_l^{fuse}}{\sqrt{C}} + B), \quad (4)$$

where B means the learnable position bias.

In the depth guided global attention module (DGGA), the calculation is conducted between the focused token and all the coarse-grained tokens. The RGB and depth tokens X^{rgb} and X^d are down-sampled to form the coarse-grained tokens. Then we use the coarse-grained RGB and depth tokens to generate the global Key and Value, *i.e.*, K_g^{rgb}, K_g^d, V_g^{rgb}, and V_g^d. The fused global embeddings can be generated as follows: $K_g^{fuse} = K_g^{rgb} + K_g^d$, $V_g^{fuse} = V_g^{rgb} + V_g^d$. The DGGA can be formulated as follows:

$$X_{DGGA} = V_g^{fuse} \cdot \text{Softmax}(\frac{{Q^{fuse}}^\top K_g^{fuse}}{\sqrt{C}} + B). \quad (5)$$

Finally, we add the local tokens X_{DGLA} and global tokens X_{DGGA} together and project the resulting tokens to generate the output depth and RGB tokens.

Complexity analysis. Let N be the number of spatial tokens, C and C_d the channel dimensions for RGB and depth, $l \times l$ the local window size in DGLA, and G the number of pooled global tokens in DGGA (with $G \ll N$ and l fixed across resolutions). For each query, DGLA attends to l^2 local tokens (cost $O(C\,l^2)$) and DGGA attends to G global tokens (cost $O(C\,G)$). Across all N queries, the joint attention cost is:

$$O\big(N\,C\,(l^2 + G)\big)\,,$$

which is linear in N since l^2 and G are resolution-independent constants. The attention map memory is likewise $O(N(l^2 + G))$. Modality-specific Q/K/V projections introduce $O\big(N\,(C^2 + CC_d)\big)$ multiply-adds, shared by all attention variants and thus omitted for relative comparison. It illustrate advantages over the prior works: (1) Concatenation + global self-attention over $(2N)$ fused tokens: $O\big((2N)^2C\big) = O(4N^2C)$ (quadratic). (2) Bidirectional cross-attention (RGB↔Depth): two passes of $O(N^2C)$ each $\Rightarrow$ $O(2N^2C)$ (quadratic). (3) Local-only fusion (no global context): $O(NCl^2)$ (linear but context-limited). (4) Full global self-attention per modality + late fusion: $2 \times O(N^2C)$ + fusion overhead (quadratic). Our DGLA+DGGA attains a hybrid receptive field (fine local + coarse global) while remaining $O\big(NC(l^2 + G)\big)$-a strict reduction from quadratic baselines without discarding global cues.

Advantage and feature analysis. We replace the three-branch CNN–Transformer hybrid with a unified depth-

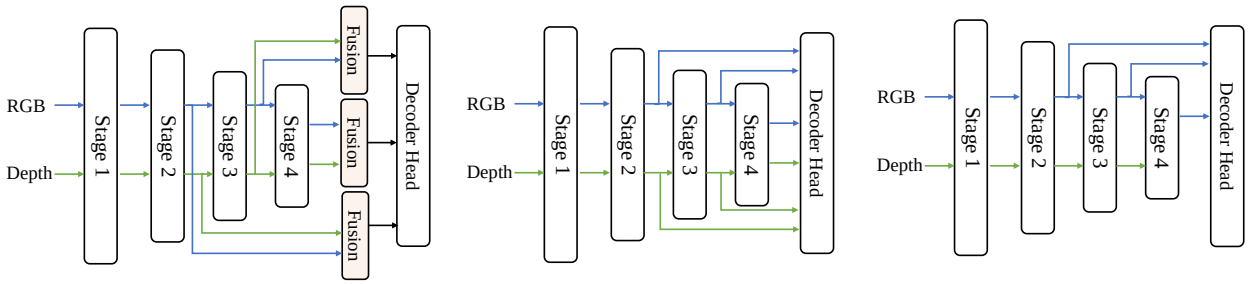


Fig. 6. Three different decoding manners. Left: RGB and depth features that generated at the last three stages of the encoder are fused together and then sent to the decoder. Middle: RGB and depth features are sent to the decoder. Right: Only RGB features are sent to the decoder.

guided attention that projects RGB/depth to Q/K/V and fuses once inside attention, removing the RGB-only base branch and conv fusion while residual routes and adaptive weights preserve RGB fidelity and down-weight depth when unnecessary. With fixed size, the paths have linear pixel complexity; addition avoids concat's channel inflation/projection and Hadamard's gating/conv overhead, and allocating fewer depth channels reduces activations—yielding a more identifiable, optimization-stable module with a superior accuracy–efficiency trade-off. In our attention module, the depth clues are utilized at fine-grained local and coarse-grained global regions to help distinguish which object each pixel belongs to and perceive the geometry relationship between different objects. This feature enables our model to utilize depth maps from different perspectives and achieve more accurate scene perception. Besides, all the adopted fusion operations including embeddings addition, DGLA in a fixed window, and DGGA in a fixed pool size are performed in a linear complexity mode. With our efficient and effective RGB-D attention modules, the model can conduct information interaction in each block with an affordable computation burden.

3.3 Overall Architecture

Fig. 5 illustrates the overall architecture of our DFormer++. The hierarchical encoder is composed of four stages, which are utilized to generate multi-scale RGB-D features. Each stage contains a stack of depth guided attention blocks. Two convolutions with kernel size 3×3 and stride 2 are used to down-sample RGB and depth features, respectively, between two consecutive stages.

Given an RGB image and the corresponding depth map with spatial size $H \times W$, they are first separately processed by two parallel stem layers consisting of two convolutions with kernel size 3×3 and stride 2. Then, the RGB features and depth features are fed into the hierarchical encoder to encode multi-scale features at $\{1/4, 1/8, 1/16, 1/32\}$ of the original image resolution. In stage 4, as the feature map size has been reduced to $\frac{H}{32} \times \frac{W}{32}$, the feature pooling module cannot function properly. We perform the multi-modal fusion at the whole regions without pooling, the modification in the last stage is widely adopted in perception models [52]. Next, we pretrain this encoder using the image-depth pairs from ImageNet-1K using the classification objective to generate the transferable RGB-D representations. Finally, we send the visual features from the pretrained RGB-D encoder to the decoder to produce predictions, *e.g.*, segmentation maps with a spatial size of $H \times W$.

We empirically observe that encoding depth features requires fewer parameters compared to the RGB ones due to their less semantic information, which is verified and illustrated in detail in the supplementary materials. To reduce model complexity in our RGB-D block, we use a small portion of channels to encode the depth information. Based on the configurations of the RGB-D blocks in each stage, we design a series of DFormer++ encoder variants, termed DFormer++-Tiny, DFormer++-Small, and DFormer++-Base, respectively, with the same architecture but different model sizes. For detailed configuration, readers can refer to supplementary. DFormer++-Tiny is a lightweight encoder for fast inference, while DFormer++-Base is the largest one for receiving better performance.

3.4 RGB-D Pretraining

The purpose of RGB-D pretraining is to endow the backbone with the ability to achieve the interaction between RGB and depth modalities and generate transferable representations with rich semantic and spatial information. To this end, we first apply a depth estimator, *e.g.*, Depth anything [58], on the ImageNet-1K dataset [41] to generate a large number of image-depth pairs. Then, we add a classification head on the top of the RGB-D encoder to build the classification network for pretraining. Particularly, the RGB features from the last stage are flattened along the spatial dimension and fed into the classifier head. In addition, we empirically found that the different depth estimators for generating depth maps have little effect on the downstream tasks after finetuning. The details are given in the experiment section. The standard cross-entropy loss is used as our optimization objective, and the network is pretrained on RGB-D data for the commonly used pretraining durations, *i.e.*, 300 epochs. Following previous works [23], [37], the AdamW optimizer [38] with learning rate 1e-3 and weight decay 5e-2 is used and the batch size is set to 1024.

3.5 Task-Specific Decoder

In RGB-D segmentation models [31], [49], [61], [62], their encoders are mostly pretrained on ImageNet. To capture the multi-level features, a decoder is usually necessary, which is built upon the encoder. In this paper, we explore three decoding paradigms on the encoder features, which have been shown in Fig. 6. The first paradigm, mostly adopted in RGB-D segmentation models like CMX [61], first fuse the RGB

TABLE 1
Results on NYU Depth v2 [45] and SUN-RGBD [46]. We split the models to three sets according to the parameters. Some methods do not report the results or settings on the SUN-RGBD datasets, so we reproduce them with the same training config. † indicates our implemented results. All the backbones are pre-trained on ImageNet-1K. We calculate the MACs for all the models using the same testing code.

Model	Backbone	Params	NYU Depth v2			SUN-RGBD			Code
			Input size	MACs	mIoU	Input size	MACs	mIoU	
Omnivore$_{22}$ [21]	Swin-T	28.9M	480 × 640	28.6G	49.7	530 × 730	—	—	Link
TokenFusion$_{22}$ [53]	MiT-B2	26.0M	480 × 640	54.9G	53.3	530 × 730	70.7G	50.3†	Link
DFormer-T$_{24}$ [60]	DFormer-T	6.0M	480 × 640	14.7G	51.8	530 × 730	18.9G	48.8	Link
DFormer-S$_{24}$ [60]	DFormer-S	18.7M	480 × 640	28.7G	53.6	530 × 730	37.0G	50.0	Link
DFormer-B$_{24}$ [60]	DFormer-B	29.5M	480 × 640	45.3G	55.6	530 × 730	58.4G	51.2	Link
DFormer++-Tiny	DFormer++-T	17.3M	480 × 640	29.5G	**57.0**	530 × 730	39.3G	**51.5**	Link
ESANet$_{21}$ [43]	ResNet-34	47.0M	480 × 640	45.1G	50.3	480 × 640	34.9G	45.0	Link
EMSANet$_{22}$ [42]	ResNet-34	46.9M	480 × 640	45.1G	51.0	530 × 730	58.7G	48.4	Link
TokenFusion$_{22}$ [53]	MiT-B3	45.9M	480 × 640	94.0G	54.2	530 × 730	121.6G	51.0†	Link
Omnivore$_{22}$ [21]	Swin-S	50.6M	480 × 640	56.0G	52.7	530 × 730	—	—	Link
DFormer-L$_{24}$ [60]	DFormer-L	39.0M	480 × 640	69.3G	57.2	530 × 730	89.2G	52.5	Link
AsymFormer$_{24}$ [16]	B0+Conv-T	33.0M	480 × 640	39.4G	55.3	530 × 730	52.6G	49.1	Link
Sigma$_{24}$ [49]	VMamba-T	48.3M	480 × 640	89.5G	53.9	480 × 640	89.5G	50.0	Link
DFormer++-Small	DFormer++-S	37.2M	480 × 640	56.4G	**58.1**	530 × 730	74.6G	**52.7**	Link
ACNet$_{19}$ [29]	ResNet-50	115.6M	480 × 640	115.6G	48.3	530 × 730	150.6G	48.1	Link
SGNet$_{20}$ [8]	ResNet-101	64.7M	480 × 640	108.5G	51.1	530 × 730	151.5G	48.6	Link
SA-Gate$_{20}$ [9]	ResNet-101	110.9M	480 × 640	192. 6G	52.4	530 × 730	248.7G	49.4	Link
CEN$_{20}$ [54]	ResNet-101	118.2M	480 × 640	617.9G	51.7	530 × 730	789.2G	50.2	Link
CEN$_{20}$ [54]	ResNet-152	133.9M	480 × 640	663.3G	52.5	530 × 730	848.3G	51.1	Link
ShapeConv$_{21}$ [6]	ResNext-101	106.8M	480 × 640	168.4G	51.3	530 × 730	217.8G	48.6	Link
FRNet$_{22}$ [68]	ResNet-34	87.8M	480 × 640	109.5G	53.6	530 × 730	142.7G	51.8	Link
PGDENet$_{22}$ [67]	ResNet-34	107.4M	480 × 640	162.9G	53.7	530 × 730	212.2G	51.0	Link
MultiMAE$_{22}$ [3]	ViT-B	95.4M	640 × 640	403.8G	56.0	640 × 640	403.8G	51.1†	Link
Omnivore$_{22}$ [21]	Swin-B	90.1M	480 × 640	99.5G	54.0	530 × 730	—	—	Link
CMX$_{22}$ [61]	MiT-B2	66.6M	480 × 640	66.9G	54.4	530 × 730	85.9G	49.7	Link
CMX$_{22}$ [61]	MiT-B4	139.9M	480 × 640	134.0G	56.3	530 × 730	173.2G	52.1	Link
CMX$_{22}$ [61]	MiT-B5	181.1M	480 × 640	167.5G	56.9	530 × 730	216.9G	52.4	Link
CMNeXt$_{23}$ [62]	MiT-B4	116.6M	480 × 640	131.4G	56.9	530 × 730	169.6G	51.9†	Link
GeminiFusion$_{24}$ [31]	MiT-B3	75.8M	480 × 640	138.2G	56.8	530 × 730	179.0G	52.7	Link
Sigma$_{24}$ [49]	VMamba-S	69.8M	480 × 640	138.9G	57.0	480 × 640	138.9G	52.4	Link
DFormer++-Base	DFormer++-B	66.7M	480 × 640	97.5G	**59.0**	530 × 730	129.0G	**53.0**	Link

and depth features at each stage and then send the fused features to the decoder. The second paradigm, adopted in MultiMae [3], sends the RGB and depth features from the encoder to the decoder together. Our DFormer++ only sends the enhanced RGB features to the decoder as shown in the right part of Fig. 6. Through experiments, we found that our decoder is the most efficient decoding manner and is adequate to achieve good perception of the RGB-D scenes with our DFormer++ encoder. Specifically, our decoder only uses the X^{rgb} features, while other methods [53], [61], [62] mostly design modules that fuse both modalities features X^{rgb} and X^{d} for final predictions. We argue that our DFormer++ achieves better interaction between RGB-D clues compared to previous works and there is no need to incorporate depth features in the decoder. We show that our X^{rgb} features can efficiently extract the 3D geometry clues from the depth modality thanks to our powerful RGB-D pretrained encoder in our experiments. Delivering the depth features X^{d} to the decoder is not necessary.

For the applications of our DFormer++ to downstream tasks, we adopt the right decoding paradigm in Fig. 6 and use a lightweight decoder head on the top of the pretrained RGB-D backbone to build the task-specific network. After finetuning on corresponding benchmark datasets, the task-specific network is able to generate great predictions, without using extra designs, like fusion modules [9], [61].

TABLE 2
Results on Stanford2D3D [2] dataset.

Model	Backbone	Params	MACs	mIoU
Depth-aware CNN [51]	VGG-16	47.0M	–	39.5
MMAF-Net-152 [19]	ResNet-152	122.3M	134.4G	52.9
Shapeconv-101 [6]	ResNet-101	106.8M	126.3G	60.6
CMX [61]	MiT-B2	66.6M	50.2G	61.2
CMX [61]	MiT-B4	139.9M	100.5G	62.1
DFormer++-T	Ours-T	17.3M	21.5G	60.6
DFormer++-S	Ours-S	37.2M	41.3G	62.1
DFormer++-B	Ours-B	66.7M	72.1G	63.8

Specifically, we adopt a lightweight Hamburger head [20] to aggregate the multi-scale RGB features from the last three stages of our pretrained encoder.

4 Experiments

4.1 RGB-D Semantic Segmentation

Datasets & implementation details. Following the common experiment settings of RGB-D semantic segmentation methods [22], [57], we finetune and evaluate our DFormer++ on three widely used datasets, *i.e.*, NYU Depth v2 [45], SUN-RGBD [46], and Stanford2D3D [2]. NYU Depth v2 contains 1,449 RGB-D images with size 480 × 640, divided into 795 training images and 654 test images with annotations on 40 semantic categories. SUN-RGBD has 10,335 RGB-D

TABLE 3
Comparison with other state-of-the-art methods on Cityscapes [14]. We adopt the same training and testing settings as CMX [61].

Depth estimtor	**Params**	**MACs**	mIoU
ESANet [43]	46.9M	45.1G	80.0
SA-Gate-ResNet101 [9]	110.9M	192.6G	81.7
CMX-B2 [61]	66.6M	67.6G	81.6
CMX-B4 [61]	139.9M	134.3G	82.6
DFormer++-T	17.3M	29.5G	81.8
DFormer++-S	37.2M	56.4G	83.0
DFormer++-L	66.7M	97.5G	**84.8**

images with 37 classes and 5,285/5,050 for training/test. Stanford2D3D is a large-scale dataset and has 70,496 RGB-D images with 13 object categories. Following the data splitting [6], [61], areas of 1,2,3,4,6 in Stanford2D3D are used for training, and area 5 is for test. During finetuning, we only adopt two common data augmentation strategies, *i.e.*, random horizontal flipping and random scaling (from 0.5 to 1.75). The training images are cropped and resized to 480×640, 480×480, and 480×480, respectively, for NYU Depth v2, SUN-RGBD, and Stanford2D3D benchmarks. Cross-entropy loss is utilized as the optimization objective. We use AdamW [32] as our optimizer with an initial learning rate of 6e-5 and the poly decay schedule. Weight decay is set to 1e-2. During test, we employ mean Intersection over Union (mIoU), which is averaged across semantic categories, as the primary evaluation metric to measure the segmentation performance. Following recent works [53], [61], [62], we adopt multi-scale (MS) flip inference strategies with scales $\{0.5, 0.75, 1, 1.25, 1.5\}$ for NYU Depth v2 and SUN-RGBD and use single scale inference for Stanfor2D3D.

Comparisons with state-of-the-art methods. We compare our DFormer++ with 18 recent RGB-D semantic segmentation methods on the NYU Depth v2 [45], SUN-RGBD [46], Stanford2D3D [2] datasets. These methods are chosen according to three criteria: a) recently published, b) representative, and c) with open-source code. As shown in Tab. 1, our DFormer++ achieves new state-of-the-art performance across these two benchmark datasets. DFormer++ also achieves edge-cutting performance on Stanford2D3D, as shown in Tab. 2. We also plot the performance-efficiency curves of different methods in Fig. 2. It is clear that DFormer++ achieves much better performance and computation trade-off compared to other methods. Particularly, our DFormer++-S can achieve equal performance to the current state-of-the-art methods with less than half of the parameters and MACs, *i.e.*, 37.2M and 56.4G. DFormer++-B further yields 59.0% mIoU with 66.7M parameters and 97.5G MACs and surpasses the recent state-of-the-art methods by a large margin. In addition, the experiments on SUN-RGBD and Stanford2D3D also present similar advantages of our DFormer++ over other methods. These consistent improvements indicate that our RGB-D backbone can more efficiently build interactions between the RGB and depth features, and hence yields better performance with even lower computational cost. Moreover, the qualitative comparisons between the semantic segmentation results of our DFormer++ and other state-of-the-art models in Fig. 8 further demonstrate the advantage of our method.

TABLE 4
Latency (ms) on mobile and edge devices, *i.e.*, iPhone 15ProMax and NVIDIA Jetson Orin NX.

Model	**iPhone 15PM**	**Orin NX**	NYU Depth V2
CMX-B2	285.4	47.9	54.4
CMX-B4	355.6	87.5	56.3
CMNeXt-B4	342.8	71.5	56.9
GemniFusion-B3	317.5	58.1	56.9
DFormer++-L	**215.4**	**34.3**	**59.0**

TABLE 5
Performance of the RGB pretrained and depth pretrained backbone for segmentation using only depth maps. The two backbones adopt the same pretraining settings and architecture but are pretrained on the ImageNet images and their depth maps, respectively.

Backbone	**#Params**	**MACs**	**NYU mIoU(%)**
RGB	13.8M	14.4G	29.6
Depth	13.8M	14.4G	44.8

Speed on diverse hardware. We benchmark DFormer++ in deployment settings by exporting to ONNX for TensorRT on Jetson Orin and converting to Core ML for iOS devices, using a unified protocol (batch size 1, 480×640). As summarized in Tab. 4, DFormer++ delivers lower latency than recent state-of-the-art methods on iPhone 15 ProMax and Jetson Orin, confirming that its architectural efficiency transfers to real runtimes. From a memory–bandwidth perspective, our design employs fixed local windows and global grids (linear-time attention), embedding-level addition rather than concatenation with projection, a smaller depth channel ratio, and an RGB-only decoder—choices that collectively reduce peak activations and feature movement, easing the bandwidth bottleneck common on mobile/edge devices. In practice, the measured latencies on Orin and iOS confirm that DFormer++ sustains its accuracy–efficiency advantage under realistic deployment constraints.

Diversity of real-world scenes. To assess robustness under diverse real-world conditions, we report results across indoor (NYU Depth V2, SUN-RGBD) and large-scale mixed indoor scenes (Stanford2D3D), and additionally include outdoor evaluations. Specifically, DFormer++ is further tested on KITTI-360 [35] and MFNet [24], which exhibit different sensor characteristics and noise patterns, and we also conduct experiments on Cityscapes [14] to broaden the evaluation scope (Tab. 3). Despite being pretrained with synthetic depth, our pretraining paradigm transfers effectively to real-world scene perception across these datasets, consistently improving performance under varied scene complexities and sensor behaviors. This diversity of benchmarks strengthens the empirical evidence that DFormer++ is not only accurate and efficient on standard RGB-D settings, but also resilient across heterogeneous real-world conditions.

4.2 Ablation Study and Analysis

In this subsection, we perform comprehensive ablation analyses on our DFormer++ from different perspectives. Because NYU Depth v2 [45] is the most widely-used RGB-D segmentation dataset and contains various scenes, the subsequent ablation experiments are mainly carried out on it.

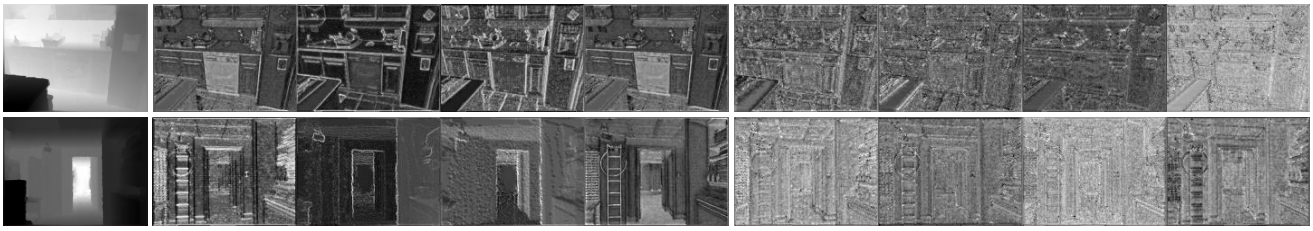


Fig. 7. Encoding depth maps with backbones pretrained on different types of modality, *i.e.,* pretraining with RGB data and pretraining with depth maps. During finetuning, we only take the depth maps as input to see which backbone works better. Obviously, the backbone pretrained on depth maps can generate more expressive feature maps.

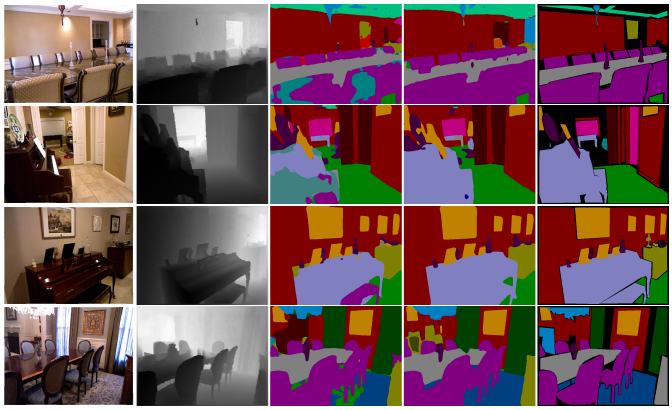


Fig. 8. Qualitative comparison of DFormer++ and our conference version.

TABLE 6
Effect of different pretraining manners. RGB-only: pretraining on RGB images; Syn-RGB-D: pretraining on synthetic RGB-D data; Hybrid: hybird pretraining on synthetic RGB-D and real RGB + estimated depth; Ours: pretraining on real RGB + estimated depth.

Model	Pretraining manner	NYU Depthv2	SUN RGBD
DFormer++-T	RGB-only	54.3	48.4
DFormer++-T	Syn-RGB-D	55.8	50.0
DFormer++-T	Hybrid	56.5	50.9
DFormer++-T	Ours	57.0	51.5
DFormer++-S	RGB-only	55.9	49.9
DFormer++-S	Syn-RGB-D	57.0	51.4
DFormer++-S	Hybrid	57.8	52.2
DFormer++-S	Ours	58.1	52.7

4.2.1 Analysis of RGB-D Pretraining

Why involving depth information in pretraining? Existing state-of-the-art RGB-D perception models [31], [61], [62] mainly use models pretrained on RGB images to extract 3D geometry information from depth maps. We argue that the huge representation distribution shift caused by using the RGB backbone to encode the depth maps may influence the extraction of the 3D geometry. To demonstrate this, we respectively use RGB and depth data to pretrain the RGB and depth backbone and then take only the depth maps as input for segmentation. As shown in Tab. 5, we can see under the same network architecture, the model pretrained on RGB images obviously falls behind the one pretrained on depth maps. We also visualize some features within the RGB pretrained and depth pretrained backbones in Fig. 7. Before finetuning, the backbone pretrained on RGB data is not able to extract expressive features from the depth maps. After finetuning, the model using RGB backbones still struggles to extract diverse features from the depth maps. In contrast, the features of the backbone pretrained on depth data are better. These experiments indicate that there exist significant differences between RGB and depth maps and it is difficult to process the depth data with RGB pretrained weights. This observation motivates us to involve depth data during ImageNet pretraining.

Effect of different pretraining schemes. To illustrate the necessity of the RGB-D pretraining within DFormer++, we attempt to replace the depth maps with RGB images during pretraining, dubbed as RGB pretraining. In particular, RGB pretraining modifies the input channel of the depth stem layer from 1 to 3. Note that for the finetuning setting, the modalities of the input data and the model structure are the same, *i.e.,* DFormer++-T. As shown in Tab. 7, our RGB-D pretraining brings 2.8% improvement for DFormer++-T compared to the RGB pretraining in terms of mIoU on NYU Depth v2. We argue that this is because our RGB-D pretraining avoids the mismatch encoding of the 3D geometry features of depth maps caused by the use of pretrained RGB backbones and enhances the interaction efficiency between the two modalities. Tab. 5 and Fig. 7 also demonstrate the mismatch problem. These experimental results indicate that the RGB-D representation capacity learned during the RGB-D pretraining is crucial for semantic segmentation accuracy.

To better understand the role of the proposed RGB-D pretraining, we further compare four pretraining regimes under the same backbone, pretraining schedule, and finetuning recipe: (1) RGB-only pretraining on ImageNet-1K, (2) synthetic-only RGB-D pretraining on large-scale synthetic RGB-D data (Syn-RGB-D), (3) hybrid pretraining that first uses synthetic RGB-D and then real RGB + estimated depth (Hybrid), and (4) our RGB-Pseudo-D pretraining on real RGB + estimated depth for ImageNet-1K (Ours). We adopt SceneNet [25] as the large-scale synthetic RGB-D data. As summarized in Tab. 6, we observe a consistent ordering across NYU Depth v2 and SUN RGB-D and for both DFormer++-T and DFormer++-S: *Ours* > *Hybrid* > *Syn-RGB-D* > *RGB-only*. For example, on NYU Depth v2, RGB-Pseudo-D pretraining improves over RGB-only pretraining by up to $+2.7$ mIoU for DFormer++-T and $+2.2$ mIoU for DFormer++-S, and still brings clear gains over synthetic-only and hybrid pretraining.

These results indicate that real RGB + estimated depth provides more effective supervision than purely synthetic RGB-D or hybrid schedules, likely because it combines

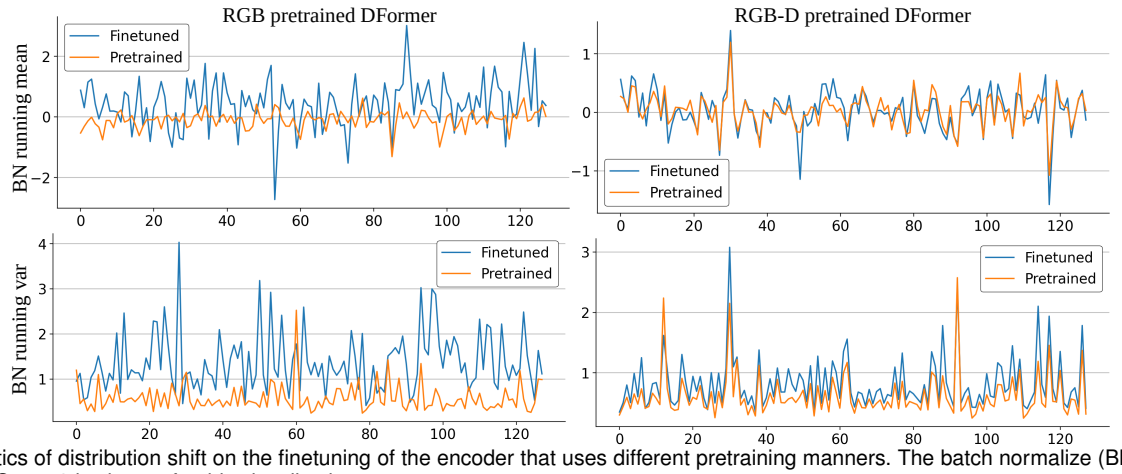


Fig. 9. Statistics of distribution shift on the finetuning of the encoder that uses different pretraining manners. The batch normalize (BN) layer at the first block of Stage 2 is chosen for this visualization.

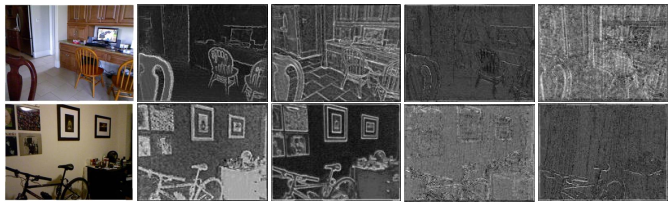


Fig. 10. Visualization of the features with the finetuned models that load RGB-pretrained and RGBD-pretrained weights and RGB-D input.

realistic appearance and semantic statistics with explicit geometric cues. Together with the analyses in Fig. 9 and Fig. 10 (BN statistics, feature visualizations), this supports our claim that learning RGB–depth interaction priors on large-scale real-image RGB-Pseudo-D data is a practical and strong alternative to synthetic-only or hybrid pretraining schemes.

Observations towards the distribution shift. Interacting the RGB and depth features within the RGB-pretrained encoders would bring drastic changes in the feature distribution, which makes the previous statistics of batch normalization incompatible with the input features. Following [10], we visualize the statistics of the BN layers of the encoder during the finetuning process to reflect the distribution shift. Specifically, we visualize the BN layer for a random layer in Fig. 9 to observe the statistics of the fused features. For the RGB-D pretrained encoder, the running means and variance of the BN layer only have slight changes after finetuning, illustrating the learned RGB-D representations are transferable for the RGB-D segmentation tasks. In contrast, for the RGB pretrained one, the statistics of the BN layer are changed sharply after finetuning, which indicates the encoding is mismatched. The situation forces RGB pretrained weights to adapt the input features fused by the two modalities. The redesigned unified depth-guided attention (local+global) and RGB-D pretraining embed geometric cues directly into the token stream, so the geometry is propagated through deeper layers rather than being carried by a separate depth branch. This architectural choice encourages RGB-D interactions within attention at multiple scales, which theoretically reduces modality drift and helps preserve depth-related structure in deep layers. We also visualize some features of DFormer++ that use the RGB and RGB-D pretraining, as shown in Fig. 10. Our method preserves the border and shape information within depth cues.

Analysis of domain gaps. Two major domain gaps arise in RGB-D pretraining: (1) the image/semantic gap between fully synthetic RGB-D corpora (e.g., SceneNet) and real-world scenes, and (2) the depth noise/artifact gap between predicted depth (dense, hole-free, low speckle) and real sensor depth (holes, quantization, range-dependent noise, multipath, sparsity). The first gap is largely addressed by pretraining on real ImageNet images rather than synthetic renderings. The second gap remains but is gradually reduced through fine-tuning on real RGB-D benchmarks. Although the depth maps are estimated, recent predictors (e.g., Depth Anything) reliably preserve relative geometry and ordinal structure, which are the main cues exploited by our depth-guided attention. Fine-grained metric accuracy and raw noise characteristics are less critical for delineating semantic boundaries. Moreover, predicted depth maps are spatially complete, providing consistent cross-modal correspondences that facilitate robust RGB-depth alignment transferable to noisy sensor data. Compared with synthetic pretraining, this strategy maintains authentic appearance diversity, realistic object co-occurrence, and material statistics, thus alleviating the semantic distribution shift that synthetic-based pretraining still suffers from. Consistent improvements across SUN-RGBD, Stanford2D3D, and NYU Depth v2 further demonstrate that fine-tuning effectively bridges the remaining domain gap.

Impact of depth map quality. We pretrain DFormer++ on ImageNet-1K [41] using monocularly estimated depth paired with RGB. To assess sensitivity to the quality of pseudo-depth, we generate depth maps with DepthAnything [58], OmniData [17], AdaBins [4], and DepthPro [5]. These four estimators are representative of different periods and modeling paradigms in monocular depth estimation, providing sufficient coverage for our study. At the pretraining stage, we observe modest differences in ImageNet Top-1 accuracy across estimators (*e.g.,* 84.9/84.7/84.0/83.7 for DepthAnything/DepthPro/OmniData/AdaBins). However, after fine-tuning for RGB-D semantic segmentation, the downstream results are highly consistent: all the models reported above all reach ∼57.0 mIoU on NYU Depth V2. These findings indicate that DFormer++'s RGB-D pre-

training is robust to the choice and quality of the depth estimator—suggesting that the gains stem from learning a geometry–RGB interaction prior rather than exploiting estimator-specific artifacts.

Our RGB-D pretraining is intended to endow the encoder with RGB-D representational capability at scale. Although the pretraining depth maps are estimated, they preserve the depth information patterns needed to learn robust RGB–depth interactions, which then consistently translate into gains on downstream segmentation with real sensor depth.

What does RGB-Pseudo-D pretraining actually learn? Our experiments on different depth estimators and depth qualities suggest that the proposed RGB-Pseudo-D pretraining is not simply distilling a particular monocular depth estimator, but learning to exploit structural guidance from the depth maps for segmentation. The target task is semantic segmentation rather than 3D reconstruction, and the role of the estimated depth during pretraining is to provide coarse geometric structure and object contours that complement RGB appearance. Although monocular depth estimators may hallucinate fine-scale artifacts (e.g., internal boundaries on wall paintings or errors on mirrors), they generally agree on the global layout and major depth discontinuities. The depth-guided attention in DFormer++ learns how to fuse these structural cues with RGB features to sharpen boundaries and improve region consistency, as reflected by the feature visualizations and segmentation results in Fig. 8 and Fig. 10. The robustness to different estimators and even highly perturbed depth maps is thus less an indication that the model ignores geometry, and more an indication that it focuses on stable, geometry-like structure (layout, contours) shared across estimators, while down-weighting estimator-specific artifacts that are not consistent with the semantic supervision.

Discussion on RGB-D pretraining. Within the RGB-D semantic segmentation community, it is now well established that both RGB-only pretraining (e.g., on ImageNet or ADE20K) and RGB-D pretraining can improve downstream performance [18], [42], [43], including our conference version [60]. Our goal in this work is therefore not to claim the idea of RGB-D pretraining itself as novel, but to provide a principled pretraining–architecture combination and to clarify why it is effective. Conceptually, our RGB-D pretraining is designed to reduce the mismatch between pretraining and finetuning: instead of pretraining on RGB-only data and introducing depth and fusion only at finetuning time, we directly pretrain a single RGB-D backbone on large-scale real-image RGB–depth pairs synthesized from ImageNet-1K using a strong monocular estimator. The proposed depth-guided local–global attention enables RGB and depth to interact inside every encoder block, so that the modalities and fusion patterns seen during pretraining are aligned with those used in downstream segmentation. Compared to prior RGB-D pretraining strategies that mainly rely on synthetic RGB-D corpora, weak modality pairings, or relatively small, task-specific pretraining regimes, our pipeline leverages the appearance diversity and realistic semantics of ImageNet together with geometry-preserving estimated depth, and scales to full-image pretraining while keeping computation manageable. The ablation studies in Tab. 7 further show that this design consistently yields non-trivial gains over RGB-only pretraining and over variants that do not involve depth during pretraining, supporting our claim that aligning pretraining and finetuning in a unified RGB-D encoder is key to the observed improvements.

TABLE 7
Semantic segmentation performance with different models under the RGB-only pretraining and RGB-D pretraining on NYU Depth v2 [45].

Model	Params.	MACs	RGB	RGB-D
TokenFusion-B3	45.9M	94.0G	54.2	56.1 (+1.9)
CMX-B2	66.6M	38.8G	54.4	56.0 (+1.6)
CMNeXt-B4	116.6M	131.4G	56.8	58.3 (+1.5)
DFormer++-T	17.3M	29.5G	54.3	57.0 (+2.7)

TABLE 8
Effect of fusion manners within DFormer++-T under the RGB-only pretraining and RGB-D pretraining on NYU Depth v2 [45].

Model	Params.	MACs	RGB	RGB-D
sparse	16.9M	27.3G	54.2	55.9 (+1.7)
dense (ours)	17.3M	29.5G	54.3	57.0 (+2.7)

4.2.2 *Generalization of DFormer++*

RGB-D pretraining on other models. To verify the effect of our RGB-D pretraining on other methods, we pretrain a recent popular model CMX (MiT-B2) with our RGB-D pretraining pipeline and it obtains about 1.4% mIoU improvement, as shown in Tab. 7. Under the RGB-D pretraining, DFormer++-T still outperforms CMX (MiT-B2) by a large margin, which should be attributed to that the pretrained fusion weight within DFormer++ can achieve better and more efficient fusion between RGB-D data. Besides, we provide the RGB pretrained DFormer++ to provide more insights in Tab. 7. A similar phenomenon appears under the RGB-only pretraining. While RGB-D pretraining consistently improves all models, the absolute gain for DFormer++-T on NYU Depth v2 is notably larger (+2.7 mIoU) than for the dual-branch baselines (+1.5–1.9 mIoU). To probe whether this effect is tied to our architectural design, Tab. 8 contrasts a *sparse-fusion* variant of DFormer++-T, where RGB–D interaction is performed only in the last block of each stage, with our standard *dense-fusion* design, where depth-guided attention operates in every encoder block. Under RGB-D pretraining, dense fusion not only achieves higher absolute accuracy but also enjoys a substantially larger gain (+2.7 vs. +1.7 mIoU) at similar parameter counts and MACs. These results support our hypothesis that a joint RGB-D encoder with frequent, multi-scale RGB–depth interactions allows a much larger fraction of the backbone parameters to directly absorb the additional geometric supervision during pretraining, making DFormer++ more "pretraining-effective" than conventional two-branch designs that rely on sparse fusion points.

Generalization ability to other modalities. Through RGB-D pretraining, our DFormer++ is endowed with the capacity to fuse the RGB and depth features during pretraining. To verify whether the interaction method still works when replacing depth with another modality, we finetune our RGB-D pretrained DFormer++ to some benchmarks with

TABLE 9
Results on the RGB-T semantic segmentation benchmark MFNet [24] and RGB-L semantic segmentation benchmark KITTI-360 [35]. 'RGB' and 'RGBD' mean RGB-only and RGB-D pretraining, respectively. '*' represents our implementations via their official codes.

Model	Pretrain	**Params**	**MACs**	MFNet	KITTI
CMX-B2	RGB	66.6M	67.6G	58.2	64.3
CMX-B4	RGB	139.9M	134.3G	59.7	65.5*
CMNeXt-B2	RGB	65.1M	65.5G	58.4*	65.3
CMNeXt-B4	RGB	135.6M	132.6G	59.9	65.6*
DFormer-L	RGB	39.0M	65.7G	59.5	65.2
DFormer-L	RGBD	39.0M	65.7G	60.3	66.1
DFormer++-S	RGB	37.2M	56.4G	59.8	65.6
DFormer++-S	RGBD	37.2M	56.4G	60.9	67.2
DFormer++-L	RGB	66.7M	97.5G	60.0	65.9
DFormer++-L	RGBD	66.7M	97.5G	**61.4**	**67.7**

TABLE 10
Results on the standard and misalignment settings on NYU Depth V2.

Model	**Params**	**MACs**	standard	misalignment
CMX-B2	66.6M	67.6G	54.4	53.1
CMX-B4	139.9M	134.3G	56.3	55.0
CMNeXt-B4	135.6M	132.6G	56.9	55.4
GemniFusion-B3	75.8M	138.2G	56.9	55.6
DFormer++-L	66.7M	97.5G	**59.0**	**58.2**

other modalities, *i.e.*, RGB-T semantic segmentation on MFNet [24] and RGB-L semantic segmentation on KITTI-360 [35]. As shown in Tab. 9, our pretraining framework still improves the performance. However, the improvement is limited compared to that on RGB-D scenes.

Thermal images are not depth maps, but they still provide structured, spatially coherent signals that strongly correlate with object boundaries and shapes: temperature distributions typically change across object surfaces and at region transitions, producing clear gradients along contours and between materials. In our framework, the second modality (depth in the RGB-D setting) is mainly used as a geometry-like structural cue: the depth-guided attention block leverages this modality to highlight boundaries, separate overlapping objects, and refine region consistency, rather than relying on its absolute physical semantics. When depth is replaced by thermal in the RGB-T setting, the thermal channels still supply high-frequency structural information (object contours, homogeneous regions, etc.), which the same attention mechanism can exploit in a similar way. Consistent with this interpretation, RGB-D pretraining brings moderate but positive improvements on MFNet—smaller than on RGB-D benchmarks, as expected given the semantic gap between thermal and depth—indicating that the learned fusion pattern and structural priors are not strictly tied to metric depth but can generalize to other structure-bearing modalities.

We further frozen depth-related components, where only the RGB branch and decoder are finetuned. In the frozen setting, RGB-D pretraining still outperforms RGB-only pretraining on DFormer++-S (57.8 *vs.* 56.5), indicating that part of the structural prior learned from RGB-D pretraining is embedded in the RGB branch and can transfer to RGB-T. However, both settings perform clearly worse than full finetuning, suggesting that a significant portion of the improvement on RGB-T tasks comes from jointly adapting the depth-related and fusion modules to the thermal modality.

TABLE 11
Results on ADE20K using RGB-D input via single scale testing on resolution of 512×512.

Model	Pretrain	**Params**	**MACs**	ADE20K
DFormer++-T	RGB	17.3M	30.4G	51.4
DFormer++-T	RGBD	17.3M	30.4G	53.8

TABLE 12
Effect of additional RGB-D ADE20K pretraining on DFormer++-T.

Pretrain	**Params**	**MACs**	NYU
RGB	17.3M	29.5G	54.3
RGBD Imagenet	17.3M	29.5G	57.0
RGBD Imagenet + ADE20K	17.3M	29.5G	57.6

Expansion to RGB segmentation dataset. To further test the scalability of our approach, we apply monocular depth estimation to the RGB-only ADE20K [66] training set and train DFormer++-T on ADE20K with RGB-D inputs. As shown in Tab. 11, RGB-D ImageNet pretraining followed by RGB-D ADE20K training improves the ADE20K test performance from 51.4 to 53.8 mIoU compared with RGB-only pretrain. We then use these ADE20K-trained checkpoints as initialization for NYU Depth v2. As reported in Tab. 12, additional RGB-D training on ADE20K further boosts NYU Depth v2 performance from 57.0 to 57.6 mIoU on top of RGB-D ImageNet pretraining. These results show that our RGB-D pretraining framework is scalable: augmenting more large-scale RGB datasets with estimated depth and incorporating them into the pretraining stage can provide additional gains as the pretraining corpus grows.

RGB-D misalignment. In Tab. 10, we conducted a misalignment stress test on NYU Depth V2 by randomly shifting the depth maps 0-20% in both horizontal and vertical directions. As misalignment increases, all methods exhibit notable performance drops; our method remains the best performance, though it also declines. This trend is expected because current RGB-D pipelines, including ours, rely on position-wise fusion that implicitly assumes spatial alignment between RGB and depth, and most architectures do not explicitly model cross-modal misregistration, which inevitably harms accuracy under strong shifts. Despite the unavoidable degradation, our approach shows the strongest robustness and retains the best overall performance under misalignment.

5 Conclusions

In this work, we propose a novel RGB-D pretraining framework to learn transferable representations for RGB-D downstream tasks. The core of our model architecture is a tailored depth guided attention mechanism. Thanks to the proposed RGB-D attention, our model is able to utilize the 3D geometry within the depth modality from coarse- and fine-grained scales. This avoids the use of individual branches for encoding depth information and achieves efficient and effective RGB-D encoding. We combine them to propose a powerful and highly robust RGB-D foundation model, DFormer++, which achieves new state-of-the-art performance in RGB-D semantic segmentation benchmarks, with far less computational cost compared to existing methods. We hope this

work could bring the multi-modal segmentation community new insights for advanced RGB-D training pipelines and model designs.

REFERENCES

[1] Hassan Akbari, Liangzhe Yuan, Rui Qian, Wei-Hong Chuang, Shih-Fu Chang, Yin Cui, and Boqing Gong. Vatt: Transformers for multimodal self-supervised learning from raw video, audio and text. In *Adv. Neural Inform. Process. Syst.*, volume 34, pages 24206–24221, 2021.

[2] Iro Armeni, Sasha Sax, Amir R. Zamir, and Silvio Savarese. Joint 2D-3D-semantic data for indoor scene understanding. *arXiv preprint arXiv:1702.01105*, 2017.

[3] Roman Bachmann, David Mizrahi, Andrei Atanov, and Amir Zamir. MultiMAE: Multi-modal multi-task masked autoencoders. In *Eur. Conf. Comput. Vis.*, pages 348–367, 2022.

[4] Shariq Farooq Bhat, Ibraheem Alhashim, and Peter Wonka. Adabins: Depth estimation using adaptive bins. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 4009–4018, 2021.

[5] Aleksei Bochkovskii, Amaël Delaunoy, Hugo Germain, Marcel Santos, Zhou Yichao, Stephan R. Richter, and Vladlen Koltun. Depth pro: Sharp monocular metric depth in less than a second. In *Int. Conf. Learn. Represent.*, pages 1–36, 2025.

[6] Jinming Cao, Hanchao Leng, Dani Lischinski, Danny Cohen-Or, Changhe Tu, and Yangyan Li. ShapeConv: Shape-aware convolutional layer for indoor RGB-D semantic segmentation. In *Int. Conf. Comput. Vis.*, pages 7088–7097, 2021.

[7] Lluis Castrejon, Yusuf Aytar, Carl Vondrick, Hamed Pirsiavash, and Antonio Torralba. Learning aligned cross-modal representations from weakly aligned data. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 2940–2949, 2016.

[8] Lin-Zhuo Chen, Zheng Lin, Ziqin Wang, Yong-Liang Yang, and Ming-Ming Cheng. Spatial information guided convolution for real-time RGBD semantic segmentation. *IEEE Trans. Image Process.*, 30:2313–2324, 2021.

[9] Xiaokang Chen, Kwan-Yee Lin, Jingbo Wang, Wayne Wu, Chen Qian, Hongsheng Li, and Gang Zeng. Bi-directional cross-modality feature propagation with separation-and-aggregation gate for RGB-D semantic segmentation. In *Eur. Conf. Comput. Vis.*, pages 561–577, 2020.

[10] Xinghao Chen, Chang Xu, Minjing Dong, Chunjing Xu, and Yunhe Wang. An empirical study of adder neural networks for object detection. In *Adv. Neural Inform. Process. Syst.*, volume 34, pages 6894–6905, 2021.

[11] Yuming Chen, Xinbin Yuan, Ruiqi Wu, Jiabao Wang, Qibin Hou, and Ming-Ming Cheng. Yolo-ms: Rethinking multi-scale representation learning for real-time object detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 47(6):4240–4252, 2025.

[12] Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. Uniter: Universal image-text representation learning. In *Eur. Conf. Comput. Vis.*, pages 104–120, 2020.

[13] Bowen Cheng, Alexander G. Schwing, and Alexander Kirillov. Per-pixel classification is not all you need for semantic segmentation. In *Adv. Neural Inform. Process. Syst.*, volume 34, pages 17864–17875, 2021.

[14] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 3213–3223, 2016.

[15] Alexey Dosovitskiy et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *Int. Conf. Learn. Represent.*, pages 1–12, 2021.

[16] Siqi Du, Weixi Wang, Renzhong Guo, Ruisheng Wang, and Shengjun Tang. Asymformer: Asymmetrical cross-modal representation learning for mobile platform real-time rgb-d semantic segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog. Worksh.*, pages 7608–7615, 2024.

[17] Ainaz Eftekhar, Alexander Sax, Jitendra Malik, and Amir Zamir. Omnidata: A scalable pipeline for making multi-task mid-level vision datasets from 3d scans. In *Int. Conf. Comput. Vis.*, pages 10786–10796, 2021.

[18] Söhnke Benedikt Fischedick, Daniel Seichter, Robin Schmidt, Leonard Rabes, and Horst-Michael Gross. Efficient multi-task scene analysis with rgb-d transformers. In *Proc. Int. Jt. Conf. Neural Netw*, pages 1–10, 2023.

[19] Fahimeh Fooladgar and Shohreh Kasaei. Multi-modal attention-based fusion model for semantic segmentation of RGB-depth images. *arXiv preprint arXiv:1912.11691*, 2019.

[20] Zhengyang Geng, Meng-Hao Guo, Hongxu Chen, Xia Li, Ke Wei, and Zhouchen Lin. Is attention better than matrix decomposition? *Int. Conf. Learn. Represent.*, pages 1–24, 2021.

[21] Rohit Girdhar, Mannat Singh, Nikhila Ravi, Laurens van der Maaten, Armand Joulin, and Ishan Misra. Omnivore: A single model for many visual modalities. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 16102–16112, 2022.

[22] Meng-Hao Guo, Cheng-Ze Lu, Qibin Hou, Zhengning Liu, Ming-Ming Cheng, and Shi-Min Hu. Segnext: Rethinking convolutional attention design for semantic segmentation. *Adv. Neural Inform. Process. Syst.*, 35:1140–1156, 2022.

[23] Meng-Hao Guo, Cheng-Ze Lu, Zheng-Ning Liu, Ming-Ming Cheng, and Shi-Min Hu. Visual attention network. *Comp. Visual Media*, 9(4):733–752, 2023.

[24] Qishen Ha, Kohei Watanabe, Takumi Karasawa, Yoshitaka Ushiku, and Tatsuya Harada. MFNet: Towards real-time semantic segmentation for autonomous vehicles with multi-spectral scenes. In *IEEE Int. Conf. Intell. Robots Syst.*, pages 5108–5115, 2017.

[25] Ankur Handa, Viorica Pătrăucean, Simon Stent, and Roberto Cipolla. Scenenet: An annotated model generator for indoor scene understanding. In *IEEE Int. Conf. Robot. Autom.*, pages 5737–5743, 2016.

[26] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 770–778, 2016.

[27] Qibin Hou, Cheng-Ze Lu, Ming-Ming Cheng, and Jiashi Feng. Conv2former: A simple transformer-style convnet for visual recognition. *IEEE Trans. Pattern Anal. Mach. Intell.*, 46(12):8274–8283, 2024.

[28] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 7132–7141, 2018.

[29] Xinxin Hu, Kailun Yang, Lei Fei, and Kaiwei Wang. ACNet: Attention based network to exploit complementary features for RGBD semantic segmentation. In *IEEE Int. Conf. Image Process.*, pages 1440–1444, 2019.

[30] Keli Huang, Botian Shi, Xiang Li, Xin Li, Siyuan Huang, and Yikang Li. Multi-modal sensor fusion for auto driving perception: A survey. *arXiv preprint arXiv:2202.02703*, 2022.

[31] Ding Jia, Jianyuan Guo, Kai Han, Han Wu, Chao Zhang, Chang Xu, and Xinghao Chen. Geminifusion: Efficient pixel-wise multi-modal fusion for vision transformer. In *Inter. Conf. Mach. Learning*, pages 21753–21767, 2024.

[32] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *Int. Conf. Learn. Represent.*, pages 1–15, 2015.

[33] Zhong-Yu Li, Shanghua Gao, and Ming-Ming Cheng. Sere: Exploring feature self-relation for self-supervised transformer. *IEEE Trans. Pattern Anal. Mach. Intell.*, 45(12):15619–15631, 2023.

[34] Zhong-Yu Li, Bo-Wen Yin, Shanghua Gao, Yongxiang Liu, Li Liu, and Ming-Ming Cheng. Enhancing representations through heterogeneous self-supervised learning. *IEEE Trans. Pattern Anal. Mach. Intell.*, 47(7):5976–5989, 2025.

[35] Yiyi Liao, Jun Xie, and Andreas Geiger. Kitti-360: A novel dataset and benchmarks for urban scene understanding in 2d and 3d. *IEEE Trans. Pattern Anal. Mach. Intell.*, 45(3):3292–3310, 2023.

[36] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Int. Conf. Comput. Vis.*, pages 10012–10022, 2021.

[37] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 11976–11986, 2022.

[38] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *Int. Conf. Learn. Represent.*, pages 1–19, 2019.

[39] Nicolas Marchal, Charlotte Moraldo, Hermann Blum, Roland Siegwart, Cesar Cadena, and Abel Gawel. Learning densities in feature space for reliable segmentation of indoor scenes. *IEEE Robot. Autom. Lett.*, 5(2):1032–1038, 2020.

[40] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *Inter. Conf. Mach. Learning*, pages 8748–8763, 2021.

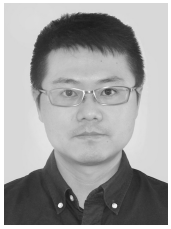
[41] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. ImageNet large scale visual recognition challenge. *Int. J. Comput. Vis.*, 115(3):211–252, 2015.

[42] Daniel Seichter, Söhnke Benedikt Fischedick, Mona Köhler, and

Horst-Michael Groß. Efficient multi-task rgb-d scene analysis for indoor environments. In *Proc. Int. Jt. Conf. Neural Netw*, pages 1–10, 2022.
[43] Daniel Seichter, Mona Köhler, Benjamin Lewandowski, Tim Wengefeld, and Horst-Michael Gross. Efficient RGB-D semantic segmentation for indoor scene analysis. In *IEEE Int. Conf. Robot. Autom.*, pages 13525–13531, 2021.
[44] Dai Shi. Transnext: Robust foveal visual perception for vision transformers. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 17773–17783, 2024.
[45] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from RGBD images. In *Eur. Conf. Comput. Vis.*, pages 746–760, 2012.
[46] Shuran Song, Samuel P. Lichtenberg, and Jianxiong Xiao. SUN RGB-D: A RGB-D scene understanding benchmark suite. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 567–576, 2015.
[47] Xiaoshuai Sun, Xuying Zhang, Liujuan Cao, Yongjian Wu, Feiyue Huang, and Rongrong Ji. Exploring language prior for mode-sensitive visual attention modeling. In *ACM Int. Conf. Multimedia*, pages 4199–4207, 2020.
[48] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Adv. Neural Inform. Process. Syst.*, 30:5998–6008, 2017.
[49] Zifu Wan, Yuhao Wang, Silong Yong, Pingping Zhang, Simon Stepputtis, Katia Sycara, and Yaqi Xie. Sigma: Siamese mamba network for multi-modal semantic segmentation. In *IEEE Winter Conf. Appl. Comput. Vis.*, pages 1734–1744, 2025.
[50] Hengyi Wang, Jingwen Wang, and Lourdes Agapito. Co-slam: Joint coordinate and sparse parametric encodings for neural real-time slam. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 13293–13302, 2023.
[51] Weiyue Wang and Ulrich Neumann. Depth-aware cnn for rgb-d segmentation. In *Eur. Conf. Comput. Vis.*, 2018.
[52] Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In *Int. Conf. Comput. Vis.*, pages 568–578, 2021.
[53] Yikai Wang, Xinghao Chen, Lele Cao, Wenbing Huang, Fuchun Sun, and Yunhe Wang. Multimodal token fusion for vision transformers. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 12186–12195, 2022.
[54] Yikai Wang, Wenbing Huang, Fuchun Sun, Tingyang Xu, Yu Rong, and Junzhou Huang. Deep multimodal fusion by channel exchanging. *Adv. Neural Inform. Process. Syst.*, 33:4835–4845, 2020.
[55] Mingrui Wu, Xuying Zhang, Xiaoshuai Sun, Yiyi Zhou, Chao Chen, Jiaxin Gu, Xing Sun, and Rongrong Ji. Difnet: Boosting visual information flow for image captioning. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 18020–18029, 2022.
[56] Zongwei Wu, Guillaume Allibert, Christophe Stolz, and Cédric Demonceaux. Depth-adapted cnn for rgb-d cameras. In *Asian Conf. Comput. Vis.*, 2020.
[57] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M. Alvarez, and Ping Luo. SegFormer: Simple and efficient design for semantic segmentation with transformers. In *Adv. Neural Inform. Process. Syst.*, volume 34, pages 12077–12090, 2021.
[58] Lihe Yang, Bingyi Kang, Zilong Huang, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything: Unleashing the power of large-scale unlabeled data. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 10371–10381, 2024.
[59] Bowen Yin, Xuying Zhang, Deng-Ping Fan, Shaohui Jiao, Ming-Ming Cheng, Luc Van Gool, and Qibin Hou. Camoformer: Masked separable attention for camouflaged object detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 46(12):10362–10374, 2024.
[60] Bowen Yin, Xuying Zhang, Zhong-Yu Li, Li Liu, Ming-Ming Cheng, and Qibin Hou. Dformer: Rethinking rgbd representation learning for semantic segmentation. In *Int. Conf. Learn. Represent.*, pages 1–14, 2024.
[61] Jiaming Zhang, Huayao Liu, Kailun Yang, Xinxin Hu, Ruiping Liu, and Rainer Stiefelhagen. Cmx: Cross-modal fusion for rgb-x semantic segmentation with transformers. *IEEE Trans. Intell. Transport. Syst.*, 24(12):14679–14694, 2023.
[62] Jiaming Zhang, Ruiping Liu, Hao Shi, Kailun Yang, Simon Reiß, Kunyu Peng, Haodong Fu, Kaiwei Wang, and Rainer Stiefelhagen. Delivering arbitrary-modal semantic segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 1136–1147, 2023.
[63] Xuying Zhang, Xiaoshuai Sun, Yunpeng Luo, Jiayi Ji, Yiyi Zhou, Yongjian Wu, Feiyue Huang, and Rongrong Ji. Rstnet: Captioning with adaptive attention on visual and non-visual words. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 15465–15474, 2021.
[64] Xuying Zhang, Bowen Yin, Zheng Lin, Qibin Hou, Deng-Ping Fan, and Ming-Ming Cheng. Referring camouflaged object detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 47(5):3597–3610, 2025.
[65] Xuying Zhang, Bo-Wen Yin, Yuming Chen, Zheng Lin, Yunheng Li, Qibin Hou, and Ming-Ming Cheng. Temo: Towards text-driven 3d stylization for multi-object meshes. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 19531–19540, 2024.
[66] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 633–641, 2017.
[67] Wujie Zhou, Enquan Yang, Jingsheng Lei, Jian Wan, and Lu Yu. Pgdenet: Progressive guided fusion and depth enhancement network for rgb-d indoor scene parsing. *IEEE Trans. Multimedia*, 25:3483–3494, 2023.
[68] Wujie Zhou, Enquan Yang, Jingsheng Lei, and Lu Yu. Frnet: Feature reconstruction network for rgb-d indoor scene parsing. *IEEE J. Sel. Top. Signal Process.*, 16(4):677–687, 2022.

Bo-Wen Yin is a Ph.D. student from the College of Computer Science, Nankai University. He is supervised by Prof. Qibin Hou. His research interests include computer vision and multi-modal scene perception.

Jiao-Long Cao received his B.E. degree in the Shiing-Shen Chern Class from the School of Mathematical Sciences of Nankai University in 2024. He is currently a Ph.D. candidate at the Media Computing Lab at Nankai University, under the supervision of Prof. Qibin Hou and Prof. Ming-Ming Cheng. His research interests include computer vision and machine learning.



Dan Xu is an Assistant Professor in the Department of Computer Science and Engineering at HKUST. He was a Postdoctoral Research Fellow in Visual Geometry Group (VGG) at the University of Oxford. He was a Ph.D. in the Department of Computer Science at the University of Trento. He was also a student research assistant in MM Lab at the Chinese University of Hong Kong. He received the best scientific paper award at ICPR 2016, and a Best Paper Nominee at ACM MM 2018. He served as Area Chair/Senior PC at multiple main-stream conferences including NeurIPS, ICML, ICLR, CVPR, ICCV, ECCV, AAAI, ACM Multimedia, WACV, and ACCV.

Ming-Ming Cheng received his PhD degree from Tsinghua University in 2012. Then, he did 2 years research fellow, with Prof. Philip Torr in Oxford. He is now a professor at Nankai University, leading the Media Computing Lab. His research interests include computer graphics, computer vision, and image processing. He received research awards, including the National Science Fund for Distinguished Young Scholars and the ACM China Rising Star Award. He is on the editorial boards of IEEE TPAMI and IEEE TIP.

Qibin Hou received his Ph.D. degree from the School of Computer Science, Nankai University. Then, he worked at the National University of Singapore as a research fellow. Now, he is an associate professor at the School of Computer Science, Nankai University. He has published over 40 papers in top conferences/journals, including T-PAMI, CVPR, ICCV, NeurIPS, etc. His research interests include deep learning, image processing, and computer vision.