Attention-based Dual Supervised Decoder for RGBD Semantic Segmentation

Yang Zhang [*]
Nanjing University
https://yangzhangcst.github.io/Homepage/

Yang Yang
Nanjing University
yyang_nju@outlook.com

Chenyun Xiong
Hubei University of Technology
cyx@hbut.edu.cn

Guodong Sun[†]
Hubei University of Technology
sgdeagle@163.com

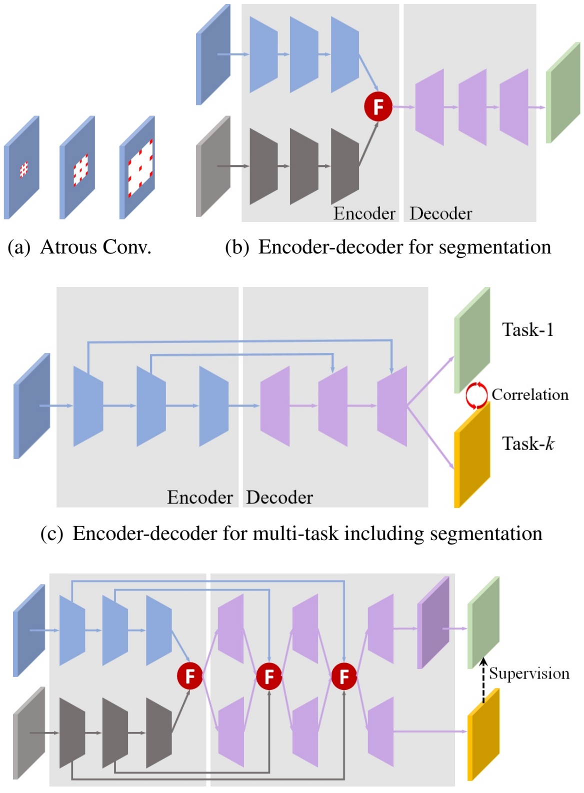
Yanwen Guo [‡]
Nanjing University
ywguo@nju.edu.cn

Abstract

Encoder–decoder models have been widely used in RGBD semantic segmentation, and most of them are designed via a two-stream network. In general, jointly rea-soning the color and geometric information from RGBD is beneficial for semantic segmentation. However, most existing approaches fail to comprehensively utilize multi-modal information in both the encoder and decoder. In this paper, we propose a novel attention-based dual supervised decoder for RGBD semantic segmentation. In the encoder, we design a simple yet effective attention-based multi-modal fusion module to extract and fuse deeply multi-level paired complementary information. To learn more robust deep representations and rich multi-modal information, we introduce a dual-branch decoder to effectively leverage the correlations and complementary cues of different tasks. Extensive experiments on NYUDv2 and SUN-RGBD datasets demonstrate that our method achieves superior performance against the state-of-the-art methods.

1. Introduction

In recent years, scene understanding has received considerable attention due to the wide applications in AR/VR [38], autonomous driving [2, 53], UAVs [46], simultaneous localization and mapping (SLAM) [32], Robotics [33], and other artificial intelligence fields. As a result, semantic segmentation for scene understanding becomes extremely important. However, there still exists many challenges in RGBD semantic segmentation caused by the complexity of the environment, the influence of inaccurate depth, and the joint reasoning of multi-modal information.



(a) Atrous Conv. (b) Encoder-decoder for segmentation

(c) Encoder-decoder for multi-task including segmentation

(d) Dual supervised decoder for segmentation (ours)

Figure 1: Examples of typical structures for RGBD semantic segmentation. The blue color and gray color indicate the RGB and depth streams, separately. The Ⓕ denotes the combination operation.

Deep learning technique has been applied to the semantic segmentation problem with great success. Though different architectures are developed, the convolutional neural networks (CNNs) are still prevalent due to their ability to model non-linear, high-dimensional functions. Generally, atrous/dilated convolution-based methods [3, 14, 29, 30] allow us to effectively enlarge the field-of-view of filters to incorporate multi-scale context (see Fig. 1(a)), especially for

[*] Work done while interning at Hubei University of Technology
[†] Corresponding author
[‡] Corresponding author

atrous spatial pyramid pooling (ASPP) [4]. But there exists a 'gridding' problem [48], and they fail to capture small objects with accurate boundaries. Furthermore, it is computationally intensive if denser output features are extracted for this type of models.

The encoder-decoder models [36, 1, 24, 28, 45] allow for faster computation in the encoder path and recovering sharp object boundaries in the decoder. These models, however, only use RGB data for semantic segmentation which cannot achieve a satisfactory performance. Compared with color, the depth data provide geometric cues to reduce the uncertainty of the segmentation of objects in which the color is similar to the background [17]. It is thus meaningful and crucial to develop effective models to combine these complementary modalities for segmentation. To achieve this goal, numerous works [17, 8, 23, 21, 6, 50, 7] focus on designing a two-stream network which processes the RGB and geometry information in terms of depth or HHA, separately. As shown in Fig. 1(b), the features from two modalities are further fused by various mechanisms such as the element-wise summation [17, 23], gate [8, 7], and attention [21, 44] in the encoder. Such approaches only process the paired complementary cues in the encoder, but ignoring the cross-modal information during decoding. Moreover, training such a model is usually difficult to converge due to this imbalance of the encoder and decoder.
Since other related tasks such as depth estimation could facilitate semantic segmentation, recent works [13, 26, 55, 35, 56, 58] have attempted to solve the segmentation problem via a multi-task learning framework. Fully convolutional encoder-decoder networks have become the mainstream. During the joint learning, different task-specific decoders explore the correlations between these tasks as shown in Fig. 1(c). Note that these methods perform the multi-task distillation at a fixed scale (*i.e.* backbone features) with specific receptive field in the decoder. However, in fact, the influence between two tasks is different for various sizes of receptive field [47]. Furthermore, the capacity of fully convolutional encoder-decoder, whose encoder and decoder are simply integrated together (*e.g.* skip connection [55, 35, 58], multi-scale feature aggregation [56]), is limited for such a complex task of semantic segmentation.

In this paper, we design a simple symmetric yet effective network (in Fig. 1(d)) to efficiently use the multi-level cross-modal information for RGBD semantic segmentation. Motivated by the above observations, we first propose an attention-based multi-modal fusion module to process the multi-level paired complementary information in a two-stream encoder. To learn cross-modal information during decoding, we introduce a novel dual-branch decoder in which the primary is designed for semantic segmentation supervised by another task-guided branch. Such design enables us to incorporate multi-scale context by the ASPP module at the end of primary-branch, which contains the pyramid supervision for enhancing the deep representation. This specific dual-branch decoder is capable of improving the performance of semantic segmentation through multi-task distillation, while facilitating the convergence of training to solve the imbalance problem of the encoder and decoder. We conduct experiments on the NYUDv2 and SUN-RGBD datasets to validate the superior performance of our method in comparison with the state-of-the-arts.

Our contributions are summarized as follows.

- We propose a novel attention-based dual supervised decoder to utilize the complementary information across modalities for RGBD semantic segmentation.
- We design a simple yet effective attention multi-modal fusion module to extract and fuse deeply multi-level paired complementary information.
- We propose a dual-branch decoder to learn more robust deep representations and rich multi-modal information for the improvement of semantic segmentation performance and the efficiency of training.
- The proposed method achieves superior performance against the state-of-the-art methods on public benchmark datasets.

2. Related work

In recent years, CNN-based methods have been successfully applied to the RGBD semantic segmentation[1]. In terms of structure, these methods can be roughly divided into the following three groups.

Atrous/dilated Convolution. Several works [3, 14, 29, 39, 49, 30] utilized the atrous/dilated convolution to incorporate multi-scale context for RGBD semantic segmentation. For example, Chen *et al.* [3] proposed a dilated convolution which can enhance the receptive field while keep the resolution of the feature map. Qi *et al.* [39] introduced a 3D graph neural network (3DGNN) to model accurate context with geometry cues provided by depth based on the dilated convolution. Lin *et al.* [30] presented RefineNet, a generic multi-path refinement network that explicitly exploits all the information available along the down-sampling process to enable high-resolution prediction using long-range residual connections. However, dilated convolution can result in losing the continuity of feature maps. In addition, it is only effective for some large objects and invalid for small objects, which is not helpful to extract accurate edges.

Encoder-decoder. Many efforts [36, 1, 8, 17, 24, 42, 28, 23, 45, 57, 6, 50, 7, 44] concerning encoder-decoder

[1] https://github.com/Yangzhangcst/RGBD-semantic-segmentation

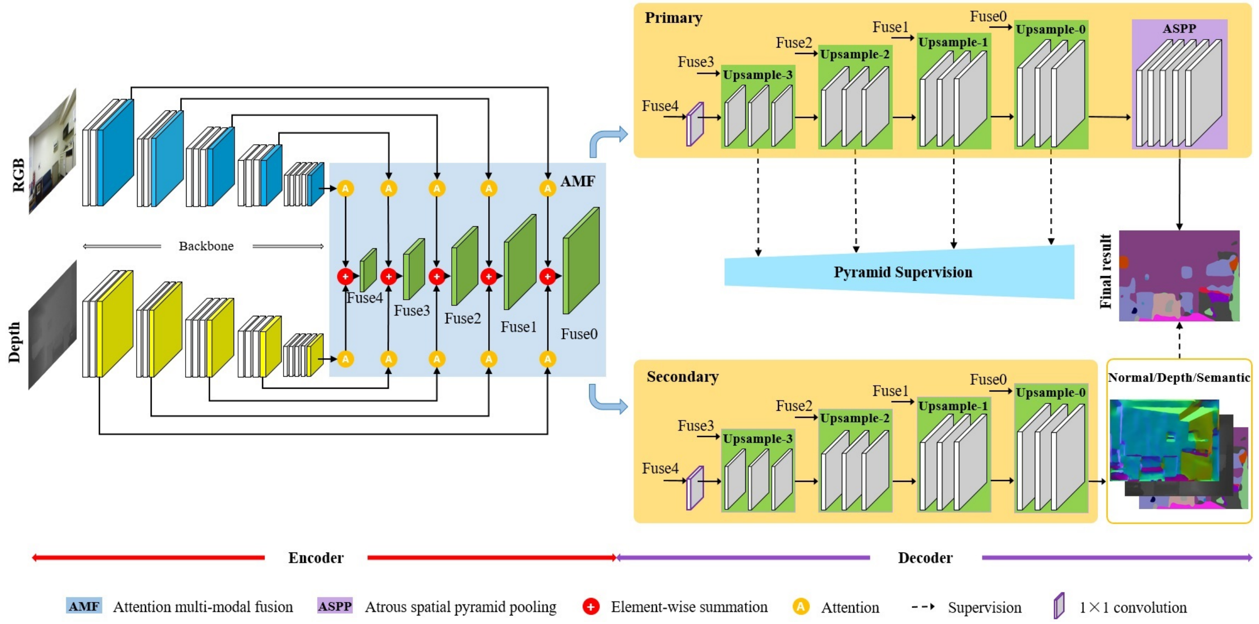


Figure 2: Overview of the proposed ADSD architecture. We employ a two-stream encoder and a dual-branch decoder. The input of the network is a pair of RGB-Depth images. The feature maps of backbone encoders are fused through AMF module, which are further used to output the results through upsampling modules in the dual-branch decoder. At the end of primary branch, the ASPP is introduced to improve the final segmentation performance. Meanwhile, each upsampling block predicts a side output for pyramid supervision. In addition to the semantic supervision, the secondary branch requires supervision from normal estimation, depth estimation, or semantic segmentation task.

architectures have been devoted to RGBD semantic segmentation. For instance, DeconvNet [36] used stacked deconvolutional layers to produce high-resolution prediction and more semantic details. SegNet [1] shared a similar idea using indices in pooling layers to promote the recovery process. To learn the optimal fusion of multi-modal features, RDFNet [28] extended the core idea of residual learning to RGBD semantic segmentation. Hu *et al.* [21] proposed a architecture ACNet with three parallel branches and a channel attention-based module that extracts weighted features from RGB and depth branches. Chen *et al.* [6] proposed a spatial information guided convolution network (SGNet) which allows to integrate 2D and 3D spatial information. ESANet [44] used two ResNet-based encoders with an attention-based fusion for incorporating depth information, and a decoder utilizing a learned upsampling. However, these methods only perform the multi-modal information in the encoder, but ignore the cross-modal cues in the decoder. Moreover, when a large number of encoder parameters are passed to the decoder, it is difficult to train such a model to converge quickly.

Multi-task Learning. Numerous works [13, 26, 52, 55, 35, 56, 58, 47] have also explored the idea of combining networks for complementary tasks to improve learning efficiency and generalization across different tasks. For example, Eigen *et al.* [13] proposed a single multi-scale network (MSCNN) to address three different computer vision tasks. Zhang *et al.* [55] proposed a joint task-recursive learning (TRL) framework to refine the results of both semantic segmentation and monocular depth estimation through serialized task-level interactions. Zhang *et al.* [56] proposed a pattern affinitive propagation (PAP) method to utilize the matched affinity information across tasks. Zhou *et al.* [58] proposed intra-task and inter-task pattern-structure diffusion (PSD) to learn long-distance propagation and transfer cross-task structures. Different from the previous works, we incorporate multi-modal information in the both encoder and decoder through attention-based dual supervised decoder to provide a unified pixel-wise scene understanding.

3. Method

In this section, we describe the proposed attention-based dual supervised decoder (ADSD) in detail. First, we briefly describe the overall architecture. Then, we discuss multi-level fusion strategy and attention block used in attention-based fusion module for multi-modal features in the encoder. Moreover, we give a detailed depiction of our dual-branch decoder which significantly improves the performance of semantic segmentation. Finally, we introduce the objective function for optimizing the network.

3.1. The Network Architecture

The entire network architecture of our ADSD is presented in Fig. 2. For clear illustration, we use blocks with different colors to indicate different layers. Note that each convolution layer in our network is followed by a batch normalization layer [22] before the activated function of rectified linear unit (ReLU), and it is omitted in the figure for simplification. The whole network can be divided into a two-stream encoder and a dual-branch decoder. In the decoder, the primary branch with pyramid supervision is designed for semantic segmentation, and the secondary branch requires supervision from the other task such as normal estimation, depth estimation, or semantic segmentation.

In the encoder part, we design two independent branches to extract features from RGB and depth images separately. In these two branches, we simply choose ResNet-50 [18] as the backbone to extract multi-scale hierarchical feature maps from inputs. The output features from RGB and depth branch are combined to produce fusion features (Fuse0∼Fuse4) through the attention-based multi-modal fusion (AMF) module, where the details are given in Section 3.2. It is worth noting that there is no connection between fusion features at different scales.

In the decoder part, we feed the above fusion features into each task-branch to decode pixel-level information. To produce high resolution predictions, we decode these convolutional features and then combine with the same scale fused features by upsampling blocks to produce task-specific features as shown in Section 3.3. Specially, at the end of the primary branch, the ASPP is introduced to improve the final segmentation performance. Meanwhile, each upsampling block predicts a side output for pyramid supervision, which are introduced in Section 3.4.

3.2. Encoder

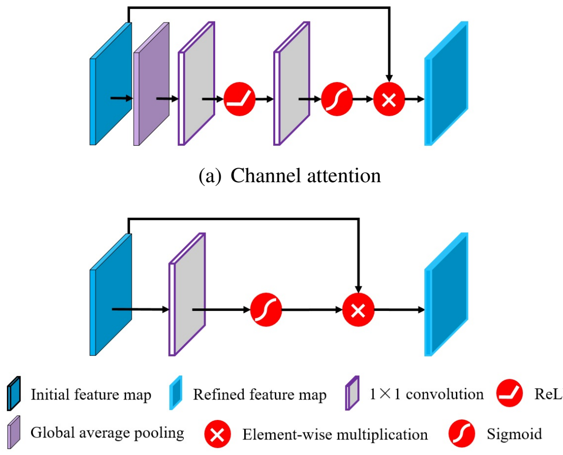
The conventional fusion branch [21, 45] integrates multi-scale features by coarse-to-fine CNNs and general attention mechanisms. Such approaches are computationally expensive leading to information redundancy easily. Considering the complementarity between paired RGB and depth cues in multiple layers, we design a simple yet effective AMF module to fully extract and fuse multi-level paired complementary information. As illustrated in the middle part of Fig. 2, we show the main process of AMF while leveraging the high performance of the attention block. In our implementation, our AMF includes all five scales (*i.e.*1/2, 1/4, 1/8, 1/16, 1/32) of the backbone network.

To improve the performance of semantic segmentation, the channel attention [54] allows the network to concentrate on more useful channels and flattens the distribution of information among channels with the effective utilization of complementary features. The architecture of the channel attention is illustrated in Fig. 3(a). Assuming an input feature map $U = [u_1, u_2, ..., u_C] \in \mathbb{R}^{C\times H\times W}$ that passes through channel attention block $A_C(\cdot)$ to generate output feature map $V_{ac} \in \mathbb{R}^{C\times H\times W}$. Here, H and W are the height and width respectively, with C being the number of channels. Channel attention is firstly performed by a global average pooling to produce a vector $Z \in \mathbb{R}^{C\times 1\times 1}$ with its t-th element

$$Z_t = \frac{1}{H\times W}\sum_{i}^{H}\sum_{j}^{W} u_t(i,j)\ . \qquad (1)$$

Then Z is transformed to $\hat{Z} = W_{1\times1}(\delta(W_{1\times1}(Z)))$, with $W_{1\times1}$ being the weight of a 1×1 convolutional layer and the ReLU operator $\delta(\cdot)$. A sigmoid $\sigma(\hat{Z})$ is applied to activate the convolution result, constraining the value of weight vector to the interval [0,1]. Finally, we perform an element-wise multiplication, and the result V_{ac} can be expressed as:

$$V_{ac} = A_C(U) = [\sigma(\hat{Z}_1)u_1, \sigma(\hat{Z}_2)u_2, ..., \sigma(\hat{Z}_C)u_C] \qquad (2)$$


(a) Channel attention

(b) Spatial attention

Figure 3: Detailed structure of channel attention and spatial attention.

In contrast to channel attention, spatial attention [41, 20] has fewer parameters with a simpler structure. The architecture of the spatial attention is illustrated in Fig. 3(b). We consider an alternative slicing of an input tensor $U = [u^{1,1}, ..., u^{i,j}, ..., u^{H,W}]$ that passes through the spatial attention block $A_S(\cdot)$ to generate output V_{sc}, where $u^{i,j} \in \mathbb{R}^{C\times 1\times 1}$ corresponding to the spatial location (i,j). The spatial attention is firstly performed by a 1×1 convolution to generate a projection tensor $Q \in \mathbb{R}^{H\times W}$. Each $Q_{i,j}$ of the projection describes the linearly combined representation of a spatial location (i,j). This projection is then performed on a sigmoid $\sigma(\cdot)$ to rescale activations to [0,1].

And the result V_{sc} can be expressed as

$$V_{sc} = A_S(U) = [\sigma(Q_{1,1})u^{1,1}, \sigma(Q_{1,2})u^{1,2}, ..., \\ \sigma(Q_{i,j})u^{i,j}, ..., \sigma(Q_{H,W})u^{H,W}] \ . \tag{3}$$

Specifically, this operation provides more importance to relevant spatial locations and ignores irrelevant ones.

3.3. Decoder

Benefit from the exploration of correlation between different tasks in multi-task learning [52, 56, 58], we propose a novel dual-branch decoder to learn more robust deep representations and multi-modal information. It is well-known that low-level layers of the CNNs usually have more positional information, while high-level layers contain more semantic cues. Both the positional and semantic cues play a key role in semantic segmentation. Inspired by upsampling strategy in [21] and skip connection like [18], we use transposed convolutional layers to upsample the features at different pyramid scales, as illustrated in Fig. 4.

In particular, the fused feature map V'_K of AMF is firstly calculated by a 1×1 convolution $W_{1\times1}$ to project the feature map $W_{1\times1}(V'_K)$ with lower channel, allowing the decoder to have a lower memory consumption. And it passes through upsampling block $B_U(\cdot)$ to generate the feature map S_K of K-th slide output.

$$S_K = B_U[W_{1\times1}(V'_K)] \ . \tag{4}$$

Then S_K is used to produce the next slide output as follows:

$$S_{K-1} = S_K \oplus B_U[W_{1\times1}(V'_{K-1})] \ , \tag{5}$$

where $\oplus$ is element-wise summation. Repeatedly, we continue to upscale feature maps and perform the above decoding process to produce a higher scale of feature maps. The scale factor of each upsampling block is set to 2. All slide outputs are employed for pyramid supervision which will be introduced in Section 3.4. In particular, the ASPP is introduced to incorporate multi-scale context at the end of this branch. In our experiments, the dilated convolution rate is set as 12, 24, and 36 in the ASPP.

In secondary branch of decoder, we repeat the operations on the primary branch to upsample the fused feature map V'_a. The final upsampling feature map S_0 is directly used to generate the predict which can be surface normal, estimated depth, or segmentation result. In practice, we propose a more efficient training method that takes advantage of multi-modal feature sharing during training. Inspired by the training strategy in [15], we train the model for a depth-guided branch decoder at the pre-training stage and the semantic-guided branch decoder at the fine-tuning stage.

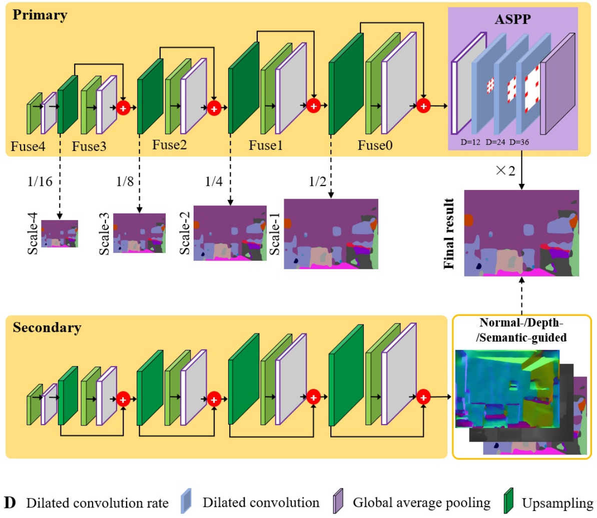


Figure 4: Detailed diagram of the proposed dual-branch decoder. The primary branch computes multi-scale fused features through 1×1 convolutional layers and upsampling blocks. The final features is refined by the ASPP to predict segmentation result, which is supervised via the output generated by normal-/depth-/semantic-guided branch.

3.4. Objective Function

Pyramid Supervision. The pyramid supervised training scheme alleviates the gradient disappearance problem by introducing supervised learning at different levels [23]. As shown in Fig. 4, the primary-branch of decoder computes K slide outputs by upsampling blocks with different spatial resolutions. In our implementation, the K is set to 4, and the slide outputs are defined as Scale 1 to 4. The resolution scales are 1/2, 1/4, 1/8, and 1/16, and the final result is a full resolution. We calculate the score map of each output through a 1×1 convolution, and then feed it into a softmax layer and cross-entropy function to build the loss function L_{P_k} ($k \in [1, K]$).

Loss Function. For semantic segmentation, most methods utilize cross-entropy to measure the difference between the prediction and ground-truth. However, for existing datasets, the distribution of semantic labels is extremely imbalanced. This will bias the learning towards the dominant samples and lead to low accuracy in minority categories. To alleviate the data imbalance issues, we re-weight the training loss of each class in the cross-entropy function using the median frequency setting proposed in [13, 23]. That is, we weight each pixel by a factor of $\alpha_c = p_m/p_c$, where c denotes the ground-truth category. p_c is the pixel probability of that category, p_m is the median of all the probabilities of these categories.

For different task supervision, we use task-guided loss functions defined as L_T which can be normal L_N, depth L_D

or semantic L_S. Following the depth estimation algorithms, we use berHu loss [27] for the depth supervision:

$$L_D = \sum_i \begin{cases} |d_i - D_i|, & |d_i - D_i| \leqslant \beta \\ \frac{(d_i - D_i)^2 + \beta^2}{2\beta}, & |d_i - D_i| > \beta \end{cases}, \tag{6}$$

where d_i is the predicted depth for pixel i, and D_i is the ground-truth. $\beta = \frac{1}{5} max(|d_i - D_i|)$. Such a loss function can provide more obvious gradients at the locations where the depth difference is low, and thus can help to better train the network. As for surface normal, we also use the berHu loss [27]. Together with the above pyramid supervision loss L_{P_k} for semantic prediction at intermediate layers, the total loss L can be defined as:

$$L = L_S + L_T + \sum_{k=1}^{K} L_{P_k} \ . \tag{7}$$

Finally, a fully end-to-end optimization is computed by using gradient back-propagation.

4. Experiments

To evaluate our proposed method, we conduct extensive experiments on NYUDv2 dataset [34] and SUN-RGBD dataset [43]. We start with the introduction of experimental setup such as implementation details, datasets, and evaluation metrics. We then conduct ablation experiments to determine whether our network improve performance. Finally, we compare our method with the existing methods for semantic segmentation on these datasets.

4.1. Implementation Details

We implement our method using the publicly available Pytorch. For the optimizer, we use Adam [25] with $(\beta_1, \beta_2) = (0.9, 0.999)$. For NYUDv2 dataset, we train the model for 600 epochs and fine-tune 50 epochs with a learning rate of 0.0002 and 0.00002, respectively. For SUN-RGBD dataset, we train the model for 300 epochs and fine-tune it for 30 epochs with the same learning rate. We adopt the step learning rate policy whose learning rate is updated after each 300 epochs. Specifically, all experiments are trained with batch size 8 on a single NVIDIA Tesla V100 GPU. To avoid overfitting, similarly with [7, 12, 30], we employ general data augmentation strategies, including random scaling in the range of [0.8, 1.4], random horizontal flipping, and random cropping. In particular, we resized the inputs to a resolution of 480×640 for the above datasets. During the inference, we only obtain the prediction results from the primary decoder for semantic segmentation.

4.2. Datasets and Metrics

We use the NYUDv2 dataset [34] for the main evaluation of our method and further use the SUN-RGBD dataset [43] for extensive comparison with the state-of-the-arts. The NYUDv2 dataset consists of 1449 RGBD images showing interior scenes. We use the segmentation labels provided in [16], in which all labels are mapped to 40 classes. We use the standard training/test split with 795 and 654 images, respectively. The SUN-RGBD contains 10335 RGBD images labeled with 37 classes. We use the official training set with 5285 images to train our network, and the official testing set with 5050 images for evaluation. Compared with NYUDv2, SUN-RGBD has more complex scene and depth conditions, which are probably more suitable to measure the generality of our method. For the evaluation of semantic segmentation results, we follow the recent works [7, 12, 30, 56, 58] and use three common metrics for evaluation, including pixel accuracy (PixAcc.), mean accuracy (mAcc.), and mean intersection over union (mIoU).

Table 1: Performance analysis of different task-guided branches in the secondary decoder on NYUDv2 dataset. During the inference, we only obtain the prediction results from the primary decoder for semantic segmentation.

Decoder	PixAcc.	mAcc.	mIoU
Semantic-guided	75.9	61.6	49.0
Depth-guided	76.8	64.6	51.2
Normal-guided	**77.3**	**64.7**	**51.5**
Depth-guided+ Normal-guided	76.8	64.0	51.0

Table 2: Performance analysis for the location of ASPP module (at the end of different Fuse modules) in the primary decoder on NYUDv2 dataset.

ASPP Location	PixAcc.	mAcc.	mIoU
with Fuse2	76.2	63.5	50.1
with Fuse1	76.4	64.2	50.4
with Fuse0 (Fig. 4)	**77.3**	**64.7**	**51.5**

4.3. Ablation study

To discover the functionality of each component in our method, we conduct an ablation study on the NYUDv2 dataset. Taking the network consisting of a two-stream encoder and a simple decoder (see Fig. 1(b)) as a baseline. In the encoder, the combination operation of multi-modal features is the element-wise summation (like FuseNet [17]). We evaluate the effectiveness of our dual-branch decoder by choosing different task-guided branches and changing the location of AASP module. The results can be found in Table 1 and Table 2. For task-guided branches of the

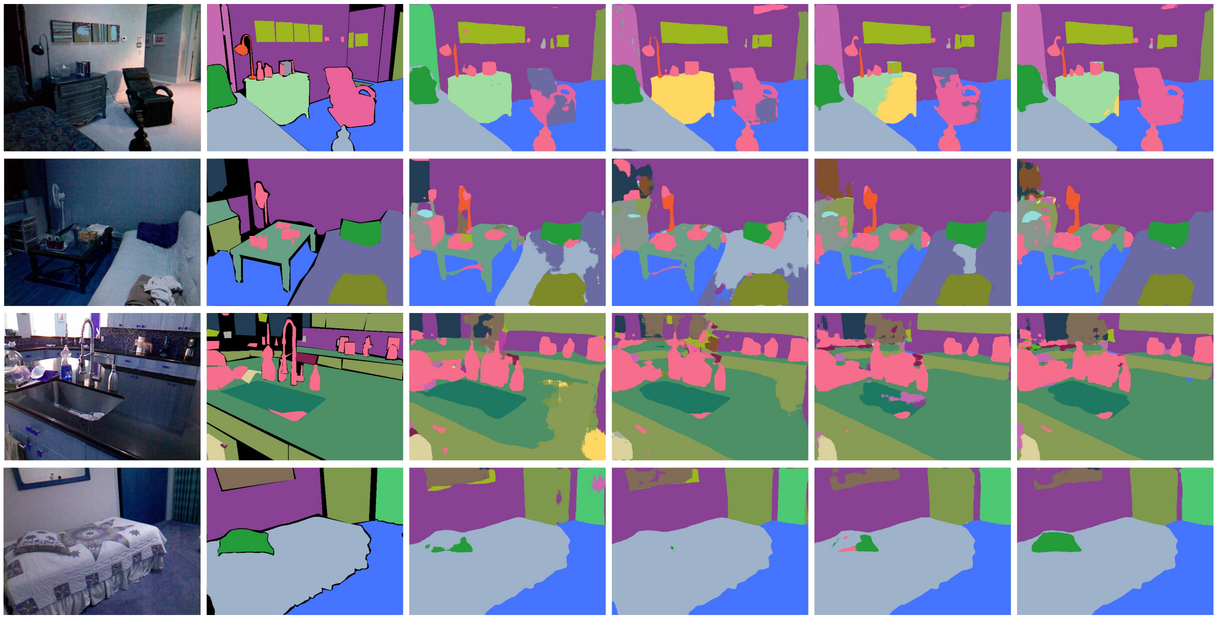


Figure 5: The visual results of ablation analysis on NYUDv2 dataset. From left to right, we show the inputs, ground-truths, the results of baseline, with AMF, with AMF and dual-branch decoder, and our method, respectively.

Table 3: Ablation study of the proposed method on NYUDv2 dataset. The Dual-decoder means dual-branch decoder. The FT means fine-tuning stage in our training method.

Method	PixAcc.	mAcc.	mIoU
Baseline (Fig. 1(b))+L_S	76.4	61.9	49.3
+AFF [10]+L_S	77.1	63.9	50.9
+SA-Gate [7]+L_S	73.3	58.3	45.4
+AMF(SA [41])+L_S	75.9	61.1	48.2
+AMF(CA [54])+L_S	77.2	63.3	51.0
+AMF(BAM [37])+L_S	76.7	64.5	50.8
+AMF(CA)+Dual-decoder+L	77.3	64.7	51.5
+AMF(CA)+Dual-decoder+FT+L	**77.5**	**65.3**	**52.5**

secondary decoder, the normal-guided performs better than the depth-guided, semantic-guided and the combination of depth-guided and semantic-guided. For the location of ASPP (at the end of different Fuse modules) in the primary decoder, the ASPP with Fuse0 (in Fig. 4) performs better than the ASPP with Fuse1 and Fuse2 modules.

The results of ablation analysis are shown in Table 3. For the multi-model feature fusion solutions (Fig. 3), our AMF with channel attention (CA) performs better than the element-wise summation (*i.e.* baseline), separation-and-aggregation gate [7], attentional feature fusion (AFF) [10], bottleneck attention module (BAM) [37], and spatial attention (SA). We owe this to the representation of channel attention that ignores less important channels of fused features and emphasizes the important ones. The results demonstrate that our AMF can significantly improve the performance of semantic segmentation. This observation also clarifies that incorporating depth information can greatly improve the performance, which reveals the effectiveness of reasoning color and geometry information together. By introducing the dual-branch decoder, the performance is further improved. The final fine-tuning (FT) stage of training strategy (see details in Section 3.3) for our model gives another rise in the performance.

We show qualitative results of our method on NYUDv2 dataset for semantic segmentation in Fig. 5. For comparison, we also include the visual results of baseline, with the proposed AMF, with the AMF and dual-branch decoder, with the AMF, dual-branch decoder and fine-tuning (FT) stage (our method). The results show that the geometry information is well distilled by our AMF, which can distinguish the objects with similar color. Moreover, when incorporating the dual-branch decoder, our network can recover more context information and more accurate object masks. As mentioned before, we argue that the training is going faster and more robust along with our dual-branch decoder. This decoder can also deal with the imbalance problem caused by the phenomenon that the encoder uses multimodal information, while the decoder dose not. To verify this statement, we report the loss values of our method and

Table 4: Comparison with state-of-the-arts on each category of the NYUDv2 dataset. Percentage (%) of IoUs are shown for evaluation, with best performance marked in **bold**.

Method	wall	floor	cabinet	bed	chair	sofa	table	door	window	bookshelf	picture	counter	blinds	desk	shelves	curtain	dresser	pillow	mirror	floormat
DeepLab [3]	67.9	83.0	53.1	66.8	57.8	57.8	43.4	19.4	45.5	41.5	49.3	58.3	47.8	15.5	7.3	32.9	34.3	40.2	23.7	15.0
FCN [42]	69.9	79.4	50.3	66.0	47.5	53.2	32.8	22.1	39.0	36.1	50.5	54.2	45.8	11.9	8.6	32.5	31.0	37.5	22.4	13.6
Mutex Constraints [11]	65.6	79.2	51.9	66.7	41.0	55.7	36.5	20.3	33.2	32.6	44.6	53.6	49.1	10.8	9.1	47.6	27.6	42.5	30.2	32.7
BI (3000) [14]	61.7	68.1	45.2	50.6	38.9	40.3	26.2	20.9	36.0	34.4	40.8	31.6	48.3	9.3	7.9	30.8	22.9	19.5	13.9	16.1
LSD-GF [8]	78.5	87.1	56.6	70.1	65.2	63.9	46.9	35.9	47.1	**48.9**	54.3	66.3	51.7	20.6	13.7	49.8	43.2	50.4	48.5	32.2
STD2P[19]	72.7	85.7	55.4	73.6	58.5	60.1	42.7	30.2	42.1	41.9	52.9	59.7	46.7	13.5	9.4	40.7	44.1	42.0	34.5	35.6
RDFNet [28]	79.7	87.0	60.9	73.4	64.6	65.4	50.7	39.9	**49.6**	44.9	61.2	67.1	**63.9**	**28.6**	14.2	59.7	49.0	49.9	54.3	**39.4**
DeepLab-LFOV [4]	70.2	85.2	55.3	68.9	60.5	59.8	44.5	25.4	47.8	42.6	47.9	57.7	52.4	20.7	9.1	36.0	36.9	41.4	32.5	16.0
DeepLabV3 [5]	78.8	83.4	56.7	61.9	57.0	59.4	41.3	39.9	44.5	45.1	60.3	56.9	54.9	22.9	14.2	52.4	40.6	40.1	31.3	30.8
DCN [9]	77.0	83.0	56.4	64.7	57.0	60.8	39.9	35.5	44.6	44.7	59.3	55.8	59.9	20.3	12.3	55.9	51.2	39.8	36.2	34.2
VCD [51]	78.2	83.7	57.4	66.1	57.2	60.9	40.1	39.5	45.1	46.8	59.4	58.1	56.6	21.9	16.0	55.2	47.0	42.7	36.2	34.3
ADSD (Ours)	**82.3**	**87.7**	**66.5**	**78.2**	**66.1**	**68.3**	**48.0**	**44.4**	48.8	47.1	**63.9**	**71.6**	58.4	28.5	**19.7**	**66.9**	**60.0**	**51.7**	**58.4**	33.7

Method	clothes	ceiling	books	fridge	tv	paper	towel	shower	box	board	person	nightstand	toilet	sink	lamp	bathtub	bag	ot. struct.	ot. furn.	ot. props.
DeepLab [3]	20.2	55.1	22.1	30.6	49.4	21.8	32.1	6.4	5.8	14.8	55.3	37.7	57.9	47.7	40.0	44.7	6.6	18.0	12.9	33.8
FCN [42]	18.3	59.1	27.3	27.0	41.9	15.9	26.1	14.1	6.5	12.9	57.6	30.1	61.3	44.8	32.1	39.2	4.8	15.2	7.7	30.0
Mutex Constraints [11]	12.6	56.7	8.9	21.6	19.2	28.0	28.6	22.9	1.6	1.0	9.6	30.6	48.4	41.8	28.1	27.6	0	9.8	7.6	24.5
BI (3000) [14]	13.7	42.5	21.3	16.6	30.9	14.9	23.3	17.8	3.3	9.9	44.7	15.8	53.8	32.1	22.8	19.0	0.1	12.3	5.3	23.2
LSD-GF [8]	24.7	62.0	34.2	45.3	53.4	27.7	**42.6**	23.9	11.2	58.8	53.2	54.1	**80.4**	59.2	45.5	52.6	15.9	12.7	16.4	29.3
STD2P[19]	22.2	55.9	29.8	41.7	52.5	21.1	34.4	15.5	7.8	29.2	60.7	42.2	62.7	47.4	38.6	28.5	7.3	18.8	5.1	31.4
RDFNet [28]	**26.9**	69.1	**35.0**	**58.9**	63.8	**34.1**	41.6	38.5	11.6	54.0	**80.0**	45.3	65.7	62.1	**47.1**	57.3	**19.1**	30.7	20.6	39.0
DeepLab-LFOV [4]	17.8	58.4	20.5	45.1	48.0	21.0	41.5	9.4	8.0	14.3	67.0	41.8	69.7	46.8	40.1	45.1	2.1	20.7	12.4	33.5
DeepLabV3 [5]	20.7	69.8	30.3	42.8	52.5	27.7	33.2	24.5	13.6	68.9	73.3	37.7	65.1	51.3	39.2	36.4	12.5	27.7	15.2	36.6
DCN [9]	22.3	63.3	26.9	52.8	58.7	29.9	39.8	40.4	14.9	65.3	76.2	39.9	67.1	50.3	38.7	40.1	7.3	26.7	16.5	36.9
VCD [51]	22.2	67.0	30.0	50.9	57.0	30.7	36.7	40.6	**15.6**	72.6	77.5	41.2	69.1	51.8	43.0	39.4	9.5	27.7	18.3	37.0
ADSD (Ours)	24.0	**76.0**	32.9	57.8	**70.8**	28.6	40.3	**48.2**	12.1	**78.3**	67.3	**57.1**	77.9	**63.2**	46.5	**62.2**	9.6	**33.4**	**22.2**	**39.6**

the methods without a dual-branch decoder during training. As shown in Fig. 6, we observe that the multi-loss L of our method is rapidly reduced only after 150 epochs from the beginning. However, the loss of the methods without a dual-branch decoder waves violently due to the imbalance of encoder and decoder. We also find that the dual-branch decoder can facilitate the convergence of training, which is capable of reducing the adverse effects on this imbalance.

To evaluate the performance of our model on the imbalanced distributed data, we also show the results on each category, as shown in Table 4. Clearly, our method performs better than other methods in most categories. Specially, our method still achieves a relatively higher IoU on some "hard" categories such as bed, curtain, dresser, shower, and board, etc. Following previous methods [43, 31, 40, 8], we also report the mACC. of our method on SUN-RGBD dataset. As shown in Table 5, we achieve 62.1% mean accuracy with 4.1% improvement over the recent method [8]. Specifically, we yield performance gains over 26 classes, which demonstrates the effectiveness of the proposed approach. We owe the robustness among almost all the categories to the effectively learned multi-modal cues in the encoder, and the cross-modal information in the decoder. Note that our method achieves unsatisfactory performance on some categories (*e.g.* blinds, person, bag) of NYUDv2 dataset, which may due to our joint reasoning on color and geometric cues, as the depth may vary greatly compared with the corresponding color appearance in different scenes.

4.4. Compared with State-of-the-arts

NYUDv2. The comparison results on the NYUDv2 dataset with 40-category are shown in Table 6. We use ResNet-50 and single-scale inference strategy for a fair comparison. Following our training method mentioned in Section 3.3, we use a normal-guided branch decoder at the pre-training stage and the semantic-guided branch decoder at the fine-tuning stage. Our method can still achieve 77.5% PixAcc., 65.3% mAcc., and 52.5% mIoU, which is better than the

Table 5: Comparison with state-of-the-arts on each category of the SUN-RGBD dataset. Percentage (%) of IoUs are shown for evaluation, with best performance marked in **bold**.

Method	wall	floor	cabinet	bed	chair	sofa	table	door	window	bookshelf	picture	counter	blinds	desk	shelves	curtain	dresser	pillow	mirror
Song *et al.* [43]	36.4	45.8	15.4	23.3	19.9	11.6	19.3	6.0	7.9	12.8	3.6	5.2	2.2	7.0	1.7	4.4	5.4	3.1	5.6
Liu *et al.* [31]	37.8	48.3	17.2	23.6	20.8	12.1	20.9	6.8	9.0	13.1	4.4	6.2	2.4	6.8	1.0	7.8	4.8	3.2	6.4
Ren *et al.* [40]	43.2	78.6	26.2	42.5	33.2	40.6	34.3	33.2	43.6	23.1	57.2	31.8	42.3	12.1	18.4	59.1	31.4	49.5	24.8
DeconvNet [8]	90.4	92.7	57.7	75.9	83.0	61.2	64.2	43.0	64.7	42.3	59.8	42.5	48.3	29.5	17.5	64.9	54.0	61.7	51.3
LSD-GF [8]	91.9	94.7	61.6	82.2	**87.5**	62.8	68.3	47.9	68.0	48.4	69.1	49.4	51.3	**35.0**	**24.0**	68.7	60.5	**66.5**	57.6
ADSD (Ours)	**92.1**	**96.0**	**70.9**	**84.0**	86.7	**74.5**	**72.5**	**58.5**	**70.4**	**51.7**	**71.8**	**57.0**	**54.3**	29.6	21.6	**78.1**	**67.2**	64.9	**64.0**

Method	floormat	clothes	ceiling	books	fridge	tv	paper	towel	shower	box	board	person	nightstand	toilet	sink	lamp	bathtub	bag	mACC.
Song *et al.* [43]	0	1.4	35.8	6.1	9.5	0.7	1.4	0.2	0.0	0.6	7.6	0.7	1.7	12.0	15.2	0.9	1.1	0.6	9.0
Liu *et al.* [31]	0	1.6	49.2	8.7	10.1	0.6	1.4	0.2	0.0	0.8	8.6	0.8	1.8	14.9	16.8	1.2	1.1	1.3	10.1
Ren *et al.* [40]	**5.6**	27.0	84.5	35.7	24.2	36.5	26.8	19.2	9.0	11.7	51.4	35.7	25.0	64.1	53.0	44.2	47.0	18.6	36.3
DeconvNet [8]	0.4	39.8	78.3	55.0	43.9	59.6	29.4	45.2	1.5	35.9	47.7	45.3	36.0	77.6	66.6	51.2	66.1	35.8	51.9
LSD-GF [8]	0	44.7	**88.8**	**61.5**	51.4	**71.7**	37.3	**51.4**	2.9	**46.0**	54.2	49.1	44.6	82.2	74.2	**64.7**	77.0	**47.6**	58.0
ADSD (Ours)	0	**55.2**	87.6	59.6	**66.9**	68.6	**43.4**	49.8	**29.5**	45.9	**64.6**	**62.6**	**46.8**	**88.1**	**76.4**	62.8	**84.2**	42.6	**62.1**

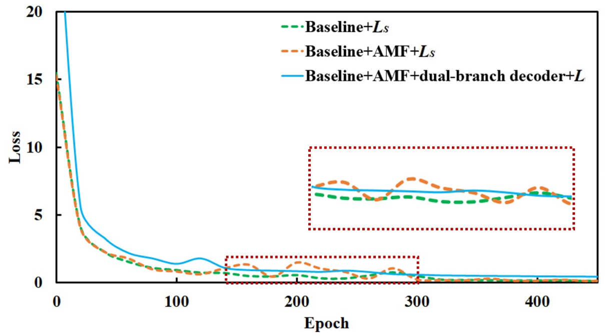


Figure 6: Statistics of loss values during a training procedure on NYUDv2 dataset. Our dual-branch decoder can reduce the adverse effects on imbalance between encoder and decoder to facilitate the convergence of training.

state-of-the-art methods. Specifically, we can find that utilizing depth and normal as extra supervision could make network more robust than general RGBD methods that take both RGB and depth as inputs. Besides, it can be observed that the methods try to use atrous/dilated convolution or gate fusion to extract complementary feature, which are more implicit than our model in selecting valid feature from complementary information.

SUN-RGBD. We also compare our method with the state-of-the-arts on the large-scale SUN-RGBD dataset. Due to the lacking of surface normal ground-truths, we use a depth-guided branch decoder at the pre-training stage and the semantic-guided branch decoder at the fine-tuning stage. As summarized in Table 6, our ADSD achieves 81.8% PixAcc., 62.1% mAcc., and 49.6% mIoU, which is the best results on mAcc. in comparison with the pervious methods. Moreover, our method obtains the superior performance than the approaches based on atrous/dilated convolution and encoder-decoder network, suggesting its superiority and high performance for RGBD semantic segmentation. We can observe that our proposed ADSD is slightly weaker than multi-task learning based methods such as PAP [56] and PSD [58] on both PixAcc. and mIoU metrics. The main reason is that we perform the dual-supervised decoder with a depth-guided branch at the pre-training stage on SUN-RGBD dataset. Note that there are many low-quality depth maps in SUN-RGBD dataset caused by the capture device [43, 28], which may affect the auxiliary utility from the depth. More details of qualitative results are shown in the supplementary material.

5. Conclusions

In this paper, we have proposed a novel encoder-decoder framework for RGBD semantic segmentation, which can take full advantage of the complementary information across modalities. The color and depth data were jointly reasoned by forming a two-stream encoder. The multi-level paired complementary cues can be processed by our pro-

Table 6: Comparison with state-of-the-arts on NYUDv2 test set in 40-class and SUN-RGBD test set in 37-class. Percentage (%) of PixAcc., mAcc., and mIoU are shown for evaluation. In category, the 'AC', 'ED', and 'MT' denote atrous/dilated convolution, encoder-decoder, and encoder-decoder for multi-task, respectively. In scale, the 'S', and 'M' denote single-scale inference strategy and multi-scale inference strategy, respectively.

Method	Category	Data	Backbone	Scale	NYUDv2 (40-class)			SUN-RGBD (37-class)		
					PixAcc.	mAcc.	mIoU	PixAcc.	mAcc.	mIoU
DeepLab [3]	AC	RGBD	VGG-16	M	68.7	46.9	36.8	–	–	–
BI (3000) [14]	AC	RGBD	VGG-16	S	58.9	39.3	27.7	–	–	–
CFN[29]	AC	RGBD	VGG-16	M	–	–	41.7	–	–	42.5
3DGNN [39]	AC	RGBD	VGG-16	M	–	55.7	43.1	–	54.6	42.3
DeepLab-LFOV [4]	AC	RGBD	VGG-16	M	70.3	49.6	39.4	71.9	42.2	32.1
D-CNN [49]	AC	RGBD	VGG-16	S	–	56.3	43.9	–	53.5	42.0
RefineNet [30]	AC	RGB	ResNet-152	M	74.4	59.6	47.6	81.1	57.7	47.0
DeconvNet [36]	ED	RGB	VGG-16	S	–	–	–	66.1	32.3	22.6
FCN [42]	ED	RGBD	VGG-16	S	65.4	46.1	34.0	68.2	38.4	27.4
SegNet [1]	ED	RGB	VGG-16	S	–	–	–	72.6	44.8	31.8
B-SegNet [24]	ED	RGB	VGG-16	S	68.0	45.8	32.4	71.2	45.9	30.7
FuseNet [17]	ED	RGBD	VGG-16	S	–	–	–	76.3	48.3	37.3
LSD-GF [8]	ED	RGBD	VGG-16	S	71.9	60.7	45.9	–	58.0	–
RDFNet-152 [28]	ED	RGB	ResNet-152	M	76.0	62.8	50.1	81.5	60.1	47.7
RedNet [23]	ED	RGBD	ResNet-50	S	–	62.6	47.2	81.3	60.3	47.8
ACNet [21]	ED	RGBD	ResNet-50	S	–	63.1	48.3	–	60.3	48.1
CANet [57]	ED	RGBD	ResNet-101	S	76.6	63.8	51.2	82.5	60.5	49.3
SGNet [6]	ED	RGBD	ResNet-101	S	76.4	62.7	50.3	81.0	59.6	47.1
Malleable 2.5D [50]	ED	RGBD	ResNet-101	M	76.9	–	50.9	–	–	–
SA-Gate [7]	ED	RGBD	ResNet-101	M	–	–	52.4	–	–	49.4
ESANet [44]	ED	RGBD	ResNet-50	S	–	–	50.5	–	–	48.3
MS CNN [13]	MT	RGB	VGG-16	S	65.6	45.1	34.1	–	–	–
PU-Loop [26]	MT	RGB	ResNet-50	S	72.1	–	44.5	80.3	–	45.1
TRL [55]	MT	RGB	ResNet-50	S	76.2	56.3	46.4	83.6	58.2	49.6
PAD-Net [52]	MT	RGB	ResNet-50	S	75.2	62.3	50.2	–	–	–
RTJ-AA [35]	MT	RGB	MobileNetV2	S	–	–	42.0	–	–	–
PAP [56]	MT	RGB	ResNet-50	S	76.2	62.5	50.4	83.8	58.4	50.5
PSD [58]	MT	RGB	ResNet-50	S	77.0	58.6	51.0	**84.0**	57.3	**50.6**
MTI-Net [47]	MT	RGB	HRNet48-V2	S	75.3	62.9	49.0	–	–	–
ADSD (Ours)	ED	RGBD	ResNet-50	S	**77.5**	**65.3**	**52.5**	81.8	**62.1**	49.6

posed AMF in the encoder. We then introduced a dual-branch decoder to effectively leverage the correlation and complementation of different tasks. In the decoder, the primary branch was used to incorporate multi-scale context by the ASPP with pyramid supervision. In addition, it was further supervised by another task-branch like normal estimation to improve the performance of segmentation and training convergence speed. Experiments on NYUDv2 and SUN-RGBD datasets demonstrated the superiority of our method compared with the previous approaches on RGBD semantic segmentation. In the future, we will generalize our method on more vision tasks and improve its efficiency.

References

[1] V. Badrinarayanan, A. Kendall, and R. Cipolla. SegNet: a deep convolutional encoder-decoder architecture for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(12):2481–2495, 2017. 2, 3, 10

[2] F. Brickwedde, S. Abraham, and R. Mester. Mono-sf: Multi-view geometry meets single-view depth for monocular scene flow estimation of dynamic traffic scenes. In *IEEE Inter-*

national Conference on Computer Vision, pages 2780–2790, 2019. 1

[3] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Semantic image segmentation with deep convolutional nets and fully connected CRFs. In *International Conference on Learning Representations*, 2015. 1, 2, 8, 10

[4] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. DeepLab: semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(4):834–848, 2018. 2, 8, 10

[5] L. Chen, G. Papandreou, F. Schroff, and H. Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv:1706.05587*, 2017. 8

[6] L.-Z. Chen, Z. Lin, Z. Wang, Y.-L. Yang, and M.-M. Cheng. Spatial information guided convolution for real-time rgbd semantic segmentation. *arXiv:2004.04534*, 2020. 2, 3, 10

[7] X. Chen, K.-Y. Lin, J. Wang, W. Wu, C. Qian, H. Li, and Z. Gang. Bi-directional cross-modality feature propagation with separation-and-aggregation gate for rgb-d semantic segmentation. In *European Conference on Computer Vision*, 2020. 2, 6, 7, 10

[8] Y. Cheng, R. Cai, Z. Li, X. Zhao, and K. Huang. Locality-sensitive deconvolution networks with gated fusion for rgb-d indoor semantic segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1475–1483, 2017. 2, 8, 9, 10

[9] J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, and Y. Wei. Deformable convolutional networks. In *IEEE International Conference on Computer Vision*, pages 764–773, 2017. 8

[10] Y. Dai, F. Gieseke, S. Oehmcke, Y. Wu, and K. Barnard. Attentional feature fusion. In *IEEE Winter Conference on Applications of Computer Vision*, pages 3559–3568, 2021. 7

[11] Z. Deng, S. Todorovic, and L. Latecki. Semantic segmentation of rgbd images with mutex constraints. In *IEEE International Conference on Computer Vision*, pages 1733–1741, 2015. 8

[12] H. Ding, X. Jiang, B. Shuai, A. Q. Liu, and G. Wang. Semantic segmentation with context encoding and multi-path decoding. *IEEE Transactions on Image Processing*, 29:3520–3533, 2020. 6

[13] D. Eigen and R. Fergus. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In *IEEE International Conference on Computer Vision*, pages 2650–2658, 2015. 2, 3, 5, 10

[14] R. Gadde, V. Jampani, M. Kiefel, and P. V. Gehler. Superpixel convolutional networks using bilateral inceptions. In *European Conference on Computer Vision*, 2016. 1, 2, 8, 10

[15] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *EEE Conference on Computer Vision and Pattern Recognition*, pages 580–587, 2014. 5

[16] S. Gupta, P. Arbelaez, and J. Malik. Perceptual organization and recognition of indoor scenes from rgb-d images. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 564–571, 2013. 6

[17] C. Hazirbas, L. Ma, C. Domokos, and D. Cremers. FuseNet: incorporating depth into semantic segmentation via fusion-based cnn architecture. In *Asian Conference on Computer Vision*, pages 213–228, 2017. 2, 6, 10

[18] K. He, X. Zhang, S. Ren, and S. Jian. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016. 4, 5

[19] Y. He, W.-C. Chiu, M. Keuper, and M. Fritz. STD2P: rgbd semantic segmentation using spatio-temporal data-driven pooling. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 7158–7167, 2017. 8

[20] J. Hu, L. Shen, S. Albanie, G. Sun, and E. Wu. Squeeze-and-excitation networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(8):2011–2023, 2020. 4

[21] X. Hu, K. Yang, L. Fei, and K. Wang. ACNet: Attention based network to exploit complementary features for rgbd semantic segmentation. In *IEEE International Conference on Image Processing*, pages 1440–1444, 2019. 2, 3, 4, 5, 10

[22] S. Ioffe and C. Szegedy. Batch normalization: accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learning*, pages 448–456, 2015. 4

[23] J. Jiang, L. Zheng, F. Luo, and Z. Zhang. RedNet: residual encoder-decoder network for indoor rgb-d semantic segmentation. *arXiv:1806.01054*, 2018. 2, 5, 10

[24] A. Kendall, V. Badrinarayanan, and R. Cipolla. Bayesian SegNet: Model uncertainty in deep convolutional encoder-decoder architectures for scene understanding. In *British Machine Vision Conference*, 2017. 2, 10

[25] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv:1412.6980*, 2014. 6

[26] S. Kong and C. Fowlkes. Recurrent scene parsing with perspective understanding in the loop. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 956–965, 2018. 2, 3, 10

[27] I. Laina, C. Rupprecht, V. Belagiannis, F. Tombari, and N. Navab. Deeper depth prediction with fully convolutional residual networks. In *International Conference on 3D Vision*, pages 239–248, 2016. 6

[28] S. Lee, S. Park, and K. Hong. RDFNet: rgb-d multi-level residual feature fusion for indoor semantic segmentation. In *IEEE International Conference on Computer Vision*, pages 4990–4999, 2017. 2, 3, 8, 9, 10

[29] D. Lin, G. Chen, D. Cohen-Or, P.-A. Heng, and H. Huang. Cascaded feature network for semantic segmentation of rgb-d images. In *IEEE International Conference on Computer Vision*, pages 1320–1328, 2017. 1, 2, 10

[30] G. Lin, F. Liu, A. Milan, C. Shen, and I. Reid. RefineNet: multi-path refinement networks for dense prediction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(5):1228–1242, 2020. 1, 2, 6, 10

[31] C. Liu, J. Yuen, and A. Torralba. SIFT Flow: dense correspondence across scenes and its applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(5):978–994, 2011. 8, 9

[32] L. Ma, C. Kerl, J. Stückler, and D. Cremers. CPA-SLAM: consistent plane-model alignment for direct rgb-d slam. In

IEEE International Conference on Robotics and Automation, pages 1285–1291, 2016. 1

[33] N. Marchal, C. Moraldo, H. Blum, R. Siegwart, C. Cadena, and A. Gawel. Learning densities in feature space for reliable segmentation of indoor scenes. *IEEE Robotics and Automation Letters*, 5(2):1032–1038, 2020. 1

[34] S. Nathan, H. Derek, K. Pushmeet, and F. Rob. Indoor segmentation and support inference from RGBD images. In *European Conference on Computer Vision*, 2012. 6

[35] V. Nekrasov, T. Dharmasiri, A. Spek, T. Drummond, C. Shen, and I. Reid. Real-time joint semantic segmentation and depth estimation using asymmetric annotations. In *International Conference on Robotics and Automation*, pages 7101–7107, 2019. 2, 3, 10

[36] H. Noh, S. Hong, and B. Han. Learning deconvolution network for semantic segmentation. In *IEEE International Conference on Computer Vision*, pages 1520–1528, 2015. 2, 3, 10

[37] J. Park, S. Woo, J. Lee, and I. S. Kweon. BAM: bottleneck attention module. In *British Machine Vision Conference*, 2018. 7

[38] J. Ping, Y. Liu, and D. Weng. Comparison in depth perception between virtual reality and augmented reality systems. In *IEEE Conference on Virtual Reality and 3D User Interfaces*, pages 1124–1125, 2019. 1

[39] X. Qi, R. Liao, J. Jia, S. Fidler, and R. Urtasun. 3d graph neural networks for rgbd semantic segmentation. In *IEEE International Conference on Computer Vision*, pages 5209–5218, 2017. 2, 10

[40] X. Ren, L. Bo, and D. Fox. RGB-(D) scene labeling: Features and algorithms. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2759–2766, 2012. 8, 9

[41] A. G. Roy, N. Navab, and C. Wachinger. Concurrent spatial and channel ‘squeeze & excitation’ in fully convolutional networks. In *Medical Image Computing and Computer Assisted Intervention*, pages 421–429, 2018. 4, 7

[42] E. Shelhamer, J. Long, and T. Darrell. Fully convolutional networks for semantic segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(4):640–651, 2017. 2, 8, 10

[43] S. Song, S. P. Lichtenberg, and J. Xiao. SUN RGB-D: A rgb-d scene understanding benchmark suite. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 567–576, 2015. 6, 8, 9

[44] Q. Tang, F. Liu, J. Jiang, Y. Z. D. Seichter, M. Kohler, B. Lewandowski, T. Wengefeld, and H.-M. Gross. Efficient rgb-d semantic segmentation for indoor scene analysis. *arXiv:2011.06961*, 2020. 2, 3, 10

[45] Q. Tang, F. Liu, J. Jiang, and Y. Zhang. Attention-guided chained context aggregation for semantic segmentation. *arXiv:2002.12041*, 2020. 2, 4

[46] L. Teixeira, M. R. Oswald, M. Pollefeys, and M. Chli. Aerial single-view depth completion with image-guided uncertainty estimation. *IEEE Robotics and Automation Letters*, 5(2):1055–1062, 2020. 1

[47] S. Vandenhende, S. Georgoulis, and L. Van Gool. MTI-Net: multi-scale task interaction networks for multi-task learning. In *European Conference on Computer Vision*, 2020. 2, 3, 10

[48] P. Wang, P. Chen, Y. Yuan, D. Liu, Z. Huang, X. Hou, and G. Cottrell. Understanding convolution for semantic segmentation. In *IEEE Winter Conference on Applications of Computer Vision*, pages 1451–1460, 2018. 2

[49] W. Wang and U. Neumann. Depth-aware cnn for rgb-d segmentation. In *European Conference on Computer Vision*, pages 144–161, 2018. 2, 10

[50] Y. Xing, J. Wang, and Z. Gang. Malleable 2.5D convolution: Learning receptive fields along the depth-axis for rgb-d scene parsing. In *European Conference on Computer Vision*, 2020. 2, 10

[51] Z. Xiong, Y. Yuan, N. Guo, and Q. Wang. Variational context-deformable convnets for indoor scene parsing. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3991–4001, 2020. 8

[52] D. Xu, W. Ouyang, X. Wang, and N. Sebe. PAD-Net: multi-tasks guided prediction-and-distillation network for simultaneous depth estimation and scene parsing. In *IEEE Conference on Computer Vision and Pattern Recognition*, June 2018. 3, 5, 10

[53] Y. Xu, X. Zhu, J. Shi, G. Zhang, H. Bao, and H. Li. Depth completion from sparse lidar data with depth-normal constraints. In *IEEE International Conference on Computer Vision*, pages 2811–2820, 2019. 1

[54] Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, and Y. Fu. Image super-resolution using very deep residual channel attention networks. In *European Conference on Computer Vision*, pages 8–14, 2018. 4, 7

[55] Z. Zhang, Z. Cui, C. Xu, Z. Jie, X. Li, and J. Yang. Joint task-recursive learning for semantic segmentation and depth estimation. In *European Conference on Computer Vision*, 2018. 2, 3, 10

[56] Z. Zhang, Z. Cui, C. Xu, Y. Yan, N. Sebe, and J. Yang. Pattern-affinitive propagation across depth, surface normal and semantic segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 4101–4110, 2019. 2, 3, 5, 6, 9, 10

[57] H. Zhou, L. Qi, Z. Wan, H. Huang, and X. Yang. Rgb-d co-attention network for semantic segmentation. In *Asian Conference on Computer Vision*, pages 519–536, 2020. 2, 10

[58] L. Zhou, Z. Cui, C. Xu, Z. Zhang, C. Wang, T. Zhang, and J. Yang. Pattern-structure diffusion for multi-task learning. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2020. 2, 3, 5, 6, 9, 10