CCASeg: Decoding Multi-Scale Context with Convolutional Cross-Attention for Semantic Segmentation

Jiwon Yoo* Dami Ko* Gyeonghwan Kim†
Department of Electronic Engineering, Sogang University, Seoul 04107, Republic of Korea
{z1u0131, dibidimi, gkim}@sogang.ac.kr

Abstract

Capturing multi-scale context within feature maps is crucial for semantic segmentation. With the success of the Vision Transformer (ViT), recent models have been designed with transformer decoders to capture it. However, these models face limitations in utilizing diverse contextual information due to the inherent nature of the attention mechanism and structural constraints. Typically, multi-head attention which leads to similar receptive fields for each token feature is achieved at the expense of significantly increased computational cost. The nature of the structure can cause inconsistent combination of the information across different levels. To address this issue, in this paper, we propose a novel and effective decoding scheme, CCASeg, which is based on convolutional cross-attention (CCA). The proposed CCA along with the decoding structure is devised not only to capture both local and global context through convolutional kernels of various sizes, but also to achieve high efficiency by effective utilization of the cheap convolution operations. Moreover, the decoding structure, which ensures the successive combination of information across various levels, facilitates understanding of diverse contexts. Consequently, this novel decoding scheme enables feature maps to effectively learn the relationships between objects of different sizes. In this way, our proposed CCASeg outperforms previous state-of-the-art methods on popular semantic segmentation benchmarks, including ADE20K, Cityscapes, COCO-stuff, and iSAID.

1. Introduction

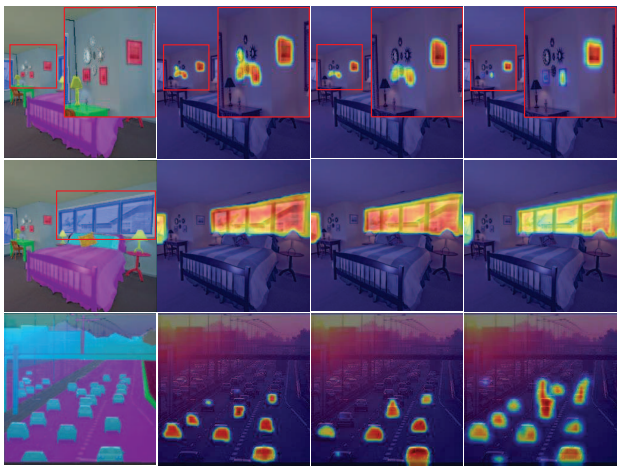
Semantic segmentation is a fundamental task in computer vision and is widely employed in numerous real-world applications, such as autonomous driving [21] and satellite imagery [29]. However, accurate pixel-wise prediction across the entire image is highly challenging, as it requires careful consideration of both global and local relationships between pixels. Recently, this difficulty has been largely alleviated with the emergence of powerful hierarchical backbones [9, 16, 24, 27] as encoders. Hierarchical backbones typically generate feature maps with different sizes across four levels, which allows the model to learn contextual diversity obtainable from the levels. Even with these advancements, these models often rely on simple multilayer perceptron (MLP) or convolutional blocks as decoders. The decoders are limited in effectively utilizing the multi-level feature maps from hierarchical backbones.

To address these issues, recent studies have designed transformer-based decoders [1, 4, 18, 25]. These methods utilize multi-head attention (MHA) to separate tokens from the multi-level feature maps and progressively update these token features within the transformer decoders. The MHA can effectively model long-range dependencies of the tokens and gradually expand the sizes of their receptive fields by aggregating information from other tokens.

However, the transformer decoder models often overlook the multi-scale attributes of objects, making them vulnerable to images with objects of distinct sizes. These limitations are mainly attributed to the fundamental nature of the attention mechanism and the structural properties of the models. The existing MHA-based attention mechanisms depend on static receptive fields for tokens and uniform information granularity within the attention layer. In addition, the structure of these models limits the consistent combination of contextual information throughout the decoder. Hence, these issues restrict the ability to capture contexts across various scales.

In this paper, we introduce a novel and comprehensive decoding scheme: **CCASeg**-**C**onvolutional **C**ross-**A**ttention for semantic **Seg**mentation. The key idea in CCASeg is to capture multi-scale context through convolutional cross-attention (CCA) and a unique structure devised to accomodate the mechanism. In contrast to the MHA approach, which depends on the limited receptive field of tokens and requires a large amount of computation proportional to the number of tokens, CCA not only captures a wide range of

*These authors contributed equally.
†Corresponding author.



(a) Ground Truth (b) SegFormer (c) FeedFormer (d) CCASeg

Figure 1. Attention maps of SegFormer [24], FeedFormer [18], and CCASeg: CCASeg captures multi-scale context effectively using convolutional kernels of various sizes and has a structure that ensures the successive combination of this context. In contrast, simple MLP in SegFormer and token-based MHA in FeedFormer capture limited context and have structures that restrict further combination.

contextual information using convolutional kernels of various sizes but also achieves computational efficiency by effective utilization of the cheap convolutional operations. Moreover, our structure leads to successive combination of diverse contextual information. These advancements enable CCASeg to capture objects of various sizes more effectively, increasing segmentation accuracy while also considerably reducing the computational cost.

Fig. 1 illustrates the objectives of the proposed scheme by comparing the attention maps of CCASeg with those of SegFormer [24] (simple MLP decoder) and FeedFormer [18] (transformer decoder). All models effectively attend to large objects (e.g., windows) in images with varying sizes of different objects. However, both the simple MLP and the transformer decoder struggle to accurately identify small objects, such as the red paintings and the silver clock, which causes difficulty in attending to the red paintings. In the bottom image in Fig. 1, which contains a class of objects with different sizes (cars in this case), CCASeg outperforms the two methods with more accurate attention map for cars. As a result, our proposed CCASeg is carefully designed to robustly attend to both large and small objects

We conduct experiments and extensive ablation studies on various public datasets to demonstrate the efficiency and effectiveness of our proposed method. To further confirm its compatibility as a decoder, we utilize various hierarchical backbones as encoders. The results show that our architecture offers advantages in both accuracy and computational cost. Our contributions can be summarized as follows.

- We propose convolutional cross-attention (CCA), which effectively and efficiently captures multi-scale objects through convolutional kernels of various sizes.
- Based on CCA, we designed a novel architecture, CCASeg, which enables the combination of diverse contextual information and enhances the ability to recognize a wide range of objects.
- We demonstrate the effectiveness of our model by showing its compatibility with powerful hierarchical backbones across various datasets.

2. Related Work

2.1. Semantic Segmentation

The design of hierarchical backbones stands out as a popular approach in segmentation tasks. A typical hierarchical backbone consists of four distinct levels, with each level responsible for learning feature maps of various sizes. Following the success of Vision Transformer (ViT) [8], various works adopted hierarchical vision transformer backbones based on MHA to enhance the performance of semantic segmentation. Swin Transformer [16] was designed a powerful backbone using shifted window attention, which divides feature maps into windows. LVT [26] was designed a lite backbone using recursive atrous attention with a recursive mechanism to reduce the number of parameters. SegFormer [24], optimized for segmentation tasks, was developed a backbone that uses efficient attention with fewer computations compared to the traditional MHA. More recent approaches are not limited to transformers. They aim to overcome the high computational demands of transformer-based attention methods. SegNeXt [9] was designed a more effective backbone using a convolution-based approach. PoolFormer [27] introduced an efficient backbone using pooling instead of MHA. Unfortunately, in semantic segmentation, these powerful hierarchical backbones still face limitations in efficiency and effectiveness, owing to their reliance on simple decoders for feature maps.

2.2. Multi-Scale Decoder

The multi-scale decoder combines multi-level feature maps from hierarchical backbones. Notably, Feature Pyramid Network (FPN) [15], UPerNet [23], and MLP-decoder [24] are widely employed. However, they face limitations in effectively combining feature maps due to their reliance on simple MLP or convolutional blocks, which restrict the ability to adequately capture context. To overcome this limitation, recent models have been designed complex transformer-based decoders, which handle multi-level feature maps through MHA within the decoder. FeedFormer [18], which directly employs the feature maps from hierarchical backbones as query, key, and value, infuses local

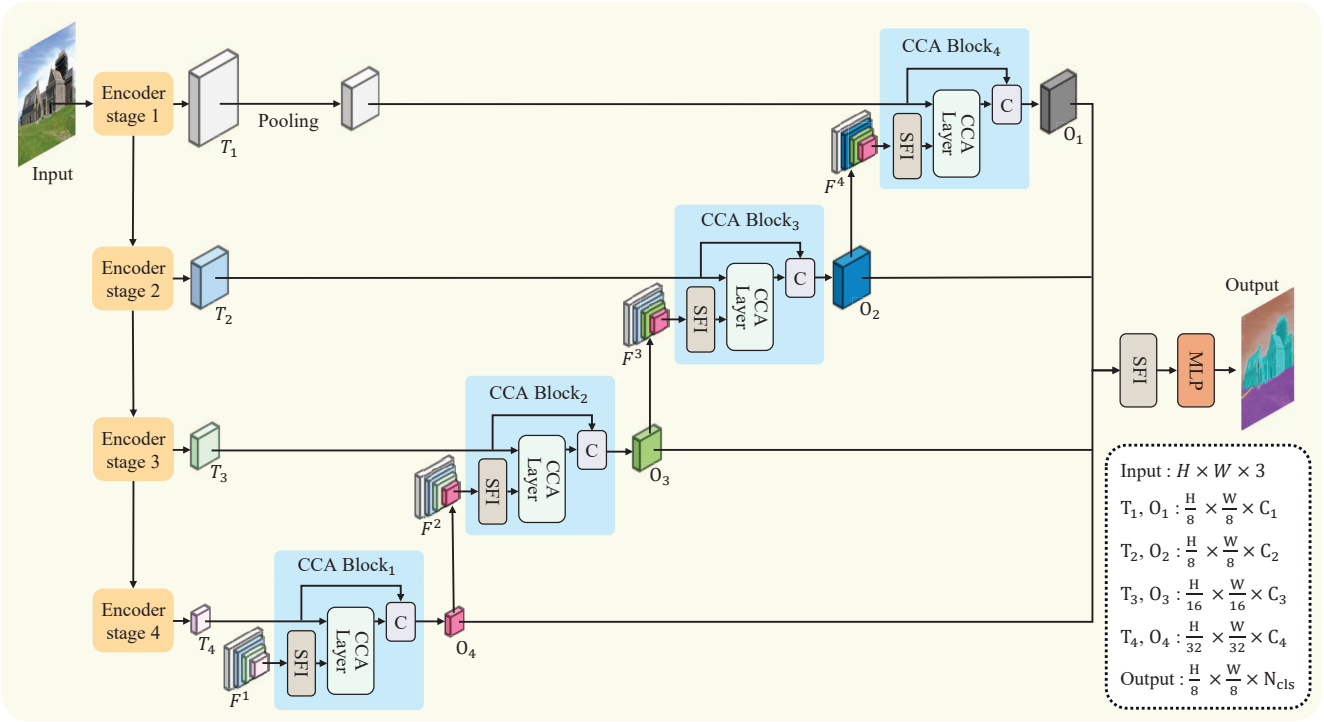


Figure 2. Overall structure of the successive combination of multi-level feature maps: The CCASeg architecture utilizes a hierarchical backbone as the encoder, which extracts the target features T_i from the input at each stage. The set of multi-level feature maps F^1 consists of all target features. The CCA Block$_i$, which composes the CCASeg decoder, takes T_{4-i+1} and F^i as inputs and generates the output O_{4-i+1}, ensuring consistent matching with multi-level information. The set F^{i+1} is formed by replacing T_{4-i+1} with O_{4-i+1}, and these gradually pass contextual information into the CCA Block$_{i+1}$.

context of the lowest-level feature map into the high-level feature maps. VWFormer [25] utilizes varying window attention to enable the feature maps from hierarchical backbones to have an appropriate receptive field. However, the reliance on the MHA-based approach, which divides feature maps into tokens of fixed sizes, limits these methods in capturing diverse contexts. Furthermore, their structure hinders the complete utilization of all multi-level contextual information within the decoder.

Therefore, CCASeg is designed to be able to effectively utilize multi-level feature maps to capture contexts of various sizes.

3. The Proposed Method

This section is for describing details of CCASeg architecture for semantic segmentation, as illustrated in Fig. 2 for the overall structure. We adopt a hierarchical backbone as the encoder to obtain multi-level feature maps. To demonstrate the compatibility of our model, we employ both CNN-based [9] and Transformer-based [24] encoders. Given an image $I \in \mathbb{R}^{H \times W \times 3}$ as input, each stage of the encoder extracts target features $T_i \in \mathbb{R}^{\frac{H}{2^{i+1}} \times \frac{W}{2^{i+1}} \times C_i}$, where $i \in \{1, 2, 3, 4\}$ and C_i denote the index of the encoder stage and the channel dimension, respectively.

To effectively capture multi-scale contexts from the feature maps extracted by the encoder, we propose a decoder composed of the CCA Blocks. The decoder successively combines the multi-level contextual information. The initial set of multi-level feature maps F^1 consists of all target features to be utilized in CCA Block$_1$. At each CCA Block, the output $O_{4-i+1} \in \mathbb{R}^{\frac{H}{2^{6-i}} \times \frac{W}{2^{6-i}} \times C_{4-i+1}}$ is generated using the target feature T_{4-i+1} and the set of multi-level feature maps F^i. As the CCA Block index increases, T_{4-i+1} in F^i is replaced with O_{4-i+1} to construct F^{i+1}. Additionally, since the size of $(\frac{H}{4} \times \frac{W}{4})$ is too large and computationally expensive, T_1 and O_1 are pooled down to $(\frac{H}{8} \times \frac{W}{8})$. The CCA Block mirrors the dimensions of the encoder stage outputs. Finally, the outputs of each CCA Block are processed by the Successive Feature Integration (SFI) and MLP to generate the segmentation map.

3.1. Convolutional Cross-Attention Block

The CCA Block, which plays a crucial role in the decoder, leverages all levels of feature maps to capture both the local and global context. As shown in Fig. 3 (a), CCA Block consists of two components, SFI and Convolutional

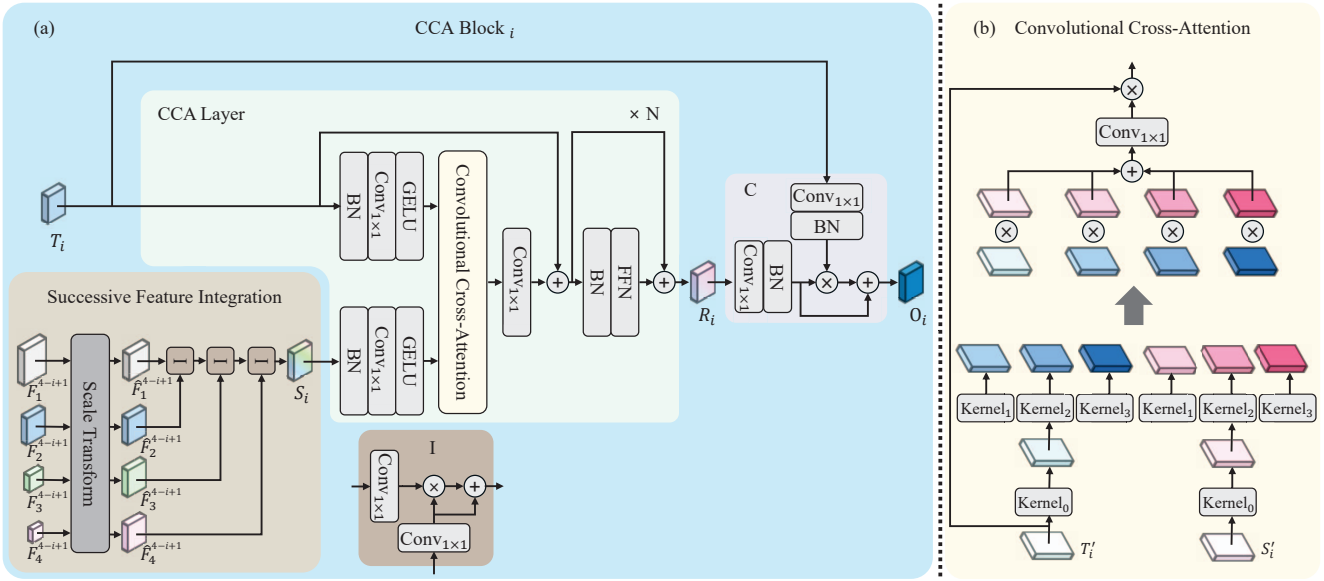


Figure 3. (a) The CCA Block$_i$ consists of the Successive Feature Integration (SFI) and the CCA Layer. In the SFI, multi-level feature maps in F^{4-i+1} are integrated successively to synthesize the source features S_i. Since S_i are required to match the size of the target features T_i, the feature maps $F_1^{4-i+1}, F_2^{4-i+1}, F_3^{4-i+1}$ and F_4^{4-i+1} are resized accordingly. In the CCA Layer, convolutional cross-attention is employed between the target features T_i and the source features S_i. Finally, refined features R_i, generated through the CCA Layer, are combined with target features T_i to produce the outputs O_i. (b) Convolutional cross-attention enhances contextual granularity of target features by matching them with source features. The kernel represents depth-wise convolution.

Cross-Attention Layer. The set of multi-level features F^i, employed as input for the CCA Block$_i$, are formalized as follows:

$$F^i = \begin{cases} \{T_j\}_{j=1}^{4} & \text{if } i = 1 \\ \{T_j\}_{j=1}^{4-i+1} \cup \{O_j\}_{j=4-i+2}^{4} & \text{otherwise} \end{cases} \quad (1)$$

In the first CCA Block$_1$, all feature maps from the encoder are selected. For subsequent blocks, the outputs from previous CCA Block are passed by replacing the encoder feature maps in F^i. The feature maps in the set F^i are processed through the SFI to generate the source features S_i, defined as:

$$S_i = \text{SFI}(F^{4-i+1}), \quad i \in \{1, 2, 3, 4\} \quad (2)$$

Then, the target features T_i and the source features S_i pass through the CCA Layer (CCAL) to generate the refined features R_i, expressed as:

$$R_i = \text{CCAL}(T_i, S_i), \quad i \in \{1, 2, 3, 4\} \quad (3)$$

Finally, the refined features R_i are combined with the target features T_i to preserve the original information and produce the outputs O_i, formulated as:

$$\text{Q}(A) = \text{BN}(\text{Conv}_{1\times1}(A)) \quad (4)$$

$$C(A, B) = \text{Q}(A) \times \text{Q}(B) + \text{Q}(B) \quad (5)$$

$$O_i = C(T_i, R_i), \quad i \in \{1, 2, 3, 4\} \quad (6)$$

Our CCA Block employs source features S_i from multi-level feature maps, enabling the target features T_i to be align with this diverse information, which leads to enhanced contextual granularity.

3.2. Successive Feature Integration (SFI)

Our method facilitates robust segmentation through successive integration of multi-level feature maps. Typically, this process involves resizing the feature maps to a uniform size and then concatenating them along the channel dimension. While combining information within the feature maps, this approach restricts the ability to account for contextual consistency from the perspective of hierarchical information integration. To address this issue, we propose a new method that integrates feature maps in successive manner.

As illustrated in the SFI of Fig. 3 (a), our approach resizes all feature maps to a uniform size. Smaller feature maps are up-sampled to match the size of the target feature, whereas larger feature maps are down-sampled. This procedure can be formalized as follows for $i, j \in \{1, 2, 3, 4\}$:

$$\hat{F}_j^i = \begin{cases} \text{AvgPool}\left(F_j^i\right), & \text{if } j < 4 - i + 1 \\ \text{Upsample}\left(F_j^i\right), & \text{if } j > 4 - i + 1 \end{cases} \quad (7)$$

The feature maps in $\hat{F}^i$ are successively integrated from low-level features. The two feature maps are first processed

	Method	Encoder	Params (M)	ADE20K		Cityscapes	
				GFLOPs	mIoU	GFLOPs	mIoU
Light-weight config.	SegFormer [24]	MiT-B0	3.8	8.4	37.4	125.5	76.2
	RTFormer [19]	RTFormer-slim	4.8	17.5	36.7	-	76.3
	VWFormer [25]	MiT-B0	3.7	5.8	38.9	-	77.2
	SegFormer-LVT [26]	LVT	3.9	10.6	39.3	140.9	77.6
	FeedFormer [18]	MiT-B0	4.5	7.8	39.2	107.4	77.9
	IFA [12]	ResNet-50	27.8	-	-	186.9	78.0
	FeedFormer-LVT [18]	LVT	4.6	10.0	41.0	124.6	78.6
	SegNext [9]	MSCAN-T	4.3	6.6	41.1	50.5	79.8
	CCASeg-B0 (ours)	**MiT-B0**	**6.2**	**7.2**	**42.6**	**115.8**	**78.7**
	CCASeg-T (ours)	**MSCAN-T**	**7.3**	**8.2**	**44.8**	**64.9**	**81.4**
Middle-weight config.	SegFormer [24]	MiT-B1	13.7	15.9	42.2	243.7	78.5
	VWFormer [25]	MiT-B1	13.7	13.2	43.2	-	79.0
	HRFormer-S [28]	HRT-S	13.5	19.5	44.0	835.7	80.0
	SegNeXt [9]	MSCAN-S	13.9	15.9	44.3	124.6	81.3
	Swin UPerNet-T [16]	Swin-T	60.0	236.0	44.4	-	-
	SenFormer [1]	Swin-T	144.0	179.0	46.0	-	-
	MaskFormer [5]	Swin-T	42.0	55.0	46.7	-	-
	Maks2Former [4]	Swin-T	47.0	74.0	47.7	-	82.1
	CCASeg-B1(Ours)	**MiT-B1**	**24.1**	**25.2**	**46.0**	**317.0**	**80.4**
	CCASeg-S(Ours)	**MSCAN-S**	**23.9**	**24.2**	**47.7**	**192.8**	**82.3**

Table 1. Performance comparison of the proposed scheme on ADE20K-val. and Cityscapes-val. with the state-of-the-art models. Boldfaced numbers, which indicate the best performance, demonstrate the overall superiority of CCASeg in terms of the computational efficiency and the accuracy. The number of FLOPs (G) is calculated on the input size of 512×512 for ADE20K, and 2,048×1,024 for Cityscapes.

with convolution and matched to the channel dimension of high-level feature. Afterward, the two features are multiplied, and the high-level feature map is added via a residual connection, formalized as:

$$\mathrm{I}(A,B) = \mathrm{Conv}_{1\times1}(A)\times\mathrm{Conv}_{1\times1}(B)+\mathrm{Conv}_{1\times1}(B) \quad (8)$$

$$S_{4-i+1} = \mathrm{I}\big(\mathrm{I}\big(\mathrm{I}(\hat{F}_1^i,\hat{F}_2^i),\hat{F}_3^i\big),\hat{F}_4^i\big), \quad i\in\{1,2,3,4\} \quad (9)$$

This method ensures contextual consistency from the perspective of hierarchical information integration, enabling the generated outputs to preserve detailed specifics.

3.3. Convolutional Cross-Attention Layer

The proposed attention mechanism, CCA, aims to efficiently and effectively capture objects of various sizes. As illustrated in Fig. 3 (a), CCA Layer enables the target features T_i to generate refined features R_i with enhanced contextual granularity. This refinement is achieved by matching the target features T_i with source features S_i containing information from different levels. The operation of CCA can be formalized as follows:

$$T_i' = \mathrm{GELU}\left(\mathrm{Conv}_{1\times1}\left(\mathrm{BN}(T_i)\right)\right) \quad (10)$$

$$S_i' = \mathrm{GELU}\left(\mathrm{Conv}_{1\times1}\left(\mathrm{BN}(S_i)\right)\right) \quad (11)$$

$$W_i = \mathrm{Conv}_{1\times1}\Big(\textstyle\sum_{k=0}^{3}\mathrm{Kernel}_k(T_i')\times\mathrm{Kernel}_k(S_i')\Big) \quad (12)$$

$$N_i = \mathrm{Conv}_{1\times1}\left(W_i\times T_i'\right) + T_i \quad (13)$$

$$R_i = \mathrm{FFN}\left(\mathrm{BN}(N_i)\right) + N_i \quad (14)$$

As shown in the Fig. 3 (b), T_i' and S_i', which represent the target features and source features, respectively, are obtained through convolution, as defined in Eqs. (10) and (11). The symbol $\times$ denotes the element-wise matrix multiplication operation. Kernel_k, $k\in\{0,1,2,3\}$, denotes the various branch in the CCA and represents a depth-wise convolution. Kernel_0 serves as the identity connection to aggregate local information, while the remaining focus on capturing the multi-scale context of the feature maps. The kernel sizes are 3×3, 5×5, 9×9, and 13×13 in CCA Block_4, and 5×5, 7×7, 11×11, and 21×21 in other Blocks, respectively. Following [11, 17], we approximate standard depth-wise convolutions with large kernels by applying two depth-wise strip convolutions, such as using 11×1 and 1×11 to achieve a 11×11 kernel. Strip convolutions are not only more lightweight than standard convolutions, but they are also particularly effective at capturing strip-like objects in complex images. Therefore, we utilize strip convolutions in the CCA to effectively and efficiently capture a variety of objects. In the Eq. (12), W_i represents the weight generated through the CCA between the target and source features. Then, this weight is used to reweigh the target features to enhance the multi-scale context as stated in Eq. (13). Finally, the reweighed target features are then passed through a Feed Forward Network (FFN) in Eq. (14).

	Method	Encoder	Params (M)	COCO-Stuff	
				GFLOPs	mIoU
Light-weight	SegFormer [24]	MiT-B0	3.8	8.4	35.6
	VWFormer [25]	MiT-B0	3.7	5.8	36.2
	HRFormer-S [28]	HRT-S	13.5	109.5	37.9
	SegNext [9]	MSCAN-T	4.3	6.6	38.7
	CCASeg-B0 (ours)	**MiT-B0**	**6.2**	**7.2**	**38.8**
	CCASeg-T (ours)	**MSCAN-T**	**7.3**	**8.2**	**40.3**
Middle-weight	SegFormer [24]	MiT-B1	13.7	15.9	40.2
	VWFormer [25]	MiT-B1	13.7	13.2	41.5
	SegNeXt [9]	MSCAN-S	13.9	15.9	42.2
	HRFormer-B [28]	HRT-B	56.2	280.0	42.4
	CCASeg-B1(Ours)	**MiT-B1**	**24.1**	**25.2**	**43.0**
	CCASeg-S(Ours)	**MSCAN-S**	**23.9**	**24.2**	**43.8**

Table 2. Performance comparison of the proposed scheme on COCO-Stuff with the state-of-the-art models. The number of FLOPs (G) is calculated on the input size of 512×512.

Method	Encoder	Params (M)	iSAID	
			GFLOPs	mIoU
Deeplab v3+ [3]	ResNet-50	26.6	112.4	61.9
HRNet [20]	HRNet-W18	9.6	56.6	62.7
SFNet [14]	ResNet-50	31.3	254.3	63.3
UPerNet [16]	Swin-T	50.0	709.1	66.2
FarSeg++ [28]	Swin-T	50.3	349.4	66.3
PFNet [13]	ResNet-50	32.9	244.6	66.9
FarSeg++ [29]	ResNet-50	37.8	158.0	67.1
SegFormer [24]	MiT-B2	24.7	102.1	67.2
AerialFormer-T [10]	Swin-T	42.7	49.0	67.5
CCASeg-B0 (ours)	**MiT-B0**	**6.2**	**28.4**	**67.3**
CCASeg-T (ours)	**MSCAN-T**	**7.3**	**24.8**	**68.7**

Table 3. Performance comparison of the proposed scheme on the remote sensing dataset iSAID with the state-of-the-art models.The number of FLOPs (G) is calculated on the input size of 896×896.

Method	Decoder	Params(M)	ADE20K	
			GFLOPs	mIoU
Swin-T [16]	UPerNet [23]	60.0	236.0	44.4
	CCASeg(Ours)	**49.2**	**54.0**	**45.8**
PoolFormer-S12 [27]	FPN [15]	15.7	31.0	37.2
	CCASeg(Ours)	**22.3**	**24.2**	**42.7**
PoolFormer-M48 [27]	FPN [15]	77.1	82.3	42.7

Table 4. An ablation study on the impact of our proposed decoder and its compatibility with other hierarchical backbones.

4. Experiment

4.1. Experimental Settings

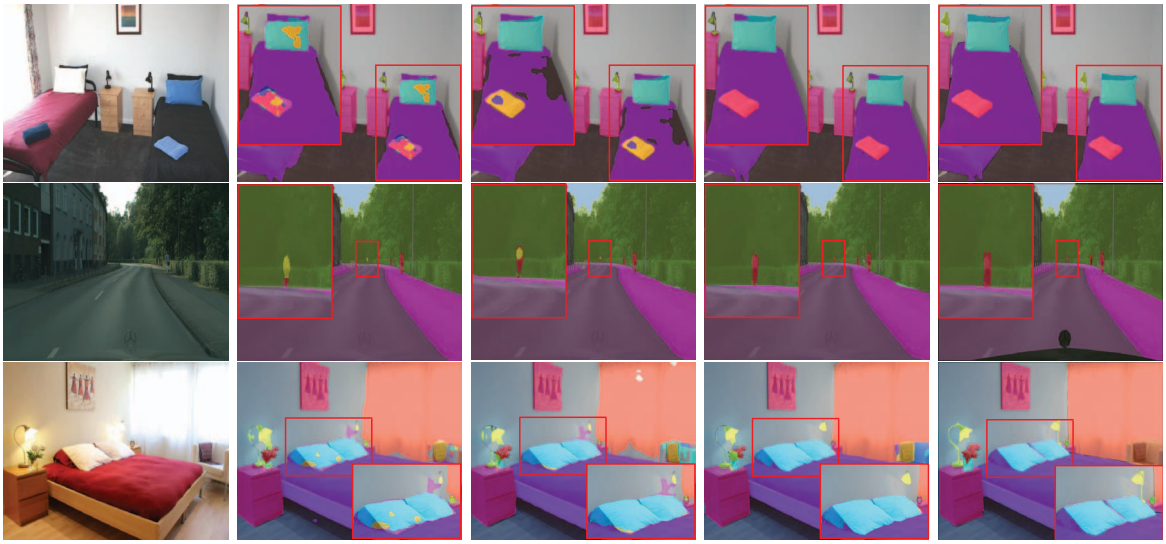
Datasets. We conducted experiments on four popular datasets: ADE20K [30], Cityscapes [7], COCO-Stuff [2], and iSAID [22]. ADE20K contains 150 semantic classes with 20,210/2,000/3,352 images for training, validation, and testing. Cityscapes, focused on urban scenes, has 19 categories across 5,000 high-resolution images split into 2,975/500/1,525 for training, validation, and testing. COCO-Stuff includes 172 categories and 164k images. iSAID, a large-scale aerial segmentation benchmark, has 15 foreground classes and 1,411/458/937 images for training, validation, and testing.

Implementation Details. In our implementation, models with MiT-B0 [24], MiT-B1 [24], MSCAN-T [9] and MSCAN-S [9] encoder backbones are named as CCASeg-B0, CCASeg-B1, CCASeg-T, and CCASeg-S, respectively. We used different numbers of CCA layer according to our variants. Each model employed one more CCA layer than the minimum number of layers used in the stages of the encoder. We used the default setting based on the public codebase mmsegmentation [6]. We used 2 RTX 3090 or 4080 GPUs for all training throughout the experiments. We used encoders pre-trained on ImageNet-1K dataset. During the training, we applied common data augmentation using random horizontal flipping, random scaling (from 0.5 to 2) and random cropping. We trained our models 160K iterations for ADE20K, Cityscapes and iSAID datasets and 80K iterations for COCO-Stuff. We set the batch size to 8 for the Cityscapes dataset and 16 for all the other datasets. We set the learning rate to an initial value of 6e-5, and then, used a polynomial learning rate decay schedule with factor 1.0 by default. We report all our main semantic segmentation results in mean Intersection over Union (mIoU) under the single scale inference setting.

4.2. Comparison with State-of-the-Art Methods

ADE20K, Cityscpaes. In the top part of Tab. 1, we show the performance of lightweight models. As shown in the table, CCASeg-B0 and CCASeg-T were compared with state-of-the-art models on the ADE20K and Cityscapes datasets. Compared to SegFormer-B1 [24] (in the middle-weight table), which uses the more advanced MiT-B1 encoder, CCASeg-B0 with the lighter MiT-B0 encoder achieved 54.7% reduction in parameters, 0.4% higher mIoU with 54.7% less computation on ADE20K, and 0.2% higher mIoU with 52.5% less computation on Cityscapes. Compared to SegNext-S [9] (in the middle-weight table) with MSCAN-S encoder, CCASeg-T using the lighter MSCAN-T encoder achieved 47.5% reduction in parameters, along with 48.4% decrease in computation and 0.5% increase in mIoU on ADE20K, as well as 47.9% reduction in computation and 0.1% improvement in mIoU on Cityscapes.

In the bottom part of the table, we indicate the performance of the middle-weight models. For ADE20K, CCASeg-B1 achieved 83.3% reduction in parameters and 85.9% reduction in computation compared to SenFormer [1], while maintaining the same accuracy. Similarly, CCASeg-S achieved 49.1% reduction in parameters and 67.3% reduction in computation compared to Mask2Former [4], with same accuracy. Using CCASeg as the decoder, as shown in Tab. 1, results in significant accuracy improvements, while also reducing the number of parameters and computation compared to models with similar accuracy.



(a) Image (b) SegFormer (c) FeedFormer (d) CCASeg (e) Ground Truth

Figure 4. Qualitative results on ADE20K and Cityscapes dataset. Compared to SegFormer [24] and FeedFormer [18], our CCASeg result in more precise segmentation in the boxed areas where multi-scale objects are present.

Model	Attention	Params(M)	ADE20K	
			GFLOPs	mIoU
CCASeg-B0	MHCA	5.9	13.3	40.8
	CCA(ours)	**6.2**	**7.2**	**42.6**
CCASeg-T	MHCA	6.3	9.5	43.5
	CCA(ours)	**7.3**	**8.2**	**44.8**

Table 5. Ablation study on the performance comparison of applying CCA and MHCA to our light-weight models.

COCO-Stuff, iSAID. As shown in Tab. 2, our light- and middle-weight models demonstrate outstanding performance compared to state-of-the-art models. On the COCO-Stuff dataset, our models, CCASeg-B1 and CCASeg-S, achieved 57.1% and 57.5% reduction in parameters, 91.0% and 91.4% reduction in computational load, along with 0.6% and 1.4% increase in accuracy, respectively, compared to HRFormer-B [28]. As listed in Tab. 3, on the iSAID dataset, CCASeg-B0, utilizing the lighter MiT-B0 encoder, achieved 74.9% reduction in parameters and 72.2% decrease in computational cost compared to SegFormer-B2 [24], which employs the more advanced MiT-B2 encoder, all while maintaining similar accuracy. Additionally, CCASeg-T achieved 82.9% reduction in parameters and 49.4% decrease in computational load compared to AerialFormer [10], which uses a more powerful Swin-T encoder, while attaining 1.2% higher accuracy. These results, achieving reduced computational load and improved accuracy, demonstrate that our model is effective in segmentation tasks not only for indoor and outdoor images but also for aerial images.

S 4	S 3	S 2	S 1	Params(M)	ADE20K	
					GFLOPs	mIoU
				3.5	3.7	37.7
✓				5.0	4.2	40.1
✓	✓			5.7	5.0	40.8
✓	✓	✓		6.0	6.3	41.5
✓	✓	✓	✓(64×64)	**6.2**	**7.2**	**42.6**
✓	✓	✓	✓(128×128)	6.4	11.2	41.8

Table 6. Ablation study on applying our proposed CCA Block to different stages and comparison based on the size of feature map. S represents the stage.

4.3. Qualitative Results

Fig. 4 represents the effectiveness of the proposed scheme by comparing the segmentation results of CCASeg with those of SegFormer [24] and FeedFormer [18] on ADE20K and Cityscapes. As shown in the red boxes, our CCASeg has the ability to effectively recognize object details near boundaries compared to other models. Due to the this capability, our method not only precisely segments large regions (e.g., roads and beds), but also accurately predicts small-sized objects (e.g., people and pillows). This indicates that our model effectively captures objects of various sizes by leveraging multi-scale contextual information through the decoder. In Fig. 5, we compared our segmentation predictions with AerialFormer [10] on the iSAID dataset. Our method segments detailed areas with small objects more accurately within the red boxes. Therefore, these qualitative results demonstrate that CCASeg is effective across various types of datasets.

(a) Image (b) AerialFormer (c) CCASeg

Figure 5. Qualitative results on iSAID: CCASeg delivers more precise segmentation than AerialFormer [10] in red-boxed areas.

4.4. Ablation Study

Effectiveness of CCASeg Decoder for Various Powerful Hierarchical Backbones. We experimented with a powerful hierarchical backbone as the encoder to evaluate the effectiveness of our CCASeg decoder. In semantic segmentation, Swin [16] adopts UPerNet [23] and PoolFormer [27] adopts FPN [15] as the decoder, respectively. In Tab. 4, our decoder for Swin achieved 18.0% reduction in parameters, 77.1% reduction in computation, and 1.4% improvement in accuracy compared to the baseline. For PoolFormer, it resulted in 21.9% reduction in computational cost and 5.5% accuracy increase, despite a slight increase in parameters. Notably, compared to the heaviest model, PoolFormer-M48 [27], we achieved 71.1% reduction in parameters and 70.6% decrease in computational load, all with the same accuracy. These results demonstrate that the CCASeg decoder is compatible with various hierarchical backbones, while also providing an efficient and effective architecture that enhances the visual representation of multi-level feature maps.

Effectiveness of Convolutional Cross-Attention. We compared our CCA with multi-head cross attention (MHCA) to determine which method is more effective and efficient in capturing multi-scale contexts. MHCA is a widely used attention mechanism that divides feature maps into tokens, which allows each token to model information at different attention heads. In Tab. 5, our CCA not only results in a reduction of computational load by 45.9% and 13.7%, respectively, but also improves accuracy by 1.8% and 1.3% in the CCASeg-B0 and CCASeg-T models, outperforming MHCA in both aspects. This indicates that our CCA method is more efficient and effectively captures both local and global contexts.

Model	Aggregation	Prediction	Params(M)	ADE20K	
				GFLOPs	mIoU
CCASeg-B0	Concat	Concat	6.0	7.0	42.0
	SFI	SFI	**6.2**	**7.2**	**42.6**
CCASeg-T	Concat	Concat	7.1	8.0	44.1
	SFI	SFI	**7.3**	**8.2**	**44.8**

Table 7. Ablation study on methods for the integration of multi-level feature maps.

The Number of CCA Blocks. We verified the effectiveness of applying the CCA Block. In Tab. 6, we conducted experiments on various cases of applying or non-applying CCA Block to each stage. In the stage where the CCA Block is not applied, only the feature map from the encoder were utilized without combining outputs from both the encoder and the decoder. The results show that applying CCA Block to all stage is most effective structure compared to other cases. Additionally, setting the size of T_1 and O_1 to 128×128 and applying the CCA Block to all stages resulted in increased computational cost and reduced accuracy. Information loss from excessive interpolation and the additional processing required to handle it leads to this outcome. Therefore, the successive combination of feature maps within the CCA Block effectively improves performance by leveraging multi-level information.

Effectiveness of Successive Feature Integration. We compared our novel SFI with traditional concatenation in our model to determine which method more effectively combines multi-level feature maps. In Tab. 7, Aggregation denotes the process of integrating feature maps within the CCA Block, and Prediction denotes the process of integrating outputs from the CCA Blocks. Our SFI achieves improved accuracy compared to concatenation, with only a minimal increase in computational cost. This demonstrates that SFI is capable of preserving local details in terms of hierarchical information integration.

5. Conclusion

In this paper, we propose a novel decoder architecture called CCASeg, designed to explicitly account for multi-scale context. Unlike previous methods that rely on attention mechanisms and structures using only a few static feature maps, our approach leverages convolutional cross-attention to capture objects with various sizes across all levels of feature maps and ensures consistent matching through successive combination. Several experiments and ablation studies show the compatibility and superiority of our model as a decoder with various hierarchical backbones. We believe the proposed scheme can be applied to other vision tasks requiring the capture of objects of various sizes.

References

[1] Walid Bousselham, Guillaume Thibault, Lucas Pagano, Archana Machireddy, Joe Gray, Young Hwan Chang, and Xubo Song. Efficient self-ensemble for semantic segmentation. In *33rd British Machine Vision Conference Proceedings, BMVC*, 2022. 1, 5, 6

[2] Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. COCO-Stuff: Thing and stuff classes in context. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1209–1218, 2018. 6

[3] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 801–818, 2018. 6

[4] Bowen Cheng, Ishan Misra, Alexander G Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1290–1299, 2022. 1, 5, 6

[5] Bowen Cheng, Alex Schwing, and Alexander Kirillov. Per-pixel classification is not all you need for semantic segmentation. *Advances in Neural Information Processing Systems*, 34:17864–17875, 2021. 5

[6] MMSegmentation Contributors. OpenMMLab semantic segmentation toolbox and benchmark, https://github.com/open-mmlab/mmsegmentation, 2020. 6

[7] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3213–3223, 2016. 6

[8] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 2

[9] 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. *Advances in Neural Information Processing Systems*, 35:1140–1156, 2022. 1, 2, 3, 5, 6

[10] Taisei Hanyu, Kashu Yamazaki, Minh Tran, Roy A McCann, Haitao Liao, Chase Rainwater, Meredith Adkins, Jackson Cothren, and Ngan Le. AerialFormer: Multi-resolution transformer for aerial image segmentation. *Remote Sensing*, 16(16):2930, 2024. 6, 7, 8

[11] Qibin Hou, Li Zhang, Ming-Ming Cheng, and Jiashi Feng. Strip pooling: Rethinking spatial pooling for scene parsing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4003–4012, 2020. 5

[12] Hanzhe Hu, Yinbo Chen, Jiarui Xu, Shubhankar Borse, Hong Cai, Fatih Porikli, and Xiaolong Wang. Learning implicit feature alignment function for semantic segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 487–505. Springer, 2022. 5

[13] Xiangtai Li, Hao He, Xia Li, Duo Li, Guangliang Cheng, Jianping Shi, Lubin Weng, Yunhai Tong, and Zhouchen Lin. Pointflow: Flowing semantics through points for aerial image segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4217–4226, 2021. 6

[14] Xiangtai Li, Ansheng You, Zhen Zhu, Houlong Zhao, Maoke Yang, Kuiyuan Yang, Shaohua Tan, and Yunhai Tong. Semantic flow for fast and accurate scene parsing. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16*, pages 775–793. Springer, 2020. 6

[15] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2117–2125, 2017. 2, 6, 8

[16] 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 *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10012–10022, 2021. 1, 2, 5, 6, 8

[17] Chao Peng, Xiangyu Zhang, Gang Yu, Guiming Luo, and Jian Sun. Large kernel matters–improve semantic segmentation by global convolutional network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4353–4361, 2017. 5

[18] Jae-hun Shim, Hyunwoo Yu, Kyeongbo Kong, and Suk-Ju Kang. Feedformer: Revisiting transformer decoder for efficient semantic segmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 2263–2271, 2023. 1, 2, 5, 7

[19] Jian Wang, Chenhui Gou, Qiman Wu, Haocheng Feng, Junyu Han, Errui Ding, and Jingdong Wang. RTformer: Efficient design for real-time semantic segmentation with transformer. *Advances in Neural Information Processing Systems*, 35:7423–7436, 2022. 5

[20] Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, et al. Deep high-resolution representation learning for visual recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(10):3349–3364, 2020. 6

[21] Li Wang, Dong Li, Han Liu, Jinzhang Peng, Lu Tian, and Yi Shan. Cross-dataset collaborative learning for semantic segmentation in autonomous driving. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 2487–2494, 2022. 1

[22] Syed Waqas Zamir, Aditya Arora, Akshita Gupta, Salman Khan, Guolei Sun, Fahad Shahbaz Khan, Fan Zhu, Ling Shao, Gui-Song Xia, and Xiang Bai. iSAID: A large-scale dataset for instance segmentation in aerial images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 28–37, 2019. 6

[23] Tete Xiao, Yingcheng Liu, Bolei Zhou, Yuning Jiang, and Jian Sun. Unified perceptual parsing for scene understand-

ing. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 418–434, 2018. 2, 6, 8

[24] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. SegFormer: Simple and efficient design for semantic segmentation with transformers. *Advances in Neural Information Processing Systems*, 34:12077–12090, 2021. 1, 2, 3, 5, 6, 7

[25] Haotian Yan, Ming Wu, and Chuang Zhang. Multi-scale representations by varying window attention for semantic segmentation. In *The Twelfth International Conference on Learning Representations*, 2024. 1, 3, 5, 6

[26] Chenglin Yang, Yilin Wang, Jianming Zhang, He Zhang, Zijun Wei, Zhe Lin, and Alan Yuille. Lite vision transformer with enhanced self-attention. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11998–12008, 2022. 2, 5

[27] Weihao Yu, Mi Luo, Pan Zhou, Chenyang Si, Yichen Zhou, Xinchao Wang, Jiashi Feng, and Shuicheng Yan. Metaformer is actually what you need for vision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10819–10829, 2022. 1, 2, 6, 8

[28] Yuhui Yuan, Rao Fu, Lang Huang, Weihong Lin, Chao Zhang, Xilin Chen, and Jingdong Wang. HRformer: High-resolution vision transformer for dense predict. *Advances in Neural Information Processing Systems*, 34:7281–7293, 2021. 5, 6, 7

[29] Zhuo Zheng, Yanfei Zhong, Junjue Wang, Ailong Ma, and Liangpei Zhang. Farseg++: Foreground-aware relation network for geospatial object segmentation in high spatial resolution remote sensing imagery. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023. 1, 6

[30] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ADE20k dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 633–641, 2017. 6