Toward Real Ultra Image Segmentation: Leveraging Surrounding Context to Cultivate General Segmentation Model

Sai Wang, Yutian Lin, Yu Wu,[*] Bo Du
School of Computer Science, Wuhan University
{wangsai23, wuyucs, yutian.lin, dubo}@whu.edu.cn

Abstract

Existing ultra image segmentation methods suffer from two major challenges, namely the scalability issue (i.e. they lack the stability and generality of standard segmentation models, as they are tailored to specific datasets), and the architectural issue (i.e. they are incompatible with real-world ultra image scenes, as they compromise between image size and computing resources). To tackle these issues, we revisit the classic sliding inference framework, upon which we propose a Surrounding Guided Segmentation framework (SGNet) for ultra image segmentation. The SGNet leverages a larger area around each image patch to refine the general segmentation results of local patches. Specifically, we propose a surrounding context integration module to absorb surrounding context information and extract specific features that are beneficial to local patches. Note that, SGNet can be seamlessly integrated to any general segmentation model. Extensive experiments on five datasets demonstrate that SGNet achieves competitive performance and consistent improvements across a variety of general segmentation models, surpassing the traditional ultra image segmentation methods by a large margin.

1 Introduction

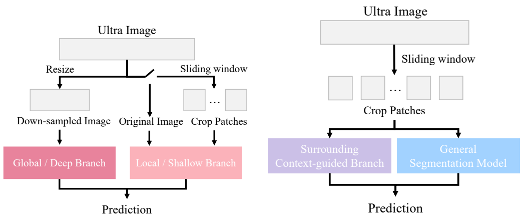
With the rapid development of computing and imaging equipment, ultra-high resolution images with millions or even billions of pixels emerge in endlessly and driving the need for more advanced analytical techniques. Accurate understanding of information conveyed by images [1, 47, 43] has become imperative in diverse fields such as remote sensing [29, 48, 44, 50] and medical analysis [37, 36, 31], and ultra image segmentation has emerged as an essential tool for achieving this goal.

To achieve ultra image segmentation, most of the methods adhere to the concept of integrating global and local information, utilizing the global context clues to aid local region refinement [6, 20]. Typically, these networks have two branches: one receives the down-sampled ultra image and extracts global context information of whole image, while the other extracts local information from sliced patches or the complete image, as shown in Fig. 1(a). The final result is generated by fusing the global and local features. Despite numerous improvements made to ultra image segmentation, we observe that it still suffer from issues in both their scalability and architecture:

(1) **Scalability Issue.** The existing ultra image segmentation methods (UIS) often rely on dataset-specific parameters, which limit their model capacity and scalability. The UIS methods struggle to scale up to larger image sizes, with performance significantly degrading as image resolution increases. Moreover, UIS has complex training procedures that need multi-stage parameter tuning [8, 24]. In contrast, general semantic segmentation (GSS) methods are more effective and demonstrate greater

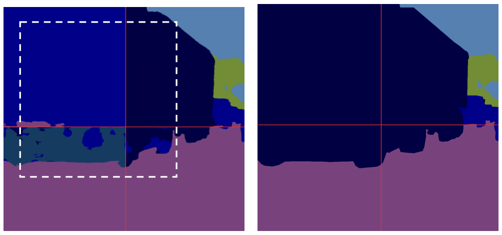
[*] Corresponding author

38th Conference on Neural Information Processing Systems (NeurIPS 2024).



(a) Specific method (b) General model

Figure 1: Comparison with specific (a) and (b). The previous architecture all suffered from **Scalability Issue** and **Architectural Issue** due to specific designs and a lack of tailored context, whereas ours is designed to leverage any general segmentation model to address ultra image segmentation.



(a) Fragmentation (b) Ground Truth

Figure 2: Directly adapting the general segmentation model to ultra image scene will cause **fragmentation phenomenon (a)**: the prediction results of the edge areas between adjacent patches (even in overlapped patches) are inconsistent.

scalability compared to UIS. With a straightforward training process, GSS can be adaptable to various ultra image datasets.

(2) **Architectural Issue.** The architecture of existing UIS is not suitable for processing ultra image scenes in real-world scenarios, as it compromises between dataset size and computing resources. Most methods utilize either the entire image or the downsampled version to capture global information. However, the two strategies are both constrained by the size of input image. When the input image is excessively large, the former strategy cannot process it directly, while the later will suffer great information loss during compression.

Building upon the above discussions, a simple solution is to introduce general segmentation model [27, 12, 18, 39] into the ultra image segmentation task using the sliding window approach, which takes only isolated patches as input due to memory limitations. However, directly adapting GSS to ultra image scenes also raises two challenges: (1) **Information bottleneck.** Using isolated patches as input prevents the model from capturing the correlations between patches, which blocks the information flow and affects the model's perception of the surrounding information. (2) **Fragmentation phenomenon.** With isolated patch input, even if overlapped patches are used, the prediction results of each patch are independent. This leads to inconsistent prediction between patches, especially at the edges of adjacent patches, as shown in Fig. 2.

Based on above considerations, we propose an end-to-end framework called Surrounding Guided Segmentation Framework (SGNet), which takes advantage of the general segmentation method and utilizes surrounding information near the local patch to guide the model. As shown in Fig. 1(b), SGNet includes two decoupled parts: a general segmentation module and a surrounding context-guided branch. Specifically, SGNet takes both the local patch and the corresponding surrounding patch as input. The surrounding patch covers a larger area around the local patch and provides more context information. In the surrounding context-guided branch, we model the context information required to segment the local patch from a larger perspective to drive the flow of information between regions of whole image. Besides, to alleviate the fragmentation phenomenon, we propose a boundary consistency loss to improve the consistency of the prediction results of adjacent patches, thereby alleviating the inconsistent predictions. Note that, unlike existing tightly coupled ultra images segmentation methods, our method can be seamlessly incorporated into any general segmentation models, and brings stable performance improvements.

Our contributions are summarized as follows:

- We excavate two essential but largely overlooked issues in UIS, which hold great value for the community. In addressing these challenges, we are the first to tackle the ultra image segmentation task from the general segmentation model perspective.

- We present a novel end-to-end ultra image segmentation framework named SGNet, which leverage surrounding context information to guide patch-based model segmentation. Our method is flexible to be added to any general segmentation model.

- Experiments show that our method achieves competitive performance and consistently improves over different general segmentation models on five public datasets, outperforming previous methods by a large margin.

2 Related Work

2.1 General Semantic Segmentation

With the development of deep learning [33, 32], the two mainstream methods based on convolution neural network [52, 35, 34, 2] and Transformer [42, 17, 51, 28] have achieved excellent performance. FCN [27] is the first fully convolutional architecture and a lot of work has been extended, such as DeeplabV3 [3] and HRNet [38]. Compared with CNN-based methods, representative Transformer-based works include SegFormer [46], Mask2Foremr [7] and SAM [22]. BiSeNetV1 [49] and STDC [12] are designed for real-time segmentation to reduce the computational overhead. The general segmentation model shows excellent stability and scalability, and we aim to leverage its strengths to address both the Scalability and Architectural issues in existing UIS methods.

2.2 Ultra Image Semantic Segmentation

GLNet [6] introduces a novel global-local architecture. PPN [45] builds on the top of GLNet by integrating a classification network to distinguish valuable patches. Furthermore, Magnet [16] employs a multi-stage pipeline where each stage corresponds to a specific magnification level. FCtL [24] exploits locality-aware contextual correlation to effectively integrate and associate contextual information of local patches. Compared to previous methods, ISDNet [14] proposes a novel framework that combines shallow and deep networks, enabling directly whole-image inference, thereby improving the segmentation effect while increasing the speed. ElegantSeg [5] proposes a end-to-end holistic learning framework from the perspective of engineering optimization. Recently, WSDNet [20] follows the architecture of ISDNet, using DWT-IWT to preserve spatial details. GPWFormer [19] also employs the global-local architecture, and propose the wavelet transformer to model semantic relations. However, the aforementioned methods all suffer from inherent framework flaws, and cannot be applied to ultra images of extremely large scale. On the contrary, our simple yet generalized solution is more adaptable and applicable than specific UIS methods in real-world scenarios.

3 Methodology

3.1 Overview

In this section, we present SGNet, a novel framework that enables existing general segmentation models for ultra image segmentation. As shown in Fig. 3, SGNet consists of two major components, the general segmentation module and surrounding context-guided branch (SCB). The two branches take the local patch and its surrounding larger area as inputs, respectively and extract their features (Section **Architecture**). In the surrounding context-guided branch, we introduce a surrounding context integration module (Section **Surrounding Context Integration Module**) to enable interaction between local and surrounding features, and selectively learn contextual information that is helpful for patch segmentation. Furthermore, we propose a boundary consistency loss to maintain the consistency of prediction results across adjacent patches in Section **Loss Function**.

3.2 Architecture

The general segmentation approaches include global inference and slide inference. The former approaches could result in a significant quality drop during the image resolution compression. Therefore, we attempt to adapt the slide inference based methods into ultra image segmentation tasks. Given an ultra image $I \in \mathbb{R}^{H \times W \times C}$, we divide it into N non-overlapping local patches $I_{local} \in \mathbb{R}^{h \times w \times c}$ and feed them into general segmentation module for prediction as our target. Next, we start from a random position within the local patch as the center and obtain a surrounding patch that is $\alpha (\alpha > 1)$ times larger than its side length. The surrounding patch includes more contextual details, which is helpful to guide the local patch training.

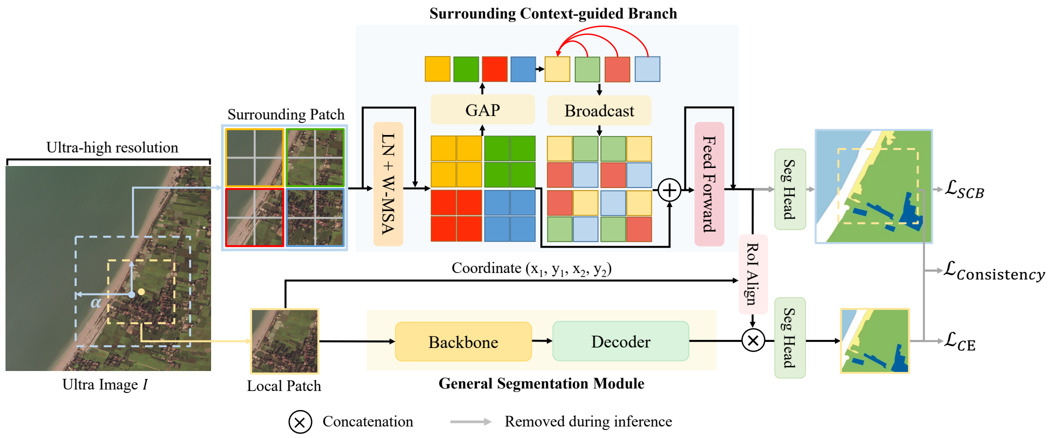


Figure 3: The Architecture of SGNet. An ultra image I is randomly cropped to obtain a local patch I_{local} and a surrounding patch I_{global} containing more context information of any size, which are respectively sent to the general segmentation module and the surrounding context-guided branch to extract features. The resulting features are aggregated through simple concatenation and used to generate high-quality predictions. General segmentation module can be applied to any segmentation model, and surrounding context-guided branch consistently achieves stable improvements on it. LN, W-MSA, and GAP stand for layer normalization, window-based multi-head self-attention, and global average pooling, respectively.

Afterward, the general segmentation module receives the local patch to extract features. Meanwhile, the surrounding context-guided branch (SCB) receives the surrounding patch to extract the surrounding information to guide the segmentation of local patch. In order to boost the processing speed of the SCB, a lightweight backbone is employed for feature extraction. The resulting feature map are subsequently transmitted to the surrounding context integration module for further relationship modeling and focused acquisition of contextual information essential for local patch prediction.

The output feature map is aligned using coordinate relative relationships, retaining only the portion corresponding to the local patch. This portion is then combined with the feature map generated by the general segmentation module to obtain the final prediction through a standard segmentation head. To achieve a more complete optimization of the SCB, we add an extra auxiliary segmentation head to predict the result of the surrounding patch. This helps us to further calculate the boundary consistency loss between the local and surrounding patches based on this prediction.

Compared to previous ultra image segmentation methods [6, 45, 24, 14, 19] that cannot handle extremely large images well, our architecture is more flexible and integrated, as it can handle an **ANY** size large image, equip with **ANY** general segmentation model, thereby well-suited for real-world scenarios.

3.3 Surrounding Context Integration Module

The information contained in the surrounding patch can serve as an extension of the local patch, providing it with more abundant decision-making guidance. Therefore, we introduce the Surrounding Context Integration Module (SCI), from the perspective of absorbing the information contained in each region, and integrating context across all the windows of surrounding patch. The structure of SCI is shown in Fig. 3.

Following [26, 25, 13], the feature map $F \in \mathbb{R}^{H\times W\times C}$ is partitioned into a set of $R \times R$ non-overlapping regions as $F_{region} \in \mathbb{R}^{\frac{H}{R}\times\frac{W}{R}\times R^2\times C}$, with each region being subsequently subdivided into $w \times w$ non-overlapping windows as $F_w \in \mathbb{R}^{\frac{H}{Rw}\times\frac{W}{Rw}\times R^2\times w^2\times C}$.

To facilitate information exchange within the region as well as absorb the contextual information that is helpful for segmentation, we perform layer normalization and window-based multi-head self-attention (W-MSA) on all patch tokens within F_w as follows:

$$F_w^{'} = \textit{W-MSA}(LN(F_w)) + F_w. \tag{1}$$

Currently, the information of all patch tokens within the region have been fully incorporated into $F_w^{'}$. Subsequently, we apply global average pooling (GAP) to acquire the global feature representation $F_w^{global} \in \mathbb{R}^{\frac{H}{Rw} \times \frac{W}{Rw} \times R^2 \times 1 \times C}$ for each window, so as to facilitate the information exchange among the windows in later stage:

$$F_w^{global} = GAP(F_w^{'}). \tag{2}$$

This process enables the absorption and integration of information from every position within the window into a unified vector representation that encapsulates the overall information of the window.

To leverage the complementarity of information and encourage information sharing among windows within the surrounding patch, we utilize self-attention (SA) mechanism to capture the interdependence among the global feature representations F_w^{global} of each window:

$$F_w^{global'} = SA(F_w^{global}). \tag{3}$$

Upon completion of the information exchange, the global feature representation of the current window incorporates the information of other windows. Subsequently, the global feature representation $F_w^{global'}$ is added to $F_w^{'}$ by broadcasting, while the information of the remaining windows is transferred to $F_w^{'}$:

$$F_{global} = F_w^{'} + F_w^{global'}. \tag{4}$$

To enhance the compatibility and generalization for subsequent operation, we apply feed forward layer (FFN) to further refine F_{global}:

$$F_{global}^{'} = FFN(F_{global}) + F_{global}. \tag{5}$$

At this time, $F_{global}^{'}$ integrates the context information of the surrounding patch, which can strengthen the features of the local patch region as a complement and solve the challenge of information bottleneck.

3.4 Loss Function

3.4.1 Boundary Consistency Loss

Despite the incorporation of local patch features and surrounding contextual information, there are still inconsistent prediction results between adjacent patches due to the lack of explicit constraints. Therefore, we propose a boundary consistency loss $\mathcal{L}_{Consistency}$ to improve the consistency of prediction results in neighboring regions and alleviate the fragmentation phenomenon. $\mathcal{L}_{Consistency}$ compels the prediction results of both the general segmentation module and SCB to be as similar as possible, thus promoting consistency in results of neighboring regions across different patches. This helps create a smoother transition between predictions of adjacent patches, harmonizing the prediction of the entire ultra image.

Concretely, we crop the predicted mask $P_{global}^{'}$ corresponding to the local patch from the prediction of the surrounding patch P_{global}. Then, we apply L1 constraints on both $P_{global}^{'}$ and the prediction result of local patch P_{local} to encourage similarity between them, even when different contexts are utilized. The loss function is defined as follows:

$$\mathcal{L}_{Consistency} = ||P_{global}^{'} - P_{local}||. \tag{6}$$

3.4.2 Overall Loss

The cross-entropy loss is used for both the general segmentation module ($\mathcal{L}_{CE}$) and SCB ($\mathcal{L}_{SCB}$). The overall loss $\mathcal{L}$ is a weighted sum of all the losses mentioned above:

$$\mathcal{L} = \lambda_1 \mathcal{L}_{CE} + \lambda_2 \mathcal{L}_{SCB} + \lambda_3 \mathcal{L}_{Consistency}. \tag{7}$$

Table 1: Comparison with baseline on five datasets. The “Mode” column denotes the specific inference modes associated with each method. The “Test size” column relates to the DeepGlobe dataset.

Method	Mode	Backbone	Test size	DeepGlobe 2448×2448	FBP 6800×7200	Aerial Inria 5000×5000	Gleason 5120×5120	Cityscapes 2048×1024
Ultra Image Segmentation Methods								
GLNet [6]	Slide+Whole	R50	508	71.60	42.05	71.20	-	-
PPN [45]	Slide+Whole	R50	512	71.90	-	-	-	75.20
MagNet [16]	Slide+Whole	R50	508	72.96	44.20	-	-	67.57
FCtL [24]	Slide	VGG16	508	72.76	48.28	72.87	-	-
ISDNet [14]	Whole	R18	2448	73.30	21.98	74.23	59.97	76.02
ElegantSeg [5]	Whole	W48	2448	74.32	61.62	-	-	-
WSDNet [20]	Whole	R50	2448	74.10	-	75.20	-	-
GPWFormer [19]	Slide+Whole	R50	500	75.80	-	76.50	-	78.10
General Semantic Segmentation Methods								
FCN [27]	Slide	R50	512	72.38	59.97	80.35	52.65	72.39
+ ***SGNet***	Slide	R50	512	75.28 **(+2.90)**	61.85 (+1.88)	80.81 (+0.46)	58.46 (+5.81)	75.84 **(+3.45)**
DeepLabV3Plus [4]	Slide	R50	512	73.22	61.85	80.88	55.45	75.24
+ ***SGNet***	Slide	R50	512	**75.44** (+2.22)	**63.18** (+1.33)	**81.21** (+0.33)	**61.21** (+5.76)	**76.72** (+1.48)
HRNet [38]	Slide	W18	512	72.87	58.55	79.17	54.31	71.20
+ ***SGNet***	Slide	W18	512	75.25 (+2.38)	61.49 (+2.94)	80.08 (+0.91)	60.50 **(+6.19)**	73.06 (+1.86)
SegFormer [46]	Slide	Mit-b0	512	72.96	57.56	76.26	49.62	67.08
+ ***SGNet***	Slide	Mit-b0	512	74.65 (+1.69)	60.35 (+2.79)	78.80 **(+2.54)**	54.86 (+5.24)	70.42 (+3.34)
STDC [12]	Slide	R50	512	72.59	54.38	75.20	54.51	66.24
+ ***SGNet***	Slide	R50	512	74.51 (+1.92)	59.40 **(+5.02)**	77.25 (+2.05)	60.36 (+5.85)	69.36 (+3.12)

4 Experiments

4.1 Datasets

To comprehensively evaluate our method, we conduct experiments on five public ultra image datasets involving general, medical and remote sensing scenarios: Cityscapes [10], DeepGlobe [11], Inria Aerial [30], Five-Billion-Pixels [40], and Gleason [21]. Since we do not have data partition details from [16], we randomly split the Gleason dataset into a training set of 195 images and a testing set of 49 images, and retrained the relevant models. All other datasets followed the official division.

4.2 Implementation Details

In both training and testing, the local patch has a size of 512×512 in all datasets, while the surrounding patch is twice as large without any resizing. During sliding inference process, we do not preserve any overlapping regions and the center of local patch and surrounding patch are the same. The first four stages of STDC, used as the lightweight backbone following [14], are initialized with ImageNet weights. For all experiments, we set $\lambda_1 = 1, \lambda_2 = 0.4, \lambda_3 = 0.1$. For the DeepGlobe dataset, the "unknown" category is ignored during training as it is not included in the evaluation [11]. We also exclude "unlabeled" category in the FBP dataset following [5].

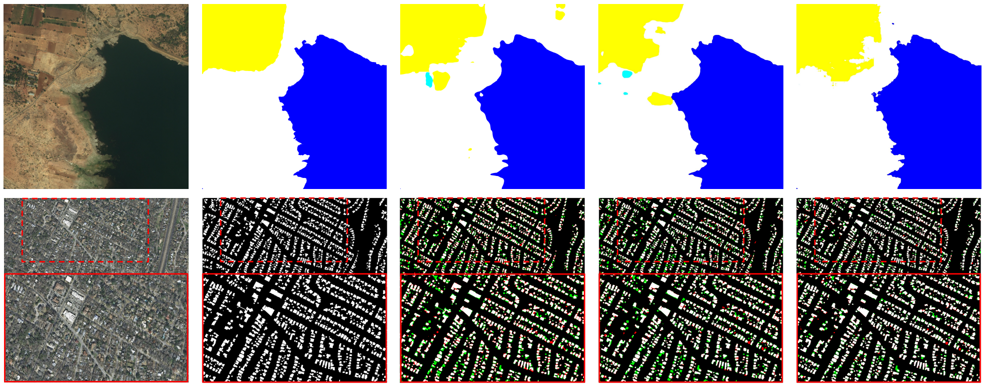
We adopt MMSegmentation [9] as our toolbox and use AdamW optimizer, which initial learning rate is set to 2×10^{-4}. All the models are trained on 4 Tesla V100 GPUs with batch size of 8, except for the Inria Aerial and Gleason dataset, which are trained for 10k iterations while the rest are trained for 30k iterations. Apart from regular operations such as multi-scale training, flipping, and rotating, we do not do any special data augmentation, and the final result does not use any test time augmentation.

4.3 Comparison Results

We classify our comparison methods into two groups: general semantic segmentation and ultra image segmentation. To perform a comprehensive evaluation, we select semantic segmentation methods that rely on CNN (FCN, DeepLabV3Plus, HRNet), Transformer (SegFormer), and lightweight architecture (STDC) for comparison. In addition, we verify the performance of classic ultra image segmentation methods, including GLNet, ISDNet, GPWFoermer, *etc*. For fair comparison, we adopt ResNet-50 [15] or its equivalent parameter amount as the backbone for all the methods we compared.

4.3.1 General *VS* Ultra Image Segmentation Methods

According to the results presented in Table 1, UIS methods only work well on specific datasets, while GSS methods achieve relatively satisfying performance on all datasets. This confirms the aforementioned *scalability issue*. The existing UIS methods show a noticeable performance decrease when transitioning from handling images in DeepGlobe dataset (2448×2448) to larger ultra images



(a) Image (b) Ground Truth (c) FCtL (d) ISDNet (e) Ours

Figure 4: The visualization of different methods in *DeepGlobe* (top row, 2448×2448) and *Inria Aerial* (bottom row, 5000×5000). We use DeepLabV3Plus as our general segmentation model and add SCB on top of it. We apply red color to mark regions where the background is misclassified as foreground and green color to denote regions where foreground is misclassified as background.

Table 2: Efficacy of proposed module. Align, SCI, Aux, Loss respectively correspond to the feature alignment operation, surrounding context integration module, auxiliary head and boundary consistency loss.

Group	Backbone	Align	SCI	Aux	Loss	mIoU
𝔸	-	-	-	-	-	73.22
𝔹	✓	-	-	-	-	73.76
ℂ	✓	✓	-	-	-	73.88
𝔻	✓	✓	✓	-	-	74.76
𝔼	✓	✓	✓	✓	-	75.23
𝔽	✓	✓	✓	✓	✓	75.44

Table 3: Analysis of Surrounding Context Integration.

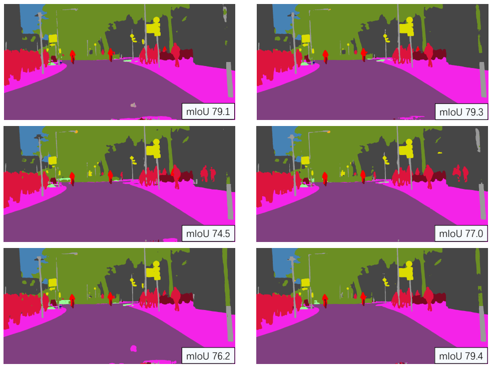
Attention		Global Interaction		
Naive SA	W-MSA	Conv	GAP	mIoU
-	-	-	-	74.40
✓	-	-	-	74.86
-	✓	-	-	74.72
-	✓	✓	-	75.17
-	✓	-	✓	75.44

like FBP (6800×7200), particularly for those utilizing whole images as input, such as ISDNet. This demonstrates that the current UIS architecture lacks flexibility and faces a trade-off between models and datasets. This reveals the aforementioned *architectural issue*.

4.3.2 Improvement over General Segmentation Model

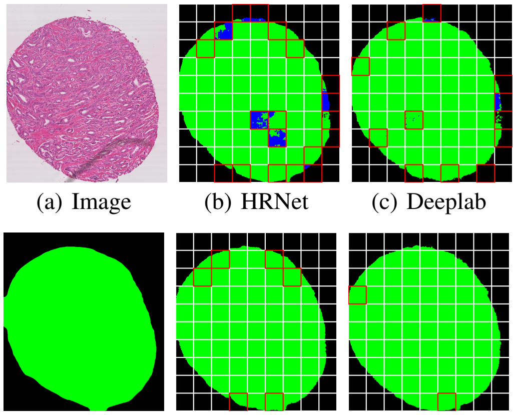
Compared with above methods, our method obtains consistent gains upon all general segmentation methods. Since our method can seamlessly be incorporated into GSS and utilizes the sliding inference strategy for prediction, it can handle ultra images of any scale while leveraging the advantages of scalability from GSS. This perfectly resolves the *scalability issue* and *architectural issue*. Taking the DeepGlobe dataset as an example, our method yields a minimum improvement of 1.69% across all general models, and a maximum of 75.44% mIoU, which is 2.22% higher than using only DeepLabV3Plus. This demonstrates that our method can make full use of the contextual information in the surrounding patch to guide local patch segmentation. It is worth emphasizing that our method is flexible and applicable to any general segmentation models. It also should be noted that all the methods reported in Table 1 are sliding window without overlap. A sliding window with overlap is essentially a test time augmentation method, where multiple predictions on overlapping regions are averaged to enhance the model's performance. By using overlapping regions that cover half of the input patch for predictions, our method can further improve the performance by 0.24% to 75.68% mIoU on DeepGlobe dataset.

Qualitative results of representative works are shown in Fig. 4, while the results of WSDNet and GPWFormer are not available due to no public code. Since we leverage the contextual clues around the current patch as a guide, our method exhibits fewer false positives and delivers more precise segmentation results than other methods.



(a) w/o SCB (b) w/ SCB

Figure 5: Comparison of adding SCB on different models (from top to bottom are DeeplabV3Plus, HRNet and FCN in 5(a) and 5(b)) in *Cityscapes* (2048×1024). The mIoU values are calculated on each shown images.



(a) Image (b) HRNet (c) Deeplab

(d) GT (e) HRNet+SCB (f) Deeplab+SCB

Figure 6: Comparison of adding SCB on different models in *Gleason* (5120×5120). The predicted patches of model are divided by the white line, while the red box indicates the area where the model failed to predict.

Table 4: Analysis of feature fusion scheme.

Early Fusion	Late Fusion	ADD	CONCAT	mIoU
✓	-	✓	-	74.68
✓	-	-	✓	74.87
-	✓	✓	-	74.93
-	✓	-	✓	75.44

Table 5: Analysis of surrounding patch size.

Scale (α)	Surrounding Patch Size	mIoU
1	512	74.75
2	**1024**	**75.44**
3	1536	74.96
4	2048	74.78

4.4 Ablation Study

4.4.1 Effectiveness of Proposed Module

To thoroughly confirm the efficacy of each module, we substitute SGNet (DeepLabV3Plus version) with six variations denoted as Group $\mathbb{ABCDEF}$. As shown in Table 2, we employ only a lightweight backbone to individually extract features from surrounding patch, and then concatenated them on last features in group $\mathbb{B}$. Group $\mathbb{B}$ achieves a mIoU of 73.76%, which exceeds group $\mathbb{A}$ that only used general segmentation module by 0.54%. In group $\mathbb{C}$, we utilize the crop operation to align features, resulting in an improvement of 0.12% compared to $\mathbb{B}$. By applying the surrounding context integration module to model the context information, and the auxiliary head to enhance the convergence of SCB in groups $\mathbb{D}$ and $\mathbb{E}$, we are able to improve performance by 0.88% and 1.35% compared to group $\mathbb{C}$. Finally, group $\mathbb{F}$ incorporates the boundary consistency loss from group $\mathbb{E}$, and we can conclude that this loss can effectively enhance the consistency between the bordering regions of adjacent patches, up to 75.44% mIoU. Additionally, when combining the logits maps from the surrounding branch and the local branch, the performance is further improved to 75.59% mIoU. The improvement in results essentially belongs to a model ensembling method. This indicates that our surrounding branch has learned effective and complementary information to the local branch, further demonstrating the validity of our approach.

Other than analyzing the effectiveness of individual components within the module, we also demonstrate its generality across different methods, as shown in Fig. 5. As observed, on the general dataset such as Cityscapes, our method still lead to significant and consistent improvements upon various models. In addition, we select objects whose area is less than 900 pixels (0.0036% of the whole image) from Inria Aerial dataset to verify the effect on small objects. Adding SCB improved the mIoU from 57.12% to 57.73%, which shows SCB also improve the tiny objects.

4.4.2 Efficacy of Surrounding Context Integration Module

To explore how the attention mechanism and global interaction affect the performance, we conduct ablation studies on surrounding context integration module in Table 3. We first analyze the difference between the naive self-attention (Naive SA) and the window self-attention (W-MSA) in modeling the attention information of the surrounding patch. We apply six residual layers to replace attention module as the baseline, achieving 74.40% mIoU. Building on this, we further utilize Naive SA and W-MSA, which improve the mIoU by 0.46% and 0.32%, respectively. This indicates that the attention mechanism can capture the correlation between different positions, promoting the flow of surrounding information within the feature map. As naive self-attention operates on every pixel of the feature map, it inherently incorporates surrounding interaction operation. We further analyze the impact of incorporating surrounding information into the W-MSA mechanism, and we utilize convolution and GAP operations to extract surrounding information from the windows. After adopting either convolution or GAP, we can get consistent improvement, indicating that introducing surrounding context is essential for W-MSA, which enables the current window to acquire information of other regions. In comparison to the convolution, using GAP to extract the surrounding representation of the window is more suitable for W-MSA, which can reach up to 75.44% mIoU.

4.4.3 Efficacy of Feature Fusion

We perform experiments to analyze the scheme of feature fusion of SCB and general segmentation module, as shown in Table 4. The fusion position including before the decoder head (early fusion) and after the decoder head (late fusion). The fusion method including ADD and CONCAT. We observe that the performance of late fusion is better than early fusion. We believe that late fusion is more flexible, shielding the complex processing of different models in the input part of the decoder head. Both ADD and CONCAT operations yield satisfactory outcomes, where the latter have a leading of 0.51%. We believe that the segmentation head captures a greater amount of information, allowing it to dynamically select the optimal feature for prediction. Hence, modifying the segmentation head of GSS can further enhance the performance of model, but it deviates from our starting point of simplicity and is outside the scope of our method.

4.4.4 Comparison of Surrounding Patch Size

We perform ablation studies on the surrounding patch size in Table 5. We observe that, as the surrounding size increases, the performance initially improves and then gradually decreases. This is attributed to the fact that only those close nearby area could provide valuable contextual guidance. Too far-away pixels in the entire images are not relevant to the local patch and may introduce additional noise. This confirms the motivation that contextual information around the local patch is beneficial.

4.4.5 Efficiency Study

We conduct experiments to examine the speed of different methods. Frames-per-second (FPS) and Memory are measured on a Tesla V100 GPU with a batch size of 1. The variation in FPS among different methods is primarily attributed to the inference framework, that is, whole inference and slide inference. Without considering the overlap, the latter theoretically requires $\lceil \frac{W}{w} \rceil \times \lceil \frac{H}{h} \rceil$ more operations than the former. Methods such as ISDNet primarily focus on model efficiency and achieve higher FPS, based on a shallow-deep architecture that directly employs the whole image for inference. Consequently, we also develop a fast version named SGNet (ISDNet-Style) by incorporating a modified SCB for a fair comparison. SGNet (ISDNet-Style) applies a single surrounding context integration module to the output feature map of the last stage of STDC to model global information. To maintain consistency with ISDNet and exclude speed differences caused by inference frameworks, we set our local patch size as same as ISDNet to simulate the whole inference. As shown in Table 6, the fast version not only surpasses the mIoU of ISDNet but also achieves a higher FPS of up to 25.59.

Table 6: Comparison of speed on DeepGlobe. We measure GPU memory using the command line tool "gpustat". "$*$" represents results we reproduced in our setting. "-" indicates that there is no publicly available result or code.

Method	mIoU	FPS	Memory(MB)
GLNet [6]	71.60	0.17	1865
CascadePSP [8]	68.50	0.11	3236
PPN [45]	71.90	12.90	1193
PointRend [23]	71.78	6.25	1593
MagNet [16]	72.96	0.80	1559
MagNet-Fast [16]	71.85	3.40	1559
FCtL [24]	72.76	0.13	4332
ISDNet [14]	73.30	22.67*	1948
GPWFormer [19]	75.80	-	2380
DeepLabV3Plus [4]	73.22	1.14	1279
DeepLabV3Plus + *SGNet*	75.44	0.66	2187
SGNet (ISDNet-Style)	74.28	25.59	2043

Table 7: Comparison of normal image and JPEG compressed image.

	Normal	JPEG compression
DeepLabV3Plus	73.22	60.24
+ *SGNet*	75.44 (+2.22)	64.16 (+3.92)

4.4.6 Robustness Study

We compressed the DeepGlobe dataset to 10% of its original image quality using JPEG compression and retrained SGNet and DeepLabV3Plus on it. As shown in Table 7, the results show that SGNet significantly outperforms DeepLabV3Plus by 3.92% (from 64.16% mIoU versus 60.24% mIoU), and this improvement is almost twice that of normal images (from 73.22% mIoU to 75.44% mIoU). This indicates that our method is relatively insensitive to noise compared to baseline models and can use surrounding information to infer damaged pixel information within the object. It also demonstrates that our method is particularly robust in scenarios involving image degradation.

4.4.7 Fragmentation Phenomenon Study

To show our advantage to alleviate the fragmentation phenomenon, we compare our module against GSS in Fig. 6. We simulate all predicted patches (non-overlapping) generated during slide inference process with white lines. It is evident that GSS result in a steep change in the boundary between adjacent patches, and the predicted results of each patch are relatively independent, lacking coherence. Our method is capable of modeling the correlation between patches, leading to smoother prediction results for adjacent patches.

5 Conclusion

In this paper, we propose the Surrounding Guided Segmentation Framework (SGNet) to address the scalability and architectural issues in existing UIS methods. SGNet leverages surrounding context to guide local patch segmentation and can be incorporated into any general segmentation model. Our method consistently improves performance across five datasets and demonstrates greater adaptability and applicability than existing methods in real-world scenarios. This work not only contributes a novel solution to the UIS domain but also emphasizes the potential of integrating general segmentation techniques to advance the field. We hope that SGNet inspires further exploration in ultra image segmentation, fostering innovations that enhance model performance and scalability.

Limitations. Noise and artifacts in ultra-high-resolution images can hinder segmentation accuracy, necessitating further research to address these challenges.

Social Impact. The proposed method has the potential to advance various fields, including medical image analysis and remote sensing image processing.

Acknowledgment. This work was supported in part by the National Key Research and Development Program of China under grant 2023YFC2705700, in part by the National Natural Science Foundation of China under grant 62372341, and in part by the Fundamental Research Funds for the Central Universities under grant 2042024kf0040.

References

[1] Xiaojun Chang, Pengzhen Ren, Pengfei Xu, Zhihui Li, Xiaojiang Chen, and Alex Hauptmann. A comprehensive survey of scene graphs: Generation and application. *IEEE TPAMI*, 45(1):1–26, 2021.

[2] Chun-Fu Chen, Rameswar Panda, and Quanfu Fan. Regionvit: Regional-to-local attention for vision transformers. *arXiv preprint arXiv:2106.02689*, 2021.

[3] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. arxiv 2017. *arXiv preprint arXiv:1706.05587*, 2, 2019.

[4] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *ECCV*, pages 801–818, 2018.

[5] Wei Chen, Yansheng Li, Bo Dang, and Yongjun Zhang. Elegantseg: End-to-end holistic learning for extra-large image semantic segmentation. *arXiv preprint arXiv:2211.11316*, 2022.

[6] Wuyang Chen, Ziyu Jiang, Zhangyang Wang, Kexin Cui, and Xiaoning Qian. Collaborative global-local networks for memory-efficient segmentation of ultra-high resolution images. In *CVPR*, pages 8924–8933, 2019.

[7] Bowen Cheng, Ishan Misra, Alexander G Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *CVPR*, pages 1290–1299, 2022.

[8] Ho Kei Cheng, Jihoon Chung, Yu-Wing Tai, and Chi-Keung Tang. Cascadepsp: Toward class-agnostic and very high-resolution segmentation via global and local refinement. In *CVPR*, pages 8890–8899, 2020.

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

[10] 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 *CVPR*, pages 3213–3223, 2016.

[11] Ilke Demir, Krzysztof Koperski, David Lindenbaum, Guan Pang, Jing Huang, Saikat Basu, Forest Hughes, Devis Tuia, and Ramesh Raskar. Deepglobe 2018: A challenge to parse the earth through satellite images. In *CVPRW*, pages 172–181, 2018.

[12] Mingyuan Fan, Shenqi Lai, Junshi Huang, Xiaoming Wei, Zhenhua Chai, Junfeng Luo, and Xiaolin Wei. Rethinking bisenet for real-time semantic segmentation. In *CVPR*, pages 9716–9725, 2021.

[13] Jiemin Fang, Lingxi Xie, Xinggang Wang, Xiaopeng Zhang, Wenyu Liu, and Qi Tian. Msg-transformer: Exchanging local spatial information by manipulating messenger tokens. In *CVPR*, pages 12063–12072, 2022.

[14] Shaohua Guo, Liang Liu, Zhenye Gan, Yabiao Wang, Wuhao Zhang, Chengjie Wang, Guannan Jiang, Wei Zhang, Ran Yi, Lizhuang Ma, et al. Isdnet: Integrating shallow and deep networks for efficient ultra-high resolution segmentation. In *CVPR*, pages 4361–4370, 2022.

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016.

[16] Chuong Huynh, Anh Tuan Tran, Khoa Luu, and Minh Hoai. Progressive semantic segmentation. In *CVPR*, pages 16755–16764, 2021.

[17] Jitesh Jain, Jiachen Li, Mang Tik Chiu, Ali Hassani, Nikita Orlov, and Humphrey Shi. One-former: One transformer to rule universal image segmentation. In *CVPR*, pages 2989–2998, 2023.

[18] Jitesh Jain, Anukriti Singh, Nikita Orlov, Zilong Huang, Jiachen Li, Steven Walton, and Humphrey Shi. Semask: Semantically masked transformers for semantic segmentation. In *ICCV*, pages 752–761, 2023.

[19] Deyi Ji, Feng Zhao, and Hongtao Lu. Guided patch-grouping wavelet transformer with spatial congruence for ultra-high resolution segmentation. In *IJCAI*, pages 920–928, 2023.

[20] Deyi Ji, Feng Zhao, Hongtao Lu, Mingyuan Tao, and Jieping Ye. Ultra-high resolution segmentation with ultra-rich context: A novel benchmark. In *CVPR*, pages 23621–23630, 2023.

[21] Davood Karimi, Guy Nir, Ladan Fazli, Peter C Black, Larry Goldenberg, and Septimiu E Salcudean. Deep learning-based gleason grading of prostate cancer from histopathology images—role of multiscale decision aggregation and data augmentation. *IEEE JBHI*, 24(5):1413–1426, 2019.

[22] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. *arXiv preprint arXiv:2304.02643*, 2023.

[23] Alexander Kirillov, Yuxin Wu, Kaiming He, and Ross Girshick. Pointrend: Image segmentation as rendering. In *CVPR*, pages 9799–9808, 2020.

[24] Qi Li, Weixiang Yang, Wenxi Liu, Yuanlong Yu, and Shengfeng He. From contexts to locality: Ultra-high resolution image segmentation via locality-aware contextual correlation. In *ICCV*, pages 7252–7261, 2021.

[25] Ze Liu, Han Hu, Yutong Lin, Zhuliang Yao, Zhenda Xie, Yixuan Wei, Jia Ning, Yue Cao, Zheng Zhang, Li Dong, et al. Swin transformer v2: Scaling up capacity and resolution. In *CVPR*, pages 12009–12019, 2022.

[26] 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 *ICCV*, pages 10012–10022, 2021.

[27] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, pages 3431–3440, 2015.

[28] Huaishao Luo, Junwei Bao, Youzheng Wu, Xiaodong He, and Tianrui Li. Segclip: Patch aggregation with learnable centers for open-vocabulary semantic segmentation. In *ICML*, pages 23033–23044. PMLR, 2023.

[29] Lei Ma, Yu Liu, Xueliang Zhang, Yuanxin Ye, Gaofei Yin, and Brian Alan Johnson. Deep learning in remote sensing applications: A meta-analysis and review. *ISPRS*, 152:166–177, 2019.

[30] Emmanuel Maggiori, Yuliya Tarabalka, Guillaume Charpiat, and Pierre Alliez. Can semantic labeling methods generalize to any city? the inria aerial image labeling benchmark. In *IGARSS*, pages 3226–3229. IEEE, 2017.

[31] Maciej A Mazurowski, Haoyu Dong, Hanxue Gu, Jichen Yang, Nicholas Konz, and Yixin Zhang. Segment anything model for medical image analysis: an experimental study. *Medical Image Analysis*, 89:102918, 2023.

[32] Gaurav Menghani. Efficient deep learning: A survey on making deep learning models smaller, faster, and better. *ACM Computing Surveys*, 55(12):1–37, 2023.

[33] Shervin Minaee, Yuri Boykov, Fatih Porikli, Antonio Plaza, Nasser Kehtarnavaz, and Demetri Terzopoulos. Image segmentation using deep learning: A survey. *IEEE TPAMI*, 44(7):3523–3542, 2021.

[34] Tuan Duc Ngo, Binh-Son Hua, and Khoi Nguyen. Isbnet: a 3d point cloud instance segmentation network with instance-aware sampling and box-aware dynamic convolution. In *CVPR*, pages 13550–13559, 2023.

[35] Yaolei Qi, Yuting He, Xiaoming Qi, Yuan Zhang, and Guanyu Yang. Dynamic snake convolution based on topological geometric constraints for tubular structure segmentation. In *ICCV*, pages 6070–6079, 2023.

[36] Muhammad Imran Razzak, Saeeda Naz, and Ahmad Zaib. Deep learning for medical image processing: Overview, challenges and the future. *Classification in BioApps: Automation of Decision Making*, pages 323–350, 2018.

[37] Dinggang Shen, Guorong Wu, and Heung-Il Suk. Deep learning in medical image analysis. *Annu Rev Biomed Eng*, 19:221–248, 2017.

[38] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-resolution representation learning for human pose estimation. In *CVPR*, pages 5693–5703, 2019.

[39] Hans Thisanke, Chamli Deshan, Kavindu Chamith, Sachith Seneviratne, Rajith Vidanaarachchi, and Damayanthi Herath. Semantic segmentation using vision transformers: A survey. *EAAI*, 126:106669, 2023.

[40] Xin-Yi Tong, Gui-Song Xia, and Xiao Xiang Zhu. Enabling country-scale land cover mapping with meter-resolution satellite imagery. *ISPRS Journal of Photogrammetry and Remote Sensing*, 196:178–196, 2023.

[41] Philipp Tschandl, Cliff Rosendahl, and Harald Kittler. The ham10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions. *SCI DATA*, 5(1):1–9, 2018.

[42] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *NeurlPS*, 30, 2017.

[43] Sai Wang, Yutian Lin, and Yu Wu. Omni-q: Omni-directional scene understanding for unsupervised visual grounding. In *CVPR*, pages 14261–14270, 2024.

[44] Yi Wang, Syed Muhammad Arsalan Bashir, Mahrukh Khan, Qudrat Ullah, Rui Wang, Yilin Song, Zhe Guo, and Yilong Niu. Remote sensing image super-resolution and object detection: Benchmark and state of the art. *Expert Syst. Appl*, page 116793, 2022.

[45] Tong Wu, Zhenzhen Lei, Bingqian Lin, Cuihua Li, Yanyun Qu, and Yuan Xie. Patch proposal network for fast semantic segmentation of high-resolution images. In *AAAI*, volume 34, pages 12402–12409, 2020.

[46] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. *NeurlPS*, 34:12077–12090, 2021.

[47] Shuo Xu, Sai Wang, Xinyue Hu, Yutian Lin, Bo Du, and Yu Wu. Mac: A benchmark for multiple attributes compositional zero-shot learning. *arXiv preprint arXiv:2406.12757*, 2024.

[48] Huang Yao, Rongjun Qin, and Xiaoyu Chen. Unmanned aerial vehicle for remote sensing applications—a review. *Remote Sens*, 11(12):1443, 2019.

[49] Changqian Yu, Jingbo Wang, Chao Peng, Changxin Gao, Gang Yu, and Nong Sang. Bisenet: Bilateral segmentation network for real-time semantic segmentation. In *ECCV*, pages 325–341, 2018.

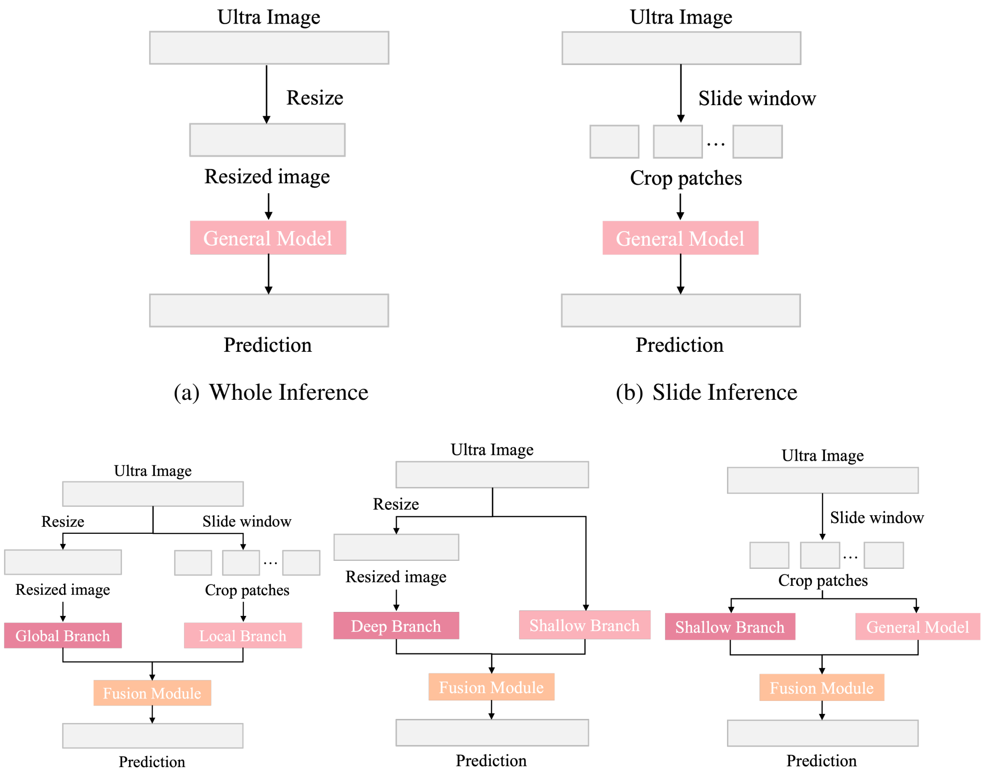
[50] Haitao Yuan, Sai Wang, Zhifeng Bao, and Shangguang Wang. Automatic road extraction with multi-source data revisited: completeness, smoothness and discrimination. *Proceedings of the VLDB Endowment*, 16(11):3004–3017, 2023.

[51] Hao Zhang, Feng Li, Xueyan Zou, Shilong Liu, Chunyuan Li, Jianwei Yang, and Lei Zhang. A simple framework for open-vocabulary segmentation and detection. In *ICCV*, pages 1020–1031, 2023.

[52] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *CVPR*, pages 2881–2890, 2017.

A Existing Architecture Analysis

In Figure 7, we show the detail architectures of existing ultra image segmentation methods. Whole inference 7(a) and slide inference 7(b) are the two most fundamental ultra image segmentation architectures. While whole inference can be applied directly to general segmentation models [27, 4, 38, 46, 12, 49], it utilizes the down-sampled image as input, which results in a loss of essential information. The latter divides the image into multiple patches by means of sliding window, and predicts the patches one by one, aggregating the results together. However, due to the isolated prediction of patches and the lack of global information, it is prone to cause **information bottleneck** and **fragmentation phenomenon**.



(a) Whole Inference

(b) Slide Inference

(c) Global & Local

(d) Shallow & Deep

(e) **Ours**

Figure 7: Comparison of existing architectures of ultra image segmentation.

To address the issues above and leverage the benefits of both whole inference and slice inference, some studies [6, 45, 24, 16, 19] have introduced the Global & Local architecture 7(c). This architecture involves one branch that takes in the down-sampled ultra image as the global cue, while the other branch extracts local information from sliced patches. However, these tasks usually require sequential processing of each patch and complex integration strategies to combine global information, resulting in poor scalability of the architecture. Alternatively, some study[14, 20] have introduced the Shallow & Deep architecture 7(d), which inputs the original image and the resized image into different branches for processing. Despite its fast speed, it is essentially a trade-off in dataset size and computing resources, and cannot be used for real ultra image segmentation scenarios.

The architectures depicted in Figure 7(c) and Figure 7(d) both use the original image after resize in an input branch, which may limit the capability of the architecture to process ultra images. On the one hand, when the image is extremely large, the resized image will inevitably lose a lot of detail information to fit the GPU memory. On the other hand, the resized ultra image cannot guarantee to provide specific global information that is helpful for local patch segmentation. To address these issues, we designed a novel and effective ultra image segmentation architecture 7(e) based on slide window. Our architecture can handle ultra images of any scale and provide surrounding information that aids in local patch segmentation. Furthermore, our architecture is highly flexible, not restricted by computing resources, and can be seamlessly integrated to any encoder-decoder based segmentation model.

B Dataset Details

To comprehensively evaluate our method, we conduct experiments on five public ultra image datasets involving general, medical and remote sensing scenarios:

Cityscapes [10]. The Cityscapes dataset is a popular street scene dataset for generic semantic segmentation. It contains 3475 images with the resolution of 1024×2048 and a total of 19 categories. We use 2975 images for training and 500 images for testing.

Gleason [21]. The Gleason dataset is a high resolution medical image dataset with the resolution of 5120×5120. It contains 244 H&E-stained histopathology images for automatic Gleason grading of prostate cancer. Since we do not have data partition details from [16], we randomly split dataset into training and testing set with 195 and 49 images.

DeepGlobe [11]. DeepGlobe is a satellite image dataset that contains 803 ultra-high resolution images (2448×2448 pixels). It contains 7 categories, of which the class named "unknown" is excluded. Following [6], we divide the training, validation, and testing sets into 454, 207, and 142 images, respectively.

Inria Aerial [30]. The Inria Aerial dataset comprises 180 images, each with a resolution of 5000×5000 pixels. Following [6], we divide the training, validation, and testing sets into 126, 27, and 27 images, respectively.

Five-Billion-Pixels [40]. The FBP dataset includes 150 high-resolution images, each with a size of 7200×6800 pixels and labeled with 24 categories. We use the same test set as in [40], and randomly divide the remaining images into training set and validation set, including 90 and 30 images.

C Comparative Analysis of SCB Branch

In order to demonstrate the proposed SCB Branch's efficacy, we used sixteen conventional convolution layers followed by four transformer blocks to form a trivial replacement branch for extracting surrounding image feature. It serving as a functionally analogous replacement for the proposed SCB branch. As shown in Table 8, we added this trivial replacement branch to all general segmentation models in Table 1 and conducted experiments on the DeepGlobe dataset using the same settings. The results show that our proposed SCB branch significantly outperforms this branch, proving the efficacy of our proposed component.

Table 8: Comparison of SCB Branch with it trivial replacement.

	Original	**+ Trivial Branch**	**+ SGNet**
FCN	72.38	72.56 (+0.28)	75.28 (+2.90)
DeepLabV3Plus	73.22	73.77 (+0.55)	75.44 (+2.22)
HRNet	72.87	73.24 (+0.37)	75.25 (+2.38)
SegFormer	72.96	73.56 (+0.60)	74.65 (+1.69)
STDC	72.59	72.88 (+0.29)	74.51 (+1.92)

D Large Scale Human Subject Segmentation Study

We conducted experiments on the well-known CelebAMask-HQ dataset to further verify the effectiveness of our method in large scale human subject segmentation. Due to the lack of extremely high-resolution human datasets, we simulate ultra-high resolution by resizing the images from the CelebAMask-HQ dataset from 1024 to 2448 pixels. We compared our SGNet with DeepLabV3Plus, which is a highly popular and widely used image segmentation model across various domains. The performance of our method significantly outperformed DeepLabV3Plus by 1.61% (from 62.93% mIoU to 64.54% mIoU). We provide more examples of qualitative results in Figure 8.

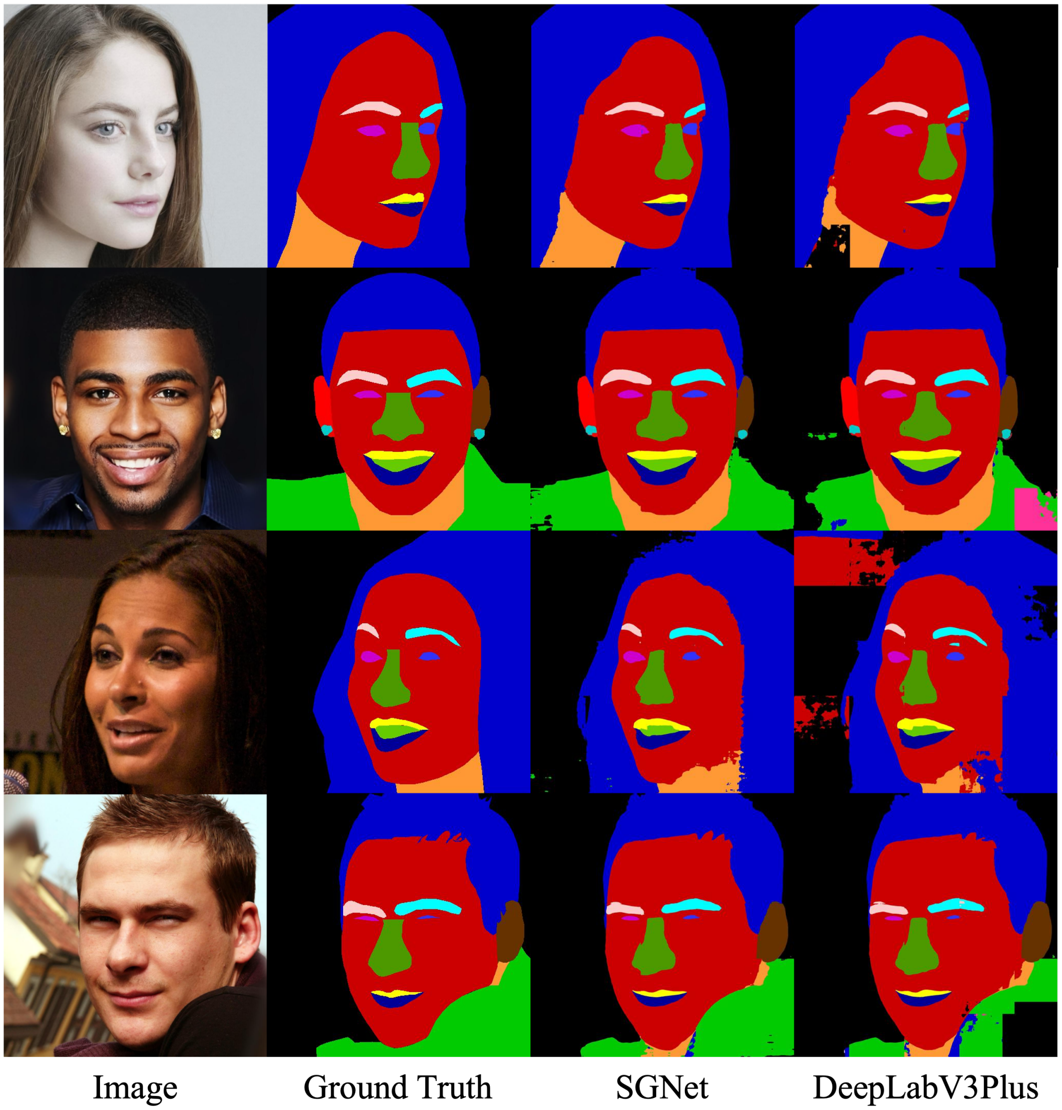
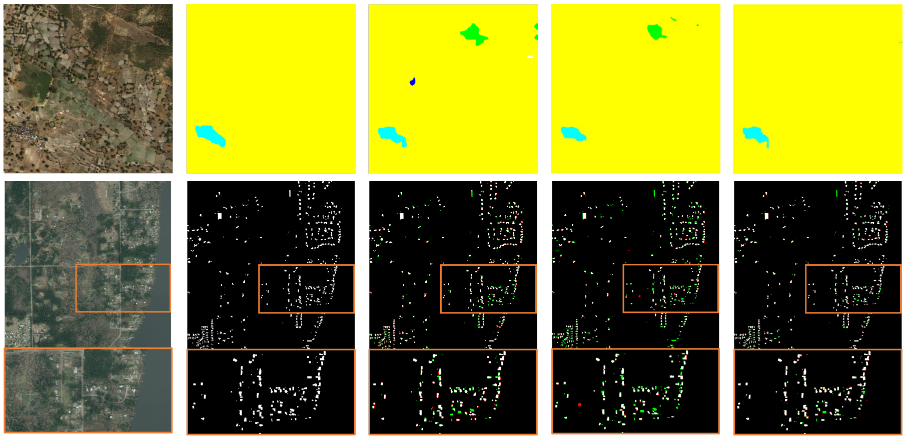


Figure 8: The visualization of different methods in CelebAMask-HQ. We resized the images from 1024 to 2448 pixels to simulate ultra-high resolution and used sliding window of size 512 without overlap for inference.

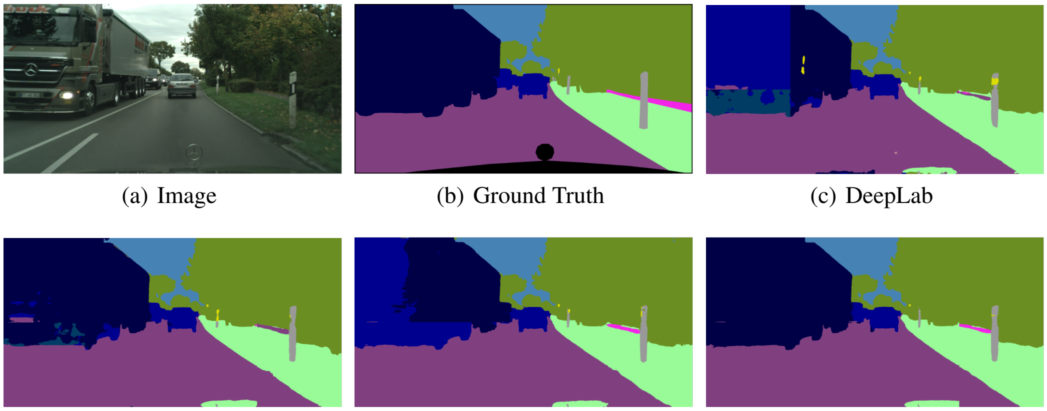
E More Qualitative Result

In Figure 9, we provide more examples of qualitative results between our method and existing ultra image segmentation methods like FCtL [24] and ISDNet [14]. These results indicate that our method is capable of achieving satisfactory quality on various challenging datasets [11, 30]. Figure 10 and Figure 11 illustrate the effectiveness of SCB by comparing it to general segmentation models such as DeepLabV3Plus and HRNet on Cityscapes [10] and Gleason [41].



(a) Image (b) Ground Truth (c) FCtL (d) ISDNet (e) Ours

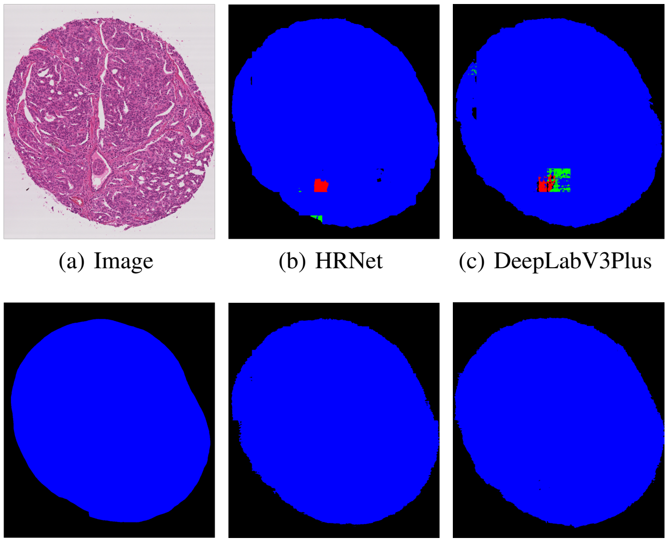
Figure 9: The qualitative results of different methods in *DeepGlobe* (top row, 2448×2448) and *Inria Aerial* (bottom row, 5000×5000).



(a) Image (b) Ground Truth (c) DeepLab

(d) DeepLab + SCB (e) HRNet (f) HRNet + SCB

Figure 10: The comparison of adding SCB on different models in *Cityscapes* (2048×1024).



(a) Image (b) HRNet (c) DeepLabV3Plus

(d) Ground Truth (e) HRNet + SCB (f) DeepLab + SCB

Figure 11: The comparison of adding SCB on different models in *Gleason* (5120×5120).

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The main claims accurately summarize the our contributions.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: In Conclusion, we discuss the limitations of the proposed method.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

Justification:

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We provide the necessary information required to replicate the main experimental results, including the model architecture and experimental settings.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: We have included our core code in an anonymized zip as part of the supplementary, and will release the complete code upon acceptance.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so “No” is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: The training and test details are provided in Sec. 4.2.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [No]

Justification: Not applicable.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).

- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: In Section 4.2, we provided detailed information about the specific computer resources.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We have conducted.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We discuss the societal impacts in conclusion.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification:

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We cite all the original papers that provided the code and dataset.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, paperswithcode.com/datasets has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification:

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification:

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification:

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.