Practical Techniques for Vision-Language Segmentation Model in Remote Sensing

Yuting Lin[1], Kumiko Suzuki[2], Shinichiro Sogo[3]

[1] Kokusai Kogyo Co., Ltd. – Tokyo, Japan - utei_rin@kk-grp.jp
[2] Kokusai Kogyo Co., Ltd. – Tokyo, Japan - kumiko_suzuki@kk-grp.jp
[3] Kokusai Kogyo Co., Ltd. – Tokyo, Japan - shinichiro2_sogo@kk-grp.jp

Keywords: Segmentation of Remote Sensing Data, Vision-Language Model, Fine-tuning, Visual Prompting.

Abstract

Traditional semantic segmentation models often struggle with poor generalizability in zero-shot scenarios such as recognizing attributes unseen in the training labels. On the other hands, language-vision models (VLMs) have shown promise in improving performance on zero-shot tasks by leveraging semantic information from textual inputs and fusing this information with visual features. However, existing VLM-based methods do not perform as effectively on remote sensing data due to the lack of such data in their training datasets. In this paper, we introduce a two-stage fine-tuning approach for a VLM-based segmentation model using a large remote sensing image-caption dataset, which we created using an existing image-caption model. Additionally, we propose a modified decoder and a visual prompt technique using a saliency map to enhance segmentation results. Through these methods, we achieve superior segmentation performance on remote sensing data, demonstrating the effectiveness of our approach.

1. Introduction

In recent years, significant progress has been made in the field of semantic segmentation using deep learning techniques. Along with that, segmentation methods have already been studied in the fields related to remote sensing data, such as land-cover / land-usage classification and building extraction using satellite or aerial images. However, traditional vision-based semantic segmentation methods are limited in their ability to recognize new categories that are not included in training datasets, i.e., zero-shot learning. Taking the building extraction model as an example, the model is usually trained to extract all buildings but can not distinguish specific types of buildings like ’commercial facilities’, ’medical facilities’, or ’buildings with a red rooftop’. Incorporating every possible specific category into training datasets is impractical due to the high cost of annotation. Therefore, a model that can be easily generalized to an arbitrary category using remote sensing data is desired.

The emergence of large Vision-Language Models (VLMs), capable of leveraging semantic information from textual data, has revolutionized computer vision. Models like CLIP (Radford et al., 2021) have demonstrated impressive performance, especially in zero-shot settings, compared to traditional vision-based methods. However, VLM models are usually trained on large-scale datasets primarily sourced from the Internet, which lack sufficient data related to remote sensing data. This limitation explains the suboptimal performance of remote sensing data (Radford et al., 2021). The annotation for remote sensing data is extremely time-consuming and costly, not even to mention the scale of datasets required for training large VLMs.

To overcome this limitation, we propose a two-stage approach to fine-tune a VLM-based segmentation model, CLIPSeg (Lüddecke and Ecker, 2022), using publicly available remote sensing datasets without further manual annotation. We create a training dataset using public remote sensing datasets to fine-tune CLIPSeg. Since these datasets often lack captions or textual annotations, we utilize the image-caption model, BLIP (Li et al., 2022), to generate captions for each remote sensing image. To further enhance performance, we introduce an effective modification to the model. We replace transposed convolutions with linear interpolation in the decoder of CLIPSeg to generate segmentation masks with clearer boundaries. Additionally, we also propose a visual prompting engineering technique to further improve performance. Prompting engineering has attracted growing attention, especially in large language models, and has been proven to effectively improve performance without requiring the model to be re-trained. Specifically, we propose a simple visual prompting that uses saliency maps to enrich the visual input information. The experiments demonstrate that our proposed methods of fine-tuning, module modification, and visual prompts effectively enhance the performance of CLIPSeg on remote sensing data.

2. Related Works

2.1 Vision-Language Model

CLIP (Radford et al., 2021) is a groundbreaking model in the realm of VLMs. By leveraging a 400 million scale training dataset consisting of image-text pairs, CLIP learns a joint embedding space of textural and visual features using a contrastive learning scheme. This joint embedding space bridges the domain gap between natural languages and vision, thus CLIP shows superior generalizability to unseen data. Furthermore, the joint embedding of CLIP also serves as a promising initialization for many downstream computer vision tasks, such as image classification, image retrieval, etc. Among these tasks, CLIPSeg (Lüddecke and Ecker, 2022) proposed a semantic segmentation method that uses the pre-trained CLIP as an encoder. By conditioning the joint text-visual embedding space of the pre-trained CLIP with a simple segmentation decoder, CLIPSeg can estimate a binary segmentation mask based on an arbitrary text prompt at a test time. Since CLIPSeg can be easily applied to various settings such as zero-shot, one-shot, or few-shot, we choose this model as our base VLMs.

2.2 Image captioning

Image captioning, which estimates a textual description of an image, has been developed along with VLMs as image captions are required to learn VLMs. Among image captioning models, BLIP (Li et al., 2022), short for Border-Layout Integrated Picture, is a novel image captioning model that excels at generating visual content of images. Through incorporating visual features with spatial layout information, BLIP can generate captions that describe the objects in an image and their relative positions. The datasets to learn image captioning in remote sensing fields, on the other hand, have not been published enough as far as we are aware. UCM-captions (Qu et al., 2016) and RSICD (Lu et al., 2017) are the most ones. However, compared to normal image captioning datasets such as Visual Genome (Krishna et al., 2017), those remote sensing images lack sufficient scale and diversity.

2.3 Visual Prompting

The design of the input, often referred to as a prompt, plays a crucial role in the performance of language-related models. Recent studies, such as prompt tuning and prompt learning, have garnered significant attention in the topics concerning language-related models. Prompt tuning involves modifying the appearance of the input while preserving its content to enhance the model's performance. In Vision-Language Models, the visual prompt also plays a pivotal role in determining the model's output. (Shtedritski et al., 2023) proposed the use of unsupervised saliency maps to reduce noise in the input. Building on these studies, we utilize saliency maps as prior information to guide the model's focus on the target category.

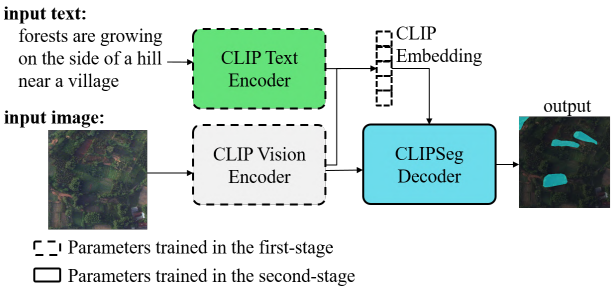


Figure 1. The Network of CLIPSeg

3. Methodology

3.1 Fine-tuning CLIPSeg

CLIPSeg comprises two main components, as illustrated in Figure 1: CLIP encoders and a CLIPSeg decoder. It takes an image and a short text caption as the input and estimates a binary segmentation mask that corresponds to the text caption. At first, the CLIP text encoder extracts textual embedding from the input text, and the CLIP vision encoder extracts visual embedding from the input image. Then, CLIP projects these embeddings to a joint embedding space. Subsequently, the CLIPSeg decoder relates the activations in joint embedding space with visual embedding to generate a binary mask. The original CLIPSeg is trained using the pre-trained CLIP as the encoder where the parameters were frozen during training.

However, the experiment results of CLIP (Radford et al., 2021) indicated its limited performance on remote sensing data due to the lack of samples from this domain in its training dataset, primarily consisting of common internet images.

To address this, we propose a two-stage fine-tuning approach for CLIPSeg to enhance its performance on remote sensing data. In the first stage, we fine-tune the CLIP encoders using remote sensing images with their corresponding captions. Subsequently, we fine-tune the decoder of CLIPSeg using remote sensing images, segmentation masks, and text captions, while keeping the CLIP encoders, fine-tuned in the first stage, fixed.

3.2 Dataset Preparation

The preparation of datasets for training Vision-Language Models (VLMs) for remote sensing tasks poses unique challenges due to the focus of existing public remote sensing datasets on vision-based tasks. Given that language-related models require large amounts of data, manually annotated captions are both time-consuming and costly, requiring skilled annotators. To address this, we leverage the image captioning model BLIP (Li et al., 2022), which excels at generating textual descriptions that describe the visual content of images.

We start by collecting eight remote sensing datasets (listed in Table 1) and use BLIP to generate captions for fine-tuning CLIPSeg. Specifically, we employ BLIP with a large Vision Transformer (ViT) as a backbone to generate captions from remote sensing images. BLIP usually takes an image and a query text token as the input and generates a text caption starting with the query text token based on the visual content of the image. However, a query text token that is not related to the image can mislead BLIP to generate a false caption, which may harm the training of the VLM. Thus, to guide BLIP in generating captions relevant to the image's content, we use the category label of the input image as the query text token. For the datasets with captions, we use the original captions as the training data. For Semantic Segmentation datasets such as WHDLD (Shao et al., 2018) and LoveDA (Wang et al., 2021), where images contain multiple categories, we extract the top-5 categories based on pixel count from the segmentation mask and use these top categories as text tokens. Examples of generated captions for fine-tuning CLIP, alongside those from the UCM image caption dataset (Qu et al., 2016), are shown in Figure 2. Examples of generated captions of LoveDA, for fine-tuning CLIPSeg, are shown in Figure 3, providing insight into the quality of the generated captions.

The generated captions are used to train CLIP and CLIPSeg, resulting in a dataset comprising approximately 0.2M image-caption pairs. This dataset is then split into two parts. Datasets other than LoveDA are used for the first training stage, which is fine-tuning CLIP. The rest of the dataset, LoveDA, is used for the second training stage, which is fine-tuning the decoder of CLIPSeg.

3.3 CLIPSeg Decoder

In the pilot experiments of CLIPSeg, we observed suboptimal segmentation masks, as illustrated in Figure 5 (b) where the boundary of the extracted target is not smooth and the score of neighboring patches is inconsistent. We attribute this suboptimal performance to the network design of the CLIPSeg decoder.

The CLIPSeg decoder comprises Transformer blocks and convolution layers, as shown in the upper part of Figure 4. The

Datasets	Training Stage	Annotation Category	Image Size	Image Number
DLRSD (Shao et al., 2018)	First Stage	Classification / Segmentation	256×256	2,100
MLRSNet (Qi et al., 2020)		Classification		109,16
PatternNet (Zhou et al., 2018)		Classification		30,400
NWPU-RESISC45 (Cheng et al., 2017)		Classification		31,500
UCM-captions (Qu et al., 2016)		Image Captioning	224×224	2,100
RSICD (Lu et al., 2017)		Image Captioning		10,921
WHDLD (Shao et al., 2020)		Segmentation	256×256	4,940
LoveDA (Wang et al., 2021)	Second Stage	Segmentation	1,000×1,000	4,191
Aerial Image Dataset(ours)		Segmentation (buildings only)		3,925

Table 1. Source of Our Generated Datasets

(a) DLRSD (b) MLRSNet (c) PatternNet (d) RESISC45 (e) UCM (f) WHDLD

Figure 2. Examples of Captions generated by BLIP are as follows. (a) baseball fields are seen from above with a few trees in the background. (b) commercial area is seen from above in this aerial photo. (c) dense residential is with a lot of trees and cars parked on the street. (d) freeway are lined with cars and utility lines on both sides. (e) An intersection with some cars parked at the roadside. (f) [”buildings are blue in color”, ”roads are in the middle of the picture”, ”pavements are shown in this aerial photo of a building”, ”vegetations are growing in a field near a blue building”, ”bare soil are seen in this aerial photo of a building”]

(a) buildings (b) agricultures (c) waters

Figure 3. Examples of LoveDA Dataset. The red regions represent the segmentation mask of the corresponding category. The captions generated by BLIP of each image are as follows. (a) buildings are in the distance of a large body of water. (b) agricultures are growing in a large area near a lake. (c) waters are seen in a large area of farmland and a lake.

Transformer blocks take text and vision embeddings from the CLIP encoder as input and project them to token embedding. Subsequently, the convolution layers reproject the token embedding to the original image size, estimating a binary segmentation mask. This lightweight network design minimizes the number of parameters, reducing training time. However, the decoder originally adopts two transposed convolution layers (kernel_size=4, stride=4) with activation functions only, to rescale the feature maps that are projected from the token embedding. The relationship between neighboring pixels is not investigated sufficiently in this manner. As a result, the boundaries are not smooth and the scores of neighboring pixels are inconsistent. In our pilot studies, we failed to improve the segmentation results by adding convolution layers after the transposed convolutions or reducing their rescaling factor (kernel_size=2, stride=2).

To address this issue, we propose replacing transposed convolutions with linear interpolation layers in the CLIPSeg decoder, similar to conventional vision-based segmentation models. The detail of the architecture of our proposed decoder is shown in the down part of Figure 4. We introduce four linear interpolation layers with a rescaling factor of 2 to increase the size of

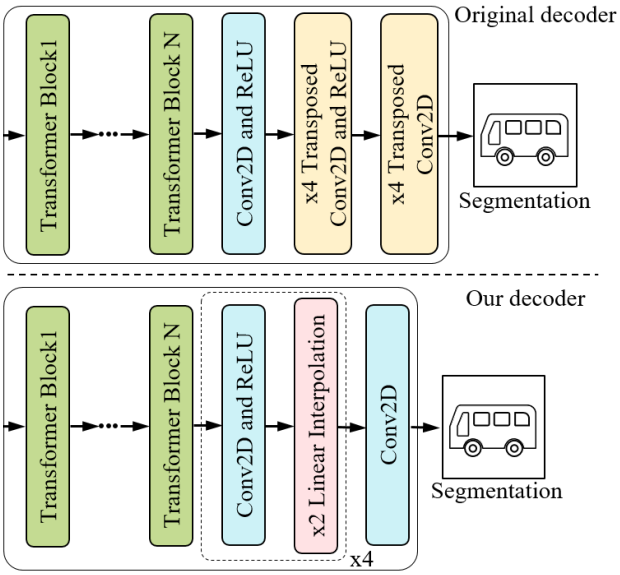


Figure 4. The Network Structure CLIPSeg Decoder

feature maps. Each interpolation layer is followed by a convolution layer and ReLU activation, except for the last interpolation layer, which is only followed by a convolution layer. This modification significantly improves segmentation mask quality, reducing inconsistency between patch images, as depicted in Figure 5 (d).

3.4 Visual Prompting

Effective prompt design can enhance model performance, particularly for language-related models. Studies (Shtedritski et al., 2023) have shown that simple visual prompts, like highlighting targets of interest with red circles, can lead to performance improvements. Motivated by these findings, we propose

generating visual prompts by multiplying the saliency map with the input image, emphasizing less salient regions.

We conducted experiments using saliency maps as supplementary information for visual prompting, exploring both unsupervised and supervised approaches. For unsupervised saliency maps, we employed TokenCut (Wang et al., 2022) and DINO (Caron et al., 2021) to generate saliency maps. DINO, a self-supervised training method for Vision Transformers (ViTs), has shown promise in extracting object features from images. TokenCut utilizes DINO to locate object regions in an image. However, these methods only extract saliency maps without semantic information, potentially leading to the inclusion of undesired objects in the prompt, contrary to our task's goal, where the segmentation mask should correspond to the query text.

For supervised saliency maps, we utilized Class Activation Map (CAM) (Zhou et al., 2016) and SmoothGradCAM++ (Omeiza et al., 2019). CAM highlights regions salient to a specific category based on the weights of the last convolutional layer of a CNN model, such as ResNet (He et al., 2016). Smooth Grad-CAM++ improves CAM by smoothing it with the average gradient while perturbing the input with random noise.

Unlike unsupervised saliency maps, CAM can extract saliency regions corresponding to the query text's category. However, most publicly available pre-trained CNN models, such as ResNet50, are trained on non-remote sensing datasets, potentially leading to performance degradation due to domain gaps and inconsistencies between the training dataset and our task. To mitigate this, we trained a ResNet50 using the classification datasets from Table 1, which have more than 70 categories, to generate saliency maps. Specifically, we estimate the activation map $CAM_i \in \mathbb{R}^{H \times W}$ of category $i \subset N$ that is related to the query text. Then, we calculate the mean of all activation maps and normalize it to 0˜1, as the following equation exhibits, to generate saliency map $S \in \mathbb{R}^{H \times W}$.

$$S = \frac{\frac{1}{N}\sum_i^N CAM_i}{max(\frac{1}{N}\sum_i^N CAM_i)}$$

4. Experiments

4.1 Datasets

We utilize a total of nine datasets for training and evaluation, as summarized in Table 1. LoveDA and our Aerial Image Dataset are employed for training the CLIPSeg decoder, while the remaining seven datasets are utilized for training the CLIP Vision Encoder and CLIP Text Encoder. To eliminate the influence of a domain gap, we also incorporate high resolution Aerial Image Dataset to train the CLIPSeg decoder in one of our experiments, which is not included in these public datasets. All images are divided into training and validation sets at a 4:1 ratio.

DLRSD: This dense labeling remote sensing dataset provides category labels and pixel-wise labels for images collected from the UC Merced Land Use Dataset (Yang and Newsam, 2010). It comprises 21 broad categories with 100 images per category, sourced from the USGS National Map Urban Area Imagery collection. The images have a pixel resolution of 1 foot.

MLRSNet: A multi-label remote sensing dataset containing 46 categories, with pixel resolutions ranging from 0.1m to 10m. The number of images per category varies from 1,500 to 3,000.

PatternNet: This dataset serves as a benchmark for remote sensing image retrieval tasks, offering 38 classes with 800 images per class. The data is collected from Google Earth imagery or via the Google Maps API for US cities, with pixel resolutions ranging from 0.062m to 4.693m.

NWPU-RESISC45: A remote sensing image scene classification dataset comprising 45 categories, with 700 images per category. Images are sourced from Google Earth imagery, with pixel resolutions ranging from 0.2m to 30m.

UCM-captions: An image captioning dataset for remote sensing data, utilizing the same images as DLRSD. Each image has 5 captions generated to describe its scene, with slight differences in captions between images of the same category.

RSICD: A remote sensing image captioning dataset comprising images from various sources with captions generated manually by experienced volunteers. The number of captions per image varies from 1 to 5, with various pixel resolutions.

WHDLD: A dense labeling remote sensing dataset with pixel-wise labels for 6 categories, consisting of images cropped from a larger remote sensing image of the urban region of Wuhan, China, with a pixel resolution of 0.3 m

LoveDA: Originally designed for domain adaptive semantic segmentation tasks, this dataset comprises images collected from 3 regions in China, with a pixel resolution of 0.3m and 7 categories.

Aerial Image Dataset: This dataset contains high-resolution aerial images ranging from 0.05m to 0.125m resolution, with 4,033 images collected from three regions of Japan. We only annotate a segmentation label for the building category in this dataset. Evaluation is performed on 108 images with a pixel resolution of 0.125m, divided into urban, rural, and mountainous regions. The remaining images, with pixel resolutions of 0.05m to 0.1m, are used for training.

4.2 Implementation Details

We utilized the CLIPSeg implementation from the Hugging Face library (Wolf et al., 2020), employing the ViT-B/16 architecture as an encoder. For comparison, we employed the officially released pre-trained CLIPSeg weights (hereinafter referred to as original CLIPSeg). The binary cross entropy loss function is adopted to optimize the decoder.

To generate supervised saliency maps for visual prompts, we additionally trained a ResNet50 classification model using datasets containing category labels, i.e., DLRSD, MLRSNet, PatternNet, and NWPU-RESISC45. Aiming at a unified classification dataset, we merged all identical categories, resulting in a dataset with 72 categories. The training of this classification CNN model was conducted using the PyTorch deep learning library. For generating visual prompts, we utilized the library of torch-cam (Fernandez, 2020). In a test time, we used visual prompts, that are generated from saliency maps, and query text prompts as input for CLIPSeg.

4.3 Evaluation of Training Strategy

We evaluated the performance of CLIPSeg using the IoU metric for building segmentation in the Aerial Image dataset. The results are presented in Tables 2, 3, and 4. Across all three

Model	Background	Building	Mean
Original CLIPSeg	33.45	36.86	35.15
Fine-tuned CLIPSeg	45.22	41.43	43.32
Fine-tuned CLIPSeg /w our decoder	57.26	45.63	51.45
Fine-tuned CLIPSeg /w our decoder /w Aerial Image	72.94	55.81	64.37

Table 2. Aerial Image Building Segmentation Evaluation Result (Urban Region)

Model	Background	Building	Mean
Original CLIPSeg	95.08	41.80	68.44
Fine-tuned CLIPSeg	97.21	52.83	75.02
Fine-tuned CLIPSeg /w our decoder	97.50	55.68	76.62
Fine-tuned CLIPSeg /w our decoder /w Aerial Image	97.59	55.94	76.77

Table 3. Aerial Image Building Segmentation Evaluation Result (Rural Region)

Model	Background	Building	Mean
Original CLIPSeg	98.64	26.62	62.63
Fine-tuned CLIPSeg	99.20	28.07	63.34
Fine-tuned CLIPSeg /w our decoder	99.33	35.93	67.63
Fine-tuned CLIPSeg /w our decoder /w Aerial Image	99.49	44.00	71.74

Table 4. Aerial Image Building Segmentation Evaluation Result (Mountainous Region)

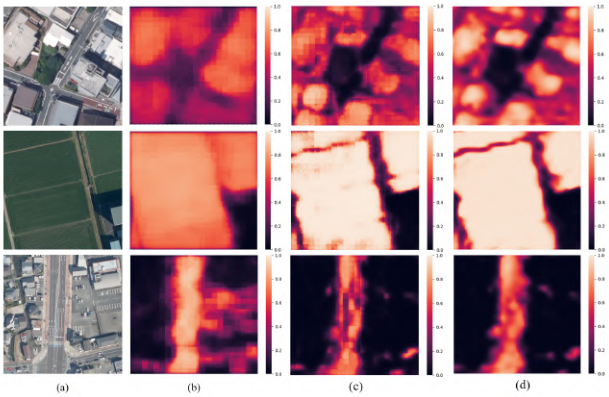


Figure 5. Examples of Segmentation Results. (a) shows the input image. (b) shows the Result of pre-trained CLIPSeg. (c) shows the result of our fine-tuned CLIPSeg w/ transposed convolutions layers. (d) shows the Result of our fine-tuned CLIPSeg w/ linear interpolation layers. The first row shows the results of the text prompt as ’buildings’. The second row shows the results of the text prompt as ’agriculture’. The third row shows the results of the text prompt as ’roads’.

evaluation regions, fine-tuning CLIPSeg significantly improved building segmentation accuracy compared to those with the original CLIPSeg.

In the urban region, the original CLIPSeg tended to over-segment the building areas, leading to noisy segmentation results. Our fine-tuned model effectively mitigated this over-segmentation issue, demonstrating an IoU improvement of over 10% when fine-tuned with aerial images. This highlights the challenge of domain gaps between the source and target data, even when trained on a large dataset. We also observed a similar improvement in the mountainous region, but the results in the rural region did not show significant enhancement. We attribute this to the fact that the proportion of buildings in the mountainous regions is less than 1%, where even minor changes can lead to substantial differences in evaluation results.

Method	Background	Building	Mean
Red Channel	70.14	50.61	60.38
Green Channel	71.78	43.01	57.40
Blue Channel	76.40	51.41	63.91
All Channels	79.77	56.21	67.99

Table 5. Aerial Image Building Segmentation Result Using Visual Prompts (CAM) on Different Channels (Urban Region).

Conversely, both the original and fine-tuned models achieved high accuracy in the background category in the rural and mountainous regions. This is due to the low density of buildings in these regions, which is less than 4%, and forests and agricultural lands being the dominant land-cover categories. In contrast, the urban region has an almost 30% building ratio.

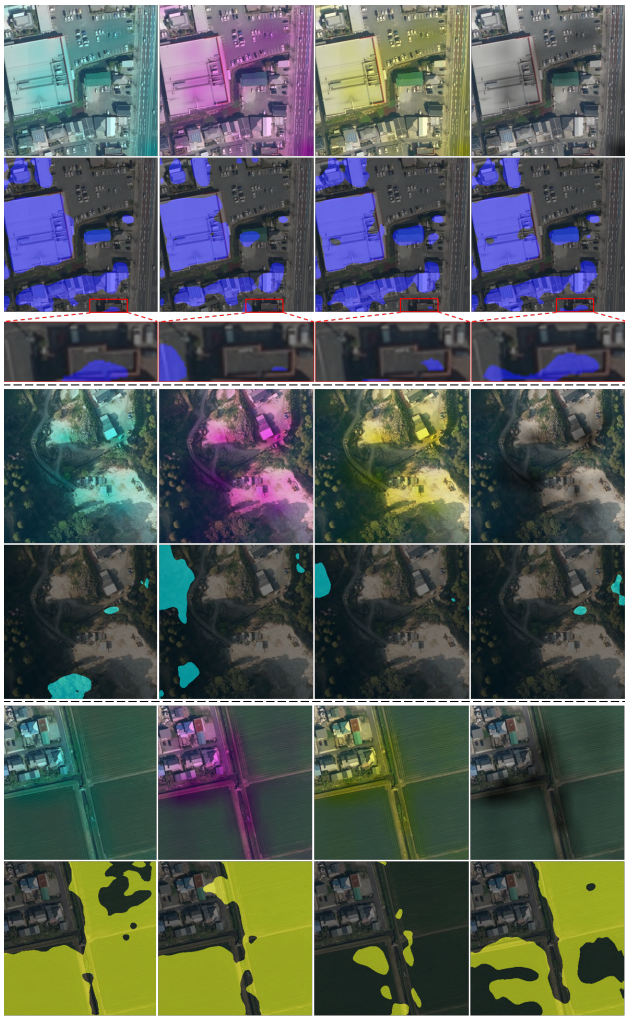
Figure 5 showcases examples of segmentation prediction scores. Columns (b) and (c) depict segmentation confidence predicted by the original decoder architecture of CLIPSeg, which estimates masks from patch images and struggles to address the inconsistency between them. This inconsistency is alleviated by using interpolation layers instead of transposed convolution layers, as shown in column (d).

Furthermore, the rows of Figure 5 demonstrate the segmentation results for query texts of buildings, agricultural lands, and roads. The original CLIPSeg exhibited rough segmentation masks, while our fine-tuned CLIPSeg produced more precise masks, extracting objects at the instance level.

4.4 Evaluation of Visual Prompt

(Shtedritski et al., 2023) suggested a method for creating visual prompts by overlaying a saliency map onto all channels of an input image. However, each channel may have varying effects on different categories. For instance, buildings, typically appearing in bright colors like light gray or white, are influenced by all channels, while the green channel tends to have a stronger impact on categories such as forests and agricultural regions.

We conducted experiments to test various methods of generating visual prompts, specifically by multiplying the real number of saliency map as a weight to the red, green, blue, or all channels of the input image. This manipulation can either lower the importance of certain colors (when applied to a single channel) or highlight certain regions of the image (when applied to all channels). We qualitatively evaluated building extraction in the urban region, forest extraction in the mountainous region, and agriculture extraction in the rural region. The generated visual prompts and their segmentation results are illustrated in Figure 6. For categories of forests and agriculture that are more closely related to the green channel, visual prompts that overlay the saliency map to the green channel tend to improve the segmentation performance. We did an additional quantitative evaluation for building extraction in the urban region. The result is shown in Table 5, which indicates that highlighting the



(a) Red Channel (b) Green Channel (c) Blue Channel (d) All Channels

Figure 6. Examples of different visual prompt overlay methods and their corresponding segmentation results using our fine-tuned CLIPSeg with our proposed decoder and aerial images. The 1st row shows the visual prompts of buildings. 2nd row shows the segmentation results of buildings. 3rd row shows the zoomed-in segmentation results of buildings. 4th row shows the visual prompts of forests. 5th row shows the segmentation results of forests. 6th row shows the visual prompts of agricultures. The last row shows the segmentation results of agricultures.

image contents across all channels yielded better building extraction performance compared to manipulating a single color channel.

Furthermore, we examined the effectiveness of visual prompts using unsupervised and supervised saliency maps, as depicted in Figure 7. Visual prompts based on unsupervised saliency maps tended to fail in cases where there were multiple categories, lacking the necessary semantic information. For example, TokenCut focused on buildings and ignored forests in the second row of Figure 7 (b), leading to a failure in forest extraction. Compared to CAM, SmoothGradCAM++ generates saliency maps with a larger contrast between foreground and background. As a result, in visual prompts, the values of the background become smaller, leading to increased difficulty in segmentation. Especially, omissions are more likely to occur.

Method	Background	Building	Mean
/wo visual prompt	72.94	55.81	64.37
TokenCut	60.80	45.62	53.21
DINO	57.12	43.05	50.09
CAM	79.77	56.21	67.99
SmoothGradCAM++	75.59	47.00	61.30

Table 6. Aerial Image Building Segmentation Result Using Visual Prompt (Urban Region)

Method	Background	Building	Mean
/wo visual prompt	97.59	55.94	76.77
TokenCut	96.27	41.49	68.88
DINO	97.27	41.55	69.41
CAM	98.19	56.98	77.58
SmoothGradCAM++	88.89	14.56	51.72

Table 7. Aerial Image Building Segmentation Result Using Visual Prompt (Rural Region)

Method	Background	Building	Mean
/wo visual prompt	99.49	44.00	71.74
TokenCut	99.25	37.73	68.49
DINO	99.46	14.13	56.80
CAM	99.41	39.46	69.44
SmoothGradCAM++	99.49	37.91	68.70

Table 8. Aerial Image Building Segmentation Result Using Visual Prompt (Mountainous Region)

We also conducted quantitative evaluations on building extraction using visual prompts generated by unsupervised and supervised saliency maps, with results shown in Tables 6 - 8. The segmentation performance is consistent with the qualitative evaluation results, that CAM can correctly react to the query text, as demonstrated in the segmentation results of Figure 8 and reduce segmentation noise compared to the original CLIPSeg. In urban and rural regions, visual prompts using CAM demonstrated the ability to enhance building extraction performance from aerial images. However, in the mountainous region, all methods failed to improve segmentation performance. This is likely due to the small and sparse distribution of buildings in the mountainous regions, causing saliency maps to overlook these objects.

5. Conclusion

In this study, we proposed a two-stage training approach to fine-tuning CLIPSeg and improving its performance using remote sensing data. Specifically, we generated a training dataset comprising approximately 0.2 million image-caption pairs using the image captioning model BLIP. We then introduced a modified decoder using linear interpolation layers instead of transposed convolution layers to address the inconsistency between patch images. Additionally, we proposed a simple visual prompt method that highlights relevant regions of query text to reduce segmentation noise. Our experimental results demonstrated that fine-tuning a Vision-Language Model (VLM) can significantly enhance segmentation performance.

The scale of the dataset, however, is still not sufficient for training a generalizable model, as we noticed that it does not perform as well as the original CLIPSeg in zero-shot cases. Furthermore, when the target object is small, such as buildings in the mountainous region, CAM tends to generate saliency maps

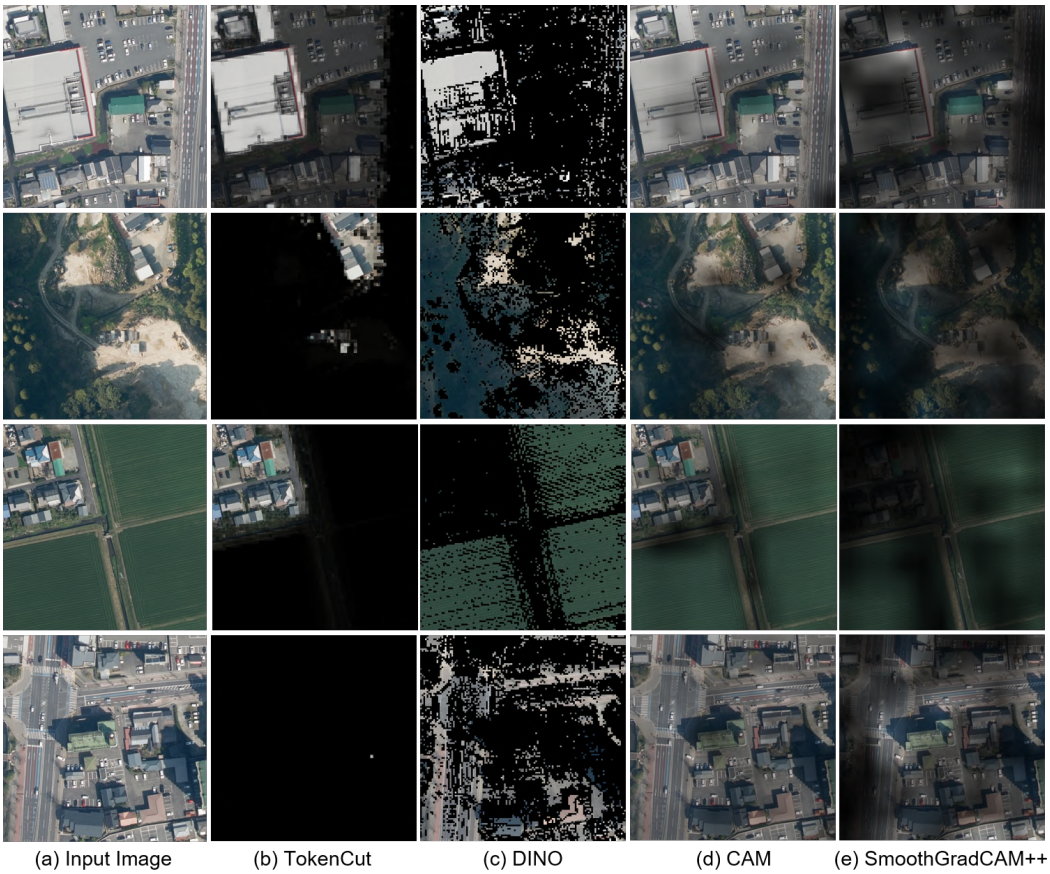


Figure 7. Examples of visual prompts generated by different saliency maps. Visual prompts of 2nd column (TokenCut) and 3rd column (DINO) are generated from unsupervised saliency maps, which do not contain semantic information. Visual prompts of 4th column (CAM) and 5th column (SmoothGradCAM++) are generated from saliency maps of one certain category. Specifically, the 1st row shows the visual prompts for buildings, the 2nd row shows the visual prompts for forests, the 3rd row shows the visual prompts for agricultures, and the 4th row shows the visual prompts for roads.

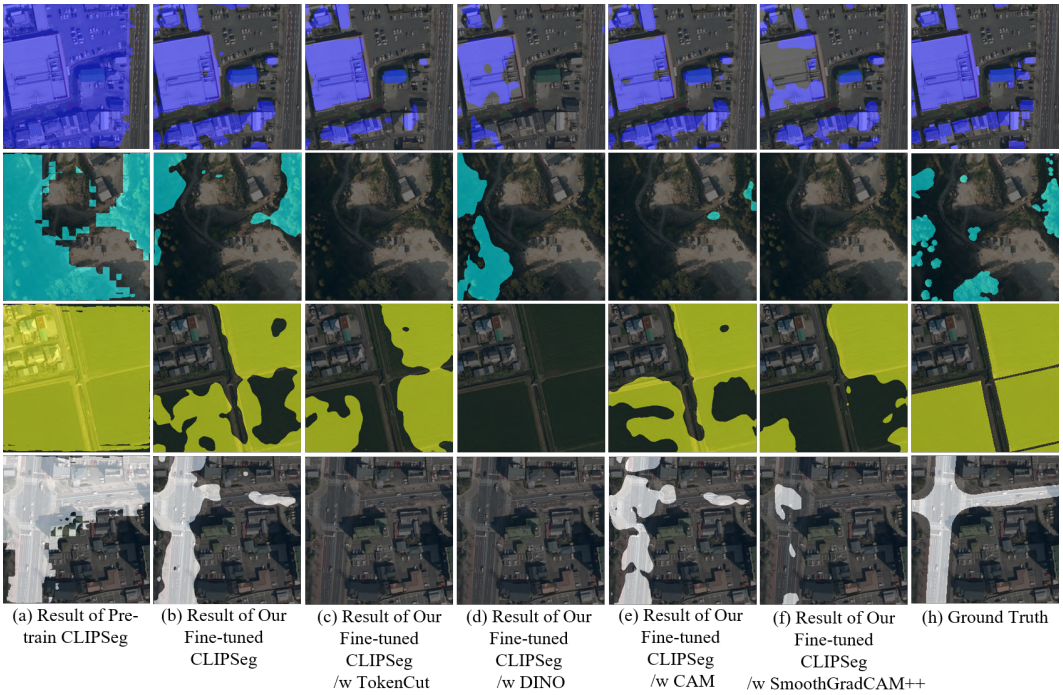


Figure 8. Examples of segmentation results using visual prompts. The 1st row shows the results of buildings. 2nd row shows the results of forests. 3rd row shows the results of agriculture. 4th row shows the results of roads.

with reduced accuracy. This is due to the small size of the attention map, which is 1/32 of the input image.

For future work, we aim to extract specific targets based on their attributes, such as color, shape, and usage. Since the captions generated by BLIP may not provide such detailed information, we plan to explore the use of larger vision-language models, such as LLaVA (Liu et al., 2023), for generating more informative captions.

References

Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., Joulin, A., 2021. Emerging properties in self-supervised vision transformers. *Proceedings of the International Conference on Computer Vision (ICCV)*.

Cheng, G., Han, J., Lu, X., 2017. Remote Sensing Image Scene Classification: Benchmark and State of the Art. *Proceedings of the IEEE*, 105(10), 1865-1883.

Fernandez, F.-G., 2020. Torchcam: class activation explorer. https://github.com/frgfm/torch-cam.

He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep Residual Learning for Image Recognition. *Proceedings of 2016 IEEE Conference on Computer Vision and Pattern Recognition*, CVPR ’16, IEEE, 770–778.

Krishna, R., Zhu, Y., Groth, O., Johnson, J., Hata, K., Kravitz, J., Chen, S., Kalantidis, Y., Li, L.-J., Shamma, D. A., Bernstein, M. S., Fei-Fei, L., 2017. Visual Genome: Connecting Language and Vision Using Crowdsourced Dense Image Annotations. *Int. J. Comput. Vision*, 123(1), 32–73. https://doi.org/10.1007/s11263-016-0981-7.

Li, J., Li, D., Xiong, C., Hoi, S., 2022. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. *International Conference on Machine Learning*, PMLR, 12888–12900.

Liu, H., Li, C., Wu, Q., Lee, Y. J., 2023. Visual instruction tuning. *NeurIPS*.

Lu, X., Wang, B., Zheng, X., Li, X., 2017. Exploring models and data for remote sensing image caption generation. *IEEE Transactions on Geoscience and Remote Sensing*, 56(4), 2183–2195.

Lüddecke, T., Ecker, A., 2022. Image segmentation using text and image prompts. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 7086–7096.

Omeiza, D., Speakman, S., Cintas, C., Weldermariam, K., 2019. Smooth Grad-CAM++: An Enhanced Inference Level Visualization Technique for Deep Convolutional Neural Network Models. *arXiv preprint arXiv:1908.01224*.

Qi, X., Zhu, P., Wang, Y., Zhang, L., Peng, J., Wu, M., Chen, J., Zhao, X., Zang, N., Mathiopoulos, P. T., 2020. MLRSNet: A multi-label high spatial resolution remote sensing dataset for semantic scene understanding. *ISPRS Journal of Photogrammetry and Remote Sensing*, 169, 337-350.

Qu, B., Li, X., Tao, D., Lu, X., 2016. Deep semantic understanding of high resolution remote sensing image. *2016 International Conference on Computer, Information and Telecommunication Systems (CITS)*, 1–5.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., Sutskever, I., 2021. Learning transferable visual models from natural language supervision. *Proceedings of the 38th International Conference on Machine Learning*, Proceedings of Machine Learning Research, 139, PMLR, 8748–8763.

Shao, Z., Yang, K., Zhou, W., 2018. Performance Evaluation of Single-Label and Multi-Label Remote Sensing Image Retrieval Using a Dense Labeling Dataset. *Remote Sensing*, 10(6). https://www.mdpi.com/2072-4292/10/6/964.

Shao, Z., Zhou, W., Deng, X., Zhang, M., Cheng, Q., 2020. Multilabel Remote Sensing Image Retrieval Based on Fully Convolutional Network. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 13, 318-328.

Shtedritski, A., Rupprecht, C., Vedaldi, A., 2023. What does clip know about a red circle? visual prompt engineering for vlms. *arXiv preprint arXiv:2304.06712*.

Wang, J., Zheng, Z., Ma, A., Lu, X., Zhong, Y., 2021. Loveda: A remote sensing land-cover dataset for domain adaptive semantic segmentation. J. Vanschoren, S. Yeung (eds), *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, 1, Curran.

Wang, Y., Shen, X., Hu, S. X., Yuan, Y., Crowley, J. L., Vaufreydaz, D., 2022. Self-supervised transformers for unsupervised object discovery using normalized cut. *Conference on Computer Vision and Pattern Recognition*.

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Le Scao, T., Gugger, S., Drame, M., Lhoest, Q., Rush, A., 2020. Transformers: State-of-the-art natural language processing. Q. Liu, D. Schlangen (eds), *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, Association for Computational Linguistics, Online, 38–45.

Yang, Y., Newsam, S., 2010. Bag-of-visual-words and spatial extensions for land-use classification. *Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems*, GIS ’10, Association for Computing Machinery, New York, NY, USA, 270–279.

Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., Torralba, A., 2016. Learning deep features for discriminative localization. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2921–2929.

Zhou, W., Newsam, S., Li, C., Shao, Z., 2018. PatternNet: A benchmark dataset for performance evaluation of remote sensing image retrieval. *ISPRS Journal of Photogrammetry and Remote Sensing*, 145, 197-209. Deep Learning RS Data.