Learnable Prompt for Few-Shot Semantic Segmentation in Remote Sensing Domain

Steve Andreas Immanuel
TelePIX
Seoul, South Korea
steve@telepix.net

Hagai Raja Sinulingga
TelePIX
Seoul, South Korea
hagairaja@telepix.net

Abstract

Few-shot segmentation is a task to segment objects or regions of novel classes within an image given only a few annotated examples. In the generalized setting, the task extends to segment both the base and the novel classes. The main challenge is how to train the model such that the addition of novel classes does not hurt the base classes performance, also known as catastrophic forgetting. To mitigate this issue, we use SegGPT as our base model and train it on the base classes. Then, we use separate learnable prompts to handle predictions for each novel class. To handle various object sizes which typically present in remote sensing domain, we perform patch-based prediction. To address the discontinuities along patch boundaries, we propose a patch-and-stitch technique by re-framing the problem as an image inpainting task. During inference, we also utilize image similarity search over image embeddings for prompt selection and novel class filtering to reduce false positive predictions. Based on our experiments, our proposed method boosts the weighted mIoU of a simple fine-tuned SegGPT from 15.96 to 35.08 on the validation set of few-shot OpenEarthMap dataset given in the challenge.

1. Introduction

Generalized Few-Shot Segmentation (GFSS) is a computer vision task wherein the model must effectively segment novel classes with limited examples alongside the base classes it has been trained on. This task holds significant relevance in remote sensing applications, where annotation costs are high and diverse user interests necessitate adaptable segmentation models. For instance, agricultural societies prioritize distinguishing between cultivated and fallow land, whereas civil registration departments require accurate house mapping for population estimation. Recognizing the critical importance of addressing these challenges, the OpenEarthMap Land Cover Mapping Few-Shot Challenge [33], jointly organized with the L3D-IVU CVPR 2024 Workshop, was convened to drive advancements in solving the problem.

The strategy for addressing the GFSS problem currently revolves around two approaches: 1) individually predicting each novel class and then merging the results using fusion techniques [13, 22], and 2) relearning the classifier so it can predict both base and novel classes simultaneously [10, 15, 28]. Our method follows the first approach but diverges from existing methods in that we train the model only once using data from the base class. For the novel classes segmentation, we only derive a prompt for each class obtained by training solely on the support set. The reason we chose this is due to the emergence of new foundation models with strong generalization capabilities [12, 23, 32]. The prompt for each novel class serves as an adaptation layer to handle a specific novel class characteristics. Therefore, our method is able to handle any number of novel classes without performance degradation on the base classes. This approach is both simple and straightforward, yet highly extensible to real-life scenarios.

Furthermore, this challenge presents characteristics commonly encountered in remote sensing, particularly varying object sizes. In remote sensing imagery, there are both large-scale objects like industrial complexes, roads, and lakes, as well as smaller objects such as trees, boats, and houses. Common strategies include employing multi-scale features [6] to capture both large and small objects, or segmenting the image into smaller patches [30]. Our approach incorporates elements of both strategies, utilizing a comprehensive foundational model comprising multi-scale layers and leveraging detailed inference results from smaller patches. Although segmenting images into patches is not widely favored due to the risk of information loss and discontinuous results along patch boundaries, our method addresses this issue by introducing a patch-and-stitch technique.

The contribution and novelty of our approach can be summarized as:

- We introduce a simple, yet effective method to handle novel classes prediction in few-shot setting using learnable prompts. Initial training is only done on the base classes while the learnable prompts are optimized using the frozen model.
- We propose a patch-and-stitch technique to smooth out the results in patch-based predictions, especially along the patch boundaries. We also incorporate similar prompt searching based on similarity and novel class filtering to further boost the performance.

2. Related Work

2.1. Semantic Segmentation

The basis of GFSS is semantic segmentation where models assign labels to individual pixels in an image. While methods like FCN [16] and encoder-decoder architectures [2, 24] have improved per-pixel predictions, incorporating context information through techniques like dilated convolutions and attention mechanisms has been proven to further enhance segmentation accuracy. Recent advancements, including pyramid pooling [11], parallel dilated convolutions [5], and the adoption of vision transformers [26], have led to significant improvements in segmentation quality. However, challenges persist in adapting these models to handle unseen classes without extensive fine-tuning using sufficiently annotated data.

2.2. Few-Shot Semantic Segmentation

Few-shot semantic segmentation (FSS) is formulated to specifically answer the challenge wherein pixel-wise labeling is required for novel classes with limited support examples, mainly focusing on the novel class prediction only without the base class. Methods like PL [8] and PANet [29] adapt prototype learning, while ASR [14] learns orthogonal prototypes. Other method like OSLSM [25] assigns weights to the final classifier and PFENet [27] leverages pre-trained backbone knowledge and addresses spatial inconsistency with a Feature Enrichment Module (FEM). Despite FSS models' effectiveness with support samples, the practical scenarios involving both base and novel classes on the target image has not been covered by these methods.

2.3. Generalized Few-Shot Semantic Segmentation

In the generalized setting, the task is to predict not only the novel classes but also the base classes. Several approaches that have been proposed can be seperated into two categories based on the prediction design setting.

Directly predict all novel and base classes

- *POP* [15], employs orthogonal loss to separate base and novel classes from the background. The training process involves two stages: initially training a PSPNet to obtain the base model, and then training the novel model using combined data from both base and novel classes, with the base data chosen through random sampling. Notably, the POP model updates labels from the base class at every epoch.
- *DIaM* [10], focuses on finetuning the classifier using a modified InfoMax framework and knowledge distillation techniques to retain base class knowledge.
- *CAPL* [28], pioneers GFSS with three classifiers: one for base class together with novel class, one for base class only, and a classifier to combine both result.

Predict classes individually

- *BAM* [13], trains a ResNet50 encoder and decoder for each novel class, utilizing weights to select support images. However, this approach is tailored for the one-class novel scenario and the performance is reported to degrade as the number of novel classes increased [10].
- *HSNet* [22], utilizes a 4D sparse correlation tensor over feature pyramids generated by a CNN-based backbone network. Although HSNet is not designed for GFSS, but adaptation of the model as reported in [15] show a competitive result.

2.4. Foundation Model

The concept of training large-scale models through semi-supervised learning on extensive datasets, referred to as an upstream task, and subsequently employing them as foundation models for fine-tuning on downstream tasks, has gained significant prominence recently [23, 31]. These large models exhibit strong generalization capabilities due to their training on diverse datasets. Some of these models have been explored for addressing general remote sensing tasks such as segmentation, object detection, change detection and super resolution [1, 3, 7, 18, 20, 21]. Notably, recent research [32] demonstrates that even without fine-tuning on specialized remote-sensing data, this approach can rival SOTA FSS methods, particularly with an increasing number of shots. This observation motivate our direction in exploring the potential of the foundation model, as no prior attempts have been made to tackle the GFSS problem using such approach.

3. Method

3.1. Preliminaries

Given an RGB image $\mathbf{X} \in \mathbb{R}^{H \times W \times 3}$, where H and W denote the height and width of the image, the goal of semantic segmentation task is to predict semantic map $\mathbf{Y} \in \mathbb{R}^{H \times W}$. Each pixel $\mathbf{Y}_{ij}$ corresponds to a class label from a predefined set $\mathcal{C} = \{c_1, ..., c_P\}$, where P is the number of classes, reflecting the semantic class of the corresponding pixel $\mathbf{X}_{ij}$. In few-shot setting, the training dataset contains only base classes, while for each novel class, we are given k samples of images and their semantic maps which only

contain the novel class.

3.2. Model Architecture

We use SegGPT [32] as the foundation model due to its strong generalizability. Internally, SegGPT uses ViT [9] as the backbone and is trained with smooth-l_1 loss.

3.3. Training

We follow the masked image modeling (MIM) approach where the objective is to reconstruct the masked regions of the input image. To this end, a pair of images is fed to the model instead of only one image. Prompt image $\mathbf{X}^p$ and target image $\mathbf{X}^t$, and their corresponding semantic maps $\mathbf{Y}^p$ and $\mathbf{Y}^t$, are provided, where certain patches of the semantic maps are masked as shown in Fig. 1. All $\mathbf{X}^p$, $\mathbf{X}^t$, $\mathbf{Y}^p$, and $\mathbf{Y}^t$ need to be of the same dimension H x W x 3. Therefore, the semantic maps are transformed into image space by mapping each class label into a color using color map $\mathcal{M} : \mathbb{R} \rightarrow \mathbb{R}^3$. This color is randomized for each data sample. The idea is to force the model to learn the contextual information in order to reconstruct the masked region, rather than exploiting the color [32]. This is particularly useful in few-shot setting, because it prevents overfitting to the base classes.

3.3.1 Base Classes

The base classes are trained following the standard MIM approach. Each data sample consists of $\mathbf{X}^p$, $\mathbf{X}^t$, $\mathbf{Y}^p$, and $\mathbf{Y}^t$. In order to select $\mathbf{X}^p$ and $\mathbf{X}^t$, we initially generate all possible pair combinations of images in the training set. Then, we adopt different masking strategy for each pair depending on the classes present in each image. If $\mathbf{X}^p$ and $\mathbf{X}^t$ contain at least one different class, we randomly mask α portion of the patches of $\mathbf{Y}^p$ and $\mathbf{Y}^t$ as shown in Fig. 1a. Alternatively, we mask the whole $\mathbf{Y}^t$ as shown in Fig. 1b if and only if $\mathbf{X}^p$ and $\mathbf{X}^t$ contain the exact same classes.

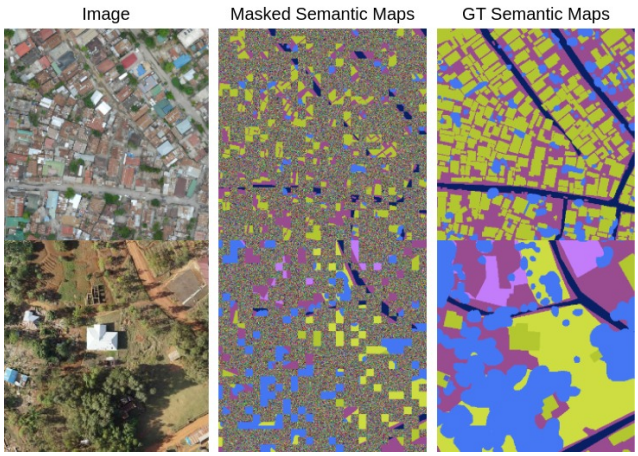
The idea is that if $\mathbf{X}^p$ and $\mathbf{X}^t$ contain the same classes, then given $\mathbf{Y}^p$, the model should be able to predict the whole $\mathbf{Y}^t$. In contrast, if their classes differ, the model should reconstruct the masked patches by leveraging contextual information from the unmasked regions.

3.3.2 Novel Classes

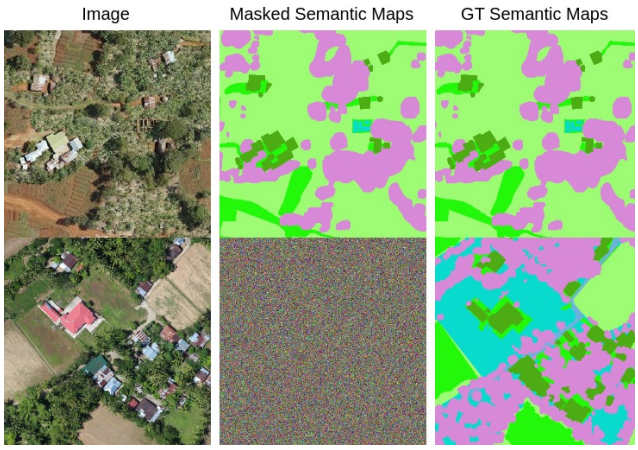
Due to the limited number of samples, the novel classes cannot be trained with the same approach as base classes. SegGPT inherently has strong few-shot capabilities by feeding the k samples as the segmentation context. However, as we show in Tab. 1, it still does not suffice for this challenge.

The primary obstacle in few-shot setting is how to make the model able to predict the novel classes given few samples, while simultaneously retaining the performance on the base classes. To this end, we use a learnable prompt $\mathbf{Z}$ which acts as $\mathbf{X}^p$ and $\mathbf{Y}^p$. After we train the model on the base classes, we freeze the whole model and optimize only $\mathbf{Z}$. Given there are N novel classes, we create $\{\mathbf{Z}^i\}_{i=1}^N$, each tailored to the characteristics of the corresponding i-th novel class, and train them independently using samples from each class. The main strength of this approach is that the introduction of novel classes does not compromise the performance of the base classes. Moreover, the optimization for $\mathbf{Z}$ using the k samples is much faster than the initial training on the base classes. Every learnable prompt for each novel class only amounts to about 5MB of model parameters, which is highly practical. To predict the novel classes, we simply utilize the corresponding learned prompt $\mathbf{Z}$ in a plug-and-play manner.

For novel classes training, we only use the half masking strategy. Since the semantic maps for the novel classes only contains the novel classes, this task reduces into a binary classification between the novel class and background.



(a) Random masking strategy



(b) Half masking strategy

Figure 1. Masked image modeling approach

As the novel class is typically confined to small regions within the image, direct prediction of the whole image would make the model trivially predict everything as background. Therefore, the learnable prompt $\mathbf{Z}$ is optimized in two phases. In phase 1, we adopt a sliding window approach to crop the image into smaller patches, utilizing only those containing the novel class to optimize $\mathbf{Z}$, thus facilitating the model's comprehension of the novel class. Subsequently, in phase 2, we incorporate all patches, including those featuring only background, to mitigate false positive predictions of the novel class. Additionally, we only use white color to represent the novel class and black color to represent background, as opposed to random color as in base classes training.

3.4. Inference

Inference works similarly as half masking strategy in training. The image $\mathbf{X}^p$ and its semantic map $\mathbf{Y}^p$ acts as the prompt to give contextual information. Then, given the target image $\mathbf{X}^t$, the model predicts $\hat{\mathbf{Y}}^t$. We generate a fixed color map $\mathcal{M}$ and use it to transform $\mathbf{Y}^p$ into image space, and its inverse $\mathcal{M}^{-1}$ to transform the prediction $\hat{\mathbf{Y}}^t$ back to class label space. The class of the ij-th pixel in $\hat{\mathbf{Y}}^t$ can be determined as follows,

$$class(\hat{\mathbf{Y}}^t_{ij}) = \arg\min_{c \in \mathcal{C}} d(\hat{\mathbf{Y}}^t_{ij}, \mathcal{M}^{-1}(c)), \qquad (1)$$

where c iterates over the set of class labels $\mathcal{C}$, and d is cosine similarity distance.

Image Similarity Search. The quality of the prompt $\mathbf{X}^p$ and $\mathbf{Y}^p$ greatly affects the prediction result $\hat{\mathbf{Y}}^t$. In general, the more similar $\hat{\mathbf{X}}^p$ and $\hat{\mathbf{X}}^t$, the better the result. Additionally, SegGPT can incorporate multiple prompts in order to generate more accurate results. We leverage CLIP-ViT [23] to extract the embeddings of each images in the training set. Then, we retrieve the top-l most similar images to $\mathbf{X}^t$ using cosine similarity and use them as the prompt.

Patch-and-Stitch. In remote sensing domain, the objects are typically small and scattered across the image. Processing the whole image directly often leads to objects not being detected. Therefore, we partition the image into 2x2 equal non-overlapping patches and perform the prediction on those patches independently. To get the result for the whole image, we can simply combine the prediction result on those patches directly. However, there might be some artefacts along the edges of the patches shown by the discontinuity of color (see Fig. 4 column 3). To mitigate this, we perform additional predictions on the middle regions that overlap adjacent patches as illustrated in Fig. 2. Instead of predicting the entire overlapping region, we focus solely on predicting the middle portion, while the remaining areas are filled using the previous predictions from the non-overlapping patches to give more context. This process effectively frames the task as an image inpainting task, enabling seamless integration of non-overlapping patch predictions.

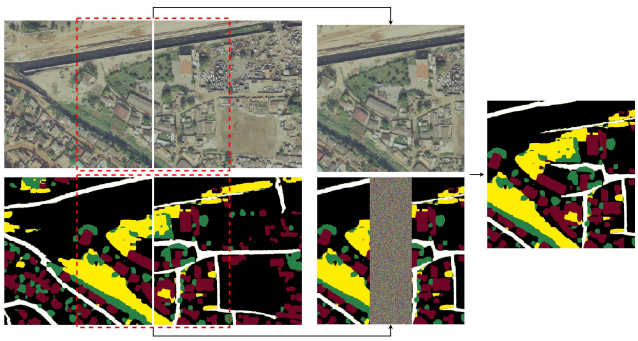


Figure 2. Seamless stitching between non-overlapping patches

To get the final prediction containing both base and novel classes, we first perform prediction for the base classes. For each of the novel class, we do not utilize the image similarity search to get similar images as the prompt is essentially replaced with $\mathbf{Z}$. Instead, we calculate the similarity between the target image $\mathbf{X}^t$ and the k given samples, analogously using CLIP-ViT and cosine similarity distance. If the similarity does not exceed a certain threshold we skip processing the corresponding novel class for the target image altogether. The idea is if $\mathbf{X}^t$ is not similar with the k given samples, then it is unlikely to contain the novel class. This approach helps to further reduce false positive prediction of novel classes. Subsequently, we simply overlay the novel classes predictions on top of the base classes prediction.

4. Experiment

4.1. Dataset

The dataset used in the challenge is a few-shot dataset consists of 408 samples of the original OpenEarthMap (OEM) benchmark dataset [33]. The challenge dataset extends the original 7 semantic classes (excluding background class) of the OEM to 15 classes, which is split into 7:4:4 for base classes, validation novel classes, and test novel classes, respectively. All base, validation novel, and test novel classes are disjointed.

The 408 samples are split into 258 as training set, 50 as validation set, and 100 as test set. Validation and test set only contains novel classes. For each novel class, 5 example images are given with their corresponding semantic maps. Therefore, 20 images are given as the support set, while 30 and 80 images are used as the query set for the validation and test set, respectively.

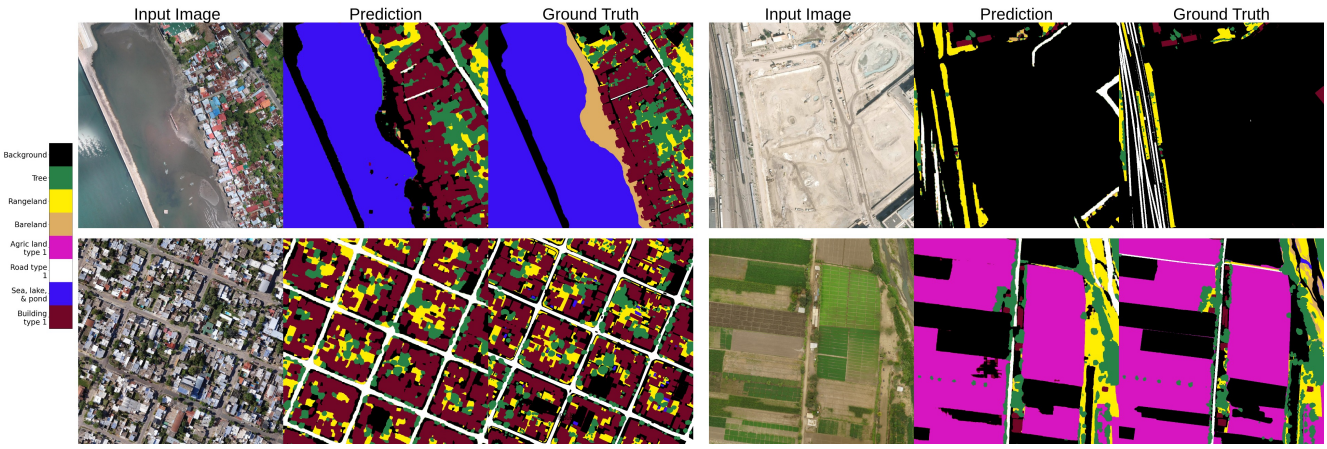


Figure 3. Semantic map prediction results on the training set

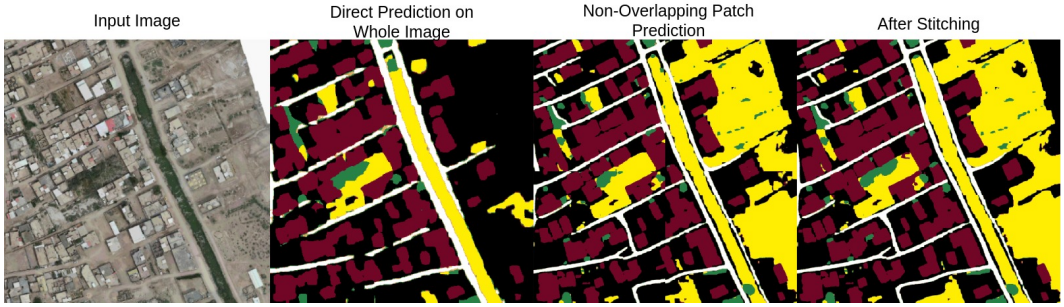


Figure 4. Seamless stitching produces more detailed and continuous result

4.2. Implementation Details

We initialize SegGPT with the pretrained checkpoint provided by the original authors [32]. Other initial foundation model checkpoints that are specialized on remote sensing domain [1, 3, 7, 18, 19, 21] can also be used, which might improve the performance. We leave this for future work. During training, we use image augmentations including, random cropping, random horizontal and vertical flip, and color jittering. We set the α value for the random masking to 0.75 empirically. During inference, we select the top-5 most similar images as the prompt. We use AdamW [17] optimizer and cosine annealing rate scheduler [4] with learning rate of 1e-4 and linear warmup for 500 steps by default. To optimize the learnable prompt, we simply treat $\mathbf{Z}$ as model parameters of size $\mathbb{R}^{H\times W\times 3}$ times two (to represent $\mathbf{X}^p$ and $\mathbf{Y}^p$). All codes are implemented in Python using PyTorch. Training and experiments are conducted using 4 Nvidia RTX A6000.

4.3. Evaluation Metric

The evaluation metric is weighted mean intersection-over-union (mIoU) over all classes excluding the background. As the focus of the challenge is in the novel class, the final evaluation metric is calculated as $0.4 * \text{base classes mIoU} + 0.6 * \text{novel classes mIoU}$.

4.4. Quantitative Results

The mIoU results[1] on the validation set of our proposed method are presented in Tab. 1. We can see incremental improvements for each of the method that we employ.

Simply using SegGPT that is finetuned on the OEM dataset only gives out mIoU of 15.96, mainly because the model is unable to detect any of the novel classes. Utilizing similar images as the prompt offers small improvement of +1.86 because it only improves the mIoU on the base classes. It is only after the integration of the learnable prompt that the model becomes capable of predicting the novel classes, leading to a substantial increase of mIoU by +7.44.

Incorporating the patch and stitch approach further enhances the mIoU to 29.41, demonstrating a significant improvement in capturing finer details and increasing overall accuracy. Filtering the novel classes based on the image similarity to the given k samples also proves to be very ef-

[1]The IoU for each class is not available because the detailed evaluation for some of our submissions cannot be viewed in the submission portal.

Table 1. The impact of each method used on the mIoU of the validation set

Method	mIoU
SegGPT Baseline [32]	15.96
+ Similar image prompts	17.82
+ Learnable prompt on novel classes	25.26
+ Patch and stitch	29.41
+ Filter on novel classes	35.08

fective, shown by another +5.67 increase in mIoU. By filtering the novel classes, we reduce the false positive prediction. Considering that we overlay the predictions of novel classes on top of the base classes, false positive predictions negatively impact not only the mIoU of the novel classes but also that of the base classes. As for the test set, we obtain a weighted mIoU of 36.52[2].

4.5. Qualitative Results

Fig. 3 shows the results of our model on the training set. Our model ables to predict well on most classes. One particular class that our model struggle with is bareland, as shown in the top-left image in Fig. 3. This is mainly due to the very limited number of samples in the training set, as well as the inconsistency of the class definitions [33].

In Fig. 4, we compare the results when directly predicting the whole image and using the patch and stitch method that we proposed. By using non-overlapping patch prediction, the model can capture finer details in the segmentation between buildings. Finally, the stitching mechanism allows aggregation of predictions from each patch seamlessly.

4.6. Impact of Color Map on Inference

While we can use any random color to fill $\mathcal{M}$, we observe empirically that some colors lead to higher performance. Specifically, we want objects that are typically located close together to have as different color as possible, e.g. bareland and sea, tree and road. Colors that are far apart from each other in RGB color space make it easier for the model to separate the close pixels between objects, reducing ambiguity. This is also similar to humans who can easily distinguish two adjacent objects that have contrasting colors.

4.7. Limitations

Due to how inference works in SegGPT, the results highly depend on the prompt that is given, therefore finding the most suitable prompt given an image is a crucial aspect to consider. The patch-and-stitch approach that we use in the inference also introduce additional computational cost as a trade-off for details and accuracy on the results.

[2]The breakdown of each method is not available for the test set as in Tab. 1 due to the limited number of submissions during the competition and the submission portal is closed once the competition ends

5. Conclusion

In this work, we proposed a method to handle novel classes prediction in few-shot setting using learnable prompt. We also introduced some additional techniques including prompt based on image similarity, patch-and-stitch, and novel class filtering which led to substantial performance improvements.

References

[1] Peri Akiva, Matthew Purri, and Matthew Leotta. Self-supervised material and texture representation learning for remote sensing tasks. In *CVPR*, pages 8203–8215, 2022. 2, 5

[2] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *PAMI*, pages 2481–2495, 2017. 2

[3] Favyen Bastani, Piper Wolters, Ritwik Gupta, Joe Ferdinando, and Aniruddha Kembhavi. Satlaspretrain: A large-scale dataset for remote sensing image understanding. In *ICCV*, pages 16772–16782, 2023. 2, 5

[4] John Bridle. Training stochastic model recognition algorithms as networks can lead to maximum mutual information estimation of parameters. In *ICLR*, pages 211–217, 1989. 5

[5] 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. 2

[6] Xiang Cheng and Hong Lei. Semantic segmentation of remote sensing imagery based on multiscale deformable cnn and densecrf. *Remote Sensing*, page 1229, 2023. 1

[7] Yezhen Cong, Samar Khanna, Chenlin Meng, Patrick Liu, Erik Rozi, Yutong He, Marshall Burke, David Lobell, and Stefano Ermon. Satmae: Pre-training transformers for temporal and multi-spectral satellite imagery. *NEURIPS*, pages 197–211, 2022. 2, 5

[8] Nanqing Dong and Eric P Xing. Few-shot semantic segmentation with prototype learning. In *BMVC*, page 4, 2018. 2

[9] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *CVPR*, 2022. 3

[10] Sina Hajimiri, Malik Boudiaf, Ismail Ben Ayed, and Jose Dolz. A strong baseline for generalized few-shot semantic segmentation. In *CVPR*, pages 11269–11278, 2023. 1, 2

[11] Qibin Hou, Li Zhang, Ming-Ming Cheng, and Jiashi Feng. Strip pooling: Rethinking spatial pooling for scene parsing. In *CVPR*, pages 4003–4012, 2020. 2

[12] 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. In *ICCV*, pages 4015–4026, 2023. 1

[13] Chunbo Lang, Gong Cheng, Binfei Tu, and Junwei Han. Learning what not to segment: A new perspective on few-shot segmentation. In *CVPR*, pages 8057–8067, 2022. 1, 2

[14] Binghao Liu, Yao Ding, Jianbin Jiao, Xiangyang Ji, and Qixiang Ye. Anti-aliasing semantic reconstruction for few-shot semantic segmentation. In *CVPR*, pages 9747–9756, 2021. 2

[15] Sun-Ao Liu, Yiheng Zhang, Zhaofan Qiu, Hongtao Xie, Yongdong Zhang, and Ting Yao. Learning orthogonal prototypes for generalized few-shot semantic segmentation. In *CVPR*, 2023. 1, 2

[16] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, pages 3431–3440, 2015. 2

[17] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint, arXiv:1711.05101*, 2019. 5

[18] Utkarsh Mall, Bharath Hariharan, and Kavita Bala. Change-aware sampling and contrastive learning for satellite images. In *CVPR*, pages 5261–5270, 2023. 2, 5

[19] Utkarsh Mall, Cheng Perng Phoo, Meilin Kelsey Liu, Carl Vondrick, Bharath Hariharan, and Kavita Bala. Remote sensing vision-language foundation models without annotations via ground remote alignment. In *ICLR*, 2024. 5

[20] Oscar Manas, Alexandre Lacoste, Xavier Giró-i Nieto, David Vazquez, and Pau Rodriguez. Seasonal contrast: Unsupervised pre-training from uncurated remote sensing data. In *ICCV*, pages 9414–9423, 2021. 2

[21] Matías Mendieta, Boran Han, Xingjian Shi, Yi Zhu, and Chen Chen. Towards geospatial foundation models via continual pretraining. In *ICCV*, pages 16806–16816, 2023. 2, 5

[22] Juhong Min, Dahyun Kang, and Minsu Cho. Hypercorrelation squeeze for few-shot segmentation. In *ICCV*, 2021. 1, 2

[23] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 1, 2, 4

[24] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, pages 234–241. Springer, 2015. 2

[25] Amirreza Shaban, Shray Bansal, Zhen Liu, Irfan Essa, and Byron Boots. One-shot learning for semantic segmentation. *arXiv preprint arXiv:1709.03410*, 2017. 2

[26] Robin Strudel, Ricardo Garcia, Ivan Laptev, and Cordelia Schmid. Segmenter: Transformer for semantic segmentation. In *ICCV*, pages 7262–7272, 2021. 2

[27] Zhuotao Tian, Hengshuang Zhao, Michelle Shu, Zhicheng Yang, Ruiyu Li, and Jiaya Jia. Prior guided feature enrichment network for few-shot segmentation. *PAMI*, (2):1050–1065, 2020. 2

[28] Zhuotao Tian, Xin Lai, Li Jiang, Shu Liu, Michelle Shu, Hengshuang Zhao, and Jiaya Jia. Generalized few-shot semantic segmentation. In *CVPR*, 2022. 1, 2

[29] Kaixin Wang, Jun Hao Liew, Yingtian Zou, Daquan Zhou, and Jiashi Feng. Panet: Few-shot image semantic segmentation with prototype alignment. In *ICCV*, pages 9197–9206, 2019. 2

[30] Xiaolei Wang, Zirong Hu, Shouhai Shi, Mei Hou, Lei Xu, and Xiang Zhang. A deep learning method for optimizing semantic segmentation accuracy of remote sensing images based on improved unet. *Scientific reports*, page 7600, 2023. 1

[31] Xinlong Wang, Wen Wang, Yue Cao, Chunhua Shen, and Tiejun Huang. Images speak in images: A generalist painter for in-context visual learning. In *CVPR*, 2023. 2

[32] Xinlong Wang, Xiaosong Zhang, Yue Cao, Wen Wang, Chunhua Shen, and Tiejun Huang. SegGPT: Segmenting everything in context. In *ICCV*, 2023. 1, 2, 3, 5, 6

[33] Junshi Xia, Naoto Yokoya, Bruno Adriano, and Clifford Broni-Bediako. Openearthmap: A benchmark dataset for global high-resolution land cover mapping. In *WACV*, pages 6254–6264, 2023. 1, 4, 6