A Study of Zero-Cost Proxies for Remote Sensing Image Segmentation

Chen Wei[1] **Kaitai Guo**[2] **Yiping Tang**[2] **Junyao Ge**[2] **Jimin Liang**[2]

[1]College of Economics and Management, Xi'an University of Posts&Telecommunications, China
[2]School of Electronic Engineering, Xidian University, China

Abstract Zero-cost proxies neural architecture search (NAS) can efficiently evaluate the performance of neural architectures and significantly reduce the search cost, but existing zero-cost proxies NAS are mainly focus on image classification. This paper investigates whether zero-cost proxies can accurately rank neural architectures used for remote sensing image segmentation. Firstly, we design a new search space for remote sensing image segmentation, denoted as SEG101, which considers enhancing the feature maps' contextual information and improving the fusion of feature maps. Secondly, a predictor-based NAS algorithm is adopted to explore SEG101 and collect neural architectures from it. Finally, zero-cost proxies are analysised by using the collected neural architectures. The preliminary experimental results illustrate that SEG101 is a promising search space and also show that zero-cost proxies can be used by predictor-based NAS for remote sensing image segmentation. [1]

1 Introduction

Neural Architecture Search (NAS) can automatically design neural architectures, and due to its convenience and superior performance, NAS has been applied to many tasks, including image classification (White et al., 2021; Liu et al., 2019b), object detection (Wang et al., 2020b; Xiong et al., 2021), semantic segmentation (Liu et al., 2019a; Zhang et al., 2021; Ding et al., 2021), and natural language processing (Klyuchnikov et al., 2020; Li et al., 2021). Search space, search strategy, and performance estimation strategy are the main components of NAS (Elsken et al., 2018). The search space defines all the potential neural architectures that can be selected, and search strategies are used to find the neural architecture with the best performance for target tasks from the search space. The search strategy uses the performance of neural architectures to explore the search space. Since the evaluation of neural architecture is time-consuming, the researchers propose performance estimation strategies to speed up the procedure of evaluating neural architectures. Zero-cost proxies (Abdelfattah et al., 2021; Mellor et al., 2021; Chen et al., 2021) are newly proposed neural architecture performance estimation strategies that can estimate the performance of neural architectures without training. Since existing zero-cost proxies estimation strategies are widely used to estimate the performance of image classification neural architectures, this paper tries to extend the scope of usage of zero-cost proxies and investigate whether zero-cost proxies can be used to accurately rank semantic segmentation neural architectures.

Semantic segmentation, which assigns a category to each pixel in an image, has been widely used for autonomous driving (Cordts et al., 2016; Chen et al., 2018b) and medical imaging analysis (Ronneberger et al., 2015a; Tang et al., 2021). Due to the convenience and efficiency of NAS, researchers have applied NAS to semantic segmentation and searched architectures that can achieve comparable performance to human-designed networks (Liu et al., 2019a; Zhang et al., 2021;

[1]https://github.com/auroua/zero_costs_nas_rs
Corresponding author: Jimin Liang

Ding et al., 2021). Recently proposed NAS algorithms for semantic segmentation mainly use the gradient-based search strategy, and the searched architecture is applied to do the segmentation of natural images. Since the distribution of categories in natural images is different from that of remote sensing images, the search space designed for remote sensing image segmentation should take into account the properties of remote sensing images. Existing studies about NAS for remote sensing images are mainly applied to scene classification (Wang et al., 2021; Peng et al., 2021; Broni-Bediako et al., 2022; Ma et al., 2021). In contrast to previous studies, this paper designs a new search space for remote sensing image segmentation and utilizes the predictor-based search strategy (Wei et al., 2022) to search the optimal architecture from the newly designed search space.

In summary, the main contribution of this paper can be summarized as follows.

- This paper designs a new search space for remote sensing image segmentation, denoted as SEG101, and it verifies that the predictor-based NAS algorithm NPENAS-NP (Wei et al., 2022) is a promising search strategy for semantic segmentation.
- In this paper, we compare different zero-cost proxies used to rank semantic segmentation architectures in SEG101 and demonstrate that zero-cost proxies exhibit different properties in terms of semantic segmentation compared to image classification.

2 Related Works

Recent studies have applied NAS for semantic segmetnation and remote sensing scene classification. Auto-DeepLab (Liu et al., 2019a) and DCNAS (Zhang et al., 2021) use a search space consisting of L different searchable layers, each containing four feature maps with different resolutions. Both Auto-DeepLab and DCNAS employ gradient-based search strategies to explore the search space. HR-NAS (Ding et al., 2021) designs a search space that contains architectures like Transformer (Vaswani et al., 2017) and also adopts gradient-based search strategy to find the optimal lightweight architecture. SceneNet (Ma et al., 2021) adopts a multi-objective neural evolution strategy for remote scene classification. RSNet (Wang et al., 2021) and Peng *et al.* (Peng et al., 2021) adopt search spaces like Auto-DeepLab and DARTS (Liu et al., 2019b), respectively, and utilize gradient-based strategies to solve the remote scene classification problem. Unlike previous work, we use RegNet (Radosavovic et al., 2020) as backbone and design a new search space that focuses on how to efficiently enhance and fusion the feature maps of the backbone network. The architecture in this search space is used for remote sensing image segmentation, and we adopt the predictor-based search strategy to explore the search space.

Abdelfattah *et al.* (Abdelfattah et al., 2021) studied the performance of zero-cost proxies in ranking neural architectures in NASBench-101 (Ying et al., 2019), NASBench-201 (Dong and Yang, 2020), and NASBench-ASR (Mehrotra et al., 2021), and experimentally demonstrated the superior performance of zero-cost proxies. TENAS (Chen et al., 2021) proposes to use the trainability and expressivity of neural networks as an estimation of the image classification architectures' performance. In contrast to the aforementioned studies, this paper applies zero-cost proxies to semantic segmentation and attempts to find out whether zero-cost proxies can accurately rank remote sensing image segmentation architectures.

3 Methodology

To investigate the characteristic of zero-cost proxies for remote sensing image segmentation, a new search space is designed and introduced in Section 3.1. Zero-cost proxies for remote sensing image segmentation are outlined in Section 3.2.

3.1 Search Space for Remote Sensing Image Segmentation

This section discusses the proposed search space SEG101, which is designed from the perspective of how to enhance the feature maps' contextual information and how to conduct effective multi-scale

feature fusion. The SEG101 contains two modules - the feature maps enhancement module and multi-head feature fusion module. An overview of SEG101 is shown in Fig. 1.

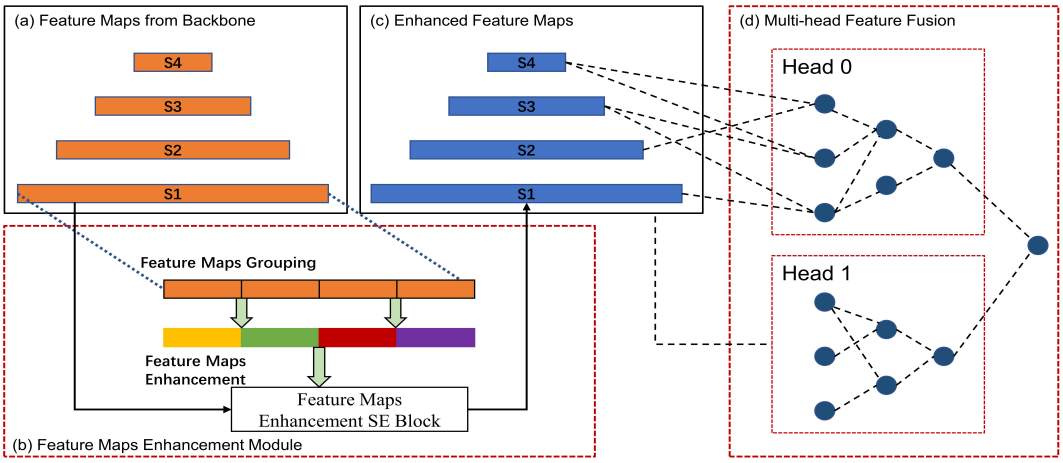


Figure 1: Overview of SEG101.

3.1.1 Feature Maps Enhancement Module. In this paper, we select RegNet (400MFX) as the backbone, which contains four different stages, and the feature maps generated by these four stages (S1, S2, S3, S4) are shown in Fig. 1(a). We divide the four feature maps into two different categories according to their spatial size, and the feature maps S1 and S2 belong to the first category (denoted as category A), while S3 and S4 belong to the second category (denoted as category B).

As shown in Fig. 1(b), taking feature maps S1 as an example, the feature maps enhancement module divieds S1 into four different groups evenly by channel, and then select four different operations to enhance the features of these four groups, respectively. The candidate operations for category A are 3×3 conv , 3×3 conv with dilation 3, and adaptive pooling layer. Since the feature maps in category B have a smaller spatial size, the candidate operations for category B are 3×3 conv with dilation 3, adaptive pooling layer, self-attention block, and SE block (Hu et al., 2020). The above operations are used to enhance the spatial contextual information of the feature maps. After the spatial enhancement, a feature maps enhancement SE block is used to improve the spatially enhanced feature maps' global information. The feature maps enhancement SE block is discussed in Supplementary Materials (Section A).

3.1.2 Multi-head Feature Fusion Module. The enhanced feature maps are illustrated in Fig. 1(c). In this paper, a multi-head feature fusion module is proposed to maintain the diversity of multi-scale feature fusion, as shown in Figure 1(d). This module employs two heads and both heads have the same structure. Each head contains three layers, and the first layer contains three nodes, and each node of the first layer can fusion any two enhanced feature maps. The nodes of the second layer of each head can fusion the outputs of any two nodes of the first layer in the same head. The outputs of the two heads are fused and resized to the size of input image, and the generated feature maps are regarded as the prediction of the segmentation network. There are totally $C_3^1 \times C_3^1 \times C_4^1 \times C_4^1 \times 5^4 \times C_6^3 \times C_3^2 \times C_6^3 \times C_3^2 \times 2 = 6.48 \times 10^8$ architectures in SEG101.

3.2 Zero-cost Proxied for Remote Sensing Image Segmentation

We select the six zero-cost proxies *grad_norm*, *snip* (Lee et al., 2019), *grasp* (Wang et al., 2020a), *synflow* (Tanaka et al., 2020), *fisher*, and *jacob_cov* (Mellor et al., 2021) discussed in Abdelfattah *et al.* (Abdelfattah et al., 2021) to analyze. For space reasons, we omit the description of these six

proxies, and detailed information about these six zero-cost proxies can be found in Abdelfattah *et al.* (Abdelfattah et al., 2021).

4 Experiments and Analysis

All the experiments in this paper are finished by using PyTorch (Paszke et al., 2019). We use the implementation of zero-cost proxies from Abdelfattah *et al.* (Abdelfattah et al., 2021) (Apache-2.0 LICENSE), and employ the predictor-based NAS algorithm NPENAS (Wei et al., 2022) (MIT LICENSE) to explore the search space SEG101.

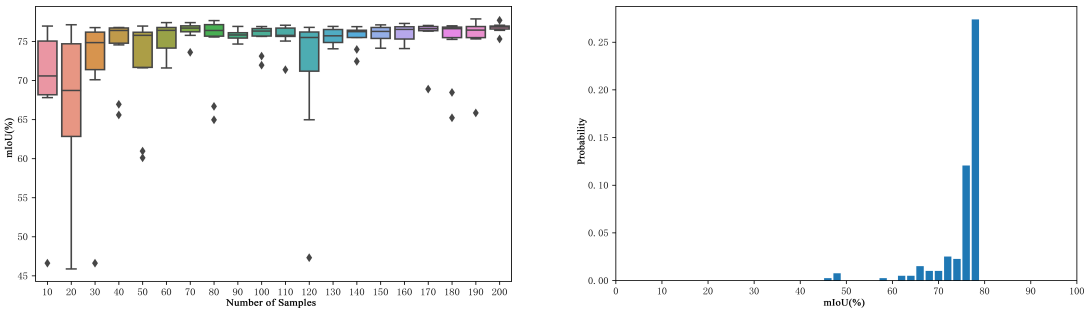
4.1 Datasets

The experiments are performed by using the Massachusetts road dataset (Mnih, 2013) and the WHU building dataset (Ji et al., 2018). The Massachusetts road dataset is a benchmark for road segmentation that contains 1171 images, and the images of this dataset are seamlessly cropped to 512×512 small images, and 8076, 224 and 784 images are selected from the cropped images for training, validation and testing, respectively. The WHU building dataset is a benchmark for building extraction. Similar to the road dataset, each image is seamlessly cropped to a 512×512 image, and the total number of cropped images is 8188, of which the training dataset, the validation dataset, and the testing dataset are 4736, 1036 and 2416, respectively.

4.2 Neural Architecture Search Analysis

In this section, the NPENAS-NP (Wei et al., 2022) algorithm is employed to find the optimal architecture from SEG101 for Massachusetts road dataset. The search budget of NPENAS-NP is 200, and the other training details of NPENAS-NP are directly adopted from the original implementation. Each searched neural architecture is trained on the training dataset for 20 epochs. Two NVIDIA 2080Ti GPUs are used to perform the search experiment, which costs 23.26 GPU days to complete.

The analysis of searched architectures by NPENAS-NP is illustrated in Fig. 2, and it verifies that the predictor-based search strategy NPENAS-NP is suitable for searching architectures applied to remote sensing image segmentation. Fig. 2(a) shows that as the search proceeds, NPENAS-NP is significantly more likely to find architecture with high performance. Fig. 2(b) confirms that most of the architectures searched by NPENAS-NP have high performance. The above experiment results also indicate that SEG101 is a promising search space for remote sensing image segmentation.



(a) Boxplot of the top ten architectures in each iteration.

(b) The mIoU histogram of the 200 searched architectures.

Figure 2: The analysis of searched architectures by NPENAS-NP.

The top-5 searched architectures are selected and fully trained for 100 epochs. The architecture with the highest validation performance is selected from the top-5 architectures and compared with other segmentation algorithms. As illustrated in Table 1, the best architecture searched by

NPENAS-NP performs better than many recently designed segmentation algorithms for remote sensing image and achieves comparable performance with DeepLab V3 plus (Chen et al., 2018b).

Table 1: Performance comparison of different segmentation algorithms on Massachusetts test dataset.

Algorithms	mIoU(%)	IoU Target(%)	FWIoU(%)	PA(%)	MPA(%)
U-Net (Ronneberger et al., 2015b)	62.34	–	93.05	96.16	66.56
FCN (Shelhamer et al., 2017)	75.71	–	95.50	97.54	81.49
SiU-Net (Ji et al., 2019)	75.33	–	94.51	96.64	**90.74**
HCN (Li et al., 2019)	77.36	–	95.78	97.69	83.55
ConDinet++ (Yang et al., 2021)	78.88	–	–	–	–
PSPNet (Zhao et al., 2017)	79.43	61.11	96.03	97.83	85.8
DeepLab V3 + (Chen et al., 2018b)	**79.71**	**61.63**	96.09	97.87	85.9
Ours	79.55	61.27	**96.1**	**97.89**	85.08

The WHU building dataset is used to test the generalization ability of the searched architecture, and the results are illustrated in Supplementary Materials.

4.3 Zero-cost Proxies Analysis

The search strategy NPENAS-NP is adopted to collect 498 neural architectures from SEG101 and the rank performance of the six zero-cost proxies is evaluated on these architectures. The comparison results are illustrated in Table 2. The high negative correlation of *grad_norm*, *snip*, and *fisher* can be utilized by search strategies to explore the search space.

Table 2: Spearman correlation coefficient of zero-cost proxies on SEG101.

	grad_norm	snip	grasp	fisher	synflow	jacob_cov
SEG101	-0.45	-0.46	0.07	-0.48	0.05	0.03

5 Limitations and Broader Impact Statement

Since this paper presents a preliminary study of the zero-cost proxies for remote sensing image segmentation, it contains many limitations, which are listed below.

(1) Since the architectures used to evaluate the rank performance of zero-cost proxies are collected by only one search strategy, the collected architectures lack diversity and may not be sufficient to reflect the properties of the search space SEG101, and the results shown in Table 2 may only be applicable for the predictor-based strategy strategies. This limitation can be eliminated by utilizing the method proposed in Surrogate NAS Benchmarks (Zela et al., 2022).

(2) Since the searched architecture could not achieve the SOTA performance, this reflects that there exists design shortages in SEG101. Design deficiencies may hinder the search strategies to find architectures that can achieve SOTA performance.

Due to the promising performance of the proposed SEG101, it can be used by other search strategies to find segmentation architectures for other datasets. The results in Section 4.3 illustrate that *grad_norm*, *snip*, and *fisher* can be combined with the predictor-based search strategy to search segmentation architectures, which can significantly reduce search costs.

6 Acknowledgements

This work was supported by the National Natural Science Foundation of China under Grant Numbers 61976167 and U19B2030.

7 Reproducibility Checklist

1. For all authors...

   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes] [See Section 3 and Section 4.]

   (b) Did you describe the limitations of your work? [Yes] [See Section 7.]

   (c) Did you discuss any potential negative societal impacts of your work? [Yes] [See Section 7.]

   (d) Have you read the ethics author's and review guidelines and ensured that your paper conforms to them? `https://automl.cc/ethics-accessibility/` [Yes] [See Section 7.]

2. If you are including theoretical results...

   (a) Did you state the full set of assumptions of all theoretical results? [N/A] [We do not include theoretical results.].

   (b) Did you include complete proofs of all theoretical results? [N/A] [We do not include theoretical results.].

3. If you ran experiments...

   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results, including all requirements (e.g., `requirements.txt` with explicit version), an instructive `README` with installation, and execution commands (either in the supplemental material or as a URL)? [No] [We will make these materials publicly available after all experiments are completed.]

   (b) Did you include the raw results of running the given instructions on the given code and data? [No] [We will include these materials after all experiments are completed.

   (c) Did you include scripts and commands that can be used to generate the figures and tables in your paper based on the raw results of the code, data, and instructions given? [No] [The scripts and commands will be include in the published code.]

   (d) Did you ensure sufficient code quality such that your code can be safely executed and the code is properly documented? [No] [This will be present in the code we release.]

   (e) Did you specify all the training details (e.g., data splits, pre-processing, search spaces, fixed hyperparameter settings, and how they were chosen)? [Yes] [See Section 4.]

   (f) Did you ensure that you compared different methods (including your own) exactly on the same benchmarks, including the same datasets, search space, code for training and hyperparameters for that code? [Yes] [See Section 4.]

   (g) Did you run ablation studies to assess the impact of different components of your approach? [Yes] [See Section 4.]

   (h) Did you use the same evaluation protocol for the methods being compared? [Yes] [See Section 4.]

   (i) Did you compare performance over time? [Yes] [See Section 4.]

   (j) Did you perform multiple runs of your experiments and report random seeds? [No] [Due to the limitation of computational resources, we did not run experiments multiple times.]

   (k) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [N/A] [We did not running experiments multiple times.]

(l) Did you use tabular or surrogate benchmarks for in-depth evaluations? [No] [Tabular or surrogate benchmarks for semantic segmentation do not exist.]

(m) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] [See Section 4.]

(n) Did you report how you tuned hyperparameters, and what time and resources this required (if they were not automatically tuned by your AutoML method, e.g. in a NAS approach; and also hyperparameters of your own method)? [No] [Hyperparameters are used directly from previous work.]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...

(a) If your work uses existing assets, did you cite the creators? [Yes] [See Section 4.]

(b) Did you mention the license of the assets? [Yes] [See Section 4.]

(c) Did you include any new assets either in the supplemental material or as a URL? [N/A] [We do not contain new assets in the supplemental materials.]

(d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A] [Our experiments were conducted on publicly available datasets.]

(e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A] [The dataset does not contains personally identifiable information or offensive content.]

5. If you used crowdsourcing or conducted research with human subjects...

(a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A] [We do not use crowdsourcing or conducted research with human subjects.].

(b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A] [We do not use crowdsourcing or conducted research with human subjects.].

(c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A] [We do not use crowdsourcing or conducted research with human subjects.].

References

Acknowledgements.

Abdelfattah, M. S., Mehrotra, A., Dudziak, L., and Lane, N. D. (2021). Zero-cost proxies for lightweight NAS. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Broni-Bediako, C., Murata, Y., Mormille, L. H. B., and Atsumi, M. (2022). Searching for cnn architectures for remote sensing scene classification. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–13.

Chen, L., Zhu, Y., Papandreou, G., Schroff, F., and Adam, H. (2018a). Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part VII*, volume 11211, pages 833–851, Berlin. Springer.

Chen, L.-C., Zhu, Y., Papandreou, G., Schroff, F., and Adam, H. (2018b). Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Computer Vision – ECCV 2018*, pages 833–851, Cham. Springer International Publishing.

Chen, W., Gong, X., and Wang, Z. (2021). Neural architecture search on imagenet in four GPU hours: A theoretically inspired perspective. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., and Schiele, B. (2016). The cityscapes dataset for semantic urban scene understanding. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3213–3223.

Ding, M., Lian, X., Yang, L., Wang, P., Jin, X., Lu, Z., and Luo, P. (2021). HR-NAS: Searching efficient high-resolution neural architectures with lightweight transformers. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2981–2991.

Dong, X. and Yang, Y. (2020). Nas-bench-201: Extending the scope of reproducible neural architecture search. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*.

Elsken, T., Metzen, J. H., and Hutter, F. (2018). Neural architecture search: A survey. *Journal of Machine Learning research*, 20:55:1–55:21.

Hu, J., Shen, L., Albanie, S., Sun, G., and Wu, E. (2020). Squeeze-and-excitation networks. *IEEE Trans. Pattern Anal. Mach. Intell.*, 42(8):2011–2023.

Ji, S., Wei, S., and Lu, M. (2018). Fully convolutional networks for multisource building extraction from an open aerial and satellite imagery data set. *IEEE Transactions on Geoscience and Remote Sensing*, 57(1):574–586.

Ji, S., Wei, S., and Lu, M. (2019). Fully convolutional networks for multisource building extraction from an open aerial and satellite imagery data set. *IEEE Transactions on Geoscience and Remote Sensing*, 57(1):574–586.

Klyuchnikov, N., Trofimov, I., Artemova, E., Salnikov, M., Fedorov, M. V., and Burnaev, E. (2020). Nas-bench-nlp: Neural architecture search benchmark for natural language processing. *CoRR*, abs/2006.07116.

Lee, N., Ajanthan, T., and Torr, P. H. S. (2019). Snip: single-shot network pruning based on connection sensitivity. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Li, J., Liu, X., Zhang, S., Yang, M., Xu, R., and Qin, F. (2021). *Accelerating Neural Architecture Search for Natural Language Processing with Knowledge Distillation and Earth Mover's Distance*, page 2091–2095. Association for Computing Machinery, New York, NY, USA.

Li, Y., Guo, L., Rao, J., Xu, L., and Jin, S. (2019). Road segmentation based on hybrid convolutional network for high-resolution visible remote sensing image. *IEEE Geoscience and Remote Sensing Letters*, 16(4):613–617.

Lin, J., Jing, W., Song, H., and Chen, G. (2019). Esfnet: Efficient network for building extraction from high-resolution aerial images. *IEEE Access*, 7:54285–54294.

Liu, C., Chen, L.-C., Schroff, F., Adam, H., Hua, W., Yuille, A. L., and Fei-Fei, L. (2019a). Auto-deeplab: Hierarchical neural architecture search for semantic image segmentation. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 82–92.

Liu, H., Simonyan, K., and Yang, Y. (2019b). DARTS: differentiable architecture search. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Ma, A., Wan, Y., Zhong, Y., Wang, J., and Zhang, L. (2021). Scenenet: Remote sensing scene classification deep learning network using multi-objective neural evolution architecture search. *ISPRS Journal of Photogrammetry and Remote Sensing*, 172:171–188.

Mehrotra, A., Ramos, A. G. C. P., Bhattacharya, S., Dudziak, Ł., Vipperla, R., Chau, T., Abdelfattah, M. S., Ishtiaq, S., and Lane, N. D. (2021). NAS-Bench-ASR: Reproducible neural architecture search for speech recognition. In *International Conference on Learning Representations (ICLR)*.

Mellor, J., Turner, J., Storkey, A. J., and Crowley, E. J. (2021). Neural architecture search without training. In Meila, M. and Zhang, T., editors, *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 7588–7598. PMLR.

Mnih, V. (2013). Machine learning for aerial image labeling. In *Ph.D. dissertation,Dept. Comput. Sci., Univ. Toronto, Toronto, ON, Canada*.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Köpf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S. (2019). Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, 8-14 December 2019, Vancouver, BC, Canada*, pages 8024–8035.

Peng, C., Li, Y., Jiao, L., and Shang, R. (2021). Efficient convolutional neural architecture search for remote sensing image scene classification. *IEEE Transactions on Geoscience and Remote Sensing*, 59(7):6092–6105.

Radosavovic, I., Kosaraju, R. P., Girshick, R. B., He, K., and Dollár, P. (2020). Designing network design spaces. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 10425–10433. Computer Vision Foundation / IEEE.

Ronneberger, O., Fischer, P., and Brox, T. (2015a). U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*.

Ronneberger, O., Fischer, P., and Brox, T. (2015b). U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention - MICCAI 2015 - 18th International Conference Munich, Germany, October 5 - 9, 2015, Proceedings, Part III*, volume 9351 of *Lecture Notes in Computer Science*, pages 234–241, Berlin. Springer.

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

Tanaka, H., Kunin, D., Yamins, D. L., and Ganguli, S. (2020). Pruning neural networks without any data by iteratively conserving synaptic flow. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H., editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Tang, Y., Yang, D., Li, W., Roth, H. R., Landman, B. A., Xu, D., Nath, V., and Hatamizadeh, A. (2021). Self-supervised pre-training of swin transformers for 3d medical image analysis. *ArXiv*, abs/2111.14791.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. (2017). Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 6000–6010, Red Hook, NY, USA. Curran Associates Inc.

Wang, C., Zhang, G., and Grosse, R. B. (2020a). Picking winning tickets before training by preserving gradient flow. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Wang, J., Zhong, Y., Zheng, Z., Ma, A., and Zhang, L. (2021). Rsnet: The search for remote sensing deep neural networks in recognition tasks. *IEEE Transactions on Geoscience and Remote Sensing*, 59(3):2520–2534.

Wang, N., Gao, Y., Chen, H., Wang, P., Tian, Z., Shen, C., and Zhang, Y. (2020b). Nas-fcos: Fast neural architecture search for object detection. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11940–11948.

Wei, C., Niu, C., Tang, Y., Wang, Y., Hu, H., and Liang, J. (2022). Npenas: Neural predictor guided evolution for neural architecture search. *IEEE Transactions on Neural Networks and Learning Systems*, pages 1–15.

White, C., Neiswanger, W., and Savani, Y. (2021). Bananas: Bayesian optimization with neural architectures for neural architecture search. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Wu, G., Shao, X., Guo, Z., Chen, Q., Yuan, W., Shi, X., Xu, Y., and Shibasaki, R. (2018). Automatic building segmentation of aerial imagery using multi-constraint fully convolutional networks. *Remote Sensing*, 10(3):407.

Xiong, Y., Liu, H., Gupta, S., Akin, B., Bender, G., Wang, Y., Kindermans, P.-J., Tan, M., Singh, V., and Chen, B. (2021). Mobiledets: Searching for object detection architectures for mobile accelerators. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3824–3833.

Yang, K., Yi, J., Chen, A., Liu, J., and Chen, W. (2021). Condinet++: Full-scale fusion network based on conditional dilated convolution to extract roads from remote sensing images. *IEEE Geoscience and Remote Sensing Letters*, pages 1–5.

Ying, C., Klein, A., Christiansen, E., Real, E., Murphy, K., and Hutter, F. (2019). NAS-bench-101: Towards reproducible neural architecture search. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97, pages 7105–7114.

Zela, A., Siems, J. N., Zimmer, L., Lukasik, J., Keuper, M., and Hutter, F. (2022). Surrogate NAS benchmarks: Going beyond the limited search spaces of tabular NAS benchmarks. In *International Conference on Learning Representations*.

Zhang, X., Xu, H., Mo, H., Tan, J., Yang, C., Wang, L., and Ren, W. (2021). Dcnas: Densely connected neural architecture search for semantic image segmentation. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13951–13962.

Zhao, H., Shi, J., Qi, X., Wang, X., and Jia, J. (2017). Pyramid scene parsing network. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 6230–6239, Los Alamitos, CA. IEEE Computer Society.

A Feature Maps Enhancement SE Block

The feature maps enhancement SE block is shown in Fig. 3. Different from the standard SE block, the feature maps enhancement SE block utilizes the channel information of the original feature maps to improve the channel-wise information of the enhanced feature maps.

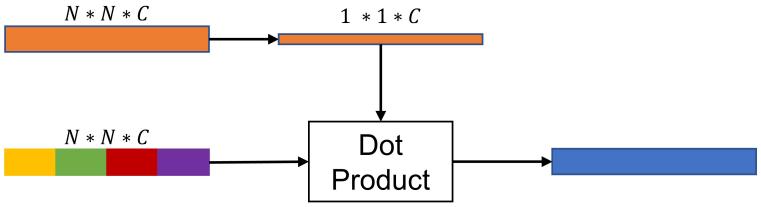


Figure 3: Feature maps enhancement SE block.

B Performance Comparison on WHU Building Dataset

The performance of searched architecture compares with other algorithms is shown in Table 3. Although the architecture is searched on the Massachusetts dataset, it achieves a performance comparable to DeepLab V3 plus, which demonstrates the good generalization ability of the searched architecture.

Table 3: Performance comparison of different segmentation algorithms on WHU test dataset.

Algorithms	mIoU(%)	IoU Target(%)	FWIoU(%)	PA(%)	MPA(%)
U-Net (Ronneberger et al., 2015b)	–	86.8	–	–	–
SiU-Net (Ji et al., 2019)	–	88.4	–	–	–
ESFNet (Lin et al., 2019)	–	85.34	–	–	–
CU-Net (Wu et al., 2018)	–	87.1	–	–	–
PSPNet (Zhao et al., 2017)	93.84	89.1	97.52	98.72	96.59
DeepLab V3 + (Chen et al., 2018a)	**94.32**	**89.96**	**97.72**	**98.83**	**96.94**
Ours	93.5	88.51	97.38	98.65	96.47

C Visualization of Segmentation Results

The Visualization of segmentation results by different algorithms on the Massachusetts road dataset and the WHU building dataset is illustrated in Fig. 4 and Fig. 5, respectively.

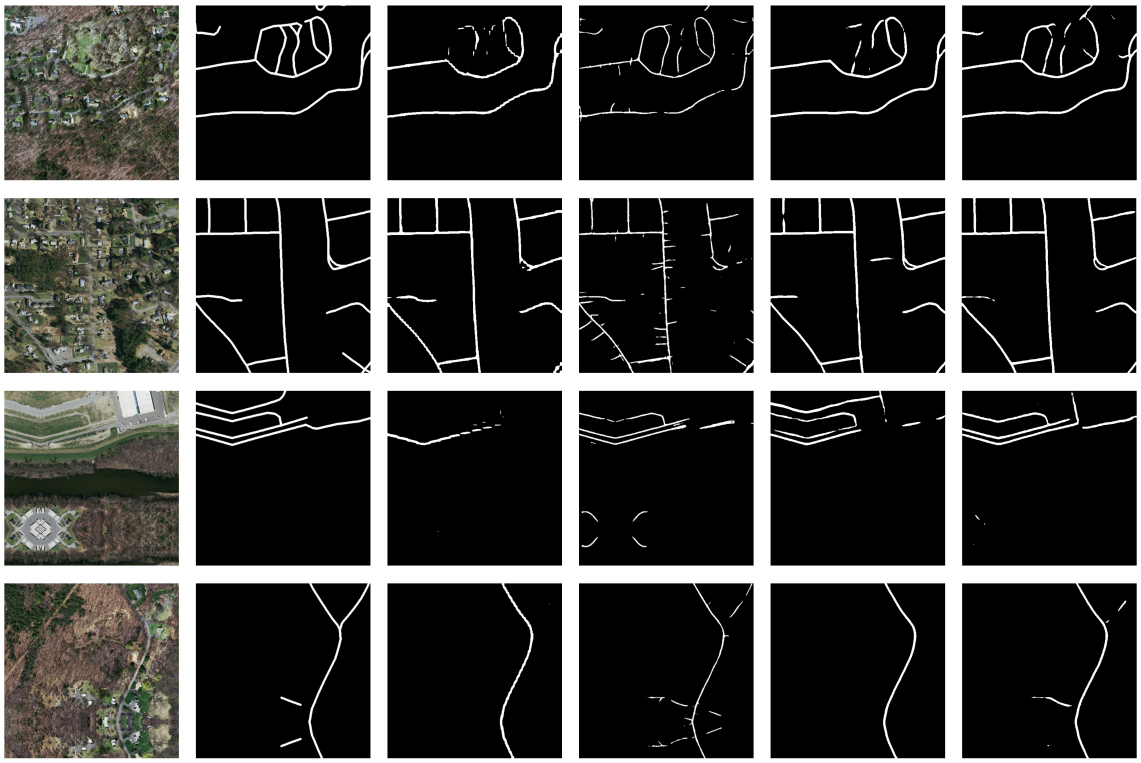


Figure 4: Visualization of segmentation results on the Massachusetts road dataset.

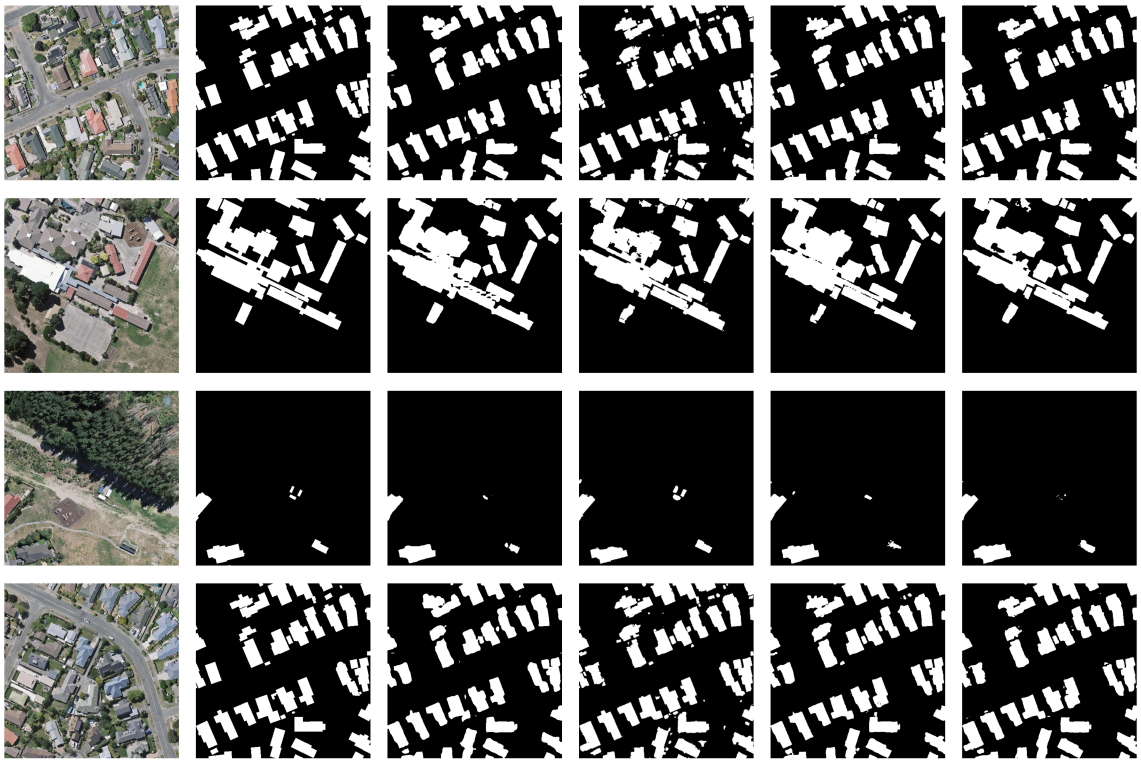


Figure 5: Visualization of segmentation results on the WHU building dataset.