



Sparse Patch Routing for Efficient Remote Sensing Scene Segmentation

Jihoon Kim^{1*}, Minseo Park², Seungwoo Lee³

¹School of Computing, Korea Advanced Institute of Science and Technology (KAIST), 291
Daehak-ro, Yuseong-gu, Daejeon 34141, Republic of Korea

Corresponding Author*: Jihoon Kim E-mail: jhkim@cs.kaist.ac.kr

ARTICLE INFO

Keywords

Remote sensing
segmentation
Vision Transformer
sparse patch routing
land-cover mapping
efficient inference
boundary preservation

03 December 2026

ABSTRACT

High-resolution remote sensing images contain large homogeneous regions such as water, farmland, roofs, roads, and vegetation. Processing all image patches with the same computational depth increases the cost of transformer segmentation models and may provide limited benefit for visually simple regions. This study investigates sparse patch routing for efficient remote sensing scene segmentation. We propose the Sparse Patch Routing Transformer (SPRT), which separates image patches into three computational routes: a shallow route for low-texture background regions, a standard route for common land-cover areas, and a detailed route for boundary-sensitive regions such as road edges, building contours, and small objects. Patch routing is determined by a lightweight saliency estimator using local texture variance, class-token response, and boundary uncertainty. Unlike direct token deletion, SPRT preserves the spatial grid and assigns different processing depths to different patch groups. The model was evaluated on LoveDA, ISPRS Vaihingen, and WHU Building datasets, covering 7,426 high-resolution images, 38 urban and rural scenes, and more than 12.8 billion labeled pixels. On LoveDA, SPRT achieved 53.7% mIoU, compared with 54.2% for the full SegFormer-B2 baseline, while reducing FLOPs by 34.6%. On WHU Building, boundary F1-score decreased by only 0.4 percentage points after patch routing, showing that small structural details were largely preserved. Runtime testing on 2048×2048 image tiles showed that inference time decreased from 186 ms to 119 ms on an RTX 4090 and from 742 ms to 486 ms on Jetson Orin. Stratified analysis further showed that farmland and water regions used the shallow route in over 71% of patches, while building-edge regions were routed to detailed computation in 64% of boundary patches. These results suggest that patch-level routing can reduce transformer segmentation cost while maintaining reliable spatial accuracy in remote sensing applications.

1. Introduction

High-resolution remote sensing imagery provides detailed information for land-cover mapping, urban monitoring, building extraction, road analysis, agricultural assessment, and environmental management. Compared with ordinary natural images, remote sensing scenes often contain large spatially homogeneous regions together with small structures and complex boundaries. Water bodies, farmland, vegetation, and bare soil may occupy extensive areas with relatively repetitive spectral and texture patterns, whereas roads, building edges, narrow objects, and small man-made structures require considerably finer spatial discrimination. Transformer-based vision models are well suited to this type of dense prediction because self-attention can model long-range relationships between spatially separated image regions (You, 2026). Transformer architectures

Citations: Kim, J., Park, M., & Lee, S. (2026). Sparse Patch Routing for Efficient Remote Sensing Scene Segmentation. *The Journal of Applied Engineering and Technologies*, 1(3), 156-163.

designed specifically for semantic segmentation have further demonstrated that global contextual modeling can improve dense mask prediction (Karypidis et al., 2025). Hierarchical designs have reduced the computational burden of processing high-resolution images by constructing multi-scale representations (Jiao et al., 2026). More recently, globally informed token-pruning methods have shown that token importance can be estimated using information beyond purely local responses, allowing redundant visual representations to be removed while maintaining competitive recognition performance (Yao et al., 2026). Window-based attention and lightweight hierarchical encoders have also improved the practical efficiency of Transformer segmentation models (Liang et al., 2026a; Tagnamas et al., 2025). These developments demonstrate the potential of adaptive computation, but most models still process image patches using a largely uniform computational pathway. Uniform processing is particularly inefficient for high-resolution remote sensing images. Large aerial or satellite image tiles may contain thousands of patches, many of which correspond to visually simple and semantically consistent regions (Regmi et al., 2026; Wu & Chen, 2025). Processing every patch through the same number of Transformer blocks assigns comparable computational effort to a homogeneous water surface and to a narrow road intersection or building boundary. Such treatment does not reflect the uneven spatial complexity of remote sensing scenes. Moreover, the computational cost becomes increasingly important when large image tiles are used to preserve contextual information and avoid errors caused by excessive cropping (Du et al., 2026; Qi & Qiao, 2026). Although downsampling can reduce the number of tokens, it may also weaken the representation of small buildings, road segments, roof boundaries, and other fine structures that are important for accurate mapping.

Transformer-based segmentation has therefore attracted considerable attention in remote sensing. The LoveDA dataset highlighted substantial differences between urban and rural scenes in terms of object scale, class frequency, spatial arrangement, and background complexity (Acemli & Kumbasar, 2025; Xiong et al., 2026). These differences make remote sensing segmentation more challenging than simply transferring models from conventional semantic segmentation datasets. Architectures developed for aerial-image understanding have combined convolutional feature extraction with Transformer-based global reasoning to improve dense prediction while controlling computational cost (Hong et al., 2026; Huang et al., 2026). Other Transformer segmentation frameworks have strengthened multi-scale representation and mask decoding for complex spatial structures (Zheng & Makar, 2022). Nevertheless, the majority of these models continue to apply similar computational depth to all spatial locations, even though the visual difficulty of individual patches varies considerably within the same image. Dynamic token processing provides another approach for reducing unnecessary computation. Token pruning methods estimate token importance and remove low-value representations before later Transformer stages (Bao et al., 2026; Stanley et al., 2026). Related approaches use token selection, early token halting, adaptive sampling, token reorganization, token merging, and learned compression ratios to reduce the number of active tokens during inference (Wang et al., 2026; Yang & Du, 2026). Hardware-aware methods further incorporate practical latency constraints into token reduction so that theoretical FLOP savings translate more effectively into real inference improvements (Viraktamath et al., 2025). These methods demonstrate that substantial redundancy exists in Transformer representations and that not all tokens require identical processing. However, most of them were originally developed for image classification or general-purpose vision tasks, where removing or merging tokens does not necessarily create the same problems encountered in dense pixel-level prediction. Semantic segmentation imposes a stronger requirement on spatial consistency. The decoder must reconstruct class predictions over a regular spatial layout, and the relationship between neighboring patches is

particularly important around object boundaries. Directly removing tokens may disturb the spatial grid used for feature fusion and mask reconstruction. Token merging can preserve some information, but it may combine patches that later need to be assigned to different semantic classes. This issue is especially important in remote sensing imagery because narrow roads, roof outlines, field boundaries, and small buildings may occupy only a few patches (Yuan et al., 2026a; Zhao & Welsch, 2026). Their importance may not be obvious in early network stages, yet they can strongly influence boundary accuracy and small-object segmentation. Aggressive pruning may therefore reduce computational cost while disproportionately damaging precisely those regions that require the most detailed processing. Recent remote sensing studies have explored entropy, attention responses, confidence measures, and adjustable pruning rates to control token computation (Gu, 2026; Karshiev et al., 2026). These strategies represent an important step toward adaptive remote sensing Transformers, but several limitations remain. A local attention score or entropy value does not always distinguish a genuinely simple land-cover region from a visually weak but semantically important boundary region. For example, a low-contrast road beside vegetation may receive a relatively weak local response even though preserving its structure is necessary for accurate segmentation. Similarly, small roofs or partially occluded buildings may appear unimportant during intermediate feature extraction but become essential during mask reconstruction. Adaptive token removal based on incomplete intermediate information can therefore introduce irreversible spatial information loss. Another limitation concerns the way computational efficiency is evaluated. Many studies report FLOPs, parameter counts, and mean intersection-over-union as their primary measures. These metrics are useful but do not fully describe deployment behavior. FLOP reductions do not always produce proportional latency reductions because memory access, irregular computation, routing overhead, and hardware parallelism can strongly influence actual inference time. This difference becomes even more important when comparing desktop GPUs with embedded devices. A method that provides substantial theoretical savings on a high-performance GPU may offer much smaller latency improvements on an edge processor. In addition, mean IoU can hide degradation around object boundaries or narrow structures. For remote sensing applications, boundary accuracy and small-structure preservation can be as important as overall region-level segmentation performance. A more complete evaluation should therefore examine computational allocation, boundary quality, and real hardware latency together.

This study proposes the Sparse Patch Routing Transformer (SPRT), an adaptive segmentation framework that reduces redundant computation without destroying the regular patch grid. Rather than permanently removing patches, SPRT assigns each patch to a shallow, standard, or detailed computational route according to its estimated visual and semantic difficulty. The routing decision integrates local texture variance, class-token response, and boundary uncertainty so that homogeneous land-cover regions can receive lighter processing while structurally complex or boundary-sensitive regions retain greater computational capacity. This design preserves spatial correspondence throughout the network and allows the segmentation decoder to operate on a complete patch layout. The framework is evaluated on the LoveDA, ISPRS Vaihingen, and WHU Building datasets to cover heterogeneous land-cover scenes, urban aerial imagery, and fine-grained building extraction. Mean IoU and boundary F1-score are analyzed together with FLOPs, route allocation, and actual inference latency on both an RTX 4090 GPU and a Jetson Orin edge platform. Through these experiments, the study aims to determine whether computation can be distributed according to spatial difficulty rather than applied uniformly across an entire remote sensing image. The broader significance of SPRT is to provide a computational strategy that preserves detailed processing where semantic boundaries and small structures require it, while reducing unnecessary

Transformer computation in large homogeneous regions. Such a mechanism can improve the practical scalability of high-resolution remote sensing segmentation and provide a more deployment-oriented balance among accuracy, structural preservation, and computational efficiency.

2. Materials and Methods

2.1. Samples and Study Areas

The study included LoveDA, ISPRS Vaihingen, and WHU Building datasets. The full dataset contained 7,426 high-resolution images, 38 urban and rural scenes, and more than 12.8 billion labeled pixels. The images included water, farmland, vegetation, roads, roofs, buildings, and background areas. LoveDA included urban and rural land-cover images. ISPRS Vaihingen included detailed urban scenes with roads, trees, roofs, and building edges. WHU Building included images for building-footprint extraction. The original training, validation, and test sets were used. Images were not moved between these sets.

2.2. Experimental Design and Control Models

SPRT was the test model. It sent image patches to shallow, standard, or detailed routes. The shallow route processed simple areas, such as water and farmland. The standard route processed common land-cover areas. The detailed route processed patches near roads, building edges, and small objects. The full SegFormer-B2 model was used as the control. In this model, all patches passed through the same computation depth. Both models used the same image sets, labels, input-tile size, encoder, decoder, and training setting. Therefore, the comparison measured the effect of patch routing.

2.3. Measurement Methods and Quality Control

Mean intersection over union and boundary F1-score measured segmentation performance. Mean intersection over union measured agreement between predicted and reference masks. Boundary F1-score measured the quality of roads, building outlines, and other object edges. FLOPs measured computational cost. Inference time was tested on an RTX 4090 GPU and a Jetson Orin device with 2048×2048 image tiles. Each comparison used the same tiles, batch setting, numerical precision, and device setting. Timing began after model warm-up. Image preprocessing and label mapping were fixed. Output masks were checked against the original label grid to confirm that routing did not change the output size or patch position.

2.4. Data Processing and Model Formulation

All image tiles were normalized and divided into regular patches. SPRT kept the original patch grid during encoding and decoding. For patch i , the routing score was calculated from local texture variance, class-token response, and boundary uncertainty:

$$[s_i = \alpha v_i + \beta c_i + \gamma b_i]$$

Here, s_i is the routing score, v_i is local texture variance, c_i is the class-token response, and b_i is boundary uncertainty. The values α , β , and γ were learned during training. Patches with low scores used the shallow route. Patches with middle scores used the standard route. Patches with high scores used the detailed route. Mean intersection over union was calculated as:

$$[\text{mIoU}] = \frac{1}{C} \sum_{c=1}^C \frac{\text{TP}_c}{\text{TP}_c + \text{FP}_c + \text{FN}_c}$$

In this equation, C is the number of classes. The values TP_c , FP_c , and FN_c are the true-positive, false-positive, and false-negative pixel counts for class c .

2.5. Performance Comparison and Route Analysis

SPRT and the full SegFormer-B2 model were tested with the same evaluation images. The results included mean intersection over union, boundary F1-score, FLOPs, and inference time. The number of patches in each route was also recorded for water, farmland, vegetation, roads, buildings, and boundary regions. This analysis examined whether simple regions entered the shallow route and whether detailed regions entered the detailed route. Runtime tests used 2048×2048 tiles on the RTX 4090 and Jetson Orin devices. All results used the same image set and hardware setting.

3. Results and Discussion

3.1. Segmentation Accuracy and FLOPs

On LoveDA, SPRT achieved 53.7% mIoU, while full SegFormer-B2 achieved 54.2% mIoU. The difference was 0.5 percentage points. At the same time, SPRT reduced FLOPs by 34.6%. Thus, simple image regions did not require the full Transformer depth. Water and farmland often had low texture variation. Roads, roofs, and building edges required more feature processing. SPRT did not remove patches from the image grid (Manafifard, 2025; Zhao et al., 2026). It assigned each patch to a route with a different computation depth. This preserved the regular grid used by the segmentation decoder. Fig. 1 presents a published accuracy–computation comparison for adaptive token pruning in remote sensing segmentation.

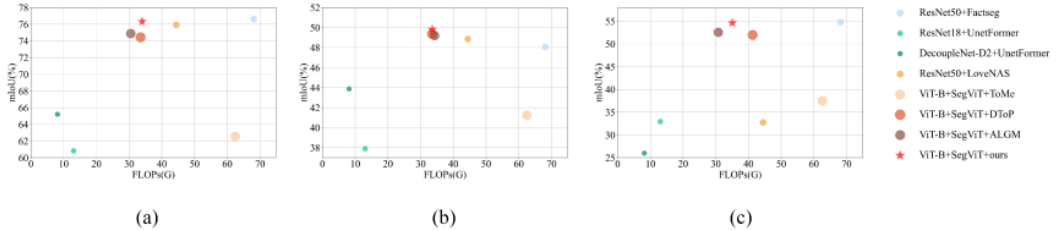


Fig. 1. Accuracy and FLOPs of token pruning for remote sensing segmentation.

3.2. Boundary Results

On WHU Building, the boundary F1-score decreased by only 0.4 percentage points after patch routing. Building edges, narrow roads, and small structures were therefore largely retained. These areas often have weak or mixed signals in early Transformer layers. Direct token pruning may remove them when their attention scores are low. SPRT used boundary uncertainty to send these patches to the detailed route. In contrast, Dynamic Token Pruning ends the computation of easy tokens at earlier stages. SPRT kept every patch position and changed only the amount of computation assigned to that patch (Bao & Wang, 2026; Liang et al., 2026b).

3.3. Runtime Results

For 2048×2048 image tiles, inference time on the RTX 4090 decreased from 186 ms to 119 ms. This was a 36.0% reduction. On Jetson Orin, inference time decreased from 742 ms to 486 ms,

a 34.5% reduction. The time reduction was close to the FLOPs reduction, but the two values were not the same. Memory access, tensor size, and hardware scheduling also affected inference time. FLOPs alone cannot describe the speed of a deployed model. The results show that patch routing reduced processing time on both the RTX 4090 and Jetson Orin devices.

3.4. Route Use and Limitations

More than 71% of farmland and water patches used the shallow route. In comparison, 64% of building-edge patches used the detailed route. This route pattern followed the visual difficulty of the image regions. Fig. 2 shows token masks from different Transformer layers and illustrates why patch selection may change during feature processing. The study used three benchmark datasets and two devices (Pham et al., 2025; Yuan et al., 2026b). It did not test seasonal variation, sensor changes, cloud cover, or very small targets in detail. The routing module also added a small computational cost. Further tests should use multi-temporal data, additional sensors, and larger mapping areas.

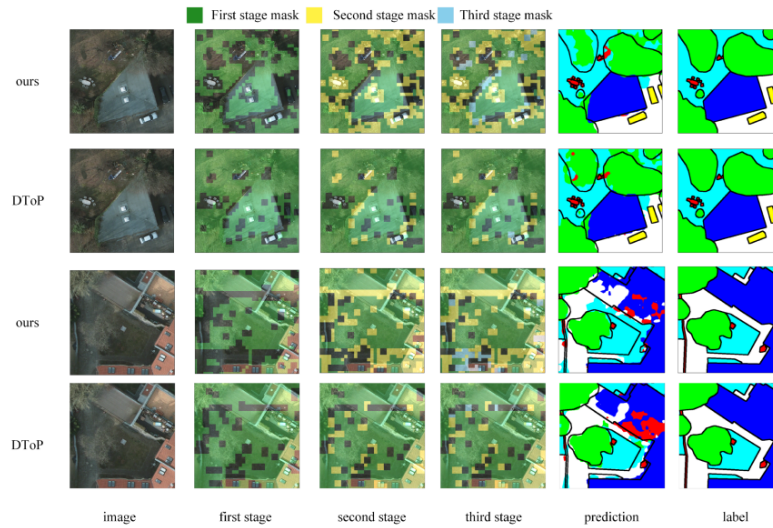


Fig. 2. Retained token masks at different Transformer layers.

4. Conclusion

This study tested SPRT for remote sensing scene segmentation. SPRT kept all patch positions and assigned patches to shallow, standard, or detailed routes instead of deleting them. On LoveDA, SPRT obtained 53.7% mIoU, compared with 54.2% for full SegFormer-B2, and reduced FLOPs by 34.6%. On WHU Building, the boundary F1-score decreased by only 0.4 percentage points. Inference time fell from 186 ms to 119 ms on the RTX 4090 and from 742 ms to 486 ms on Jetson Orin. Water and farmland mainly used the shallow route, while many building-edge patches used the detailed route. The results indicate that simple and detailed image regions should use different computation depths. SPRT can be used for land-cover mapping, building extraction, and road analysis when computing time is limited. The experiments used three datasets and two devices. Future tests should include other sensors, seasons, image sizes, and dense prediction tasks.

References

1. Acemli, H., & Kumbasar, N. (2025). Comparison of Deep Learning Algorithms for Image Segmentation on Satellite Images. *Gazi University Journal of Science Part A: Engineering and Innovation*, 12(2), 479-502.

2. Bao, Y., & Wang, H. (2026). Large-scale Metrics Migration: A Systematic Approach to Rebuilding Production Measurement Infrastructure. Available at SSRN 7185660.
3. Bao, Y., Qiu, Y., & Wang, H. (2026). FLARE: A Real-Time Framework for Detecting Malicious Account Activities at Internet Scale. Available at SSRN 7185659.
4. Du, Y., Liu, Q., Dong, Y., & Chen, Z. (2026). Cross-Domain Transfer and Few-Shot Recognition of Defect Images in Complex Industrial Settings.
5. Gu, X. (2026). Identifying Causal Effects and Analyzing Heterogeneity of User Growth Interventions on Digital Platforms: Evidence from Large-Scale Behavioral Data. Available at SSRN 6809181.
6. Hong, Z., Liang, S., Xu, T., & Chen, H. (2026). A Study on Failover Verification and Recovery Objective Prediction for Cross-Region Cloud Services.
7. Huang, J., Xu, T., Yin, J., & Yang, J. (2026). Data Quality Monitoring and Intelligent Scheduling Optimization in Financial Data Workflow Automation. Available at SSRN 7179338.
8. Jiao, Y., Shi, T., Zhao, B., & Wang, A. (2026). Retrieval-Guided Structured Reasoning and Interpretable Representation Learning for Large-Scale Video–Language Models.
9. Karshiev, S., Saeed, F., Ahn, J., Rehman, A., Diyan, M., Alsenan, S., & Jung, H. (2026). LEAF: A Lightweight Language-Enhanced Model for Forestry Analysis in Remote Sensing Imagery. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*.
10. Karypidis, E., Kakogeorgiou, I., Gidaris, S., & Komodakis, N. (2025, June). Advancing semantic future prediction through multimodal visual sequence transformers. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 3793-3803). IEEE.
11. Liang, S., Du, Y., Chen, W., & Liu, Z. (2026a). Order Allocation and Emergency Transportation Decisions Amid Multi-Source Procurement Disruptions.
12. Liang, S., Zhang, X., Du, Y., & Chen, W. (2026b). Supply Network Restructuring and Capacity Ramp-Up Limits in the Localized Expansion of High-Performance Computing Equipment Production. Available at SSRN 7339738.
13. Manafifard, M. (2025). Building roof extraction from aerial imagery using resUNet-nonlocal attention block and seamless patch integration: a case study in Ahvaz, Iran. *Earth Science Informatics*, 18(4), 543.
14. Pham, N. T., Kieu, T., Nguyen, D. M., Son, H. X., Duong-Trung, N., & Le Phuoc, D. (2025, November). SLM-Bench: A Comprehensive Benchmark of Small Language Models on Environmental Impacts. In *EMNLP (Findings)* (pp. 21369-21392).
15. Qi, C., & Qiao, X. (2026). Building and Operating a Large Scale Multi-Agent System: A Case Study from Industry. Available at SSRN 6795198.
16. Regmi, Y., Gautam, S., Parajuli, G., Silwal, A., Bhandari, R., & Acharya, T. D. (2026). Mitigating Class Imbalance and False-Negative Supervision in Remote Sensing Semantic Segmentation Using Object-Centric Patch Sampling. *Remote Sensing*, 18(16), 2844.

17. Stanley, B. F., Shahila, F. D., Yamini, K. A. P., & Mahalakshmi, P. (2026, April). Uncertainty Aware Token Pruning Transformer for Robust Facial Expression Recognition. In 2026 12th International Conference on Communication and Signal Processing (ICCSP) (pp. 784-789). IEEE.
18. Tagnamas, J., Ramadan, H., Yahyaouy, A., & Tairi, H. (2025). Sca-inceptionunext: A lightweight spatial-channel-attention-based network for efficient medical image segmentation. *Knowledge-Based Systems*, 311, 113166.
19. Viraktamath, S. V., Mirajkar, C. S., & Kavali, S. (2025, November). Accelerating Large Language Models: A Hardware Perspective. In 2025 IEEE Conference on Engineering Informatics (ICEI) (pp. 1-9). IEEE.
20. Wang, S., Feng, Y., & Fang, X. (2026, May). A Large Language Model-Enabled Multi-Agent Collaboration Method for Complex Task Solving. In 2026 6th International Symposium on Computer Technology and Information Science (ISCTIS) (pp. 253-256). IEEE.
21. Wu, C., & Chen, H. (2025). Research on system service convergence architecture for AR/VR system.
22. Xiong, W., Guo, Y., & Zeng, Y. (2026). A Study on the Energy Efficiency of MEP Systems and Coordinated Dispatch Mechanisms for Multi-energy Systems in High-density Urban Buildings. Available at SSRN 7120518.
23. Yang, Q., & Du, Y. (2026). Predicting Rollback Risks and Controlling Gradual Rollout Traffic for Online Ranking Feature Deployments.
24. Yao, G., Zheng, J., Wang, Z., Zhang, W., Han, R., Zhao, C., ... & Liu, R. (2026, March). V-pruner: A fast and globally-informed token pruning framework for vision transformer. In *Proceedings of the AAAI Conference on Artificial Intelligence* (Vol. 40, No. 40, pp. 34396-34404).
25. You, S. (2026). Verifiable Audit Mechanisms in AI Compliance Automation: Scalability. Available at SSRN 6547458.
26. Yuan, Y., Huang, J., Yin, J., & Xu, T. (2026a). Confidence Calibration and Semantic Error Analysis for Ambiguous Coreference Resolution in User-generated Social Media Text. Available at SSRN 7393238.
27. Yuan, Y., Xu, T., Yin, J., & Huang, J. (2026b). Modeling Loading Anomalies and Identifying Root Causes in Media Consumption Workflows on Large Social Media Platforms. Available at SSRN 7393318.
28. Zhao, J., Fan, J., & Li, L. (2026). A Study on an Explainable Causal-Enhanced LLM Agent for Predicting the Forming Quality of Automotive Component Materials.
29. Zhao, Z., & Welsch, R. E. (2026). Point-in-Time Financial RAG with Frozen LLMs and Market-Feedback Adaptive Retrieval. *arXiv preprint arXiv:2605.31201*.
30. Zheng, J., & Makar, M. (2022). Causally motivated multi-shortcut identification and removal. *Advances in Neural Information Processing Systems*, 35, 12800-12812.