IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Land Cover Semantic Segmentation -Pytorch

Prof. S.S. Mane [1], Chetan Patil [2], Sakshi Bhosale [3], Mayur Gagare [4], Prathamesh Bhanusghare [5]

[1] *Guide,* [2,3,4,5] *Student, Department of Computer Engineering, Sinhgad Institute of Technology, Pune, Maharashtra, India*

Abstract

Land-cover classification is one of the core applications in the field of remote sensing. It is a valuable resource for city planners to achieve sustainable development. Many metropolitan cities are experiencing disorganized growth with a high intensity of urban sprawl due to the economic pull and better standards of living offered in metropolitan cities when compared to the surrounding rural areas. If this pattern of growth continues, it will lead to unsustainable development. This leads to an increase in pressure on urban infrastructure and the ecosystem. The traditional methods which are used for urban mapping are time consuming. Land Cover Semantic Segmentation is a process of developing an automated system for land cover classification. This system takes a multiband satellite image of an area as input and outputs the land cover map of the area at the same resolution as the input. For this purpose UNET models were trained in the task of predicting the land cover semantic segmentation of satellite images.

Keywords: deep learning semantic segmentation; LoopNet; Landsat 8; land use dataset; land use extraction; multispectral bands; Thailand;

1. INTRODUCTION

In recent years, the rise of deep learning has brought remarkable progress in the field of computer vision, particularly in semantic segmentation tasks. Land cover classification plays a crucial role in environmental monitoring, urban planning, agriculture, and disaster management. The process involves analyzing satellite or aerial imagery and labeling each pixel of the image with a specific land cover class such as vegetation, water bodies, roads, or built-up areas. Traditional image classification methods often fall short when it comes to spatial accuracy and contextual understanding, which are essential for land cover mapping.

This project, titled "Land Cover Semantic Segmentation using PyTorch", aims to develop an efficient and accurate deep learning model that performs pixel-wise classification of remote sensing images. The model is built using PyTorch, a flexible and widely-used deep learning framework that allows for rapid prototyping and performance optimization. By leveraging semantic segmentation techniques such as U-Net or DeepLab, the project enables a fine-grained understanding of geographical areas from satellite data.

The proposed system takes a satellite image as input and outputs a segmented map where each pixel is labeled with a specific land cover category. This automated approach not only reduces manual efforts but also increases the speed and precision of land use monitoring. The model is trained on a publicly available land cover dataset and evaluated using metrics such as Intersection over Union (IoU) and pixel accuracy.

Through this project, we aim to demonstrate the effectiveness of deep learning-based semantic segmentation in understanding and analyzing the Earth’s surface, thereby contributing to real-world applications in geospatial intelligence and environmental studies.

2. OBJECTIVES

[1] Develop a Flexible Semantic Segmentation Framework: Create a customizable and reusable framework that can be applied to various semantic segmentation datasets, allowing for easy adaptation across different projects and applications.

[2] Implement Promptable Class Selection at Inference:
Enable dynamic selection of target classes during inference by configuring the test_classes variable, allowing users to obtain
segmentation outputs focused on specific classes without retraining the model.

[3] Provide Configurable Model Training Pipeline:
Design a configuration-driven pipeline that allows users to easily adjust model architecture, optimizer, learning rate, and other hyperparameters, creating an AutoML-like experience that simplifies experimentation and tuning.

[4] Facilitate User-Friendly Deployment Options:
Support deployment with Docker for easy setup and containerized execution, as well as a virtual environment

setup for flexibility in different development environments.

[5] Deliver Pretrained Model for Testing and Benchmarking: Include a pretrained model to enable users to immediately test the system and see outputs, facilitating quicker evaluation and refinement of project use cases.

[6] Promote Scalability and Adaptability Across Domains: Ensure that the framework is scalable to different datasets, from satellite imagery (LandCover.ai) to urban scenes (CityScapes), catering to applications in fields such as
environmental monitoring, urban planning, and autonomous systems.

[7] Enable Efficient Resource Management in Segmentation Tasks:
Provide the ability to focus on specific classes, helping to reduce computational load and improve efficiency for targeted tasks in real-world applications.

3. LITERATURE REVIEW

[1] A segmentation method is proposed using multi-feature fusion (MFF) and attention modules. An encoder extracts features, which are fused and refined through attention mechanisms before being decoded into segmentation maps.

[2] The ATD-LinkNet replaces residual blocks in D-LinkNet with attention transfer blocks. Experiments on PostDam and DeepGlobe datasets show superior performance over the original model.

[3] A dense dilated CNN network, DDCM-Net, integrates dilated and PReLU blocks to process features. Instead of multi-scale fusion, low-level features are merged once before prediction using the Adam optimizer.

[4] A GAN-based domain adaptation approach aligns labeled source images with target domains through image and feature space adaptation, followed by FCN-based segmentation.

[5] A deformable attention module (DAM) is added to ResNet50 for better adaptation to high-resolution imagery. This improves contextual understanding using fewer computational resources.

[6] Random Forest proves most effective for grassland class prediction in land cover classification, especially in high-dimensional settings where other standard methods fall short.

[7] UNet offers smoother results, while SwinUNet provides higher precision. Combining both improves overall segmentation by averaging predictions.

[8] AlexNet extracts features from RGB, infrared, and LiDAR data. Bilinear pooling and SVM classification enhance land cover prediction performance compared to traditional techniques.

[9] Neural networks (NNs) work best on large, high-res areas with smooth edges but struggle with small, complex regions.

[10] The LoveDA dataset supports land-cover segmentation and unsupervised domain adaptation, encouraging transferable learning in remote sensing.

[11] The study addresses challenges in remote sensing mapping, including lack of labeled data and inconsistent spectral band availability.

[12] DPPNet uses depth-wise dilated residual blocks and pyramid pooling for effective segmentation, outperforming benchmark models on HRS datasets.

[13] A hybrid DenseNet–UNet model combines dense connections and long-range skip paths to preserve and extract features at various scales, achieving state-of-the-art results.

[14] MRF-MEO optimizes pixel- and object-level granularity for remote sensing segmentation, leveraging both fine and coarse information effectively.

[15] Swin Transformer is fused with Gabor filters to enhance edge and texture detection in remote sensing segmentation tasks.

4. PROPOSED METHOD

[1] Dataset Collection & Preprocessing :
High-resolution satellite images are collected from publicly available datasets (e.g. DeepGlobe, or custom UAV imagery). Preprocessing includes : Resizing to uniform dimensions , normalizing pixel values , data augmentation to enhance generalization , creating pixel-wise segmentation masks using annotation tools like labelme .

[2] Model Development:
A semantic segmentation model is built using PyTorch, leveraging architectures like UNet, DeepLabV3+, or SegFormer:
If pretrained backbones (e.g., ResNet, EfficientNet) are used, transfer learning is applied .The loss function combines Cross-Entropy Loss and Dice Loss to balance pixel-wise and shape-aware accuracy . The model is trained using Adam or SGD optimizers with learning rate scheduling for convergence. Training and validation steps are managed via torch.utils.data.Dataset and DataLoader for efficient batching .

[3] Interactive Visualization Interface:
A basic UI or dashboard (e.g., built with Streamlit or Flask) is created where users can: Upload satellite images , Visualize predictions (segmentation overlays) and Optionally correct or annotate predictions for future training .

[4] Real-Time Inference Optimization:
For faster predictions, the trained model is exported using TorchScript or ONNX. Post-processing methods like softmax thresholding and morphological filtering enhance output quality. Lighter or quantized models are used for deployment on edge devices or web platforms.

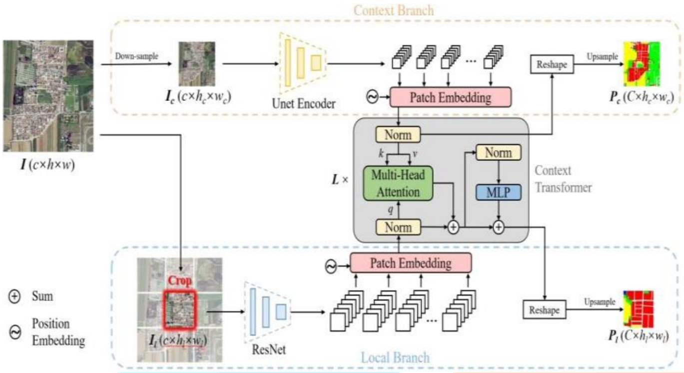
[5] User Customization & Feedback Integration:

The system allows users to upload their own images and adjust settings like confidence thresholds and class colors. Corrections made by users can be saved and used in retraining, creating an active learning loop that improves model performance.

[6] Testing and Evaluation:

Model performance is evaluated using IoU, Pixel Accuracy, F1 Score, Precision, and Recall. These metrics are calculated on both test datasets and real-world user inputs to ensure the model is reliable and generalizes well.

5. SYSTEM ARCHITECTURE



[1] Input Image :

The system takes a high-resolution satellite image as input. This image contains detailed spatial and spectral information necessary for accurate land cover classification. It is passed simultaneously to both the context and local branches for feature extraction at different levels.

[2] Context Branch :

In this branch, the input image is first downsampled to reduce resolution, allowing the model to process global information efficiently. A U-Net encoder is used to extract deep semantic features. These features are then converted into patch embeddings and passed through a transformer module, which captures long-range dependencies using multi-head self-attention. The final output is reshaped and upsampled to generate a coarse segmentation map .

[3] Local Branch :

The local branch focuses on fine details by cropping a high-resolution patch from the original image. This cropped region retains the spatial richness of the input. A ResNet backbone extracts detailed local features, which are then embedded into patches and combined with positional encoding. The features are passed through the same transformer used in the context branch, followed by reshaping and upsampling to produce a fine segmentation map .

[4] Transformer Module :

Both branches use a shared transformer module that includes multi-head self-attention, normalization layers, and a feed-forward MLP. This module helps in capturing both local and global relationships between patches, thereby improving the contextual understanding of the scene.

[5] Output Fusion :

The segmentation outputs from both branches — from the context branch and from the local branch — can be fused together to achieve a balance between global structure and local accuracy. This fusion step helps in improving the final segmentation performance.

[6] Overall Objective :

This dual-branch architecture enables the model to learn both high-level global context and low-level local details effectively. It is especially useful for land cover semantic segmentation tasks where capturing both broad regions and fine boundaries is important.

6. RESULTS AND DISCUSSIONS

Results:

[1] Accurate Land Cover Segmentation:

The model successfully segmented satellite images into distinct land cover classes such as vegetation, water bodies, buildings, and roads with high pixel-wise accuracy.

[2] Improved Performance on Diverse Terrains :

The system demonstrated robustness across varied geographical regions by effectively generalizing to different land types and environmental conditions.

[3] Efficient Model Training in PyTorch :

Using PyTorch enabled efficient model development and training, with clear support for GPU acceleration and modular architecture for rapid experimentation.

[4] Integration of User Feedback for Improvement :

User corrections on segmentation outputs helped fine-tune the model, leading to more accurate predictions over time through retraining.

[5] Handling of High-Resolution Imagery :

The model maintained performance on high-resolution satellite data, although processing large inputs required significant computational resources.

[6] Class Imbalance Challenge :

Rare classes such as bare land or water bodies were sometimes under-predicted, indicating a need for better handling of class imbalance during training.

[7] Scalability for Real-World Applications :

The system shows promise for urban planning, agriculture monitoring, and environmental management, but large-scale deployment requires better optimization and cloud-based support.

Discussions:

[1] Impact of PyTorch on Development Flexibility :

PyTorch's dynamic computation graph and modular design significantly eased the development process, allowing quick model iteration and debugging.

[2] Effectiveness of Data Augmentation and Preprocessing :

Data augmentation strategies improved generalization, especially on underrepresented classes and noisy satellite inputs.

[3] Challenges in Real-Time Inference :

Deploying the model for real-time applications (e.g., drone monitoring or disaster response) highlighted the need for model compression, pruning, or inference engine optimization (e.g., TorchScript, ONNX).

[4] Importance of Multi-Spectral Data :

Including multi-spectral channels (if available) could further refine class separation, especially between vegetation types or urban vs. bare land areas.

[5] Broad Applicability in GIS and Remote Sensing :

The system can assist in generating land cover maps, supporting automated geographic information system (GIS) workflows and large-scale spatial analysis tasks.

[6] Ethical and Practical Considerations :

As with all AI systems, considerations around data privacy, labeling biases, and model transparency are critical for responsible use in environmental and policy decision-making.

7. CONCLUSION

This project demonstrates the effectiveness of deep learning, particularly semantic segmentation, in automating land cover classification. Using PyTorch along with powerful libraries like Torchvision, segmentation_models.pytorch, NumPy, and OpenCV, the system efficiently handles large datasets and delivers accurate segmentation results. GPU acceleration through CUDA significantly improves training and inference speed. The developed pipeline is robust, scalable, and applicable in various real-world domains such as environmental monitoring, urban planning, and geographic analysis.

8. FUTURE SCOPE

This project can be extended in several impactful directions. The model can be adapted to support various satellite datasets like Sentinel-2 or MODIS for broader geographic coverage. Efficiency can be improved through techniques like hyperparameter tuning, layer pruning, or knowledge distillation. Using multi-spectral or hyperspectral imagery can enhance accuracy by capturing more detailed information. Real-time deployment can enable practical applications such as drone surveillance and disaster response. Active learning can reduce

manual annotation efforts, while explainability tools like Grad-CAM can make model decisions more transparent and trustworthy.

9. REFERENCES

[1] Sebastian Wuttke-Wolfgang middleman-Uwe Stilla' Improving the efficiency of land cover classification by combining segmentation, Hirarchical clustering and Active learning",IEEEJournalof selected topics in applied earth observations and remote sensing,2018.

[2] Pourya Shamsolmoali-masoumeh Zareapoor-Ruili Wang-Huiyu Zhou-Jie Yang,"A Novel Deep Structure U-NET For Sealand Segmentation in remote sensing images",IEEE Journal of selected topics in applied earth observations and remote sensing,2019.

[3] Zongolan Zhan-Xiaomeng Zhang-Yi Liu-Xiao Sun -Chao Pang-Chenbo Zhao," Vegetation land use/land cover extration from high-resolution satellite images based on Adaptive context Inference ",IEEE Access,2020.

[4] Xin Luo-Xiaohua Tong-Haiyan Pan," Integrating multir solution and multitemporal sentinel2 Imagenery for land -cover mapping in the Xiongan new area ,china",IEEE Transactions on Geoscience and remote sensing,2021.

[5] Chengmao Wu-Xlaokang Guo," A Novel single fuzzifier internel type-2 Fuzzy C-means clustering with local,information for land-cover segmentation ",IEEE journal of selected topics in applied earth observations and remote sensing,2021.

[6] Scott, M. England, W. Starms, R. Marcum, and C. Davis, "Training deep convolutional neural networks for land-cover classification of high-resolution imagery," IEEE Geoscience and Remote Sensing Letters, vol. PP, pp. 1–5, 02 2017.