Road Segmentation from Satellite Imagery

Ahmed Abdelmalek, Néstor Lomba Lomba, Khaled Miraoui
Department of Computer Science, EPFL, Switzerland

***Abstract*—This paper presents a pipeline for road segmentation from satellite imagery using deep learning techniques. State-of-the-art semantic segmentation models such as ResNet, DeepLabv3 and RFE-LinkNet are employed for binary segmentation of the images. The dataset, obtained from Google Maps and annotated for the EPFL ML Road Segmentation Challenge, is preprocessed into patches to ensure efficient training. The models are trained and evaluated using the Dice coefficient and F1 score, which measure the overlap between predicted masks and ground truth annotations. Our results demonstrate the strengths and weaknesses of each model, highlighting RFE-LinkNet as the most balanced approach in terms of computational cost and segmentation accuracy. Our best submission to AI Crowd is ID #278400, with F1 Score of 0.918.**

I. Introduction

Road segmentation from satellite images is a critical task in many applications such as remote sensing and urban planning, and has been a hot research topic for the last decade. Our goal is to classify an image $I \in \mathbb{R}^{H \times W \times C}$, where H is the height, W is the width, and C is the number of channels, into a binary mask $M \in \{0,1\}^{H \times W}$. Here, $M_{ij} = 1$ indicates the presence of a road pixel, while $M_{ij} = 0$ represents the background.

Given the imbalanced nature of the data, where road pixels are sparse relative to the background, traditional methods often fail to generalize effectively. To address this, we explore advanced deep learning-based semantic segmentation techniques. In this project, we:

- Preprocess and augment the dataset to improve model generalization.
- Evaluate three popular deep learning models: **RFE-LinkNet [1]**, **ResNet [2]**, and **DeepLabv3+ [3]**.
- Employ loss functions specially suited for this endeavor such as the the Dice loss in combination with Binary Cross Entropy (BCE) to address class imbalance.

Our contributions are:

1) A systematic comparison of tree segmentation models tailored for image segmentation.
2) A robust pipeline for preprocessing, training, and evaluating segmentation models.
3) Insights into the strengths and limitations of each approach based on empirical results.

II. Models and Methods

In this section, we describe the key components of our segmentation pipeline, the architectures used, and the evaluation metrics.

Dataset Preprocessing

The success of semantic segmentation tasks is critically dependent on the quality, consistency, and preprocessing of the training and testing datasets. In this project, we used multiple datasets, with particular focus on the EPFL ML Road Segmentation Dataset provided by the teaching staff. This dataset presents unique characteristics that significantly influenced our preprocessing pipeline.

Understanding the EPFL ML Road Segmentation Dataset

The EPFL ML Road Segmentation Dataset consists of:

- **Training Images**: 100 satellite images of size 400×400, each with corresponding ground truth binary masks.
- **Testing Images**: 50 satellite images of size 608×608, provided without ground truth annotations for evaluation purposes.

An important observation is that the scale between the training and testing images remains consistent, where 1 pixel ≈ 26 cm in both sets. However, the testing images are larger, and can be seen as trimmed expansions of the training images. This consistency in scale implies that resizing the images (e.g., scaling all images to 608×608) would disrupt their original pixel-level resolution, leading to mismatched scales between the datasets. Consequently, we avoided resizing altogether and adapted our preprocessing methods to preserve the spatial resolution.

Dataset Augmentation

To enhance training diversity and improve model generalization, we applied **Data Augmentation** to the training data to increase dataset variability. That includes rotations and horizontal/vertical flips.

Incorporating Additional Datasets

To increase the diversity and size of the training set, we integrated two additional datasets:

- **LSNYR Dataset**: High-resolution satellite images sourced from the paper "Reconstruction Bias U-Net for Road Extraction From Optical Remote Sensing Images" [4].
- **Chicago Dataset**: Urban satellite imagery obtained from the paper "Learning Aerial Image Segmentation from Online Maps" [5].

Experimental Configurations

To evaluate the impact of dataset selection, we experimented with three configurations:

1) Using only the EPFL-provided training images and their ground truth masks.
2) Combining the EPFL dataset with the LSNYR dataset.
3) Merging the EPFL dataset with the Chicago dataset.

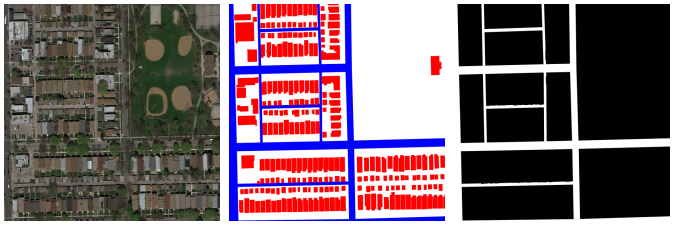
Data preprocessing Pipeline

The primary goal of pre-processing the **Chicago** dataset was to maintain its fidelity, ensuring it closely resembled the original provided dataset. We highlight in this section all the steps used to engineer this data.

1) **Classification Incompatibility:** The masks of the **Chicago** dataset don't match our requirements. Indeed, 3 labels are assigned in the masks : *road, house, and background*. For this matter, the first step was to turn the masks to a binary label format : $M \in \{0,1,2\} \implies M \in \{0,1\}$
2) **Mismatching sizes:** Not all the sizes in the model were consistent. Nonetheless, for a more stable segmentation model, the training images and labels should be of same

size. Thus the second step was to resize the images from $R^{n \times m} \implies R^{608 \times 608}$

3) **Scale incompatibility:** The yielded images from the 2 previous steps had a much bigger scale per pixel. To ensure the model efficiency, the last step was to take a patch of size according to a 2.5 zoom, and resizing it back to 608*608.



(a) Image (b) Label (c) groundtruth

Fig. 1: Resized image, original labels and groundtruth mask

The same transformations are consistently applied to the images to ensure that the mapping between pixels and their corresponding labels remains accurate.

RFE-Linknet Training and Testing Configuration

For training the RFE-Linknet model, we exclusively used the augmented dataset derived from the EPFL-provided dataset. We augmented the training images and ground truth masks by applying rotations at 8 angles (from $0°$ to $315°$ in $45°$ increments) and vertical and horizontal flips.

This augmentation pipeline resulted in a substantial increase in the effective size of the training dataset, allowing the model to learn diverse spatial patterns and reduce overfitting.

For training, we trimmed the images to 384×384 pixels so the size is divisible by 32.

The provided test images were larger, with a resolution of 608×608. To adapt these images for the 384×384 input size of the trained model, we employed a patching strategy. Each test image was divided into 4 overlapping patches of size 384×384.

Finally, the model predictions on individual patches were reassembled into a full 608×608 image, ensuring consistency with the test image dimensions. This approach allowed us to leverage the model's training resolution while effectively handling larger test images.

Model Architectures

ResNet

The ResNet architecture employed in this work uses a residual network to extract features from satellite imagery. It allows the network to learn the residual mapping $H(x) = F(x) + x$, where x is the input and $F(x)$ is the learned function.

1) The ResNet block can be expressed as:

$$y = \sigma(W_2 \cdot \sigma(W_1 \cdot x + b_1) + b_2) + x,$$

where W_1 and W_2 are weight matrices, b_1 and b_2 are biases, σ is the ReLU, and x is the input feature map. This formulation enables efficient gradient flow during backpropagation, allowing the network to learn deeper representations without degradation in performance.

2) The architecture is divided into two main stages:
   a) Feature Extraction with ResNet Backbone: The encoder extracts features from the input image using a pre-trained ResNet model.
   b) Mask Reconstruction with Decoder: The decoder progressively upsamples the encoded features to reconstruct the segmentation mask. This ensures that the outputs align with the spatial resolution of the input image.
3) **Loss Function:** To handle the inherent class imbalance in road segmentation, we use a combination of Dice loss and Binary Cross-Entropy (BCE) loss:

$$\mathcal{L}_{\text{total}} = \alpha \cdot \mathcal{L}_{\text{Dice}} + (1 - \alpha) \cdot \mathcal{L}_{\text{BCE}},$$

where $\mathcal{L}_{\text{Dice}}$ minimizes the overlap error between the predicted and ground truth masks, and $\mathcal{L}_{\text{BCE}}$ ensures pixel-wise classification accuracy.
4) **Hyperparameters and Training:**
   a) Input size: 512×512 patches extracted from original images (608×608).
   b) Batch size: 16.
   c) Learning rate: 1×10^{-4} with a learning rate scheduler reducing it to 1×10^{-7}.
   d) Optimizer: Adam with $\beta_1 = 0.9$, $\beta_2 = 0.999$.
   e) Epochs: 50.

DeepLabv3+

DeepLabv3+ is a state of the art model of deep-learning, ultimately designed for semantic segmentation tasks. The model builds upon its predecessor **DeepLabv3**, by introducing an architecture based on a *decoder* and *encoder* structure : the encoder captures rich contextual features using Atrous Spatial Pyramid Pooling and atrous convolutions, enabling multi-scale feature extraction without compromising resolution.The decoder then refines these features to produce detailed segmentation maps with sharp boundaries. This architecture excels in handling the complexities of real-world environments, making it ideal for our road segmentation project.

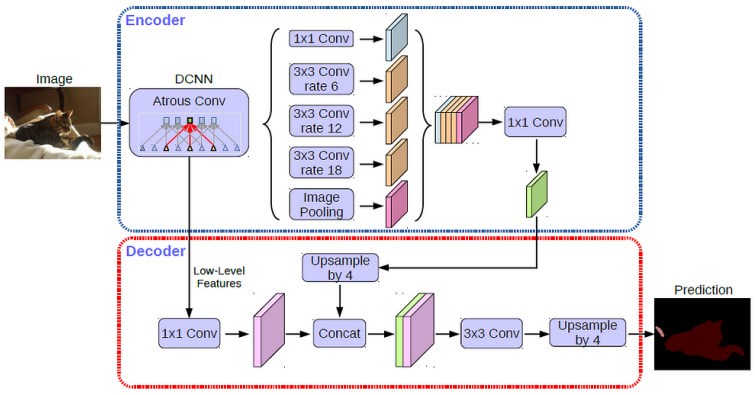


Fig. 2: Detailed architecture of DeeplabV3+

Like shown in the above figure Deeplabv3+, we briefly yet concisely explain the architecutre of the model :

1) **The encoder :** it is DCNN (Deep Convolutional Neural Network) : its goal is to extract detailed features from the input image, highlighting the following mechanisms.
   - **Atrous convolution :** It is the mechanism allowing the model to explicitly control the resolution of features, enabling multi-scale data collection. The key paramter here is r, the *atrous rate*, determining the kernel

size.For a two-dimensional, here is the mathematical formula : $y[i] = \sum_k x[i + r \cdot k]w[k]$

- **Image Concatenation** : after the parallel convolutions and the image pooling, concatenates all the outputs into a single feature representation.
- **Post ASPP** : a final 1*1 convolution to reduce the number of channels and prepare a single output for the **decoder**

2) **The decoder** : the decoder's primary job is to restore spatial resolution and ensure accurate an accurate segmentation map and alignment with the original image.

- **Low feature processing** : these features keep high details, and are extracted in the early layers of the encoder. This process is done via a 1*1 convolution that essentially selects and combines the most relevant fine details.
- **Concatenation** : Combines the low and high level features, this combination enriches the decoder with context and details, allowing the model to integrate detailed information with contextual understanding
- **segmentation mapping** : this last step is done first by a 3*3 convolution to smooth out the output and refine the segmentation boundaries, and then by upsampling by 4 the output so it matches the the spatial resolution of the input.

Concerning hyperparameters, we chose to stick the the following :

- input size : images of 608×608, combining the EPFL and processed Chicago dataset
- number of epoch : 25
- learning rate : 1×10^{-4}, with a learning rate scheduler
- batch size : 16
- Optimizer : Adam

Concerning the **Loss function**, we use the Binary Cross Entropy with Logits Loss, and since it's a binary problem, we used the sigmoid activation problem to map each pixel to its corresponding label.

$$\begin{cases} \mathcal{L}_{\text{BCE}}(p, y) = -\frac{1}{N} \sum_{i=1}^{N} \left[y_i \cdot \log\left(\sigma(p_i)\right) + (1 - y_i) \cdot \log\left(1 - \sigma(p_i)\right) \right] \\ \sigma(p_i) = \frac{1}{1+e^{-p_i}} \end{cases}$$

RFE-LinkNet

RFE-LinkNet is an extension of the traditional LinkNet architecture, designed to address challenges in road extraction from high spatial resolution imagery (HSRI). It incorporates two novel components: *Receptive Field Enhancement (RFE)* and *Dual Attention (DA)* modules. The architecture comprises four key modules:

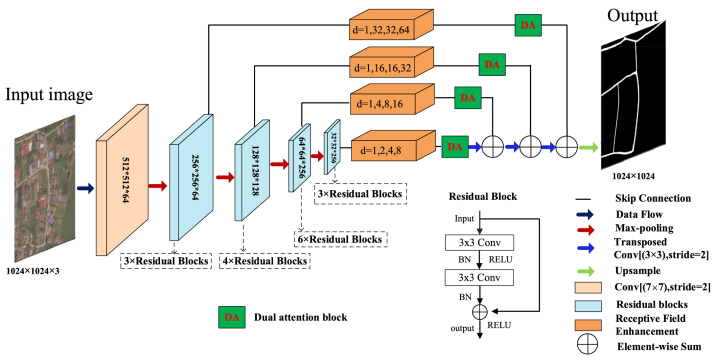


Fig. 3: Detailed architecture of RFE-Linknet

1. Multi-Scale Feature Extraction Module This module is based on the U-shaped LinkNet architecture and utilizes ResNet34 as the encoder backbone. It extracts multi-scale features by progressively downscaling the input to resolutions of $1/4$, $1/8$, $1/16$, and $1/32$. The hierarchical features captured at each level enable the network to effectively handle large HSRI inputs, such as images of size 1024×1024.

2. Receptive Field Enhancement (RFE) Module This module enhances the network's receptive field while preserving spatial resolution. It employs atrous convolutions with varied dilation rates to extract multi-scale features and includes five branches: four for capturing multi-scale features using different dilation rates and one residual branch to address gradient vanishing and grid effects. The design effectively balances global context and fine spatial details, enabling the network to capture narrow, complex, and interconnected roads in HSRI.

3. Feature Optimization Module This module refines feature representations by emphasizing important regions and suppressing noise. It incorporates a Channel Attention Module (CAM) to assign importance to feature channels using squeeze-and-excitation operations and a Spatial Attention Module (SAM) to focus on salient regions by combining max-pooling, average-pooling, and convolution operations. Additionally, residual connections are included to enhance the module's feature-fitting ability.

4. Multi-Scale Features Fusion Module Inspired by the U-Net architecture, this module aggregates multi-scale features in a bottom-up manner. It combines high-level features, which provide semantic richness, with low-level features, which retain spatial precision. This approach ensures accurate reconstruction of road structures by effectively balancing semantic and spatial details.

RFE-LinkNet uses a combined Binary Cross-Entropy (BCE) and Dice loss function:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{Dice}} + \mathcal{L}_{\text{BCE}},$$

This combination balances pixel-wise accuracy with segmentation completeness and smoothness.

Overall, the RFE-LinkNet architecture demonstrates exceptional capability in preserving road connectivity and fine details while minimizing false detections in complex backgrounds. By utilizing multi-scale features, enhanced receptive fields, and attention mechanisms, it proves to be highly effective for HSRI road extraction tasks. Its versatility and robust performance position it as a strong contender for broader semantic segmentation applications.

To optimize the performance of the RFE-LinkNet architecture, we conducted a grid search for hyperparameter tuning, focusing on the batch size and learning rate. We evaluated combinations of batch sizes $\{16, 32, 64, 128\}$ and learning rates $\{10^{-3}, 10^{-4}, 10^{-5}\}$. The best configuration was selected based on the lowest evaluation loss achieved during training. The table below presents the results of our grid search, highlighting the evaluation loss for each combination:

Based on the results, the configuration with a batch size of 4 and a learning rate of 10^{-4} yielded the lowest evaluation loss (0.85), making it the optimal setting for training the RFE-LinkNet architecture.

	10^{-3}	10^{-4}	10^{-5}
4	0.86	0.85	0.92
8	0.97	0.92	0.94
16	1.00	0.90	0.98

TABLE I: Best evaluation loss. The rows correspond to batch sized and the columns to learning rates.

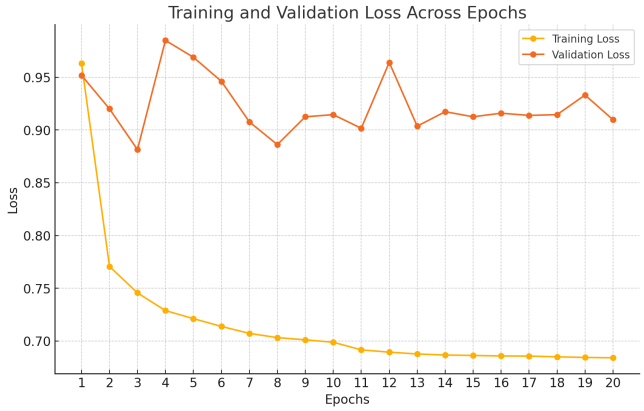


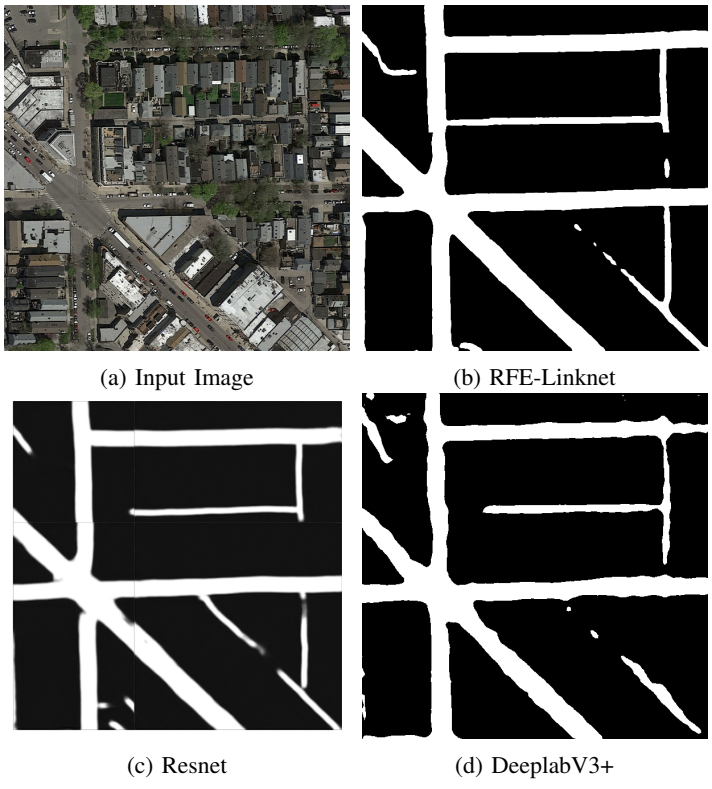
Fig. 4: Example of a training and evaluation loss curves for the RFE-LinkNet architecture.

Evaluation Metrics

We evaluate model performance using:

- **F1 Score**: $F1 = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$
- **Accuracy**

III. Results



(a) Input Image (b) RFE-Linknet

(c) Resnet (d) DeeplabV3+

Fig. 5: Input image and segmentation masks for the three models.

We evaluate the models on the validation set and compare their segmentation performance. The key results are summarized in Table 1.

Model	F1 Score	Accuracy
RFE-LinkNet	**0.918**	**0.956**
ResNet	0.909	0.952
DeepLabv3	0.887	0.946

TABLE II: Comparison of model performance on the test image set, from AI Crowd

RFE-LinkNet achieves the highest F1 score.

IV. Discussion

The results highlight the trade-offs between accuracy and computational efficiency. While DeepLabv3 performs best in terms of F1 score, U-Net offers a good balance between accuracy and inference time. ResNet struggles to capture fine-grained details, likely due to the absence of skip connections.

Strengths of our approach:

- Robust preprocessing pipeline ensures clean and consistent data.
- Use of Dice loss effectively mitigates class imbalance.

Weaknesses:

- Computationally intensive models may not scale well to larger datasets.
- Limited hyperparameter tuning for ResNet and DeepLabv3.

V. Ethical risks

One of the ethical risks we have identified is bias in model performance across regions, particularly between urban and rural areas. In particular, the EPFL provided dataset seems to contain images only of the suburbs of a single city in a single country. This can lead to bad performance in areas where the population density is much lower, such as rural or less developed areas. This imbalance is also evident in publicly available datasets: many of the most popular datasets for satellite image road segmentation such as the Chicago [5] or Massachusets[6] datasets suffer from the same bias towards wealthy urban areas.

To address this risk, we propose as future work, expanding the dataset by incorporating additional imagery from a diverse set of areas around the globe ensuring greater balance between urban and rural. An additional step could be the training of another machine learning model specifically focused on rural or underdeveloped areas. These mitigations would significantly improve the segmentation accuracy for rural areas. However, some barriers remain, such as the limited availability of high-quality labeled data for these regions and the financial cost of acquiring more diverse datasets.

VI. Summary

In this work, we implemented a robust pipeline for road segmentation using deep learning. We compared three state-of-the-art architectures (RFE-LinkNet, ResNet, and DeepLabv3) and demonstrated their strengths and limitations. Our results suggest that RFE-LinkNet achieves the best performance, while ResNet remains a computationally simpler choice for real-time applications.

Future work includes fine-tuning hyperparameters, exploring ensemble methods, and integrating post-processing techniques for further performance gains.

REFERENCES

[1] H. Zhao, H. Zhang, and X. Zheng, "Rfe-linknet: Linknet with receptive field enhancement for road extraction from high spatial resolution imagery," *IEEE Access*, vol. 11, pp. 106 412–106 422, 2023.

[2] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *arXiv preprint arXiv:1512.03385*, 2015.

[3] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," *ArXiv*, vol. abs/1706.05587, 2017. [Online]. Available: https://api.semanticscholar.org/CorpusID:22655199

[4] Z. Chen, C. Wang, J. Li, N. Xie, Y. Han, and J. Du, "Reconstruction bias u-net for road extraction from optical remote sensing images," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 14, pp. 2284–2295, 2021.

[5] P. Kaiser, J. D. Wegner, A. Lucchi, M. Jaggi, T. Hofmann, and K. Schindler, "Learning aerial image segmentation from online maps," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 7, pp. 3880–3893, 2017.

[6] A. Kulkarni, X. Wang, Z. Lin, and et al., "Efficient transformers for large-scale road segmentation in satellite imagery," *arXiv preprint arXiv:2404.02668*, Apr 2024. [Online]. Available: https://arxiv.org/abs/2404.02668