Machine Learning
Project 2 - Road Segmentation

Omar Boudarka, Adam Ezzaim, Mya Jamal Lahjouji
École Polytechnique Fédérale de Lausanne
21st December 2023

***Abstract*—This project focuses on road segmentation, which is essential for applications including traffic planning, traffic monitoring, and autonomous driving. In this paper, we introduce the use of machine learning algorithms for segmenting roads, leveraging a dataset of real-world images to understand the characteristics of road segments. Our evaluation showed a highly performing recognition of road patterns, highlighting the ability of our models to accurately distinguish road areas from non-road areas.**

I. Introduction

This project focuses on the implementation of road segmentation using satellite imagery sourced from Google Maps, with the primary objective being the categorization of image segments into two distinct classes: "road" and "background". This paper presents our approach to selecting and refining neural network models, specifically designed for pixel classification tasks.

II. Data Processing

A. Data Analysis

The training dataset comprises 100 satellite images, each of size $400 \times 400 \times 3$ pixels, with 400×400 pixel dimensions and three color channels (red, green, and blue - RGB). Accompanying each image is its respective ground-truth mask, a 400×400 gray-scale pixel image where pixel labels distinguish between "road" (white) and "background" (black).

Fig. 1: Satellite training image and its associated ground-truth

The distribution of our data exhibits a significant imbalance, with a much higher proportion of "background" labels compared to "road" labels. The numerical evidence is as follows: 77.47% of pixels are categorized as "background" against 22.53% for "road" labels.

To ensure the robustness of our training process, we partitioned our data to create a validation set, which allowed us to closely supervise and adjust the training of our model. Subsequently, to assess the effectiveness of our model, we conducted evaluations on a test set comprised of 50 images, each with dimensions of $608 \times 608 \times 3$ via the AICrowd website. The predicted masks consist of 16×16 pixel patches. The criteria for categorizing a patch as "road" is set with a threshold of 25%. This implies that a patch will be labeled as "road" if more than 25% of its pixels are identified as belonging to the road category.

To address the prevalent "non-road" versus "road" pixel imbalance in our dataset, we use the F1-score for evaluating model performance, as it effectively balances precision and recall, unlike accuracy which could be misleading.

The project focuses on daytime urban satellite images for both training and testing. For better generalization, including rural and night images is advised. Night images were not used in training due to their absence in the test set. However, data augmentation (e.g., diminished brightness to represent night images) should be applied to improve model robustness.

B. Data Augmentation

*1) **Insufficiency of Initial Data Set**:* During the initial phase of our research, we trained our neural network models on our dataset of satellite images. It quickly became apparent that the quantity and variability of the data were insufficient to achieve optimal performance and generalization in road segmentation, particularly considering that neural networks require a substantial volume of diverse data to perform effectively. This led us to explore data augmentation as a strategy to artificially expand our dataset.

*2) **Choice of Augmentation Techniques**:* In addressing the limitations of our dataset, we focused on four primary augmentation techniques: Translation, Rotation, Flip and Zoom. These techniques were selected for their particular relevance to the nature of satellite imagery and road segmentation:

Fig. 2: Image tranformations : Flip, translation, rotation

- **Translation**: By shifting the images horizontally and vertically, we can simulate the effect of satellite images taken from different altitudes and angles. This is particularly important for road segmentation as roads can appear at various positions within an image depending on the satellite's location. Translation helps in training the model to recognize roads regardless of their position in the image, enhancing the model's robustness and ability to generalize.
- **Rotation**: Rotating the images at different angles is crucial in representing the diverse orientations that roads can have. In real-world scenarios, roads are not always aligned north-south or east-west; they can run in any direction. By rotating the images, we train our model to effectively identify and segment roads regardless of their orientation. This increases the model's adaptability to real-world variations and improves its accuracy in segmenting roads in satellite imagery.

- **Flipping**: Applying both horizontal and vertical flips to the images introduces further variability, accounting for the possibility of mirrored or inverted views in satellite imagery. This helps in training the model to recognize symmetrical features and enhances its performance in diverse scenarios.
- **Zooming**: Zooming in the images allows the model to detect and segment roads at various scales. This is vital for satellite imagery as roads can be captured at different zoom levels. Zooming helps the model to generalize better across various image resolutions and distances.

*3) **Initial Approach with Keras Preprocessing Layers** [1]:* Our initial attempt at data augmentation utilized the preprocessing layers available in Keras. However, the results were not as effective as anticipated, leading us to seek more advanced solutions.

*4) **Transition to cv2 for Advanced Augmentation** [2]:* We subsequently adopted OpenCV (cv2) for its superior performance and wide array of functionalities in image processing. The advanced capabilities of cv2 enabled us to apply more complex and realistic transformations, generating a broader variety of augmented images.

*5) **Dataset Expansion and Implementation Challenges**:* Upon creating an entirely new dataset with augmented images, we faced a significant challenge: prolonged loading times. To address this, we shifted to implementing augmentation transformations dynamically after loading the data. This approach efficiently reduced loading times, maintaining the quality and diversity of our dataset.

III. Models

A. Basic Neural Network

In developing our foundational model, we drew inspiration from the helper code provided to us. Our strategy was to first segment the training dataset into patches of 16 by 16 pixels and training a model specifically on these segmented patches.

In the early stages of the project, we implemented a simple Fully Connected Feedforward (FCF) neural network. This model features a straightforward architecture of dense layers, utilizing ReLU activation functions and the He normal initializer. We used Keras Tuner for fine tuning the hyperparameters, including the size and the number of layers.

The model's architecture begins with a Flatten layer to transform the 16x16x3 image patches into a 1D array, followed by three dense layers with varying numbers of units (160, 288, and 96), and culminates in a softmax output layer for classifying each patch into road or background. This setup, while basic, aimed to capture the non-linear relationships in the data.

Despite its simplicity, this fully connected model yielded promising initial results, highlighting the potential of even rudimentary neural network architectures in segmentation tasks. However, it's important to note the inherent limitations of this approach. Fully connected networks lack spatial awareness, which is a significant consideration for segmentation tasks where the context and location of pixels are key. This limitation becomes evident as the model treats each patch independently, potentially leading to less coherent segmentation in larger images.

Therefore, while the FCF neural network served as an excellent starting point and benchmark for our project, its limitations in handling spatial context and the complexities of full-sized images indicate the need for more sophisticated, spatially aware models like CNNs or U-Nets. This initial exploration laid the groundwork for understanding the task's intricacies and set the stage for further model development.

B. Basic Convolutional Neural Network

Our second model enhanced the initial Fully Connected Feedforward (FCF) neural network by incorporating convolutional layers, transitioning to a Convolutional Neural Network (CNN). This model retained the previously fine-tuned hyperparameters of the dense layers from the FCF model, while introducing convolutional layers.

The architecture starts with convolutional layers, each followed by max-pooling, before transitioning into dense layers. This combination allows the model to first extract spatial features through the convolutional layers and then interpret these features using the dense layers. Convolutional layers are particularly effective in image-related tasks due to their focus on local receptive fields and shared weights, leading to more efficient feature extraction. This is especially beneficial in segmentation tasks like ours, where understanding the context and locality of features is crucial.

In terms of training, we used the Adam optimizer and the categorical cross-entropy loss function. The Adam optimizer is known for its efficiency in handling sparse gradients and adapting the learning rate during training, which is beneficial for complex tasks like image segmentation. The categorical cross-entropy loss is suitable for classification tasks, as it measures the performance of a model whose output is a probability value between 0 and 1.

Overall, the integration of convolutional layers in our model architecture represented a significant advancement, leveraging the strengths of CNNs in handling image data and resulting in a notable improvement in our road segmentation task.

C. U-Nets and variations

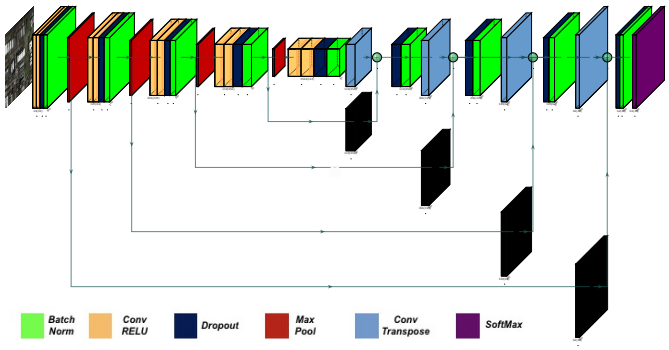

Fig. 3: U-Net 64 (32)

*1) **U-Net Architectures**:* In the progression of our road segmentation project, we implemented U-Net models [3], known for their effectiveness in image segmentation tasks. We trained two principal architectures, one starting with 32 filters and the other with 64, experimenting with various combinations of dropout and regularization. This was aimed at managing the data imbalance problem and enhancing model generalization. We adopted the Adam optimizer for its efficient handling of sparse gradients and Tversky Loss, a variant of the focal loss, specifically designed to address class imbalance by adjusting the loss calculation based on class frequencies.

The U-Net architecture is particularly well-suited for image segmentation due to its encoder-decoder structure. The encoder part, consisting of convolutional and pooling layers, progressively reduces the spatial dimensions of the image while increasing the depth, capturing the high-level contextual information. The decoder part then gradually reconstructs the spatial dimensions, using skip connections from the encoder to retain the spatial context lost during downsampling. This

structure is crucial in segmenting images with high precision, as it combines the context (from the encoder) with localization (from the decoder).

*2) **Training**:*

***Tversky Loss**:* The use of Tversky Loss [4] was a strategic choice to counter the challenge of data imbalance. Tversky Loss adds an extra level of control over the balance between false positives and false negatives, which is vital in ensuring that minority classes, such as roads in satellite images, are not overshadowed by the dominant background class.

***Dropout**:* Dropout is a regularization technique used in neural networks to prevent overfitting. It works by randomly "dropping out" a subset of neurons in a layer during training. In our U-Net models, we integrated dropout in various layers to enhance the model's generalization ability. We experimented with different dropout rates and found that a rate of around 0.1 yielded good results, striking a balance between regularizing the model and retaining sufficient information flow through the network layers.

***L2-Regularization**:* L2-Regularization is another technique to reduce overfitting. It works by adding a penalty term to the loss function, proportional to the square of the magnitude of the weights. In our U-Nets, L2-regularization was applied to the convolutional layers. We tested various regularization strengths and implemented the ones that optimized the performance, ensuring the model captures the essential features in the data without overfitting to the training set. We found that a regularization between 10^{-6} and 10^{-8} reduces overfitting of our models.

***Early Stopping**:* We used the tf.keras.callbacks.EarlyStopping [1] callback, monitoring the 'loss' metric. The training was halted if there was no improvement in the loss for 10 consecutive epochs (patience=10), and the model's weights were restored to those of the epoch with the best performance. This approach helped in avoiding unnecessary training time and also in preventing the model from overfitting to the training data.

***Input of the model**:* We trained these U-Nets on full images, moving away from the patch-based approach used in previous models. This shift allowed the models to learn from the complete spatial context of the images, leading to a more coherent understanding of the segmentation task. The models achieved excellent results, with over 97% F1-score on the training set and 86% on test submissions, underscoring the effectiveness of U-Nets in this domain.

***Data Augmentation**:* Encouraged by these results, we implemented data augmentation to further improve the model's robustness and generalization. This step was crucial in preparing the model for diverse scenarios and variations in road appearances in real-world satellite images.

*3) **Limitations and Future Work**:* While we explored different activation functions, including ReLU and experimented with models with and without dropout/regularization, we were constrained by time and computational resources, limiting our ability to test a wider range of activations like Leaky ReLU. Future work could explore these variations to potentially enhance model performance further.

IV. *Results*

In our approach to training the models for road segmentation, we carefully partitioned our dataset to ensure effective training and validation. Specifically, 85% of the dataset was allocated for training, and the remaining 15% was used as a validation set. This partitioning strategy served as a form of cross-validation.

A. Basic Neural Network and Convolutional Neural Network

The performance of the Fully Connected Feedforward (FCF) and Convolutional Neural Network (CNN) models shows distinct outcomes. The FCN model exhibits a drop in validation scores compared to its training results, suggesting a potential lack of spatial contextualisation. In contrast, the CNN demonstrates superior generalization, with consistently higher F1-scores and accuracies across training and validation sets. The CNN's robustness to new data indicates that it is the more effective model for the task, emphasizing the importance of choosing an architecture that aligns with the data's inherent structure.

Model	Epochs	T F1	T Acc	V F1	V Acc
FCF	50	0.8550	0.8891	0.6954	0.7719
CNN	30	0.9421	0.9555	0.7655	0.8234

TABLE I: Results for basic models

As illustrated in Figure 4, the basic CNN displays an ability to recognize road patterns, but its performance is inconsistent. The CNN predictions are characterized by a patchwork quality rather than a smooth, homogeneous segmentation, indicative of its struggle with complex textures and contrasts.

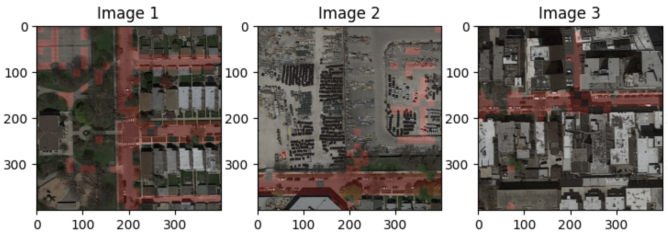

Fig. 4: Illustration of the predictions of the basic CNN

B. U-Nets

*1) **Without Data Augmentation**:* The results from the U-Net models with various configurations underscore significant differences in performance. The standard U-Net 32 with 0.5 dropout model yielded modest F1-scores. Enhancements to the model, such as additional L-2 regularization in U-Net 32R and the exclusion of dropout in U-Net 32NoD, resulted in notable improvements in both training and validation F1-scores, with U-Net 32NoD achieving an impressive training F1 of 0.9876 and validation F1 of 0.7695.

Model	L2	Dropout	T F1	V F1
U-Net 32	0	0.5	0.6605	0.5125
U-Net 32R	10^{-8}	0.2	0.9726	0.7526
U-Net 32NoD	0	0	0.9876	0.7695
U-Net 64	0	0	0.9490	0.7577
U-Net 64RD	10^{-8}	0.1	0.9851	0.7683

TABLE II: Results for U-Net models without data augmentation

Scaling up the model to U-Net 64 also improved performance, but the addition of regularization and dropout in the U-Net 64RD variant did not significantly outperform the U-Net 64 in terms of validation F1, although it did show a marginal increase. The U-Net 64RD variant allowed us to get a good F1 score of 0.854 on evaluation on AICrowd.

These results suggest that while increasing complexity and scale can lead to better training performance, it does not always translate to proportionate gains on validation data, highlighting the balance needed between model capacity and generalization.

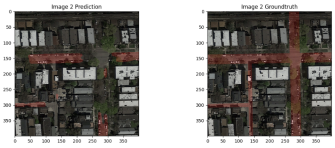

Fig. 5: Image groundtruth and U-Net 32 prediction

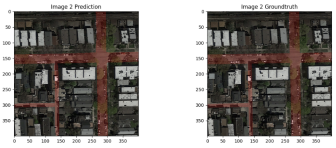

Fig. 6: Image groundtruth and U-Net 64 prediction

Fig. 7: Comparison of 2 U-Nets based on predictions trained without data augmentation

The analysis of Figure 7 reveals that the U-Net 32 model yields predictions that are notably less precise compared to those of the U-Net 64 model. Furthermore, predictions for more advanced models like U-Net 64RD are not provided, as the improvements in their performance, when contrasted with U-Net 64, are not discernible to the naked eye.

*2) **With Data Augmentation**:* We observe on Table III that U-Net A64RD and U-Net A32RDv4 show excellent score, indicating not only effective learning from the training data but also strong generalization to new, unseen data, thanks to training on augmented data. We show in Figure 8 the training and validation loss of model A64RD to demonstrate that the model avoids the issue of overfitting.

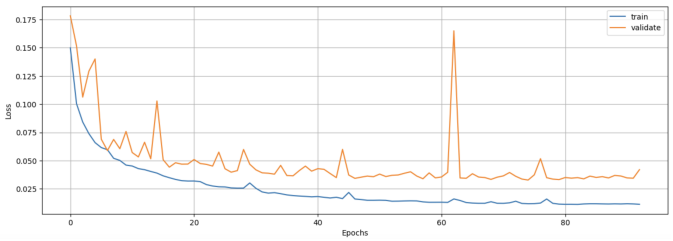

Fig. 8: Training and validation loss of U-Net A64RD

However U-Net A64 model stands out due to its extremely low F1 scores, which present anomalies compared to the other models. This is certainly caused by the absence of Droupout layers and L2-regularization in this model.

Ultimately, we found that while the A32RD may not claim the highest F1 score during validation, it outperformed others on the submission platform, achieving an impressive score of 0.903.

Model	L2	Dropout	T F1	V F1	AICrowd
A32RD	10^{-6}	0.1	0.9601	0.8989	0.903
A64RD	10^{-8}	0.1	0.9697	0.9062	0.899
A64	0	0	0.0539	0.0552	X
A32RDv3	10^{-4}	0.2	0.8553	0.8310	0.884
A32RDv4	10^{-8}	0.08	0.9698	0.9042	0.893
A16RD	10^{-6}	0.1	0.9638	0.8871	0.860

TABLE III: Results for U-Net models with data augmentation

V. Improvements

As our project concludes, we look towards future directions for enhancement. Here are some of the key ideas for future development:

1) **Incorporating More Diverse Data:** Expanding the dataset to include a wider range of scenarios, such as varying lighting conditions, weather conditions, and geographical locations, would be beneficial. This would enhance the model's robustness and its ability to accurately segment roads under diverse conditions.
2) **Experimenting with Activation Functions:** Future models could benefit from experimenting with a range of activation functions like Leaky ReLU or Parametric ReLU.
3) **Hyperparameter Optimization:** Further systematic tuning of hyperparameters, possibly through methods like Bayesian optimization using Keras Tuner, could lead to more optimal model configurations and enhanced performance reducing the overfitting of our models.

By targeting these areas for improvement, we aim not only to enhance the accuracy and F1 score of our segmentation models but also to ensure their adaptability and applicability in a variety of real-world settings.

VI. Ethics

The use of satellite images for road segmentation can raise privacy concerns. While focusing on roads, these images may also capture private properties or individuals, potentially leading to surveillance issues. **Bias and Fairness:** Machine learning models, including those used in road segmentation, can inherit biases present in their training data. For instance, a model predominantly trained on urban areas might underperform in rural or less developed regions. This could lead to unequal benefits from the technology, reinforcing existing inequalities.

VII. Conclusion

In this study, we have explored the challenging task of road segmentation using machine learning techniques, particularly focusing on the powerful U-Net architecture. Our journey began with simpler models like Fully Connected Feedforward and basic Convolutional Neural Networks, which provided us with foundational insights into the complexities of image segmentation. As we progressed, the implementation of U-Nets, with variations in filters, dropout, loss functions and regularization, marked a significant advancement in our project, demonstrating remarkable proficiency in segmenting roads from satellite imagery. Our models achieved impressive F1-scores in training, and the application of data augmentation techniques further strengthened their generalization capabilities, as evidenced by the robust performance on unseen test data.

This research not only underscores the potential of U-Net architectures in road segmentation but also highlights the importance of careful hyper-parameter tuning, data augmentation, and loss function selection in addressing specific challenges of a dataset. While our models demonstrated high effectiveness, there remains room for future exploration. Advanced architectures, diverse data incorporation, experimentation with different activation functions, and further hyper-parameter optimization present promising avenues for continued improvements.

References

[1] F. Chollet *et al.*, "Keras." https://keras.io, 2015.
[2] G. Bradski, "The OpenCV Library," *Dr. Dobb's Journal of Software Tools*, 2000.
[3] N. Tomar, "Unet implementation in tensorflow using keras api," 2021.
[4] D. E. Seyed Sadegh Mohseni Salehi and A. Gholipour, "Tversky loss function for image segmentation using 3d fully convolutional deep networks," 2017.