Road Segmentation From Satellite Images

Mahmoud Dokmak, Romain Corbel, Guilhem Destriau
CS-433 Project 2, EPF Lausanne, Switzerland

***Abstract*— Identifying road segments in satellite images requires robust Machine Learning models. In this work, we sought to develop the best-performing model, starting from a random baseline and iteratively improving through Logistic Regression, a basic Convolutional Neural Network, and finally the U-Net model. U-Net, renowned for its U-shaped architecture and success in medical image segmentation, was hypothesized to perform well for our satellite imagery task. After implementation, it achieved a very good F1 score of 0.838, demonstrating its effectiveness for precise road segmentation.**

I. Introduction

Segmenting roads from satellite images is an essential task with applications such as urban planning and navigation. In this project, we work with a dataset of 100 satellite images from Google Maps, accompanied by ground-truth labels where each pixel is classified as road (1) or background (0). The objective is to train a model capable of assigning these labels and accurately segmenting road networks.

This problem comes with several challenges. Roads can vary in color, texture, and shape, and are often partially covered by cars, trees, shadows, or features such as crosswalks. In addition, some paths, such as private driveways or dirt roads, may not always be considered roads, adding further complexity.

To address these challenges, we started with a random baseline model and progressively experimented with Logistic Regression, a basic Convolutional Neural Network, and finally the U-Net model. We implemented data augmentation strategies to improve robustness and evaluated the results under various configurations. This report presents the models and techniques used to converge towards U-Net, the results obtained, and an analysis of ethical risks related to the application of such models. The entire code for this project is available on GitHub at ml-project-2-satnet.

II. Model and Methods

Our dataset consists of 100 pairs of images. Each pair includes a colored satellite image (RGB) of 400×400 pixels and a corresponding ground truth mask of the same dimensions. In the ground truth mask, road pixels are assigned white, while background pixels (i.e., non-road parts of the image) are assigned black. Our goal is to predict whether 16×16 pixel patches in a new set of larger satellite images (612×612 pixels) belong to the road or background class.

To achieve our goal, we adopted a step-by-step approach. We started with a random prediction method to establish a baseline. Next, we implemented a simple logistic regression model to evaluate how basic machine learning techniques perform on this task. As expected, logistic regression did not perform well. Recognizing the need for a more robust approach, we experimented with a simple Convolutional Neural Network (CNN). Observing the potential of neural networks, we decided to implement a more sophisticated and widely recognized CNN architecture: U-Net.

1) *Random baseline*: The random baseline predicts road pixels based on the distribution of road proportions in ground-truth training images. We fit a Gaussian distribution to these proportions, sample a proportion for each evaluation image, and randomly assign labels (1 for road, 0 for background) to pixels according to this proportion. Although simplistic, this method serves as an essential baseline for comparing and highlighting the improvements achieved by more advanced models.
2) *Logistic Regression*: For the Logistic Regression model, we performed feature extraction by using a basic principle: calculating the mean and standard deviation of pixel values within a patch. Since the images are RGB, we explored three ways of performing this feature extraction:
   - Gray scale features (2 features): Convert the patch to gray scale and compute the mean and standard deviation.
   - RGB features (6 features): Calculate the mean and standard deviation separately for each color channel (R, G, and B).
   - Combined features (8 features): Use both gray scale (2 features) and RGB (6 features), resulting in a total of 8 features.

   For each of the three models, the F1 score was used as the primary metric for tuning. We conducted a grid search to optimize the hyperparameters and then fine-tuned the decision threshold. Since predictions were not binarized within the loss function, this approach allowed us to train the models once and subsequently adjust the thresholds to identify those that maximized the F1 score.
3) *CNN*: Since we are working with images, we looked at popular computer vision models designed for such a task and found CNN. This type of deep learning model is capable to analyze images by automatically learning features such as edges and textures through

convolution operations. We decided to implement one very simple CNN with seven layers. The first 4 layers increase in depth, whereas the last 3 decrease to a depth of 1 for a black and white image.

4) *U-Net* [1]: After experimenting with a classical CNN and observing its capabilities, we decided to extend our approach by implementing a more advanced CNN architecture: U-Net. U-Net is one of the most well-known CNN-based models, celebrated for its success in medical image segmentation and widely used in other domains, including satellite image segmentation, which perfectly aligns with our task. Its distinctive feature is its U-shaped architecture with skip connections. These connections preserve spatial information lost during encoder down-sampling and enhance the decoder's ability to accurately localize features, a crucial aspect for our task.

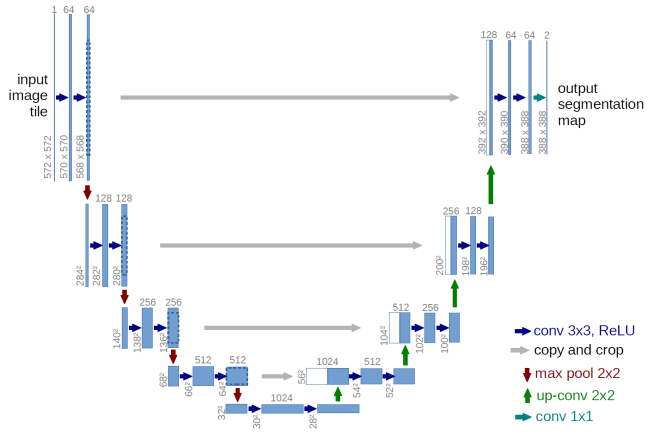

Figure 1. U-Net Architecture (Taken from: [1])

III. Data Augmentation

When training convolution neural network models, we observed a problem: the model is poor when the road is diagonal. This can be explained by the lack of images for training, but also by the lack of images containing this type of route. Moreover, given the similarity in color across all the photos, we decided to adjust the hues of the images to make our model more robust, especially when satellite images are taken at different times of the day or in different geographical areas. We therefore decided to increase the number of images on which the models were trained. From the initial 100 images we were able to create new images, which can be rotate from a certain angle. This allows for the inclusion of images with diagonal roads in the dataset, which are underrepresented in the original dataset.

- **Rotation** The image is randomly rotated by an angle ranging from $[-\pi, \pi[$ to create more routes that are neither vertical nor horizontal
- **Random Size Crop** The use the RandomResizedCrop function allow us to randomly zoom into the image,and then recreate an image of the initial size.
- **Sharpness** The sharpness of the image is modified by blurring it
- **Brightness** Randomly changes the brightness of the image
- **Contrast** Randomly changes the contrast of the image
- **Hue** Randomly changes the hue of the image

All these changes have been made thanks to torchvision's v2 importer. Using this library enabled us to carry out these transformations with a certain degree of probability. The idea is not to modify each image drastically, but rather to change them a little each time in order to make the model robust to each small change. This means that all modifications have a probability arbitrarily chosen at 0.25 of being carried out. However, given that the initial problem was diagonal roads, the rotation transformation has a higher probability of 0.75 of taking effect.

IV. Results

Before presenting the performance achieved by each of our trained models, it is essential to first outline their key parameters.

1) *Random baseline*: We analyzed the distribution of road pixel proportions in the training set and used it to generate predictions:
   - Validation size: 20 %
   - Gaussian Fit: $\mu = 0.20$, $\sigma = 0.07$

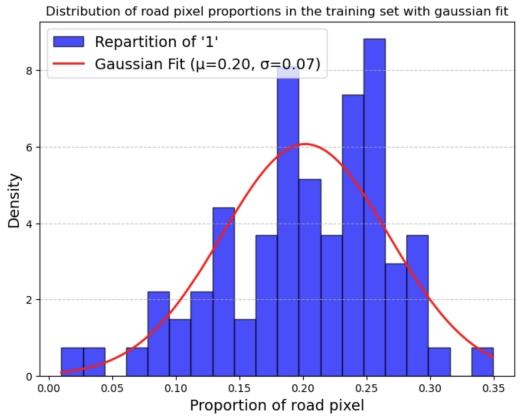

Figure 2. Distribution of road pixel proportions in the training set with gaussian fit

2) *Logistic Regression*: We performed our Logistic Regressions on both the initial training dataset and the augmented dataset of 800 images from part III. We used sklearn for both the Logistic Regression models and the Grid Search process during hyperparameters tuning with the following parameters:
   - Solver: liblinear, lbfgs, sag, newton-cg, saga

- Penalty: none, L1, L2, elasticnet
- Log space of 10 values from 10^{-6} to 10^{6}
- L1 Ratio: 0.1, 0.5, 0.9 (only for elasticnet)
- Max Iterations: 100, 1000

After the models were tuned, we implemented a straightforward decision threshold optimization by testing various thresholds to identify the ones that maximized the F1 scores on the validation test:

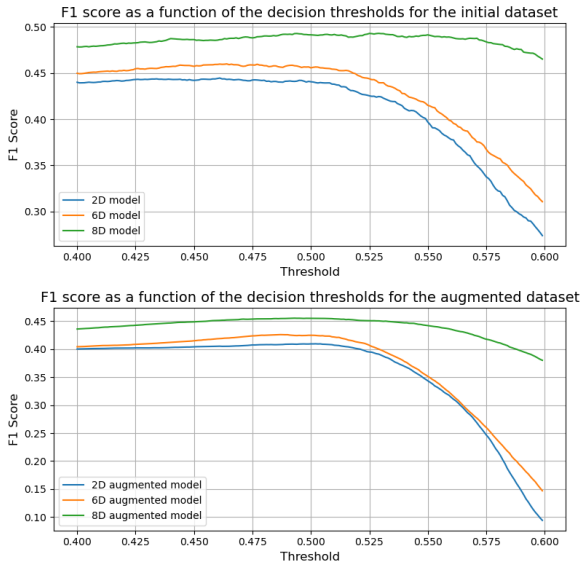

Figure 3. F1 scores of the 6 models as a function of the decision tresholds

These graphs show that augmenting the training set worsened the performance of the Logistic Regression models, which is expected since the model relies solely on the mean and standard deviation of image colors. Image augmentation alters these statistical features, introducing noise and disrupting the model's ability to classify effectively. In contrast, extracting eight features proved to be the most effective approach. The 8D model trained on the initial dataset achieved the highest F1 score of 0.49. Therefore, we will use this model for comparison with other machine learning models.

3) *CNN*: A basic Convolutional Neural Network (CNN) was tested to explore the potential of deep learning in capturing spatial features of the images.The CNN was trained on two datasets: the original dataset of 100 images, and an augmented dataset of 3000 images. In both cases, the parameters were as follows:
   - *Validation size*: 20%
   - *Number of trained epochs*: 10
   - *Learning rate*: 1×10^{-3}
   - *Weight decay*: 1×10^{-2}

   The results of this model are shown in the figure 4, reaching a f1 score of 0.59 for the augmented dataset and 0.51 for the original dataset.

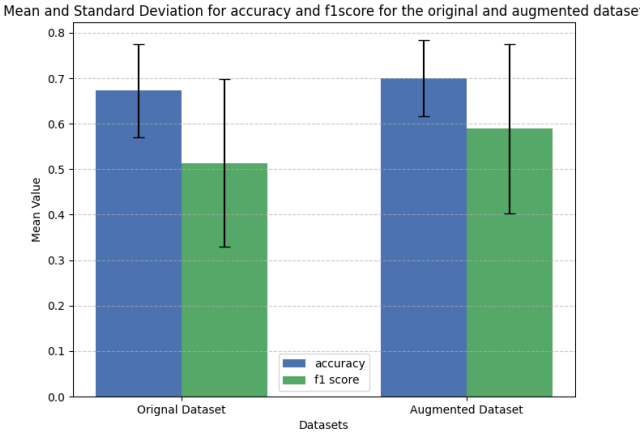

Figure 4. F1 scores and Accuracy for both orginal and augmented dataset

4) *U-Net*: We employed the U-Net architecture for both training and prediction, experimenting with two distinct training environments:
   First configuration:
   - *Number of images*: 1000
   - *Validation size*: 20%
   - *Number of trained epochs*: 10
   - *Learning rate*: 1×10^{-3}
   - *Weight decay*: 1×10^{-2}

   Second configuration:
   - *Number of images*: 3000
   - *Validation size*: 20%
   - *Number of trained epochs*: 50
   - *Learning rate*: 1×10^{-3}
   - *Weight decay*: 1×10^{-2}

For the U-Net, as it is our final proposed model, we tracked the loss, accuracy, and F1-score on both the training and validation datasets at each epoch. For display purpose, we only show the loss and the F1-score on the the training (blue) and validation (orange) set.

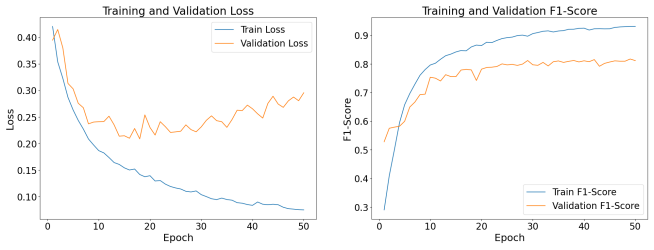

Figure 5. Training and validation loss and F1-Score evolution accros epochs

Figure 6. Example on a test image using our best model (U-Net)

V. Discussion

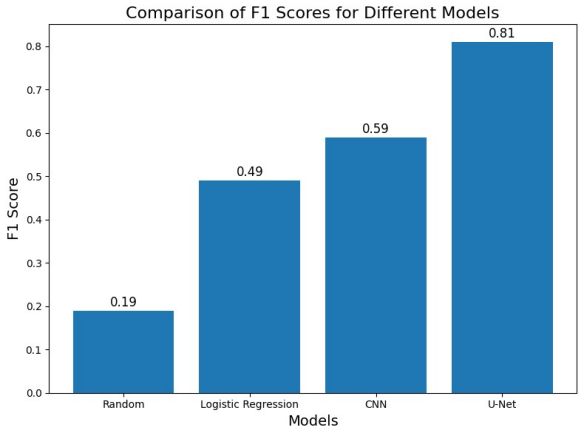

Figure 7. F1 scores of every model

As expected, the random model and logistic regression performed relatively poorly. The random model was used purely as a baseline and was not expected to achieve good performance. Regarding the Logistic Regression, this model is not designed to capture spatial features or shapes. Achieving these results based solely on the standard deviation and mean of the colors is already a good outcome.

When we introduced a simple Convolutional Neural Network (CNN), we observed a significant improvement in performance. CNNs, by design, perform convolutions that allow the model to capture local spatial structures, such as shapes and edges, making them well-suited for this type of task. Therefore, our first CNN demonstrated promising results.

Building on this, we focused our efforts on the U-Net architecture, which is specifically designed for image segmentation tasks. The U-Net model performed exceptionally well, particularly when trained on the augmented dataset consisting of 3000 images. This augmentation allowed the model to generalize better, leading to strong performance.

We observed that the U-Net model exhibited rapid overfitting on small datasets, often predicting predominantly background pixels. This issue required close scrutiny of the model to mitigate overfitting. The overfitting likely stemmed from the significant class imbalance in the dataset, where background pixels greatly outnumbered road pixels. Finally, despite augmenting the training set with rotations to include diagonal roads, the final U-Net models consistently performed better on gridded road structures compared to more complex road layouts.

VI. Ethical Risk

A key ethical risk in this project is the fairness of the model, specifically its ability to perform consistently across diverse geographic and environmental contexts.

The stakeholders affected include populations living in areas that differ significantly from the characteristics represented in the dataset. If the model underperforms in underrepresented areas, it could result in inaccurate maps and exacerbate inequalities in its performance.

The likelihood of this issue is very high, as the dataset contains only 100 images. This limited size makes it impossible to adequately capture the diversity of the world, increasing the risk of biased outcomes.

To evaluate this risk, we analyzed the training dataset to identify geographic biases. We found that all the images exhibit a strikingly uniform style, suggesting they originate from a single affluent country, likely the USA, as evidenced by the straight, grid-like road layouts typical of urban planning in such regions. This lack of diversity raises significant concerns about the model's ability to generalize to areas with different road structures, landscapes, or socioeconomic conditions. Additionally, research on similar machine learning models confirms that underrepresentation in training data is a common cause of performance disparities [2].

To mitigate this risk, we applied data augmentation techniques, introducing variations in lighting, terrain, and other factors to better simulate diverse contexts. While these methods improve the model's robustness, they cannot replace the need for a more representative dataset.

Addressing the fairness issue in its entirety was beyond the scope of this project, primarily due to limited access to diverse satellite imagery. Future efforts should prioritize sourcing and using more representative datasets to ensure the model performs consistently across all geographic contexts.

VII. Summary

In this project, we compared four machine learning models that we implemented to achieve the goal of road segmentation. As shown in Figure 7, our U-Net model significantly outperformed the baseline, logistic regression, and vanilla

CNN models, achieving a high F1 score of 0.838 and demonstrating its effectiveness for road segmentation from satellite images. However, we observed that the model's performance was impacted by class imbalance and dataset limitations, including the predominance of background pixels and the underrepresentation of complex road layouts. Finally, we addressed ethical considerations regarding the fairness of our approach, as the dataset predominantly featured gridded (American) roads, which could limit the model's ability to generalize to diverse road structures worldwide.

REFERENCES

[1] Apprendre le Deep Learning. Tout savoir sur u-net : l'architecture révolutionnaire pour la segmentation d'images. [Online]. Available: https://apprendre-le-deep-learning.com/u-net-une-architecture-pour-la-segmentation-d-images/

[2] E. Aiken, E. Rolf, and J. Blumenstock, "Fairness and representation in satellite-based poverty maps: Evidence of urban-rural disparities and their impacts on downstream policy," in *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, IJCAI-23*, E. Elkind, Ed. International Joint Conferences on Artificial Intelligence Organization, 8 2023, pp. 5888–5896, aI for Good. [Online]. Available: https://doi.org/10.24963/ijcai.2023/653

[3] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," *CoRR*, vol. abs/1505.04597, 2015. [Online]. Available: http://arxiv.org/abs/1505.04597

[4] U. Kurt, "Implement and train u-net from scratch for image segmentation - pytorch," 2023, accessed: 2024-12-09. [Online]. Available: https://www.youtube.com/watch?v=HS3Q_90hnDg