# Street Artist

Alex Thillen, Felix Schoellen, Igor Martinelli & Roger Csaky-Pallavicini
Department of Computer Science, ETH Zurich, Switzerland

***Abstract*—Significant advances in semantic image segmentation have been achieved with Convolutional Neural Networks (CNNs), notably in road segmentation from satellite images. This report introduces *Street Artist*, a novel network architecture for semantic segmentation. The primary innovation is an autoencoder that merges the predictions of multiple pretrained models, including UNet, UNet++, and DeepLab V3+, using encoders like ResNet50 and EfficientNet.**

***Street Artist* significantly improves segmentation performance, achieving an F1 score of 0.92202 in the Kaggle competition, surpassing the best individual model score of 0.90768. This demonstrates the effectiveness of our learned ensemble approach in enhancing the accuracy and robustness of urban road segmentation.**

## I. Introduction

Semantic segmentation, particularly in the context of urban environments, is a crucial task in computer vision with applications ranging from autonomous driving to urban planning. State-of-the-art methods commonly use U-Net architectures [1] due to their effectiveness in various segmentation tasks.

In this project, we aimed to classify each 16x16 pixel block of a given set of test images as either street or non-street. We were provided with 144 training images, each with corresponding ground truth labels. All these images are 400x400 pixel aerial RGB photos of urban areas.

After augmenting the training set, we trained our own custom UNet model as well as several segmentation models provided by the Segmentation Models PyTorch library [2], utilizing pretrained encoders. A key feature of our approach is the use of an autoencoder to combine predictions from these base models, which were kept frozen during training. This combination strategy, termed *Street Artist*, integrates the strengths of each model, compensates for their individual weaknesses, and introduces a regularizing effect due to the limited complexity of the autoencoder. This beneficial effect is evident even when only a single base model is utilized.

## II. Related Work

Recent advancements in deep learning have significantly improved the performance of semantic segmentation tasks. The introduction of fully convolutional networks (FCNs) by Long et al. [3] marked a significant advancement in semantic segmentation. FCNs replaced fully connected layers with convolutional layers, enabling pixel-wise classification and producing feature maps instead of scalar outputs. This led to the development of sophisticated encoder-decoder architectures, which have since become the backbone of modern segmentation models.

Building on the FCN framework, several enhancements have been proposed to refine segmentation outputs, improving the accuracy and precision of predictions. Chen et al. introduced DeepLab, which utilizes atrous convolutions and fully connected conditional random fields (CRFs) to refine segmentation outputs [4]. The inclusion of atrous convolutions expanded the receptive field without increasing the number of parameters, allowing for better context capture. Fusing high- and low-level features has proven effective in preserving spatial resolution and improving edge detection [1], [5].

The U-Net architecture, proposed by Ronneberger et al. [1], advanced the field by using skip connections to fuse features from different layers. This architecture, initially designed for biomedical image segmentation, has shown remarkable performance in various tasks. Subsequent iterations, such as UNet++ and ResNet-based UNet models, incorporated more complex encoders tailored for different domains [6].

In the specific domain of road segmentation, methodologies have evolved to address challenges such as occlusions, varying road materials, and complex urban environments. Mnih and Hinton [7] combined Boltzmann machines with tailored preprocessing and postprocessing steps to improve segmentation accuracy. Saito et al. [8] leveraged convolutional neural networks (CNNs) to extract roads directly from raw images, setting a new standard for road segmentation performance. Zhang et al. [6] utilized a deep residual U-Net, combining the robustness of residual networks with the spatial precision of U-Net, achieving superior road extraction results.

Our approach, *Street Artist* builds on these advancements by integrating predictions from multiple pretrained models, such as UNet, UNet++, and DeepLab V3+, through a novel autoencoder framework.

## III. Models

### A. Baseline models

The 3 baseline implementations ”baseline_logreg”, ”baseline_cnn” and ”baseline_unet given with the instruction set were run, tested and the output submitted without any changes to the code provided. We acknowledge that these implementations are far from optimal, in particular because

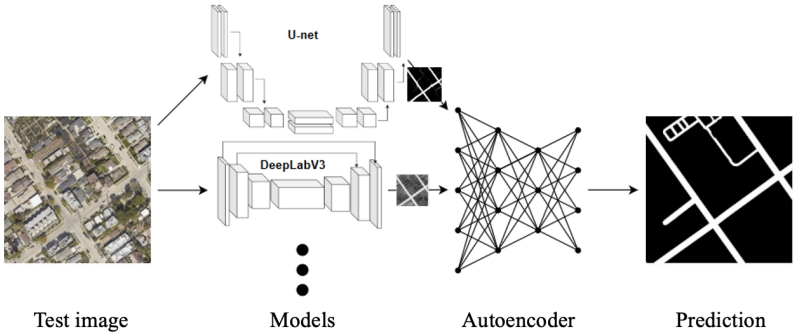


Figure 1. *Street Artist* intuition

of short training and the small training dataset, but include their results to demonstrate our progress throughout the project.

### B. Submodels

We evaluated a broad range of model and encoder combinations, all implemented using the SMP library. The best performing combinations we found are a resent 50 encoder combined with a UNet architecture (”resnet_50_unet”), an EfficientNet-B4 encoder combined with a Unet architecture (”efficientnet-b4_unet”), a resnet50 encoder combined with a deeplabv3+ model (”resnet_50_deeplabv3+”), a resnet 34 and a resnet 50 encoder combined with unet++ architectures (”resnet34_unet++” and ”resnet50_unet++”) and a xception encoder combined with a unet++ architecture (”xception_unet++”). During training, we froze the first two convolutional layers for models utilizing a ResNet-based encoder. This approach specifically enhanced generalization in ResNet models. Consequently, we did not apply freezing to other encoder types, such as EfficientNet or Xception. Because these are state of the art models, finetuned to the best of our abilities, they are also optimal baselines to compare our novel implementation to. Thus we also evaluated them all individually.

### C. Street Artist

The trained and frozen submodels are used as parts of the *Street Artist* network, where their output is fed to an autoencoder as shown in figure 1. The autoencoder used here has a very simple structure, halving and later doubling the number of neurons with each layer. The name for the model is derived from its capability to inpaint missing or noisy road predictions[9], similar to an artist restoring a damaged piece of art. This is demonstrated in figure 3. Additionally, the relatively simple structure of our autoencoder acts as a regularizer, preventing overfitting and promoting generalization.

## IV. Methodology

### A. Dataset

Our training set includes 144 RGB images of size 400x400 pixels and their corresponding ground truth masks. In these masks, each pixel is assigned a value $v \in [0, 1]$ representing the probability of belonging to a road area ($v = 1$) or the background ($v = 0$). The test set consists of an additional 144 RGB images of the same size without ground truth labels. Given the limited size of the initial dataset, data augmentation was necessary to enhance the robustness and performance of our deep learning models. To achieve this, we integrated an additional 100 images from the Deepglobe public dataset, specifically selecting those depicting urban areas to ensure consistency with the original dataset. Each image from the Deepglobe dataset was divided into four sub-images, effectively matching the zoom level of the initial dataset images. This augmentation process resulted in an expanded and diversified training dataset comprising 544 images.

### B. Preprocessing

To enhance the variability and comprehensiveness of the dataset, we employed an extensive set of data augmentation techniques, including affine geometric transformations and adjustments in image appearance (brightness, contrast, and saturation). Each image was resized to 384x384 pixels to standardize the input size, as 384 is divisible by $2^7$, aligning better with popular autoencoder structures that typically halve the number of neurons after each convolution. In contrast, a side length of 400 would result in non-integer dimensions after several layers ($400/2^5 = 12.5$).

The specific augmentations applied include:

- **Horizontal and Vertical Flips**: Randomly flipping the images and their corresponding masks horizontally or vertically to increase orientation variability.
- **Rotations**: Randomly rotating the images and masks by 90 degrees to further enhance orientation variability.
- **Random Resized Cropping**: Applying random crops to the images and masks, followed by resizing to the

original dimensions. This ensures that the model learns to identify roads at various scales and positions.

- **Brightness, Saturation and Contrast Adjustments**: Randomly modifying the brightness, contrast, and saturation of the images to simulate various lighting, camera, and meteorological conditions. This helps the model generalize to diverse real-world scenarios, enhancing its robustness to different environmental variations.
- **Affine Transformations**: Applying random affine transformations, including slight rotations, translations, scaling, and shearing. This introduces geometric distortions, making the model robust to such variations in real-world scenarios.

These augmentations were crucial in creating a robust training dataset that could simulate the diverse and unpredictable conditions found in real-world aerial imagery. The preprocessing ensured that the model was exposed to a wide range of transformations, improving its ability to generalize to new and unseen data.

### C. Training

The training process involved configuring data loaders to feed batches of images and their corresponding masks to the model. The dataset was split into training and validation sets with a 90/10 ratio.

The models were trained using a binary cross-entropy loss function and optimized with the Adam optimizer. Training consisted of multiple epochs where model weights were updated based on the loss calculated from predictions and ground truth labels. Models using a resnet encoder performed better with a second training round. We assume this is because the increased learning rate allows them to escape some local minima. Other models did not improve with a second training round, and thus we didn't apply it. An early stopping mechanism monitored validation performance and halted training if no improvement was observed to prevent overfitting. Further, the learning rate was automatically reduced upon hitting a plateau in F1 score.

A fine-tuning phase was conducted for specific models that benefited from additional training. The learning rate was reduced, and the models were trained for several more epochs exclusively on the official dataset, with a lower patience value for early stopping.

In the *Street Artist* autoencoder, the predictions from multiple pretrained models were combined by stacking them across channels. If there are $n$ pretrained models, their predictions form $n$ input channels to the autoencoder. The autoencoder then processes these channels to produce the final segmentation output. This new model integrates the strengths of different models, enhancing the overall prediction accuracy. We trained the *Street Artist* using a similar approach to that of the submodels, with data loaders providing batches of images and masks. The model was optimized using the Adam optimizer, and early stopping was employed to avoid overfitting. The training process involved a warmup round of 17 epochs with a patience of 3 epochs, followed by an additional round of 19 epochs with a patience of 4 epochs.

### D. Prediction

After training, the models were used to generate predictions on the test set. The test images were loaded and preprocessed similarly to the training images, including resizing to 384x384 pixels. Predictions were made for each test image and then post-processed to match the original image dimensions.

The *Street Artist* model combined the outputs of multiple pretrained models by stacking their predictions and passing them through an autoencoder to produce the final segmentation output. This ensemble approach leveraged the strengths of different models, improving overall prediction accuracy and robustness.

For the final output, the Kaggle competition requires a 0, 1 label for every 16×16 patch of an image. Since our models produce pixel-wise predictions, we derived per-patch labels by averaging the pixel probabilities within each 16×16 patch. A predictive label of 1 was assigned if the resulting average was higher than or equal to a threshold $T = 0.25$. This ensured that the predictions were consistent and aligned with the competition's format.

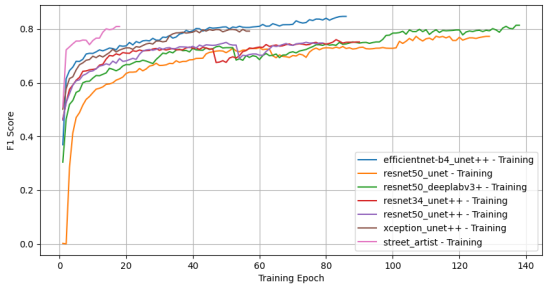


Figure 2. F1 training scores over epochs

## V. Results and Discussion

Figure 2 shows the F1 training score of all submodels during training. Because *Street Artist* was prone to overfitting we applied a more aggressive early stopping criterion, resulting in less training epochs then the submodels. This causes the submodels to overfit more to the training set, this is why *Street Artist* performs worse then the submodels in regard to F1 training score, but better with regard to validation score, as shown in figure 4. The dips of submodels using resnet encoders indicate the start of the second training round mentioned in chapter IV C. Overall these training curves match our expectation and the kaggle

scores achieved. We noted that sometimes F1 score improves suddenly after long stagnation. This can be seen for example with resnet50_deeplabv3+ arround epoch 90. If this would happen too late (and thusnot at all) this could introduce undesired early stopping and decrease performance. We suspect this could be ameliorated with larger batchsizes, but due to time constraints we decided not to pursue this issue further.

| Model | Public Score |
|---|---|
| **Public Baseline Models** | |
| baseline_logreg | 0.59893 |
| baseline_cnn | 0.81100 |
| baseline_unet | 0.81371 |
| **Street Artist Base Models** | |
| resnet_50_unet (custom + finetuned) | 0.90768 |
| efficientnet-b4_unet (finetuned) | 0.89866 |
| resnet_50_deeplabv3+ (finetuned) | 0.89157 |
| resnet_34_unet++ | 0.89126 |
| resnet_50_unet++ | 0.89275 |
| xception_unet++ | 0.89297 |
| **Street Artist** | |
| Street Artist | **0.92202** |

Table I
RESULTS REPORTED AS F1 SCORES ACQUIRED FROM THE KAGGLE COMPETITION. THE "CUSTOM" KEYWORD REFERS TO OUR OWN IMPLEMENTATION OF A CUSTOM UNET ARCHITECTURE, WHILE "FINETUNED" INDICATES ANOTHER ROUND OF TRAINING WITH A LOWER LEARNING RATE ON THE ORIGINAL TRAINING SET.

As demonstrated in Table I, while our individual base models—comprising both our custom model and those implemented in the SMP library—showed limited performance when evaluated independently, the *Street Artist* model exhibited a notable improvement. It surpassed the highest-performing base model's competition score by nearly 0.015, a considerable margin in this context. Additionally, the *Street Artist* model exceeded the average Kaggle score of the base models by over 0.025, underscoring its enhanced effectiveness. This result is particularly promising given the presence of even stronger base models, suggesting that our approach holds significant potential for further optimization and improvement.

## VI. LIMITATIONS OF THE STUDY AND FUTURE WORK

One of the primary limitations of our project is the reliance on a relatively small dataset. While data augmentation techniques were employed to increase the dataset's diversity, the initial limited number of training images may still restrict the model's ability to generalize to entirely new and unseen data. Additionally, the reliance on aerial imagery from a single source might limit the model's robustness to variations in image quality and environmental conditions encountered in different regions or seasons.

Another limitation is the computational complexity and resource requirements of the models used, particularly the ensemble approach in the *Street Artist* model. Training and fine-tuning multiple deep learning models demand substantial computational power and time, which may not be feasible in all practical applications or for all users. This constraint may affect the scalability and practical deployment of the proposed method.

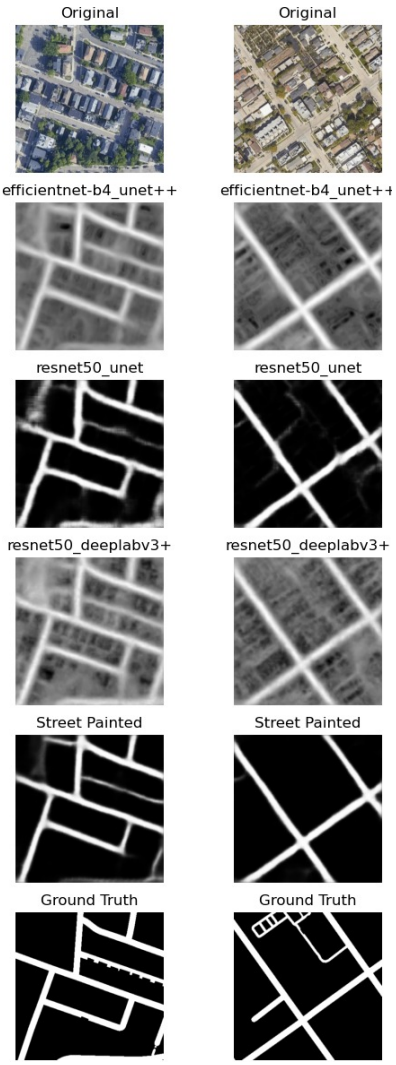


Figure 3. Submodels and *Street Artist* output examples

Future work can address these limitations by exploring the use of larger and more diverse datasets, potentially incorporating data from various geographical regions and different seasons to improve the model's robustness and generalizability. Additionally, efforts can be made to optimize the computational efficiency of the models, such as investigating model compression techniques or more efficient architectures. Further research could also explore the integration of additional data sources, such as LiDAR or multispectral imagery, to enhance the segmentation accuracy and reliability of the model in different scenarios.

## REFERENCES


[1] P. F. O. Ronneberger and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2015, pp. 234–241.

[2] P. Yakubovskiy, "Segmentation models," 2019, online; accessed 2024-07-26. [Online]. Available: https://github.com/qubvel/segmentation_models

[3] E. S. J. Long and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3431–3440.

[4] I. K. K. M. L.-C. Chen, G. Papandreou and A. L. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 4, pp. 834–848, 2017.

[5] A. H. V. Badrinarayanan and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for robust semantic pixel-wise labeling," *arXiv preprint arXiv:1505.07293*, 2015.

[6] Q. L. Z. Zhang and Y. Wang, "Road extraction by deep residual u-net," *IEEE Geoscience and Remote Sensing Letters*, vol. 15, no. 5, pp. 749–753, 2018.

[7] V. Mnih and G. E. Hinton, "Learning to detect roads in high-resolution aerial images," in *European Conference on Computer Vision*. Springer, 2010, pp. 210–223.

[8] T. Y. S. Saito and Y. Aoki, "Multiple object extraction from aerial imagery with convolutional neural networks," *Electronic Imaging*, vol. 2016, no. 10, pp. 1–9, 2016.

[9] V. C. C. B. Marcelo Bertalmío, Guillermo Sapiro, "Image inpainting," 2000.

## APPENDIX

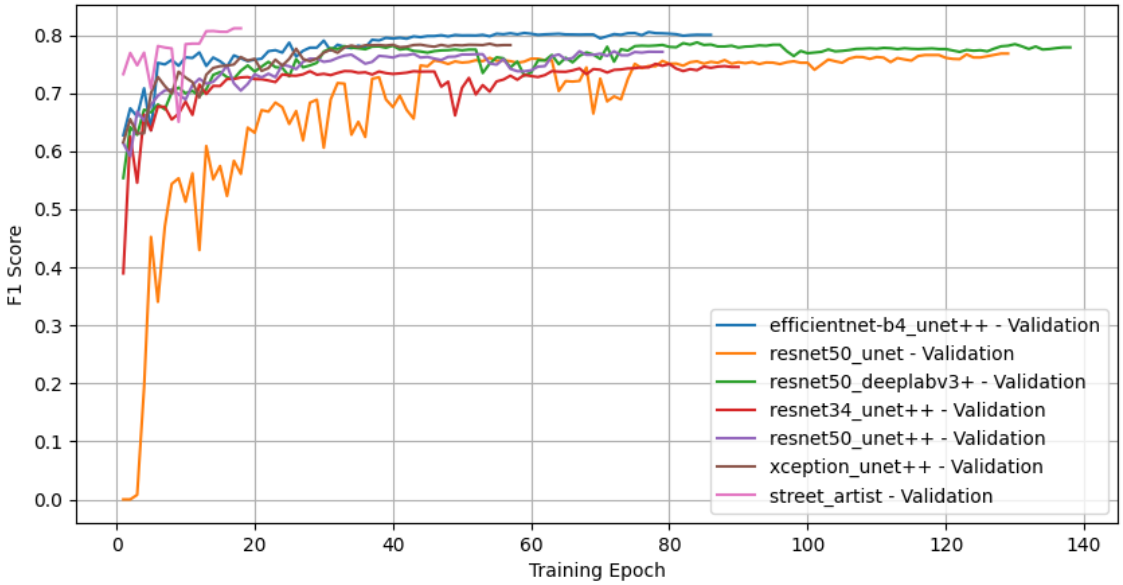


Figure 4. F1 validation scores over epochs